Proceedings of the 19th International Conference on Educational Data Mining

Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So & Young Hoan Cho (eds).

27 June–3 July, 2026
COEX Convention & Exhibition Center
Seoul, Republic of Korea

©2026 International Educational Data Mining Society

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Download copies of this and other EDM proceedings from:

International Educational Data Mining Society (IEDMS)
https://educationaldatamining.org

Proceedings of the 19th International Conference on Educational Data Mining.
Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So & Young Hoan Cho (eds).
June 27–July 3, 2026. Seoul, Republic of Korea.
ISBN 978-1-7336736-7-9

Preface

The International Conference on Educational Data Mining (EDM) is the annual flagship conference of the International Educational Data Mining Society. The 19th iteration of the conference, EDM 2026, is participating in the Festival of Learning 2026 in Seoul, Republic of Korea, from June 27 to July 3, 2026. This year's theme is Educational Data Mining Across Borders: Bridging Disciplines, Contexts, and Perspectives. Educational Data Mining has reached a stage where methods and findings are relevant across multiple areas of research and practice. Many of the challenges in learning and teaching cannot be addressed from a single perspective, and solutions are strengthened by crossing disciplinary, contextual, and cultural boundaries. This year's theme highlights work that integrates perspectives, links research and practice, and broadens the impact of educational data mining to support diverse and inclusive learning opportunities.

The scientific programming for EDM 2026 includes:

- Keynote talks by outstanding researchers in the field as part of the Festival of Learning 2026, including EDM's keynote speaker Bongshin Lee (Yonsei University, Republic of Korea).
- A plenary *Prof. Ram Kumar Educational Data Mining Test of Time Award* talk by Radek Pelánek (Masaryk University, Czech Republic), along with a discussion panel.
- A *Data Set Award* talk by Dorottya Demszky (Stanford University, USA) and Heather Hill (Harvard University, USA).
- Tutorials and workshops as part of the Festival of Learning 2026.
- Nineteen paper presentation sessions with 1 or 2 parallel tracks.
- A poster presentation session.
- An industry track session.
- A doctoral student consortium.

This year, EDM received 138 submissions to the Full Papers track (10 pages), 114 to the Short Papers track (6 pages), and 102 to the Posters and Demos track (4 pages). The program committee accepted 32 full papers (23.2% acceptance rate), 24 short papers (21.1% acceptance rate), and 46 poster and demo papers (45.1% acceptance rate). The EDM 2026 Industry track fostered exchange between industry application research and basic research, and 3 papers were included in the industry track. Furthermore, EDM 2026 also continued its tradition of providing opportunities for young researchers to present their work and receive feedback from their peers and senior researchers. The Doctoral Consortium this year featured 19 participants.

We thank all the authors who submitted their work and the program committee members for their expert inputs. We thank the members of the organization committee for their leadership that made this conference possible. We thank the sponsors of EDM 2026 and the Festival of Learning 2026

for their generous support. And, a big thank you to the local organizing committee of the Festival of Learning 2026 who made this event memorable.

Anthony Botelho	University of Florida, USA	Program Chair
Maria Mercedes T. Rodrigo	Ateneo de Manila University, Philippines	Program Chair
Adish Singla	MPI-SWS, Germany	Program Chair
Hiroaki Ogata	Kyoto University, Japan	General Chair
Hyojeong So	Ewha Womans University, Republic of Korea	General Chair
Young Hoan Cho	Seoul National University, Republic of Korea	General Chair

June 27, 2026
Seoul, Republic of Korea

Organizing Committee

General Chairs

- Hiroaki Ogata (Kyoto University, Japan)
- Hyojeong So (Ewha Womans University, Republic of Korea)
- Young Hoan Cho (Seoul National University, Republic of Korea)

Program Chairs

- Anthony Botelho (University of Florida, USA)
- Maria Mercedes T. Rodrigo (Ateneo de Manila University, Philippines)
- Adish Singla (MPI-SWS, Germany)

Equity, Diversity, Inclusion, and Accessibility Chairs

- Anna Rafferty (Carleton College, USA)
- Jade Maï Cock (LIP6 Sorbonne Université, France)

Industry Track Chairs

- Jeeheon Ryu (Chonnam National University, Republic of Korea)
- Hae-Deok Song (Chung Ang University, Republic of Korea)
- Jeongyun Han (University of Seoul, Republic of Korea)

Poster & Demo Track Chairs

- Shashi Kant Shankar (Ahmedabad University, India)
- Irena Koprinska (The University of Sydney, Australia)

Doctoral Consortium Chairs

- Jaclyn Ocumpaugh (University of Houston, USA)
- Nathaniel Blanchard (Colorado State University, USA)

JEDM Track Chairs

- Guanliang Chen (Monash University, Australia)
- Carrie Demmans Epp (University of Alberta, Canada)

Workshop & Tutorials Chairs

- Jina Kang (University of Illinois Urbana-Champaign, USA)
- Benjamin Paaßen (Bielefeld University, Germany)

Awards Chairs

- Tanja Käser (EPFL, Switzerland)
- Nigel Bosch (University of Illinois Urbana-Champaign, USA)
- Mingyu Feng (WestEd, USA)

Web Chairs

- Paul Salvador Inventado (California State University Fullerton, USA)
- Hongming (Chip) Li, Student Volunteer (University of Florida, USA)

Proceedings Chairs

- Miguel Portaz (UNED, Spain)
- Ana Serrano Mamolar (University of Burgos, Spain)

IEDMS Officers

Luc Paquette	President	University of Illinois Urbana-Champaign, USA
Anna Rafferty	Treasurer	Carleton College, USA

IEDMS Board of Directors

Rakesh Agrawal	Data Insights Laboratories, USA
Tiffany Barnes	North Carolina State University, USA
Nigel Bosch	University of Illinois Urbana-Champaign, USA
Neil Heffernan	Worcester Polytechnic Institute, USA
Sharon Hsiao	Santa Clara University, USA
Tanja Käser	EPFL, Switzerland
Kenneth Koedinger	Carnegie Mellon University, USA
Ramkumar Rajendran	Indian Institute of Technology Bombay, India

Program Committee

Senior Program Committee Members

Jill-Jênn Vie	Inria Lille
Anna Rafferty	Carleton College
Sébastien Lallé	Sorbonne University
Roger Nkambou	Université du Québec à Montréal
Sidney D'Mello	University of Colorado Boulder
Andrew Lan	University of Massachusetts Amherst
Tiffany Barnes	North Carolina State University
Mingyu Feng	WestEd
Benjamin Paaßen	Bielefeld University
Jaclyn Ocumpaugh	University of Houston
Irena Koprinska	The University of Sydney
Ashish Gurung	Carnegie Mellon University
Yang Shi	Utah State University
Neil Heffernan	Worcester Polytechnic Institute
Paul Inventado	California State University Fullerton
Sherry Sahebi	University at Albany - SUNY
Ranilson Paiva	Universidade Federal de Alagoas
Collin Lynch	North Carolina State University
Bita Akram	North Carolina State University
Thomas Price	North Carolina State University
Yo Ehara	Tokyo Gakugei University
Agathe Merceron	Berliner Hochschule für Technik
Dragan Gasevic	Monash University
Cristina Conati	UBC
Tanja Käser	EPFL
Atsushi Shimada	Kyushu University

Enkelejda Kasneci	Technical University of Munich
Agoritsa Polyzou	Florida International University
Niels Seidel	FernUniversität in Hagen
Srecko Joksimovic	Education Future, University of South Australia
Ryan Baker	University of Pennsylvania
Caitlin Mills	University of Minnesota
Kirsty Kitto	University of Bergen, Norway
Tiffany Tang	Wenzhou-Kean University
Michel Desmarais	Ecole Polytechnique de Montreal
Carol Forsyth	Educational Testing Service
James Lester	North Carolina State University
Fabiano Dorça	Universidade Federal de Uberlandia
Stephen Fancsali	Carnegie Learning, Inc.
Alexandra I. Cristea	Durham University
Stephan Weibelzahl	Private University of Applied Sciences Göttingen
Johan Jeuring	Utrecht University
Maomi Ueno	The University of Electro-Communications
Jeonghyun Lee	Georgia Institute of Technology
Mirko Marras	University of Cagliari
Nigel Bosch	University of Illinois Urbana-Champaign
Radek Pelánek	Masaryk University Brno
Giora Alexandron	Weizmann Institute of Science
Ramkumar Rajendran	IIT Bombay
Niels Pinkwart	Humboldt-Universität zu Berlin
Roger Azevedo	University of Central Florida
Alex Bowers	Columbia University
François Bouchet	Sorbonne Université - LIP6
Cristobal Romero	University of Cordoba
Jesus G. Boticario	UNED
Zach Pardos	University of California, Berkeley
Sebastián Ventura	University of Cordoba
Ratnavel Rajalakshmi	VIT University, Chennai Campus

Regular Program Committee Members

Rahul Yedida	LexisNexis
Christine Lourrine Tablatin	Pangasinan State University - Urdaneta City Campus
Eric Qiu	Cornell University
Namrata Srivastava	Vanderbilt University
Andres Felipe Zambrano	University of Pennsylvania
Ali Keramati	University of California, Irvine
Ben Khalifa Ghada	PRINCE, Research Lab , ISITCOM
Jacob Whitehill	Worcester Polytechnic Institute
Conrad Borchers	Carnegie Mellon University
Min Zhuang	Phd Student at NCSU
Maria de Los Angeles Constantino González	Tecnológico de Monterrey Campus Laguna
Yiqiu Zhou	University of Illinois at Urbana-Champaign
Daniela Rotelli	Sorbonne Université
Rosalyn Shin	University of Maryland, College Park
Danielle R Thomas	Carnegie Mellon University
Ricky Gole	Morgan State University
Fan Zhang	University of Florida
Briane Paul Samson	De La Salle University
Ivica Boticki	FER UNIZG
Teresa Ober	Educational Testing Service
Aditya Rajbongshi	University of Frontier Technology
Victor Menendez-Dominguez	Universidad Autónoma de Yucatán
Aaron Wong	University of Minnesota
Alona Strugatski	Weizmann Institute of Science
Hagit Gabbay	School of Education, Tel Aviv University
Mariano Albaladejo-González	Universidad de Murcia
Ashwin Tudur Sadashiva	Vanderbilt University
Hyeongdon Moon	Carnegie Mellon University
Arno Mel	IDLab, Ghent University
Ching Nam Hang	Saint Francis University
Tanya Nazaretsky	EPFL
Huy Nguyen	University of Pittsburgh
Divya Mereddy	Vanderbilt University
Xiaoxue Zhou	University of Maryland College Park
Ivan Luković	University of Belgrade, Faculty of Organizational Sciences
Sabine Graf	Athabasca University
Anett Hoppe	Philipps-Universität Marburg & Hessian Center for AI
Hongming Li	University of Florida
Jade Mai Cock	EPFL

Anan Schütt	University of Augsburg
Tai Le Quy	University of Koblenz
Moriah Ariely	weizmann institue of science
Alessandro Vivas	UFVJM
Vishal Kuvar	University of Minnesota
Ahana Ghosh	Max Planck Institute For Software Systems
Amine Boulahmel	Université de Rennes
Judith Azcarraga	De La Salle University
Álvaro Sobrinho	Federal University of the Agreste of Pernambuco
Huiyong Li	Research Institute for Information Technology, Kyushu University, Japan
Maria Cutumisu	McGill University
Deliang Wang	The University of Hong Kong
Shan Zhang	University of Florida
Jinnie Shin	University of Florida
Manh Hung Nguyen	MPI-SWS
Anuradha Kumari Singh	Banaras Hindu University
Shubham Gandhi	Research Staff
Duaa Baig	National Institute of Applied Sciences (INSA)
Kerstin Wagner	Berliner Hochschule für Technik
Husni Almoubayyed	Extragalactic Technologies
Eamon Worden	Worcester Polytechnic Institute
Lamgarraj Mohamed	UPJV
Chunyang Wang	Shanghai Jiao Tong University
Boxuan Ma	Kyushu University
Anupom Mondol	Texas Tech University
Sami Heikkinen	LAB University of Applied Sciences
Farhan Ali	National Institute of Education
Paul Hur	Freie Universität Berlin
Eason Chen	Carnegie Mellon University
Vitomir Kovanovic	The University of South Australia
Stephen Hutt	University of Denver
Adam Gaweda	North Carolina State University
Chao Wen	Max Planck Institute for Software Systems
Ming Gao	Shanghai Normal University
Aneng He	Universiti Putra Malaysia
Luke Eglington	Amplify Education Inc.
Isidro Butaslac	Nara Institute of Science and Technology
Wu Wen Hsiu	NTHU
Anurata Hridi	North Carolina State University
Charles Koutcheme	Aalto University

Bakhtawar Ahtisham	Cornell University
Fengjiao Tu	University of North Texas
Masaki Uto	The University of Electro-Communications
Anis Bey	La Rochelle University
Ji-Eun Lee	Singapore University of Technology and Design
Yingbin Zhang	South China Normal University
Sreecharan Sankaranarayanan	Amazon.com, Inc.
Kirk Vanacore	Cornell University
Rwitajit Majumdar	Kumamoto University
Jionghao Lin	The University of Hong Kong
Chelsea Chandler	University of Colorado Boulder
Maciej Pankiewicz	University of Pennsylvania
Aicha Bakki	Ibn Zohr University
Ange Adrienne Nyamen Tato	Université Laval
Donatella Merlini	Università di Firenze
Jorge Parraga-Alava	Universidad Técnica de Manabí
Kazuma Fuchimoto	The University of Electro-Communications
Carson Cook	Amplify Education
Markel Vigo	The University of Manchester
Yucheng Chu	Michigan State University
Patricia Angela R. Abu	Ateneo de Manila University
Vladimir Ivančević	University of Novi Sad, Faculty of Technical Sciences
Amanda La Hadi	Monash University
Hollis Lai	University of Alberta
Frank Stinar	University of Illinois Urbana-Champaign
Rémi Venant	Le Mans Université - LIUM
Jing Fan	Aalto university
Victor-Alexandru Pădurean	Max Planck Institute for Software Systems
Ting-Chia Hsu	National Taiwan Normal University
Fuzheng Zhao	Jilin Universtiy
Guan-Yun Wang	Fu Jen Catholic University
Minju Park	University of British Columbia
Lei Shi	Newcastle University
Elad Yacobson	Technion - Israel Institute of Technology
Vishesh Kumar	Vanderbilt University
Jingyi Zhao	Imperial College London
Wanjing Anya Ma	Stanford University
Filippo Sciarrone	Universitas Mercatorum
Vijay Prakash	Indian Institute of Technology Bombay
Chiao-Wei Yang	National Taiwan Normal University
Jiayi Zhang	University of Pennsylvania

Maureen Villamor	University of Southeastern Philippines
Oswaldo Velez-Langs	Universidad de Cordoba
Liang Zhang	University of Michigan
Nigel Fernandez	University of Massachusetts Amherst
Craig Zilles	University of Illinois at Urbana-Champaign
Craig Thompson	The University of British Columbia
Yann Hicke	Cornell University
Nicholas Lytle	Georgia Institute of Technology
Diego Zapata-Rivera	Educational Testing Service
Mélina Verger	INSA Lyon, LIRIS
Andrei Coronel	Ateneo de Manila University
Luis Alberto Morales Rosales	Conacyt-Universidad Michoacana de San Nicolás de Hidalgo
May Marie P. Talandron-Felipe	University of Science and Technology of Southern Philippines
Ruth Cobos	Universidad Autónoma de Madrid
Seiyon Lee	University of Florida
Alistair Willis	The Open University
Emmanuel Ayedoun	Kansai University
Yael Feldman-Maggor	The Open University
Amruth Kumar	Ramapo College of New Jersey
Tounwendyam Frédéric Ouedraogo	Université Norbert ZONGO
Phaedra Mohammed	The University of the West Indies
Yuxuan Hu	the University of Hong Kong
Tsvetomila Mihaylova	Aalto University
Mikel Larrañaga	University of the Basque Country UPV/EHU
Ella Haig	School of Computing, University of Portsmouth
Maria Bolsinova	Tilburg University
Amit Paikrao	Indian institute of Technology Bombay
Lan Jiang	University of Illinois Urbana-Champaign
Judith Azcarraga	De La Salle University
Jiawei Li	Nanyang Technological University
Yu Lu	Beijing Normal University
Pedro Manuel Moreno-Marcos	Universidad Carlos III de Madrid
Alistair Windsor	Memphis University
John Paul Vergara	Ateneo de Manila University
Thomas Christie	Digital Harbor Foundation
Henry Adorna	Dept of Computer Science, UPDiliman
Arun Balajiee Lekshmi Narayanan	University of Pittsburgh
Qianou Ma	Carnegie Mellon University

Aum Pandya	North Carolina State University
Rex Bringula	University of the East
Kole Norberg	Carnegie Learning
Juho Leinonen	Aalto University
Seyed Parsa Neshaei	EPFL
Paulo Carvalho	Carnegie Mellon University
Tung Phung	MPI-SWS
Zhikai Gao	Western Carolina University
Heeryung Choi	University of Minnesota
Jeffrey Matayoshi	McGraw Hill ALEKS
Naina Chaturvedi	Rutgers University
Yixiao Li	The Pennsylvania State University
Nachiket Kotalwar	Carnegie Mellon University
Avery Closser	University of Florida
Plaban Kumar Bhowmick	Indian Institute of Technology Kharagpur
Ed Gehringer	NCSU
Bahar Shahrokhian	Arizona State University
Thomas Trask	Georgia Institute of Technology
Margaret Perkoff	University Of Pennsylvania
Iman Mohammadi	University of California, Irvine
Wei Li	University of Florida
Roney Nascimento	USP
Denilson Barbosa	University of Alberta
Boyuan Guo	Carnegie Mellon University
Rania Ait Chabane	Université de Lorraine, LORIA
Xintian Gao	University of Florida
Elham Tajik	Albany at university
Hasnain Heickal	University of Massachusetts Amherst
Márcia Fernandes	Federal University of Uberlândia
Gamze Türkmen	Manisa Celal Bayar University
Shinyoung Lee	Independent Researcher
Mark Anthony Tolentino	Ateneo de Manila University
Chee-Kit Looi	Education University of Hong Kong
Badmavasan Kirouchenassamy	Lip6
Venkatesan Srikanth	Indian Institute of Technology, Madras
Abhishek Kumar	Independent Researcher
Jewoong Moon	University of Alabama
Ju-Ling Shih	National Central University
Naga Buddarapu	north carolina state university
Huey-Min Wu	National Taichung University of Education
Joseph Benjamin Ilagan	Ateneo de Manila University

Sylvio Rüdian	Humboldt-Universität zu Berlin
Shashi Kant Shankar	School of Arts and Sciences, Ahmedabad University
Lu Wang	Institude for Infocomm Research A*STAR
Richard Lee Davis	Stanford University
Marc Ericson Santos	Canva
Videep Venkatesha	Colorado State University
Ravi Ranjan	Florida International University
Md Akib Zabed Khan	Christopher Newport University
Remy Pages	State of Hawaii
Jason Weber	University of California, Irvine
Ran Bi	SAS Institute
Paola Mejia Domenzain	EPFL
Pauline Aguinalde	University of Florida
Nidhi Nasiar	University of Pennsylvania
Cunling Bian	Ocean University of China
Sumin Hong	Seoul National University
Jiani Wang	Worcester Polytechnic Institute
Meryem Yilmaz Soylu	Georgia Institute of Technology
Saminur Islam	North Carolina State University
Daniel Rasheed	The University of the West Indies, St Augustine
Grégory Smits	IMT Atlantique/Lab-STICC/MOTEL
Lea Cohausz	University of Mannheim
Léo Nebel	Sorbonne Université
Cecilia Bibbò	University at Albany, SUNY
Jui Bhattacharya	National Rural Livelihood Mission
Wei Qiu	Nanyang Technological University
Evanfiya Logacheva	Aalto University
Davide Fossati	Emory University
Haejin Lee	University of Illinois at Urbana-Champaign
Nicolas Hernandez	Nantes Université
Sébastien Iksal	LIUM - Le Mans Université, France
Marie Lefevre	LIRIS - Université Lyon 1
Effat Farhana	Auburn University
Ling Tan	Australian Council for Educational Research
Paul Stefan Popescu	University of Craiova
M Ali Akber Dewan	Athabasca University
Peter Wulff	Ludwigsburg University of Education
José Raúl Romero	University of Cordoba
Jun-Ming Su	National University of Tainan
Vasile Rus	The University of Memphis
Sotiris Kotsiantis	University of Patras

Alfredo Zapata González	Universidad Autonoma de Yucatan
Aditi Mallavarapu	North Carolina State University
Feifei Han	Griffith University
Michael Liut	University of Toronto Mississauga
Surina He	University of Alberta
Jason Harley	McGill University
Irene-Angelica Chounta	University of Duisburg-Essen
Sonsoles López-Pernas	University of Eastern Finland
Jeremiah Folsom-Kovarik	Soar Technology, Inc.
Ekaterina Kochmar	MBZUAI
Liu Haiqiao	Kyushu University
Michelle Taub	University of Central Florida
Zhexiong Liu	University of Pittsburgh
Yang Jiang	Columbia University
Guher Gorgun	University of Georgia
Laura Allen	University of Minnesota
Arnulfo Azcarraga	De La Salle University
Jiangang Hao	Educational Testing Service
Qiang Ma	Kyoto Institute of Technology
Claudia Antunes	Universidade de Lisboa
Charibeth Cheng	De La Salle University
Alwyn Vwen Yen Lee	Nanyang Technological University
Cheng-Yu Chung	Chunghwa Telecom Laboratories (CHTTL)
David Pritchard	Massachusetts Institute of Technology
Nghia Duong-Trung	German Research Centre for Artificial Intelligence
Hyeji Jang	Ewha Womans University
Tsunenori Mine	Kyushu University
Petra Sauer	beuth university of applied sciences
Abhinava Barthakur	University of South Australia
Nathaniel Blanchard	Colorado State University
Martin Hlosta	The Swiss Distance University of Applied Sciences
Simón Pedro Arguijo	Tecnológico Nacional de México campus Misantla
Mohammad Khalil	University of Bergen
Yumou Wei	Carnegie Mellon University
Mirka Saarela	University of Jyväskylä
Ean Teng Khor	Nanyang Technological University
Tuyet-Trinh Vu	SOICT-HUST
Khushboo Thaker	University of Pittsburgh
Armelle Brun	LORIA - Université de Lorraine
Wenbin Gan	National Institute of Information and Communications Technology

Faruk Ahmed	The University of Memphis
Püren Öncel	University of Minnesota Twin Cities
Laia Albó	Universitat de Vic
Raphael Alampay	Ateneo de Manila University
Pham-Duc Tho	Vietnam National University
Julien Broisin	IRIT, Université Toulouse III
Yanyan Li	Beijing Normal University
Jon Fernandez	Ateneo de Manila University
David Joyner	Georgia Institute of Technology
Shiyao Wei	Florida State University
Philip I. Pavlik Jr.	University of Memphis
Chenglu Li	University of Utah
Clara Belitz	University of Illinois Urbana-Champaign
Sébastien Lallé	Sorbonne University, CNRS, LIP6
Ziwei Wang	The University of Sydney
Muhammad Sajjad Akbar	University of Sydney
Haejin Lee	University of Sydney
Damilola Babalola	North Carolina State University
Muntasir Hoq	North Carolina State University
Jia Zhu	Northeastern University
Geoffray Bonnin	Université de Lorraine - LORIA
Swathi Krishnaraja	University of Potsdam
Brahim Hmedna	ibn zohr university science agadir
Renza Campagni	Università degli Studi di Firenze
Korinn Ostrow	Edmentum
Hibiki Ito	Kyoto University
Eliana Scheihing	Universidad Austral de Chile
Syaamantak Das	Indian Institute of Technology Bombay
Sein Minn	INRIA
Kaiqi Yang	Michigan State University
Keith Brawner	United States Army Research Laboratory
Yihong Yuan	The University of Sydney
Ruikun Hou	University of Tuebingen
Hagit Gabbay	Weizmann Institute of Science
Aditi Singh	Cleveland State University
Beverly Woolf	University of Massachusetts
Püren Öncel	University of Valencia

Sponsors

Platinum

Gates Foundation

Google

Silver

/*elice*/

Bronze

multicampus Eedi VitalSource®

Award Support



Table of Contents

Frontmatter

Preface . . . iii
Organizing Committee . . . v
Sponsors . . . xvii

Abstracts

Keynote Talk . . . 1
Prof. Ram Kumar EDM Test of Time Award Talk. . . . 2
EDM Data Set Award Talk . . . 2
JEDM Talks . . . 3

Full Papers

A Causal Framework for Estimating Heterogeneous Effects of On-Demand Tutoring . . . 6
Kirk Vanacore, Danielle R Thomas, Digory Smith, Bibi Groot, Justin Reich and Rene Kizilcec

A New Domain-Informed Learner Model with Uncertainty-Aware Knowledge Mastery Propagation . . . 18
Rania Ait Chabane, Armelle Brun and Azim Roussanaly

A Psychometric Framework for Context-Conditional Evaluation of LLM Tutoring in Programming Education . . . 29
Jinnie Shin, Zhe Li, Pauline Aguinalde and Laura Cruz Castro

CantoTalk: Probing Teacher Expertise From Fine-Tuned Talk Move Representations . . . 44
Mayank Sharma, Xinman Liu and Gordon Yeung

Characterizing Students' LLM Usage Behaviors and Their Association with Learning in Critical Thinking Tasks . . . 57
Minju Park, Ivan Orozco Vasquez and Cristina Conati

Confusion-Aware Rubric Optimization for LLM-based Automated Grading . . . 67
Yucheng Chu, Hang Li, Kaiqi Yang, Yasemin Copur-Gencturk, Joseph Krajcik, Namsoo Shin and Jiliang Tang

Do interviews change participants' behaviors?: Using log data to investigate student reactivity during data-driven classroom interview . . . 79
Hyeongjo Kim, Luc Paquette, Yiqiu Zhou, Qianhui Liu, Jaclyn Ocumpaugh, Andres Felipe Zambrano and Amanda Barany

Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations . . . 90
E. Margaret Perkoff, Megan Lyons, Alexandria Mulqueen, Ryan S. Baker, Alan Gee, Jaclyn Ocumpaugh, Adam Porsch, Neville Ryant, Julia Parish-Morris and Lyle Ungar

Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression 104
Megan Caruso, Robert Moulder, Rosy Southwell and Sidney D'Mello

From Heuristics to Analytics: Forecasting Effort and Progress in Online Learning 117
Eric S. Qiu, Danielle R Thomas, Boyuan Guo, Vincent Aleven and Conrad Borchers

Improving Hybrid Human-AI Tutoring by Differentiating Human Tutor Roles Based on Student Needs 130
Ashish Gurung, Ge Gao, Boyuan Guo, Alex Houk, Erin Gatz, Danielle R Thomas, Shivang Gupta, Lee Branstetter, Emma Brunskill, Vincent Aleven, Kenneth Koedinger and Jordan Gutterman

LLM Reasoning Predicts When Models Are Right: Evidence from Coding Classroom Discourse 143
Bakhtawar Ahtisham, Kirk Vanacore, Zhuqian Zhou, Jinsook Lee and Rene F. Kizilcec

Leveraging Privileged Multimodal Data to Model Cognitive States in an Interactive Learning Platform 155
Deniz Sonmez Unal, Theresa Mowad, Catherine Arrington, Erin Solovey and Erin Walker

MTBU-Bench: a Multimodal Benchmark for Evaluating Video-LLMs in Teacher Behavior Understanding 167
Zhiyi Duan, Jiangshan Guan, Lei Gao and Qianli Xing

Measuring Math or Mouse Skills? Behavioral Dimensions and Multiple Pathways to Test-Taking Efficiency in Computer-Based Testing 180
Xinman Liu and Eunjung Myoung

Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory 193
Amit Paikrao, Vishwas Badhe, Indrayani Nishane, Jyoti Shaha, Ramkumar Rajendran and Ritayan Mitra

No Single Metric Tells the Whole Story: Interpreting Multiple Pattern Mining Metrics . . . 205
Andres Felipe Zambrano, Jaclyn Ocumpaugh, Ryan S. Baker, Zhanlan Wei, Xiner Liu, Hyeongjo Kim, Qianhui Liu, Jeffrey Ginger, Luc Paquette and Amanda Barany

Pattern-based Knowledge Component Extraction from Student Code Using Representation Learning 217
Muntasir Hoq, Griffin Pitts, Tirth Bhatt, Aum Pandya, Andrew Lan, Peter Brusilovsky and Bita Akram

Personalization in Educational Data Mining and Learning Analytics: A Systematic Review (2015 - 2025) 229
Sylvio Rüdian

Real Enough to Matter? Implications of Synthetic Data for Reproducible Learning Analytics 243
Elena Tiukhova, Grzegorz Meller, Dimitri Van Landuyt, Tinne De Laet, Bart Baesens and Monique Snoeck

Stability and Contextual Sensitivity of Keystroke Process Features in Longitudinal Student Writing Portfolios 256
Damilola Babalola, Naga Buddarapu, Piotr Mitros, Paul Deane and Collin Lynch

Stable and Privacy-Preserving Synthetic Educational Data with Empirical Marginals: A Copula-Based Approach 267
Gabriel Diaz Ramos, Lorenzo Luzi, Debshila Basu Mallick and Richard Baraniuk

Temperature and Persona Shape LLM Agent Consensus With Minimal Accuracy Gains in Qualitative Coding 280
Conrad Borchers, Bahar Shahrokhian, Francesco Balzan, Elham Tajik, Sreecharan Sankaranarayanan and Sebastian Simon

The Cognitive Cost of Poor Item Design: How Item-Writing Flaws Shape Response Time and Difficulty Across Assessment Domains 293
Steven Moore

The Semantic Gap in Behavioral Embeddings: Why Linear Methods Fail for Educational RAG in Mathematics 306
Ricky Gole and Jamell Dacon

Time as a geometric property of the latent space: An empirical study of Time-and-Order RoPE in Knowledge Tracing 317
Willian Costa, Filipe Dwan Pereira, Gabriel Alves, André Nascimento, Luiz Rodrigues, Mario A. P. Santos, Tyagi M. Lima and Rafael Ferreira Mello

To What Extent Do Keystroke Dynamics Differentiate Correct from Incorrect Problem Solving 329
Oscar Deho, Ryan Baker, Digory Smith and Simon Woodhead

Toward LLM-supported Automated Assessment of Critical Thinking Subskills 340
Marisa Peczuh, Nischal Ashok Kumar, Ryan Baker, Blair Lehman, Danielle Eisenberg, Caitlin Mills, Payu Wittawatolarn, Kushaan Naskar, Keerthi Chebrolu, Sudhip Nashi, Cadence Young, Brayden Liu, Sherry Lachman and Andrew Lan

UNVaMP: Neural Knowledge Tracing with Variational Regularization of Latent Knowledge Dynamics 353
Carson Cook, Ahmed Zerouali, Anthony Schmidt, Reginald Ziedzor, Paul Lin and Luke Eglington

Utility-Preserving De-Identification for Math Tutoring: Investigating Numeric Ambiguity in the MathEd-PII Benchmark Dataset 365
Zhuqian Zhou, Kirk Vanacore, Bakhtawar Ahtisham, Doug Pietrzak, Daryl Hedley, Jorge Dias, Jinsook Lee, Rene Kizilcec, Chris Shaw and Ruth Schäfer

What Changes When the Interlocutor Is an AI? Interactional Fluency and Linguistic Uptake in L2 Spoken Dialogue 376
Russell Scheinberg, Ameeta Agrawal, Tetyana Sydorenko, Kalab Kahsay, Nina Vyatkina and Griet Boone

What Do LIWC Features Capture in Multilingual Classroom Talk? A Systemic Functional Linguistic Analysis 388
Kaycie Barron, Ananya Ganesh, Shamya Karumbaiah and Mariana Castro

Short Papers

A Human-in-the-Loop Pipeline for Multi-Class SRL Classification 400
Andreas Sebastian Kissmehl, Olga Viberg, Olov Engwall and Richard Lee Davis

A Three-GenAI Consensus Model for Deductive Analysis of Response Data 408
Amruth Kumar, Renée McCauley, Bonnie MacKellar and Tammy Vandegrift

An Ensemble Stacking with Perception-aware Meta-learning (ESPM) for Apprenticeship Dropout Prediction in France 415
Duaa Baig, Diana Nurbakova, Sylvie Calabretto and Baba Mbaye

Augmenting Student Profiles and Course Attributes with Large Language Models for Course Recommendation 422
Tianyuan Yang, Ren Baofeng, Chenghao Gu, Feike Xu, Boxuan Ma and Shinichi Konomi

Cross-Context Urgency Detection in MOOC Forums 430
Shreya Singhal

Estimating IRT Parameters from Text via an ICC-Based Supervised and Contrastive Loss . . 437
Yoshiki Hori, Takumi Itamiya, Iwao Tanuma and Emiko Yoshihara

Fair When Sparse, Accurate When Rich: RL-Gated Conversational Fusion for Temporally Equitable Knowledge Tracing 446
Fan Zhang, Rui Guo and Hai Li

Federated Code-Aware Knowledge Tracing for Privacy-Preserving Student Performance Forecasting 453
Anurata Prabha Hridi, Muntasir Hoq, Seyyedali Hosseinalipour, Rajeev Sahay and Bita Akram

From Flat to Structural: Enhancing Automated Short Answer Grading with GraphRAG . . . 462
Yucheng Chu, Haoyu Han, Shen Dong, Hang Li, Kaiqi Yang, Yasemin Copur-Gencturk, Joseph Krajcik, Namsoo Shin and Hui Liu

How Do Analytic Design Choices Shape MLLM-Based Detection of Non-Verbal Behaviors in 360-Degree Classroom Video 469
Yiqiu Zhou, Chris Palaguachi, Jina Kang, Luc Paquette, Noah Schroeder, Zhenhailong Wang and Yuji Zhang

Identifying Items on Which Humans and Chatbots Diverge Using Differential Item Functioning 479
Licol Zeinfeld, Alona Strugatski, Ziva Bar-Dov, Ron Blonder, Shelley Rap and Giora Alexandron

Improved Power in a Reanalysis of an ASSISTments Efficacy Trial by Leveraging Auxiliary Data 487
Adam Sales, Kevin Huang, Mingyu Feng and Johann Gagnon-Bartsch

Improving Speech Recognition of Named Entities in Classroom Speech with LLM Revision and Phonetic-Semantic Context 494
Viet Anh Trinh, Xinlu He and Jacob Whitehill

Mining how curiosity spreads in student-led online discussions 500
Farhan Ali

Multimodal LLM-based approach for detecting Disengagement from Task Goal (DTG) in Math problem solving 508
Nidhi Nasiar, Ryan S. Baker, Jaclyn Ocumpaugh, Zeyneb S. Sarioglan, Caitlin Mills and Samuel Rhodes

Multimodal Learning Analytics for Programming: Cognitive Difficulties, Attention Dynamics, and Physiological Regulation 516
Huiyong Li and Boxuan Ma

Simulating Learners' Task-Selection Strategies and System Constraints in Mastery Learning 524
Haley Noh, Aarna Chowdhary, Jeroen Ooge, Vincent Aleven and Conrad Borchers

Simulating Students' Java Programming Errors with Large Language Models 532
Ali Keramati, Jie Cao, Iman Mohammadi, Mark Warschauer and Yang Shi

Spatiotemporal Link Formation Prediction in Social Learning Networks Using Graph Neural Networks 541
Ali Mohammadiasl, Bita Akram, Seyyedali Hosseinalipour and Rajeev Sahay

TS-Interact: A Multimodal Dataset for Characterizing Classroom Teacher- and Student-Led Instructional Behaviors 549
Weigang Lu, Yaoxuan Peng, Saisai Ye and Cunling Bian

Teaching Language Models How to Code Like Learners: Conversational Serialization for Student Simulation 556
Charles Koutcheme, Juho Leinonen and Arto Hellas

The Medium is the Message: How Lecture Q&A Channel Structure Shapes Student Questioning 564
Meenakshi Mittal, Kaden Tang, Christopher Mach and Narges Norouzi

To What Extent Can Early Keystroke Patterns Predict Knowledge Gain in Mathematics . . . 571
Oscar Deho, Ryan Baker, Digory Smith, Jiayi Zhang and Simon Woodhead

Tracing Conceptual Overgeneralization in Middle School Students' Reasoning about Quantum Randomness 578
Jihyun Rho, Jiwon Kim, Zeynep G. Akdemir-Beveridge and Muhsin Menekse

Poster Demo Papers

A demonstration of the Quick Red Fox (QRF) system: An application for conducting Data-Driven Classroom inter-views 585
Luc Paquette and Jaclyn Ocumpaugh

Accounting for Classroom Effects in Student-Level Randomized Online Experiments: The Ordered LOOP Estimator 589
Yanping Pei, Adam Sales and Johann Gagnon-Bartsch

Automating Literature Screening with Multi-Agent Consensus Seeking: A Preliminary Study 594
Deliang Wang, Xian Chen, Joy Yu and Gaowei Chen

Beyond Idealized Cooperation: Epistemic Noise in Collaborative Problem Solving 600
Videep Venkatesha, Ethan Seefried, Changsoo Jung and Nathaniel Blanchard

Bias in MOOC Completion Prediction: A Fairness Analysis Across African Countries and Socio-Economic Contexts 605
Jade Mai Cock, Sébastien Lallé, Mélina Verger and François Bouchet

Cognitive and Metacognitive Process Differences in High and Low Writing Performers: A Trace-Based SRL Analysis of Web Log Data 611
Hyeeun Kim, Minju Kang, Jeongyun Han and Young Hoan Cho

Comparing RAG and GraphRAG for Page-Level Retrieval Question Answering on a Math Textbook 616
Eason Chen, Chuangji Li, Shizhuo Li, Zimo Xiao, Jionghao Lin and Ken Koedinger

Constrained Multimodal Neural Networks for Interpretable Adaptation Rule Extraction in Game-Based Learning 622
Ange Tato and Roger Nkambou

Coordination Regimes in StudentInteraction: The Integration Trap and Toward Enactment-Aware Scaffolding 627
Fangwei Lin, Mengchen Zhang, Zhihan Guo, Long Zhang and Khe Foon Hew

Detecting Self-Reported Cognitive-Affective States in Collaborative Problem Solving from Prosodic and Linguistic Features 632
Videep Venkatesha, Sifatul Anindho and Nathaniel Blanchard

Divergent Patterns of Attention and Exploratory Behavior Under Increasing Task Difficulty in an XR Diagnostic 637
Kukhyeon Kim and Jeeheon Ryu

Do Online Quiz-Taking Behaviors Matter? An Event-Level Log Analysis in Computer Science Courses 642
Steve Geinitz

Does Math Reason the Same Across Languages? Prompting Trade-offs in Multilingual Math Support 647
Louis Alfred Antoine Colbus, Christoph Schommer and Nina Hosseini-Kivanani

Enhanced Student Engagement through Supervised Assessment: Evidence from Kenya . . . 653
Amar Lalwani, Aidan Frieberg, Pedro Freitas, Guillermo Romero, Sharnic Djaker and Noam Angrist

Enhancing Diversity of LLM-Generated Educational Tasks 657

Manh Hung Nguyen, Sebastian Tschiatschek and Adish Singla

Evaluating Private Training in Educational Prediction Beyond Utility and Privacy Audit . . . 664
Xianghui Meng, Yujing Zhang, Xian Chen and Jionghao Lin

Extracting Curriculum Rules: Formal Modelling of the Study Entry and Orientation Phase . 669
Selina Reinhard, Tom Breuer and René Röpke

Faster, Cheaper, More Accurate: Specialised Knowledge Tracing Models Outperform LLMs 674
Prarthana Bhattacharyya, Joshua Mitton, Ralph Abboud and Simon Woodhead

From Tool to Teammate: LLM Coding Agents as Collaborative Partners for Behavioral Labeling in Educational Dialogue Analysis 679
Eason Chen, Yumo Wang, Mingyu Yuan, Sophia Judicke, Kayla Beigh and Xinyi Tang

Identifying Learning Progression Profiles in a Psychomotor Task Using Sequence Clustering 684
Emily Jensen

Investigating Patterns of Iterative Code Changes in the VEXcode VR environment by Combining Heuristic Pat-terns with Hidden Markov Modeling 689
Hyeongjo Kim, Luc Paquette and Jennie Lee

MCAR-IITA: A Procedure for Identifying Structures among Items from Incomplete Data . . 694
Peter Steiner, Aliaksei Badnarchuk, Debora De Chiusole, Milan Segedinac, Goran Savić, Luca Stefanutti and Jan Hochweber

Measuring Feature-Level Changes in Preprocessing Unfairness Mitigation 699
Hanning Lin and Frank Stinar

MetaPal: A New Paradigm for Human-Computer Interaction —From Command Response to Emotional Empathy 705
Hui Shao, Zizhuo Wu, Jing Chen, Shiyu Wang, Jiaqi Song and Yan Wang

Mining the Sequential Dependencies in Learning Events with a Self-Attentive Hawkes Process 710
Sijing Yu, Tianwei Peng, Sijie Zhang, Zengcan Xue, Su Mu and Yingbin Zhang

Multi-Feature Knowledge Tracing with Time-Series Interpretability for Mathematics Teacher Micro-Credentials 715
Hai Li and Chenglu Li

PIAS: Transparent Multimodal Assessment of Presentation Delivery 721
Nina Hosseini-Kivanani, Nafiseh Taghva and Oliver Niebuhr

PedRAG: Do LLM Tutors Practice What They Preach? Preventing Behavioral Hallucinations through Theory-Grounded Runtime Control 727
Roger Nkambou, Esperance Nfor and Daril Kengne

Permutation-Based Significance Testing for Temporal Sequential Patterns in Learning Event Logs 732
Tianwei Peng, Jinwang Song, Sijing Yu, Sijie Zhang and Yingbin Zhang

Persistence or Wheel-Spinning? Problem Solving During Computational Thinking Activities in Minecraft . . . 737
Samuel Hum, Chris Palaguachi, Kobe Duda, Jeff Ginger and H. Chad Lane

Positive or Negative? Exploring Graph-based Augmentation for Contrastive Learning in Knowledge Tracing . . . 743
Soobin Ahn, Shinyoung Lee, Lee Hyojun and Shin Insub

Predicting Disagreement with Human Raters in LLM-as-a-Judge Difficulty Assessment without Using Generation-Time Probability Signals . . . 748
Yo Ehara

Process-Level Analysis of Block-Based Robotics Programming Using Markov Chains . . . 754
Liljana Puskar, Leon Stjepan Uroic and Ana Sovic Krzic

Prompt Sensitivity in LLM-Based Essay Scoring is Model-Specific . . . 760
Giulio Corsi, Alexandru Marcoci, Maryam Abo-Tabik, Georgiana Thorpe Apreutesei, Lyba Razzaq, Roni Tibon, Yael Benn and Deborah Talmi

Scalable Argumentative Writing Support: The Efficacy of Small Open-Source Models in K-12 Writing Instruction . . . 765
Venkata Nagaraju Buddarapu, Collin Lynch and Damilola Babalola

Separating Math Errors from Language Barriers: The AI Whisperer Effect in Cross-Lingual Knowledge Tracing . . . 770
Mario Pureco-Razo and Antonieta Ceron-Ponce

Simulating Programming Traces and State Spaces for Hint Generation . . . 774
Jesper Dannath and Benjamin Paa

Temporal Clustering of Flight Simulator Data for Learner Profiling: A Methodological Comparison . . . 780
Cheker Neffati, Ange Adrienne Nyamen Tato, Georges Ghazi and Hakim Trachet

The Trap": Why LLMs Underestimate Misconception-Driven Difficulty . . . 786
Amanda La Hadi, Johan Alibasa, Guanliang Chen and A. Taufiq Asyhari

Toward a Cyclic EvidenceDesign Framework: A Preliminary Study of Generating Formative Scaffolding Tasks from Solution Logs with a Large Language Model . . . 793
Yoonji Song, Insub Shin and Yunjoo Yoo

Unsupervised Anomaly Detection of Learning Behaviors Considering Contextual Information 798
Yoshihiro Yasuda, Takayoshi Yamashita, Tsubasa Hirakawa and Hironobu Fujiyoshi

Vizart: A Multimodal LLM System to Support Visual Learning . . . 803
Susanna Atanessian and Narges Norouzi

When the Finish Line Moves: Immediate Corrective Practice and Implications for Design in Digital Math Learning . . . 808
Wen Chiang Lim, Yanping Pei, Adam Sales and Neil Heffernan

Who Benefits from Looking-for-Team Forums? Examining Expert-Novice Differences in Online Team Formation 813
Ran Bi and Shiyao Wei

Would a Short Explanation Immediately Help? Examining Causal Effects of Students' Help-Seeking in Math 818
Shan Zhang, Eamon Worden, Walter Leite, Neil T. Heffernan and Anthony F. Botelho

Doctoral Consortium Papers

AI-Assisted Ultrasound Instruction for Novices: Can We Avoid Overreliance? 823
Jack Fitzgerald and Nathaniel Blanchard

Astronauts' Cognitive Mapping: Evolving Identity through the Chronospatial Frontiers . . . 826
Ginger Chen and Joo Young Park

Beyond Seeing: Alternative Representations for Image-Dependent Mathematics 832
Ethan Croteau and Neil Heffernan

Cognitive Agent Compilation for Explicit Problem Solver Modeling 835
Hyeongdon Moon, John Stamper and Carolyn Rosé

Doctoral Consortium Application: Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression 836
Megan Caruso

Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations 837
Margaret Perkoff

Exploring Disagreement as a Dialogue Policy 851
Mariah Bradford

Identifying When Support Matters Most: Longitudinal LMS Behaviors To Predict Failure Trajectories 855
Christopher Steadman

Identifying and Modeling Epistemic Emotions in Collaborative Problem-Solving 860
Sifatul Anindho and Nathaniel Blanchard

Modeling Epistemic Vigilance in Collaborative Problem Solving: A Multimodal Approach Using Social Deduction as a Proxy Testbed 865
Videep Venkatesha and Nathaniel Blanchard

Modeling Learner-AI Interaction in Time-Constrained Programming Tasks 870
Omkar Joshi

Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory 875
Amit Paikrao and Ramkumar Rajendran

Quantifying Effects of New Advanced Placement STEM Courses through LLM-Assisted Dataset Linkage at Scale 876
Daniela Ganelin

Real-Time Scaffolding of Critical Thinking: AI Agents for Cognitive Presence Detection in Discussions 881
Fengjiao Tu, Ji Hyun Yu and Haihua Chen

Scaling, Demonstrating and Personalizing: the Benefits of LLMs in Educational Contexts . . 884
Eamon Worden and Neil T. Heffernan

Toward Vision-Language Models as AI Tutors in Physical STEM Manipulation 889
Changsoo Jung and Nathaniel Blanchard

Towards Modeling Affecto-Cognitive-Metacognitive Interactions in Machine Learning Education 892
Aditya Rajmane, Ramkumar Rajendran and Kshitij Sharma

When and How Does LLM Support Help Learning? Toward Data-Driven Design Principles for AI-Enhanced Educational Interfaces 896
Eason Chen

Industry Papers

Assessing Early Literacy at Scale: Format Effects in Unsupervised Digital Assessment . . . 901
Harriet Crisp and Amar Lalwani

Decoding Chinese Pinyin Keystrokes by Comparing Transcription and Composition Tasks . 909
Qizhi Xu, Zedi Lin, Zongting Ge, Mo Zhang and Mengxiao Zhu

Early Prediction of Student Dropout in a Large-Scale Cram School Using Longitudinal Test Performance Data 917
Masanori Yamada, Xuewang Geng, Ryosuke Oogushi, Kouki Nakao, Hiroyuki Iwashita, Koji Takano, Yuji Hirata, Masaki Miyazono, Hiroshi Kamio and Toshihide Tsutsui

Keynote Talk

Empowering People to Learn: Inclusive Human-Data Interaction in an AI-Driven World

Bongshin Lee Yonsei University, Republic of Korea

Data and AI increasingly shape how we make decisions, manage our health, understand the world around us, and learn. From wearable sensors and health trackers to personalized dashboards and AI-powered assistants, data has become deeply embedded in everyday life. Data visualization has long served as a powerful means of understanding data and a compelling medium for data-driven storytelling. In this talk, I introduce Human-Data Interaction (HDI) as an essential bridge between data, meaningful insight, and informed action, reflecting on the evolution of my research from data visualization toward a broader exploration of HDI. I will discuss how HDI can empower people to learn from data through reflection, sensemaking, and action. I will also highlight opportunities to help make these data experiences more accessible and inclusive for people with diverse abilities and needs.

Prof. Ram Kumar EDM Test of Time Award Talk

Metrics for Evaluation of Student Models

Radek Pelanek Masaryk University, Czech Republic

This talk is based on the paper "Metrics for Evaluation of Student Models", initially published in Journal of Educational Data Mining (Vol. 7, No 2., 2015). Researchers use many different metrics for evaluation of performance of student models. The aim of this paper is to provide an overview of commonly used metrics, to discuss properties, advantages, and disadvantages of different metrics, to summarize current practice in educational data mining, and to provide guidance for evaluation of student models. In the discussion we mention the relation of metrics to parameter fitting, the impact of student models on student practice (over-practice, under-practice), and point out connections to related work on evaluation of probability forecasters in other domains. We also provide an empirical comparison of metrics. One of the conclusion of the paper is that some commonly used metrics should not be used (MAE) or should be used more critically (AUC).

EDM Data Set Award Talk

NCTE Transcript Data

Dorottya Demszky Stanford University, USA
Heather Hill Harvard University, USA

The NCTE Transcript Data focuses on observations of teachers and students in 4th and 5th grade elementary mathematics classrooms. The anonymized transcripts include significant metadata, such as annotation of discourse moves for each turn, information about teacher background and classroom practices, and demographics. It has been used in a variety of applications, including exploring how features of teacher and teacher-student language promote inquiry and growth mindset as well as evaluating and building generative AI systems.

JEDM Talks

Seeing Is Solving: MLLMs, Reasoning, and Refusal in Visual Math

Ethan Croteau Worcester Polytechnic Institute
Neil Heffernan Worcester Polytechnic Institute

Many middle-school math problems are image-dependent: the diagram or graph carries essential information. This matters for intelligent tutoring and accessibility, where systems must reason over figures and also decline responsibly when figures are missing. We evaluate six contemporary multimodal large language models (MLLMs)—three reasoning models and three non-reasoning models—on 376 Illustrative Mathematics (IM) items labeled as image-role Required (the figure contains task-critical information not recoverable from text alone without added assumptions). Each model attempts every item three times with and without the figure under a shared prompt and scoring protocol. To reduce image-role label subjectivity, we classify items as not Required when they are solvable from text alone without additional assumptions. With images, the top-performing reasoning models achieve accuracy in the mid-50%, while non-reasoning models fall in the mid-30s to low-40s. Without images, models overwhelmingly refuse rather than guess, with only rare correct-by-chance answers. Models show moderate agreement on which items are solvable, and we release two benchmark subsets of items solved consistently across models. A qualitative audit of 83 items shows that visual misreading is the dominant failure mode for non-reasoning models, while reasoning models more often produce correct answers accompanied by adequate explanations. These results suggest tutoring systems should gate automated scoring and learner-model updates on visual-evidence availability and use scaffolds that require explicit visual-evidence binding before algebra. For accessibility, systems should treat no-image refusals as missing-context signals and elicit the figure or a structured description, enabling description-substitution experiments. We release code, prompts, and summary artifacts for replication. Code and data: https://osf.io/ct7bg/.

Using Bayesian Knowledge Tracing to Evaluate Perceptual Scaffolding and Desirable Difficulties in a Math Intervention

PUYUAN Zhang Worcester Polytechnic Institute
Siddhartha Pradhan Worcester Polytechnic Institute
Morgan Lee Worcester Polytechnic Institute
Yanping Pei Worcester Polytechnic Institute
Adam Sales Worcester Polytechnic Institute
Erin Ottmar Worcester Polytechnic Institute

The present study examines the assistance dilemma of providing combinations of perceptual supports in math education by fitting a Bayesian knowledge tracing model (BKT) to an open-source, online order-of-operations intervention dataset. This online intervention dataset contained 688 U.S. middle schoolers' accuracy data who solved order-of-operations problems in pre-test, intervention, post-test, and delayed post-test sessions. Combinations of spacing and color cues were embedded in intervention materials, alongside immediate correctness feedback and worked examples, creating nine between-subject cue conditions (congruent, incongruent, or neutral spacing and/or color). To examine the effect of perceptual cues on immediate performance and long-term learning, we trained a BKT variant, conditioning guess, slip, learn, and forget parameters on perceptual cue conditions. The estimated cue-specific BKT parameters suggest two main findings. First, the presence of congruent perceptual cues (color and/or spacing, e.g., 2 - 8 + 3 x 11) increased guessing rate and reduced slipping rate, suggesting that visually highlighting high-order suboperations facilitated students' immediate problem solving. Second, students who underwent intervention with incongruent color cues (e.g., 2 - 8 + 3 x 11) showed reduced guessing rate but exhibited the highest learning rate and the lowest forgetting rate. These results imply that providing misleading visual highlights may serve as desirable difficulties, which hinder immediate performance but benefit long-term learning. Furthermore, the impact of perceptual cues on immediate performance and long-term learning indicated by BKT parameters approximately aligned with the patterns revealed by behavioral measures (i.e., intervention, post-test, and delayed post-test accuracy). Such consistencies demonstrate the feasibility of using the BKT framework to understand the assistance dilemma of perceptual supports. The code and data are available at this link.

Personalizing Second Language Learning: Integrating AI with Learner Preference, Proficiency, and Engagement

Alireza Gharahigheh	Itec, imec research group at KU Leuven, Belgium
Pedro Ilídio	Itec, imec research group at KU Leuven, Belgium
Sameh Said-Metwaly	Itec, imec research group at KU Leuven, Belgium
Ann-Sophie Noreillie	Itec, imec research group at KU Leuven, Belgium
Robbe D'hondt	Itec, imec research group at KU Leuven, Belgium
Anaïs Tack	Itec, imec research group at KU Leuven, Belgium
Felipe Kenji Nakano	Itec, imec research group at KU Leuven, Belgium
Changsheng Chen	Itec, imec research group at KU Leuven, Belgium
Ann Fastré	Itec, imec research group at KU Leuven, Belgium
Lucy Van Kleunen	Itec, imec research group at KU Leuven, Belgium
Helena Van Nuffel	Centre for Language and Education at KU Leuven, Belgium
Fien Depaepe	Itec, imec research group at KU Leuven, Belgium
Wim Van Den Noortgate	Itec, imec research group at KU Leuven, Belgium
Celine Vens	Itec, imec research group at KU Leuven, Belgium
Piet Desmet	Itec, imec research group at KU Leuven, Belgium
Frederik Cornillie	Itec, imec research group at KU Leuven, Belgium

Personalization has become a cornerstone of online learning platforms, offering tailored experiences that enhance learner engagement, satisfaction, and performance. Moving beyond one-size-fits-all approaches, personalized systems can provide flexible access, improved efficiency, and support both cognitive and non-cognitive development. In language learning, personalization affords increased motivation, self-efficacy, confidence, and technology acceptance, while reducing instructor workload. Despite these benefits, many language learning platforms remain non-personalized. This study explores the integration of personalization in an online language learning platform taking into account three learner variables: preference, proficiency, and engagement. Preference captures reported thematic and content choices, proficiency reflects performance-based knowledge levels, and engagement measures sustained interaction or dropout risk. By personalizing based on these learner variables, we aim to tailor learning materials that align with learners' interests, match their abilities, and foster sustained participation. To our knowledge, this is the first study to investigate these three variables collectively in an online language learning platform, contributing to the advancement of personalized technology-enhanced language learning and personalized language learning. Across the three personalization variables, the experiments reveal that lightweight recommender systems outperform deep models for preference prediction, DeepIRT offers the strongest while interpretable proficiency estimates, and random survival forest, especially when informed with proficiency estimates, most effectively models engagement. In the context of NedBox as a use case, and in line with teachers' views on meaningful personalization, these models enable homepage learning material recommendations driven by learner preference and engagement, as well as proficiency-aligned exercise entry points. The source code is provided as a supplementary file for this submission.

A Causal Framework for Estimating Heterogeneous Effects of On-Demand Tutoring

Kirk Vanacore
Cornell University
kpv27@cornell.edu

Danielle R Thomas
Carnegie Mellon University
dchine@andrew.cmu.edu

Digory Smith
Eedi Labs
digory.smith@eedi.com

Bibi Groot
Eedi Labs
bibi.groot@eedi.com

Justin Reich
Massachusetts Institute of Technology
jreich@mit.edu

René Kizilcec
Cornell University
kizilcec@cornell.edu

ABSTRACT

This paper introduces a scalable causal inference framework for estimating the immediate, session-level effects of on-demand human tutoring embedded within adaptive learning systems. Because students seek assistance at moments of difficulty, conventional evaluation is confounded by self-selection and time-varying knowledge states. We address these challenges by integrating principled analytic sample construction with Deep Knowledge Tracing (DKT) to estimate latent mastery, followed by doubly robust estimation using Causal Forests. Applying this framework to over 5,000 middle-school mathematics tutoring sessions, we find that requesting human tutoring increases next-problem correctness by approximately 4 percentage points and accuracy on the subsequent skill encountered by approximately 3 percentage points, suggesting that the effects of tutoring have proximal transfer across knowledge components. These effects are robust to various forms of model specification and potential unmeasured confounders. Notably, these effects exhibit significant heterogeneity across sessions and students, with session-level effect estimates ranging from $-20.58pp$ to $+19.95pp$. Our follow-up analyses suggest that typical behavioral indicators, such as student talk time, do not consistently correlate with high-impact sessions. Furthermore, treatment effects are larger for students with lower prior mastery, which was true for both higher and lower socioeconomic status students. This framework offers a rigorous, practical template for the evaluation and continuous improvement of on-demand human tutoring, with direct applications to other forms of on-demand assistance, including emerging AI conversational tutoring interventions.

Keywords

Tutoring, Adaptive Learning Systems, Causal Inference.

1. INTRODUCTION

Kirk Vanacore, Danielle R Thomas, Digory Smith, Bibi Groot, Justin Reich, and Rene Kizilcec. A Causal Framework for Estimating Heterogeneous Effects of On-Demand Tutoring. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 6–17. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039738

Tutoring is widely recognized as one of the most effective educational interventions, particularly for students struggling academically [10, 15, 23, 21]. However, recent large-scale evidence suggests that the magnitude of K-12 tutoring effects varies widely across contexts, populations, and implementations, complicating efforts to generalize findings and guide policy decisions [20]. Much prior work has focused on "high-dosage" tutoring, which is typically defined as sustained, scheduled sessions delivered multiple times per week [33]. However, a growing class of interventions provides on-demand tutoring: brief support intended to deliver instruction precisely at moments of student difficulty [9, 22]. These interventions are increasingly embedded within adaptive learning platforms and are delivered by human tutors or AI-based agents [30, 22].

These embedded tutoring systems generate rich, fine-grained interaction data, creating unprecedented opportunities for continuous evaluation and improvement. However, they also introduce a fundamental methodological problem: students self-select into tutoring at moments of struggle, creating severe time-varying confounding that undermines naive observational comparisons. As a result, despite the growth of embedded tutoring systems, the field lacks scalable methods for rigorously estimating the causal impact of individual tutoring interactions in real-world learning platforms.

This paper addresses this gap by introducing a scalable causal inference framework for estimating the effect of on-demand assistance – in this case, human tutoring – within an adaptive mathematics learning platform. Rather than evaluating entire programs or long-term interventions, the framework is designed to estimate the immediate causal effect of individual tutoring sessions on subsequent student performance. Our proposed framework integrates three components into a unified pipeline: (1) principled construction of analytic samples to approximate counterfactual comparisons, (2) latent knowledge state estimation using Deep Knowledge Tracing (DKT; [29]) to address time-varying confounding, and (3) estimation of conditional average treatment effects for local, individual tutoring sessions using Generalized Random Forests (Causal Forests; [3]). In combination, these components provide a practical template for estimating average treatment effects and effect heterogeneity from large-scale learning interaction logs.

We further demonstrate various robustness tests for this framework, showing that with and without contextual variables from outside of the platform (e.g., standardized assessments and demographic variables), this pipeline produced consistent estimates of the tutoring effects. Our framework addresses central challenges in observational studies of educational interventions, including self-selection into treatment, unmeasured baseline ability that changes as students learn, and treatment effect heterogeneity. Furthermore, it provides opportunities to better understand when, for whom, and why tutoring is effective. Finally, this method could be used to estimate outcomes for evaluating AI models for tutoring.

Applying our framework to data from an embedded tutoring system, we find that requesting tutoring has a positive impact on immediate performance by increasing the likelihood that students answer the next problem correctly by an average of 4.01 percentage points (*pp*). We further find that this affects near transfer across skills: tutoring increases the probability of answering the first problem on the subsequent skill correctly by 2.73*pp*. However, there is substantial variability in these effects, with local effect estimates ranging from $-20.58pp$ to $19.95pp$ for immediate performance correctness and from $-23.25pp$ to $+18.24pp$ for near transfer. We find that these estimates are robust to potential confounding from sample selection or unmeasured confounding. However, our marginally nonsignificant placebo suggests that these effect estimates may be biased towards underestimating the effect sizes. Students with lower estimated prior knowledge tended to benefit more from tutoring, although this relationship was slightly attenuated for lower-income students. We conclude by discussing how causal approaches of this kind can help researchers and practitioners better understand and evaluate the mechanisms that drive effective on-demand tutoring, ultimately informing the design of improved human and AI tutoring systems.

2. BACKGROUND

2.1 Efficacy and Heterogeneity in Tutoring Interventions

Tutoring is widely recognized as one of the most effective academic interventions for improving student learning outcomes [10, 15, 23]. Comprehensive meta-analyses have found that tutoring consistently yields significant positive effects on student learning [21, 23]. These effects are impactful across grade levels and subject areas, often outperforming other educational interventions such as class-size reduction or extended school days [15, 23]. However, the efficacy of tutoring is rarely uniform; effects are often heterogeneous across different implementations [23]. Impact also varies significantly according to characteristics such as prior achievement, socioeconomic status, and baseline achievement [10]. Although tutoring often targets lower-performing students to close achievement gaps [15], other studies on voluntary, on-demand models suggest that students with higher baseline engagement or fewer structural barriers are sometimes more likely to access and benefit from support [2, 9]. Understanding this heterogeneity is crucial to determine whether specific tutoring interventions narrow achievement gaps or inadvertently widen them due to differential uptake.

The magnitude of tutoring effects is also inextricably linked to the implementation model. Much of the literature advocates for "high-dosage" tutoring, typically defined as sustained, scheduled sessions occurring three or more times per week, as a primary driver of efficacy [21, 23]. However, scaling such intensive models presents logistical and financial challenges. Consequently, many Adaptive Learning Systems (ALS) have adopted "on-demand" or "just-in-time" models, where brief support is triggered by students making specific errors rather than a fixed schedule [30, 38, 22]. Although scalable, these embedded interactions complicate the traditional definition of dosage, as exposure to treatment becomes a function of student agency and immediate need rather than administrative assignment.

2.2 Integrating Human Tutoring in Adaptive Learning Systems

To address the scalability and cost constraints of high-dosage human tutoring, recent approaches have integrated human support directly into ALS. "Hybrid" or "human-in-the-loop" models aim to combine the personalized, immediate feedback of AI-driven software with the motivational and complex pedagogical support of human tutors [6, 38]. Research suggests that human tutoring can enhance the benefits of ALS [14]. For instance, human tutors—guided by real-time dashboards to intervene when students struggle—yields statistically significant additional boosts in time-on-task and skill proficiency within an ALS, compared to students working in an ALS alone [14]. These hybrid approaches seem to benefit lower-performing students more than higher-performing ones, suggesting that human intervention may most benefit students who most need it [38].

Integrated tutoring models often take the form of "on-demand" chat support embedded within the learning environment. In these settings, the ALS identifies a knowledge gap or a specific misconception, and a human tutor enters the loop to provide targeted scaffolding [17]. Evaluations of such systems have shown that students who engage with this hybrid support demonstrate higher learning gains and knowledge transfer than those using the ALS in isolation [17, 22]. Furthermore, interventions explicitly designed to combine human tutoring with ALS personalization have been shown to nearly double math learning gains compared to control groups, providing a scalable mechanism to promote educational equity for marginalized students [6]. By offloading rote instruction to the ALS and reserving human capital for high-value interactions, these systems offer a promising pathway to deliver effective tutoring at scale [6, 38, 42].

2.3 Methodological Challenges in Estimating On-Demand Effects

Although the efficacy of tutoring is well-documented, estimating the causal impact of on-demand help within adaptive systems presents unique methodological hurdles. Unlike scheduled "high-dosage" tutoring, where attendance is often mandated or fixed, "on-demand" support is driven by student agency. This introduces a specific form of confounding: students are more likely to ask for help precisely when they are least likely to succeed without it [2, 33]. Consequently, naive comparisons between tutored and non-tutored problem attempts often yield negative correlations, reflecting the student's struggle rather than the intervention's failure.

To address this, prior EDM research has largely relied on two approaches: experiments/randomized controlled trials (RCTs) and observational matching strategies. While RCTs remain the gold standard, they are often implemented at the school or student level to evaluate general program access rather than specific interactions [9, 34, 22]. Randomizing individual help requests, which may require denying help to a struggling student for the sake of experimental control, is often ethically prohibitive and disruptive to the learning experience (e.g., wait-list designs where students are randomly assigned to get help right away or later are still disruptive). As a result, granular, problem-level insights must often be derived from observational data. In observational settings, researchers have traditionally employed propensity score matching to balance treatment and control groups on observed covariates such as prior achievement, demographics, and topic difficulty [6]. However, standard matching approaches typically rely on static or slowly changing variables (e.g., pre-test scores, diagnostic assessments). These measures fail to capture the highly dynamic, time-varying nature of a student's knowledge state during a learning session. A student's decision to ask for help is often a function of immediate confusion or a specific misconception, which static or slowly-changing covariates cannot detect.

More recent work has attempted to incorporate dynamic measures of mastery into causal estimation. Approaches such as the "Rebar" and "ReLOOP" method have demonstrated that using predictions from high-dimensional models as covariates can reinforce matching estimators and reduce bias [36, 25]. These methods allow for more precise effect estimation in the contexts of RCTs but are underutilized in observational studies where conditions are not randomized.

A related significant yet underdeveloped area in Educational Data Mining (EDM) is the integration of knowledge tracing (KT) into observational causal inference frameworks to account for time-varying confounding rooted in a student's historical performance [1, 24, 27]. In particular, DKT extends this by utilizing recurrent neural networks to generate high-dimensional latent states that capture complex temporal dependencies in student learning [29]. However, even with these richer representations, standard regression or linear adjustments may not fully account for the complex, nonlinear heterogeneity in how different students respond to tutoring.

Finally, most causal studies in EDM and related fields focus on average treatment effects [36, 25] or subgroup analyses [18, 19, 28, 30]. A growing body of work also examines heterogeneity by how students interact with learning environments [37, 26, 39, 40]. However, the field would benefit from methods that estimate local, session-level estimates of effectiveness. These estimates could be leveraged to flexibly explore the circumstances under which ALS elements, like on-demand tutoring, are most effective, including elements of heterogeneity based on how students respond.

3. CURRENT STUDY

The current study proposes an observational approach for evaluating tutoring sessions embedded within digital learning platforms by estimating their impact on students' subsequent performance. The approach integrates DKT–based estimates of students' latent knowledge states into a causal inference framework, based on the assumption of strong ignorability, that combines causal forests for counterfactual outcome estimation with augmented inverse probability weighting to obtain doubly robust, session-level treatment effects. Together, these methods yield estimates of individual tutoring session effectiveness and enable downstream analyses that characterize heterogeneity in tutoring impacts across students, contexts, and instructional conditions. We use this framework on data from Eedi, an ALS that focuses on helping students overcome math misconceptions, during which students can access on-demand human tutoring through brief text chats.

4. CONTEXT

4.1 Data

The data used for this analysis came from the treatment arm of a multi-year randomized controlled trial (RCT) designed to estimate the causal impact of Eedi on middle school mathematics achievement. Schools were randomly assigned to treatment or control conditions at the school level, with the treatment group receiving access to Eedi for two full academic years while control schools continued with business-as-usual instruction. Our current sample only includes the treatment arm of this RCT: the 12 schools and 2,585 students who had access to and used Eedi during the two-year study.

The original RCT was conducted by an independent professional evaluation organization that was not involved in the present study. The evaluation organization followed an ethics protocol that included school consent, safeguards to ensure that study procedures did not deviate from typical daily instruction, and legally compliant data handling. All data transferred from Eedi to the researchers were fully deidentified. The current study was also approved by Cornell's institutional review board (IRB0149250).

4.2 Eedi

This study was run in Eedi, which is a digital mathematics platform designed to support middle school students by identifying and addressing misconceptions in real time through diagnostic assessment. Eedi uses a large bank of carefully engineered multiple-choice questions, where each incorrect option maps to a specific, well-documented mathematical misconception. This design allows Eedi to pinpoint gaps in understanding at a fine grain and respond with targeted support, including short video explanations, structured practice, and follow-up diagnostic questions. Teachers can deploy the system flexibly in lessons or as homework, using its analytics dashboards to see patterns of misunderstanding across individuals and classes and to adapt instruction accordingly. Students work independently on quizzes of five questions assigned by their teachers. Based on their answers, Eedi responds adaptively with a sequence of hints, videos, and fluency practice to help students overcome misconceptions. Evidence from a large, school-level randomized controlled trial suggests that the platform can produce modest but meaningful improvements in mathematics attainment when implemented over an academic year [17].

4.3 On-Demand Tutoring in Eedi.

Within Eedi, human tutoring was implemented as on-demand, synchronous, one-to-one chat-based support integrated directly into students' ongoing problem-solving activity. At any point, students could initiate a live session that immediately connected them with an expert tutor. Tutors received structured context at the start of each session—including the problem text, the student's answer, and the associated misconception, allowing them to provide targeted, dialog-based support focused on diagnosing and resolving misunderstandings before students returned to independent work.

The intervention involved seventeen tutors, each with at least three years of teaching experience and specific training in tutoring. Sessions were designed to be brief and task-focused, enabling students to move fluidly between independent practice and individualized help. Tutors communicated exclusively through text chat and used a Socratic approach that emphasized questioning and guided reasoning rather than direct answer-giving. Sessions concluded once students demonstrated sufficient understanding or chose to resume their lesson, positioning tutoring as a scalable, just-in-time complement to the broader Eedi's learning environment.

Table 1 summarizes the characteristics of the tutoring sessions. Tutoring sessions were relatively brief, with a median duration of 4.2 minutes and 14 messages exchanged. Tutors contributed approximately 60% of messages on average, consistent with their role in guiding the dialogue.

Table 1: Tutoring Session Characteristics

Metric	Mean	Median	Q25	Q75
Total Messages	17.7	14	9	23
Tutor Messages	10.5	8	5	14
Student Messages	7.2	6	3	9
Duration (minutes)	5.4	4.2	2.4	6.9

5. METHODS: CAUSAL FRAMEWORK FOR LOCAL EFFECT ESTIMATION

Our causal inference framework consists of three interconnected stages, illustrated in Figure 1. (1) First, we construct an analytic dataset that links each tutoring intervention to its immediate outcome while defining an appropriate control group. (2) Second, we train a DKT model on a held-out sample of students to generate latent representations of student knowledge for both the treatment and control samples. (3) Third, we apply a Causal Forest estimator that jointly estimates propensity scores and treatment effects using the DKT-derived hidden states and predicted probabilities as covariates. We validate our findings through a series of robustness checks that incorporate contextual covariates (e.g., school assignment, standardized assessments, and demographics), alternative sample configurations to address potential participation bias, and a placebo outcome test. This framework addresses three central challenges in observational tutoring research: confounding from baseline knowledge differences, self-selection into treatment, and heterogeneity in treatment effects across students.

5.1 Analytic Sample Construction

Our method requires three samples: treatment, control, and hold-out. Table 2 presents the sample breakdown. When a student received tutoring on a problem, that instance entered the treatment sample. The next problem the student attempted without help served as our *immediate performance* outcome, and the next problem on a new skill served as our *near transfer* outcome. To ensure comparability, only problems for which at least one student received tutoring were included in the causal analysis. In addition, problem attempts from treated students in which they did not receive tutoring were excluded from the primary analysis. Treated observations were also excluded when no subsequent problem attempt was available to serve as an outcome.

Table 2: Analytic Sample Construction from Raw Data to Final Analytic Sample

	Students	Problem Attempts	Unique Problems
Original Usage Data	2,585	852,274	8,540
Treatment Sample			
All Treatment Usage	1,234	465,619	6,803
Excluded	(13)	(460,456)	(5,337)
Final Analytic Treated	1,221	5,163	1,466
Control Sample			
All Control Usage	1,351	386,655	8,017
DKT Training	676	197,332	7,485
Holdout			
Control Analysis	675	189,323	5,884
Excluded	(10)	(97,874)	(4,418)
Final Analytic Control	665	91,449	1,466
Total Analytic Sample	1,886	96,612	1,466

To avoid potential Stable Unit Treatment Value Assumption (SUTVA) violations [35], only students who never received tutoring were eligible for the control sample, as prior exposure to tutoring could influence later outcomes. We relax this restriction in a robustness check to assess whether this sample construction meaningfully affects results (see Section 5.4). Students who were never treated were randomly divided between the control and hold-out samples. The hold-out sample was used exclusively for DKT estimation (Section 5.2), while the control sample was reserved for causal effect estimation. The analytic control sample, therefore, consisted of students who (a) never received tutoring and (b) attempted at least one of the same problems as students in the treatment sample. Because not all problems overlapped with tutoring interventions, control-group attempts on non-overlapping problems were excluded. Students who lacked any problem attempts overlapping with the treatment sample were removed from the final analytic control sample.

5.2 Knowledge State Estimation

We implemented a DKT model using a Long Short-Term Memory (LSTM) architecture to generate student knowledge estimates used as pre-treatment covariates in the causal analysis [29]. The model uses a student's sequential history of question attempts to predict the probability of correctly answering subsequent questions. Critically, the model maintains a latent hidden state that encodes information from prior performance across time, allowing it to summarize a student's evolving knowledge state even when past interactions involve different constructs than the one being predicted.

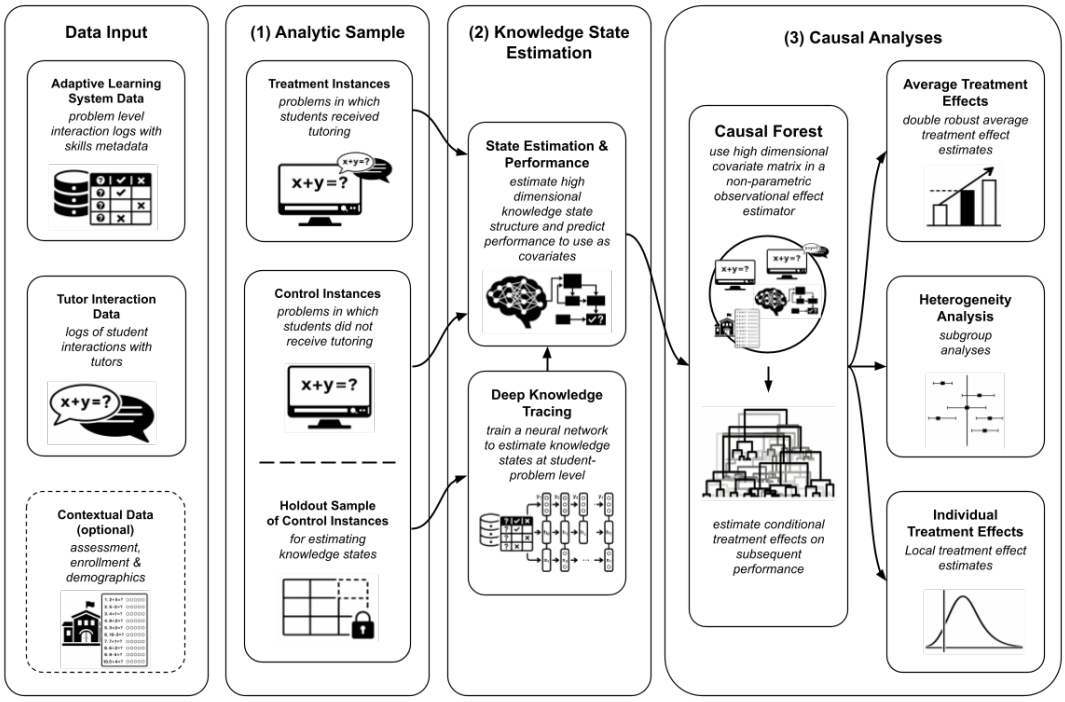


Figure 1: Causal Framework for Estimating Heterogeneous Effects of On-Demand Tutoring. The pipeline integrates ALS and tutor interaction logs to estimate student knowledge states and causal effects. These estimates drive downstream insights into effect variability. While the framework functions independently of contextual data, we incorporate it here to test robustness and identify variables for heterogeneity analysis.

The use of DKT serves a substantive causal purpose. Student knowledge is a key confounder in this setting: students are more likely to seek or be offered tutoring when they are struggling, and baseline knowledge strongly predicts future performance. By conditioning on a rich, multidimensional representation of students' latent knowledge states, we aim to reduce bias arising from time-varying confounding in tutoring assignment and outcome generation. Furthermore, DKT can provide specific point-in-time estimates of baseline knowledge that account for recent learning, which may be more robust than global pretest measures.

We implemented the DKT model in PyTorch following the architecture first introduced by Piech et al. [29]. The training data included 197,332 problem attempts across 7,485 problems representing 1,598 distinct knowledge components. Item embeddings were passed to an LSTM with a hidden dimension of 50, which updated its hidden state h_t at each time step. The resulting 50-dimensional hidden state vector $(H_1, \ldots, H_{50})$ was used as the primary knowledge representation in our causal analysis. The output layer applied a sigmoid activation function to produce probability estimates for the next construct. The model achieved an acceptable AUC of 0.72 on the analytic control sample (which was not used for training), consistent with typical performance reported for DKT models [1].

5.2.1 *Feature Extraction.*

For each observation in the treatment and control samples, we extracted the 50-dimensional LSTM hidden state calculated *just prior to the intervention.* We also included the predicted performance on the current problem, in which the student could receive tutoring support, and the subsequent problem, which serves as an outcome. Including these as covariates in causal analysis is modeled after the *Rebar* method, which has been shown to produce precise, unbiased effect estimates by applying models generated from data outside of the treatment and control sample to those samples [36].

5.3 Causal Forest Estimation

We estimated treatment effects using Generalized Random Forests (GRF) as implemented in the `grf` R package [4]. We chose to incorporate causal forests into the pipeline because of their nonparametric flexibility and their ability to model high-dimensional treatment effect heterogeneity [3].

Furthermore, causal forests offer three other advantages that are particularly relevant in our setting. First, they have been shown to perform well under targeted selection, when treatment assignment depends on predictors of the outcome itself [16, 3]. In the context of on-demand tutoring, students are more likely to seek or receive tutoring precisely when they anticipate poor performance absent intervention, rendering treatment assignment endogenous and outcome-driven. Causal forests are explicitly designed to accommodate this form of selection. Second, a key feature is *honesty*, in which separate subsamples are used for tree construction and treatment effect estimation. This sample-splitting reduces adaptive overfitting and supports valid inference by producing out-of-sample counterfactual predictions analo-

gous to leave-one-out estimation [43]. Third, causal forests directly estimate Conditional Average Treatment Effects (CATEs) as flexible functions of a high-dimensional covariate space. Given the richness of our covariates—including the 50-dimensional DKT hidden state representation—this approach enables estimation of fine-grained, unit-specific conditional effects and supports downstream analyses of treatment effect heterogeneity across students and contexts.

5.3.1 Model Specification.

The covariate matrix X included 50 DKT hidden state dimensions, cumulative historical accuracy, and DKT-based probability estimates of correct responses. The forest was trained with 500 trees, honesty enabled, and hyperparameters selected via cross-validation. We specified students as clusters in the causal forest, implementing cluster-robust estimation such that all observations from a given student are assigned to the same subsample during tree construction. As a result, treatment effect estimates account for within-student dependence, and predictions for each student are generated without using that student's own data.

5.3.2 Estimation Engine: Robinson Decomposition and Propensity Modeling

We estimate heterogeneous treatment effects using Causal Forests, which rely on the Robinson decomposition (also known as a residual-on-residual or orthogonalization strategy). Let Y_i denote the observed outcome for unit i, $Z_i \in \{0, 1\}$ the treatment indicator, and X_i the vector of pre-treatment covariates. For each outcome definition (next-problem correctness and next-skill correctness), the procedure first fits two functions using separate regression forests: the conditional mean outcome model

$$\hat{m}(X) = \mathbb{E}[Y \mid X]$$

and the propensity score model

$$\hat{e}(X) = \mathbb{P}(Z = 1 \mid X).$$

These estimates are then used to construct orthogonalized (residualized) variables:

$$\tilde{Y}_i = Y_i - \hat{m}(X_i), \qquad \tilde{Z}_i = Z_i - \hat{e}(X_i), \tag{1}$$

where $\tilde{Y}_i$ represents the deviation of the observed outcome from its covariate-expected value, and $\tilde{Z}_i$ represents the deviation of the observed treatment assignment from its predicted probability. This transformation removes outcome variation explained by baseline covariates and treatment variation explained by selection on observables, yielding Neyman-orthogonal signals for effect estimation.

The Causal Forest is then trained to model the relationship between $\tilde{Y}_i$ and $\tilde{Z}_i$ as a function of X_i, which isolates the causal signal after adjusting for baseline outcome differences ($\hat{m}$) and treatment selection ($\hat{e}$).

Propensity scores are therefore estimated internally as part of the Causal Forest procedure via a regression forest for $\hat{e}(X)$. Valid estimation requires overlap (positivity), meaning each unit must have a non-negligible probability of both treatment and control conditional on X. In our analytic sample, estimated propensity scores ranged from 0.03 to 0.89, with most observations falling between 0.05 and 0.20. Appendix A.1 presents the full propensity score distribution and associated overlap and robustness diagnostics.

5.3.3 Effect Estimates.

We report three types of effects. The Conditional Average Treatment Effect (CATE) provides unit-level predictions, where $\tau(x)$ represents the difference in expected potential outcomes for an individual with covariates x. Specifically, it captures the contrast between the expected outcome if the student receives tutoring, $Y^{(1)}$, and the expected outcome if the student does not receive tutoring, $Y^{(0)}$:

$$\tau(x) = E[Y^{(1)} - Y^{(0)} | X = x] \tag{2}$$

The Average Treatment Effect (ATE) is the population-level effect, computed using Augmented Inverse Probability Weighting (AIPW) scores:

$$\hat{\tau}^{\text{ATE}} = \frac{1}{n} \sum_{i=1}^{n} \left[\hat{\tau}(X_i) + \frac{Z_i - \hat{e}(X_i)}{\hat{e}(X_i)(1 - \hat{e}(X_i))} \big(Y_i - \hat{m}_{Z_i}(X_i)\big) \right] \tag{3}$$

The Average Treatment Effect on the Treated (ATT) is determined by restricting this calculation to treated units and represents the average effects for those who chose the treatment. The use of AIPW makes the ATE and ATT 'doubly robust,' as they combine the outcome predictions with propensity weighting to correct for bias [12]. Thus, if either the outcome model or the propensity score model is misspecified, the estimates remain consistent and asymptotically unbiased.

5.3.4 Heterogeneity Analysis.

To test for moderators of the treatment effect, we treat the estimated unit-level CATEs ($\hat{\tau}_i$) as the outcome variable in a Linear Mixed Model (MLM):

$$\hat{\tau}_i = \beta_0 + \beta_1 \cdot \text{Moderator} + \mu_s + \epsilon_i \tag{4}$$

where μ_s accounts for student-level clustering. This allows us to determine (via significance testing) which moderators explain the variability in treatment effects. Notably, we do not rely on more standard causal-forest heterogeneity summaries such as best linear projection, because our substantive interest extends to moderators that are not included in the causal forest itself, including variables derived from the tutoring sessions. Instead, we use the estimated CATEs as outcomes in follow-up analyses to examine how these external variables relate to treatment effect variation.

5.4 Robustness and Sensitivity Checks

To assess whether the model is robust to misspecification and potential unobserved confounders, we conducted a series of robustness checks. Our first sets of robustness checks determine whether adding different variables or sample configurations influences the results.

5.4.1 External Variables Added

First, we incorporated additional covariates not derived from platform usage. These included standardized pretest scores of mathematical knowledge (NWEA MAP RIT), which provide an external measure of baseline ability independent of

Table 3: Average treatment effect estimates in percentage points across different model specifications.

Model Specification	ATE (95% CI)	ATT (95% CI)
Primary Specification		
Immediate Performance (Next Problem)	4.01^{***} (2.51, 5.51)	3.98^{***} (2.43, 5.54)
Near Transfer (Next Skill)	2.73^{**} (1.12, 4.35)	2.65^{*} (0.99, 4.31)
Robustness Checks		
External Variables Added (Next Problem)	3.93^{***} (2.49, 5.37)	4.71^{***} (2.92, 6.50)
Washout Sample Included (Next Problem)	2.17^{***} (1.17, 3.16)	2.76^{***} (1.91, 3.61)
Placebo Test (Pre-Intervention)	-1.67 (-3.10, -0.24)	-0.71 (-2.15, 0.74)

Bonferroni-corrected significance: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

platform performance; a gender indicator with three categories (male, female, or neither); and Pupil Premium Funding (PPF) eligibility, a government designation for students from disadvantaged backgrounds that serves as a proxy for socioeconomic status. We also included school identifiers. The goal of this analysis was to test whether the estimated Average Treatment Effect (ATE) remains stable after adding external covariates, which would suggest that the DKT-based features may adequately capture relevant confounding and that the estimated treatment effect is not driven by unobserved demographic or institutional factors.

5.4.2 Washout Sample Included

The second robustness test evaluates potential selection bias arising from defining the control group as students who have *never* requested tutoring. These students may differ from treated students in ways not fully captured by DKT features or other observed covariates. Although restricting the control group in this way helps prevent SUTVA violations (see Section 5.1), it may introduce confounding if never-treated students are fundamentally different from those who seek tutoring. To examine this possibility, we re-estimate the model using an expanded control group that includes students who previously received tutoring, provided that the prior session did not occur within the previous two skills (i.e., a washout sample). This specification allows us to account for potential carryover effects while reducing selection bias by increasing overlap between treatment and control observations.

5.4.3 Pre-Intervention Placebo Test

Next, we conducted a placebo test using an outcome for which no causal effect is possible: student performance on a problem that occurs *before* the tutoring intervention. Placebo or "negative control" outcomes are widely used in causal inference as a diagnostic tool for detecting residual confounding or model misspecification. If the identification strategy is valid, estimated treatment effects on such outcomes should be null, since they temporally precede treatment and cannot be causally influenced by it [11]. We chose a problem three items before the intervention as our placebo outcome to avoid any interference with the intervention. Finding no significant effect on this pre-treatment outcome would therefore provide additional evidence that the method is not simply capturing baseline differences between students who choose tutoring and those who do not, but is instead identifying a genuine causal effect of tutoring.

5.4.4 Additional Checks

Finally, we employ two additional robustness checks that are common in quasi-experimental analyses. We present these in detail in Appendix A. We tested the sensitivity of our results to propensity score model specification and trimming of the distribution tails to ensure common support [8]. To quantify the robustness of our estimates to unmeasured confounding (i.e., omitted variable bias), we calculate the Robustness Value (RV). The RV represents the minimum strength of association that an unobserved confounder would need to have with both the treatment and the outcome (measured in terms of partial R^2) to reduce the estimated treatment effect to zero. This allows us to benchmark the required strength of an unmeasured confounder against observed covariates like prior mastery or student SES.

6. RESULTS

6.1 Treatment Effect Estimates

Table 3 presents the doubly robust average treatment effect estimates from the Causal Forest analysis, with p-values adjusted for multiple comparisons. We find a statistically significant positive effect of tutoring on immediate performance and near transfer. The estimated average treatment effect (ATE) of tutoring on next problem correctness was 4.01 percentage points (*pp*) (CI = [2.51, 5.51]), $p < 0.001$) and accuracy on the first attempt at a problem on the next skill was 2.73*pp* (CI = [1.12, 4.35]). The effects on the treatment group (ATT) were also significant and similar in magnitude.

6.1.1 Robustness and Sensitivity.

Table 3 presents results from several robustness checks designed to evaluate whether specific modeling choices, such as covariate selection and sample construction, influence the stability of the effect estimates. The inclusion of external demographic covariates yielded a nearly identical ATE of 3.93*pp* ($CI = [2.73, 5.51]$). This stability suggests that our Deep Knowledge Tracing (DKT)-derived features successfully capture the requisite student heterogeneity, rendering additional demographic data redundant for the purposes of confounding control. In the model specification that included the washout sample (the control included students who have been treated in the past, but not within two skills), students within the control group produced a lower but still statistically significant estimate of 2.17*pp* ($CI = [1.17, 6.16]$). The convergence and overlapping confidence intervals across these disparate design assumptions provide robust evidence for a consistent causal effect. These checks were replicated for the near-transfer outcome with similarly significant results.

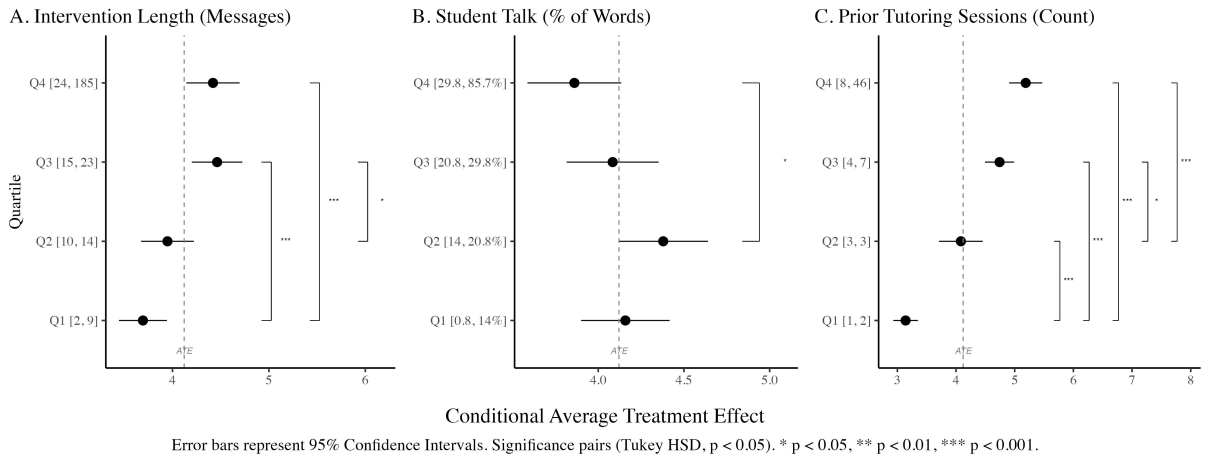


Figure 2: Effect Sizes on Immediate Performance by Tutoring Session Characteristics. Each effect is decomposed by quartiles of the characteristics. Quartile ranges are presented in brackets.

The effect estimate for the placebo outcome—a measure theoretically unaffected by the treatment—was negative and non-significant; however, this is only after applying a Bonferroni correction to account for family-wise error. This result suggests that, if anything, students in the treatment group may actually be biased toward lower baseline performance consistent with targeted selection into treatment; consequently, the observed positive treatment effects are unlikely to be artifacts of model-driven upward bias.

Two additional robustness tests are detailed in Appendix A.1. We utilized paired matching as a diagnostic tool for propensity score estimation, finding that a Random Forest propensity model achieved superior covariate balance compared to linear specifications. Furthermore, we confirmed that trimming the tails of the propensity distribution to ensure stricter overlap did not significantly alter the treatment effect estimates.

Unmeasured Confounding. Finally, as discussed above, the framework depends on a strong ignorability assumption: after conditioning on observed covariates, there are no substantial unmeasured factors that jointly influence tutoring usage and outcomes. This matters here because plausible confounders, such as student motivation, are not directly observed. We assume that some of their influence is expressed through prior performance and interaction histories and is therefore partially captured by the DKT-based covariates, though residual unmeasured confounding may still remain. To test this assumption, we conducted a formal sensitivity analysis [7] by calculating the Robustness Value ($RV_{q=1}$) required to fully explain away the estimated effect.[1] Our results indicate that an unmeasured confounder would need to be at least three times as predictive of both treatment assignment and the outcome as the stroungest predictors (DKT outcome prediction, average accuracy, or cumulative attempts) to nullify the current findings. This level of confounding is unlikely in practice, suggesting the results are robust to plausible omitted variables.

[1]See Appendix A.2

6.2 Treatment Effect Heterogeneity

While the average effects are positive, we observe substantial heterogeneity in local treatment effects across intervention sessions. The CATE for immediate performance has a standard deviation of $3.53pp$, and ranges from $-20.58pp$ to $19.95pp$. The CATE for near transfer has a standard deviation of $3.61pp$, and ranges from $-23.25pp$ to $18.24pp$. Notably, there is a moderate correlation between these effects ($r = 0.46$, $p < 0.001$), suggesting that the immediate performance CATE may be a good, however imperfect, surrogate measure for near transfer; if a tutoring session has a positive effect on next problem correctness, it is likely to have a positive effect on the students' ability to solve problems on other similar skills as well.

6.2.1 *Heterogeneity by Tutoring Session Characteristics*

To illustrate how this method can be used to examine why some tutoring sessions are more effective than others, we analyzed three potential moderators: session length, proportion of student talk, and the number of prior tutoring sessions. We chose these moderators because each of them has been studied in more traditional tutoring environments. Longer and more frequent sessions are consistently pointed to as hallmarks of high-impact tutoring [33]. Eliciting student talk is often associated with higher quality tutoring sessions [5]. Notably, we view this as a preliminary analysis into understanding what might drive effectiveness in on-demand tutoring, which will lead to future directions for this line of work.

Because these variables were non-normally distributed, each was divided into quartiles and compared using the heterogeneity model (Section 5.3.4) with Tukey HSD post-hoc tests.[2] Figure 2 presents the differences in effect size on immediate performance by quartile (quartile descriptive statistics and results tables are presented in Appendix C).

[2]We also ran the models with transformed variables and found similar results, but chose to present the results in terms of quartiles to ease interpretation.

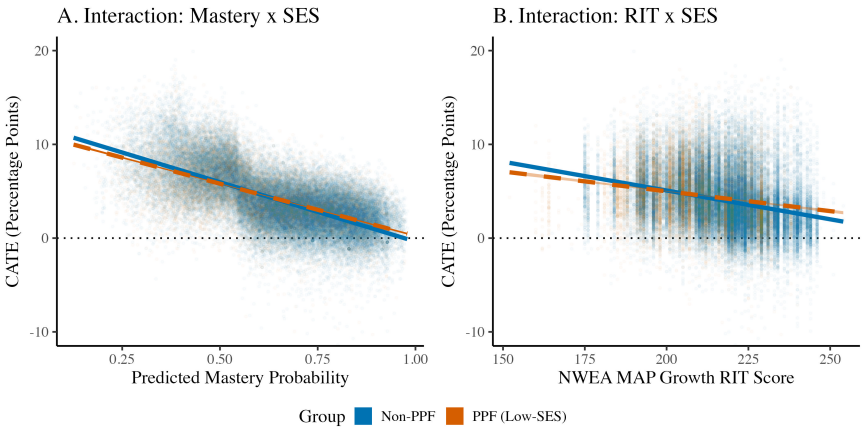


Figure 3: Interactions between prior knowledge in terms of predicted mastery (A) or standardized test scores (B) and socio-economic status (SES) based on Pupil Premium Funding (PPF) eligibility.

Session length was measured by the number of messages sent between the tutors and students that showed the clearest and most consistent pattern. Moderately long sessions (Q3) produced significantly larger effects than shorter sessions (Q1 and Q2; $p < 0.001$), indicating that brief interactions are often insufficient to support learning. Sessions in the longest quartile (Q4) also outperformed the shortest sessions (Q1, $p < 0.05$), but did not yield reliably greater benefits than moderately long sessions, suggesting diminishing returns beyond a certain point.

As a proxy for student talk, we used the percentage of total words in the session that were sent by the student. This measure of the proportion of student talk exhibited minimal and inconsistent associations with tutoring effectiveness. Only one comparison reached significance, indicating slightly lower effects in sessions with the highest levels of student talk (Q4 vs. Q2; $p < 0.05$), with all other contrasts non-significant. Students who sent between 14% and 20% of the words during the session had higher estimated effects than those who sent between 29.8% and 85.7%. This contrasts with much of the work on student talk in tutoring, which often finds that getting students to engage is an important part of tutoring [5]. However, this analysis suggests that student talk may be less important in on-demand sessions that focus on specific problems.

The number of tutoring sessions that preceded the intervention session was strongly and positively associated with impact. Almost all comparisons show that students with more previous sessions experienced substantially larger gains than first-time or infrequent users, with effect sizes increasing monotonically across quartiles. Taken together, these results suggest that tutoring is most effective when sessions are of at least moderate length and when students have prior experience engaging with on-demand support, whereas simply increasing the amount of student talk does not reliably translate into greater effects on proximal performance.

6.2.2 Heterogeneity by Student Characteristics

The CATE estimations also provide opportunities to evaluate which students benefit more from this form of tutoring. To assess this variability, we estimated a series of regression models predicting session-level CATEs (the full table of the model output can be found in Appendix B). Students with lower baseline ability exhibited larger treatment effects, with DKT mastery showing the strongest association ($\beta = -1.87$, $p < 0.001$), indicating that a one standard deviation decrease in predicted mastery corresponded to a 1.87 percentage point increase in effect size. Once DKT was included, NWEA RIT scores showed only a small positive association ($\beta = 0.08$), suggesting that the more proximal DKT-based measure of knowledge was the primary driver of heterogeneity. This pattern is consistent with both a ceiling effect—where higher-performing students have less room to improve—and the hypothesis that just-in-time tutoring is most valuable for students who are struggling.

Overall, low-SES students benefited from tutoring, but to a slightly lesser extent than their peers ($\beta = -0.21$, $p < 0.05$) and interaction terms revealed that the relationship between prior knowledge and treatment effects was weaker for these students (DKT $\times$ PPF: $\beta = 0.14$, $p < 0.001$; RIT $\times$ PPF: $\beta = 0.33$, $p < 0.001$). Figure 3 presents a visualization of this interaction. While lower-performing students generally benefited more from tutoring, this gradient was attenuated among low-SES students. However, as the figure shows, tutoring effects remain strongly related to prior knowledge: on average, lower-performing, low-SES students still appear to benefit more from tutoring than higher-performing, higher-SES students.

7. DISCUSSION & CONCLUSION

This paper introduces a scalable causal framework for estimating problem-level effects of on-demand human tutoring embedded in an adaptive learning system. Across specifications, requesting tutoring produces modest but reliable gains in immediate performance and near transfer.

One central contribution of this work is methodological. We show that time-varying knowledge representations derived from platform logs can be used as pretreatment covariates to address confounding driven by evolving mastery and task difficulty in observational studies of an opt-in feature. By integrating Deep Knowledge Tracing features [29] into a dou-

bly robust causal forest estimator [41, 3, 12], the framework targets the core selection problem that tutoring is typically requested when students anticipate poor performance. The stability of ATE estimates after adding external covariates (test scores, SES proxy, school identifiers) suggests that platform-derived learning traces capture much of the relevant confounding structure. However, the marginally non-significant negative placebo effect may suggest some bias in the approach, but, in this application, that bias is likely an underestimation of the effect.

A key substantive result is that the ATE alone masks wide heterogeneity, including strongly positive and negative session-level effects. Although the average tutoring effect is comparable to lower-cost ALS features such as on-demand hints [13, 32, 31], individual effects range from roughly +20 to −20 percentage points. This indicates that some tutoring sessions are highly effective while others are neutral or counterproductive.

Our analysis of this heterogeneity provides a preliminary, though not comprehensive, attempt to identify the factors associated with higher-impact tutoring. Evaluating session characteristics highlights several potential drivers of effectiveness. Session length exhibits the clearest pattern: moderately long sessions (outperform very short sessions, while the longest sessions do not reliably improve upon moderately long ones. This suggests a diminishing marginal return on tutor time beyond a certain threshold. It is also possible that longer conversations indicated the need for additional remediation beyond what can be provided through a chat interface.

Furthermore, prior intervention experience is strongly and monotonically associated with larger impacts, indicating that students may 'learn how to use' tutoring more effectively over time. Alternatively, students who access tutoring more frequently may possess inherent characteristics, such as higher baseline engagement, that render the tutoring more beneficial. In contrast, the proportion of student talk shows minimal associations with impact. This finding is directionally inconsistent with classic tutoring accounts that emphasize student explanation and constructive engagement as drivers of learning [5], and suggests that, in brief, problem-focused, on-demand chats, the quantity of student talk may be less informative than the specificity, timing, and instructional content of tutor moves. However, the crude metric of the proportion of words used by students vs. teachers does not capture the quality of the conversation; more work can be conducted in this area using the effect estimates from our framework. Together, these patterns point to a practical implication for embedded systems: effectiveness may depend less on maximizing interaction volume and more on ensuring sessions are long enough to resolve a misconception while supporting students in developing productive help-seeking routines.

Student-level heterogeneity shows larger benefits for learners with lower prior mastery, consistent with just-in-time support models and prior tutoring evidence [23]. However, low-SES students benefit slightly less on average and show a weaker mastery–impact gradient, aligning with prior work suggesting that on-demand tutoring may require complementary design supports to ensure equitable benefit [21, 9].

Finally, the robustness checks show both the value and the limits of using observational data to evaluate embedded tutoring interventions. When we used a broader control group, the estimated effects became smaller, but remained positive. This pattern suggests that the results are not driven by one specific control-group definition. At the same time, it shows how difficult it is to identify the right comparison group when tutoring happens at different moments over time and when earlier tutoring may affect later performance. These challenges underscore the importance of methods that account for students' changing knowledge states and include diagnostic checks, such as overlap and placebo tests, to assess possible remaining bias [35, 12].

8. LIMITATIONS & FUTURE WORK

Several limitations suggest directions for future research. First, although our identification strategy relies on conditional ignorability, sensitivity analyses indicate that unmeasured factors such as student motivation or teacher practices would need to be substantially stronger than our platform-derived measures to overturn the findings. Second, treating tutoring as a binary exposure masks variation in instructional quality and conversational strategy. Future work should use process-level data, such as transcripts or instructional moves, to identify the mechanisms driving effectiveness. Third, our proximal outcomes capture immediate impact and near transfer but not long-term learning gains. Future studies should examine whether session-level effects accumulate over time. Finally, because this study focuses on a single platform, future research should test the framework across domains, platforms, and designs that more directly examine equity and pedagogical mechanisms.

9. ACKNOWLEDGMENTS

This research was conducted as part of the National Tutoring Observatory, which was funded by the Bill Gates Foundation (Grant No. INV-068961) and the Chan Zuckerberg Initiative (Grant No. 2024-351541). The views expressed here do not necessarily reflect the official positions or policies of the funding agencies.

10. REFERENCES

[1] G. Abdelrahman, Q. Wang, and B. Nunes. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11):224:1–224:37, 2023.

[2] V. Aleven, I. Roll, B. M. McLaren, and K. R. Koedinger. Help helps, but only so much: Research on help seeking with intelligent tutoring systems. *International Journal of Artificial Intelligence in Education*, 26(1):205–223, 2016.

[3] S. Athey, J. Tibshirani, and S. Wager. Generalized random forests. *The Annals of Statistics*, 47(2):1148–1178, 2019.

[4] S. Athey and S. Wager. Estimating treatment effects with causal forests: An application. *Observational studies*, 5(2):37–51, 2019.

[5] M. T. Chi, S. A. Siler, H. Jeong, T. Yamauchi, and R. G. Hausmann. Learning from human tutoring. *Cognitive science*, 25(4):471–533, 2001.

[6] D. R. Chine, C. Brentley, C. Thomas-Browne, J. E. Richey, A. Gul, P. F. Carvalho, and K. R. Koedinger. Educational equity through combined human-ai personalization: A propensity matching evaluation. In *International Conference on Artificial Intelligence in Education*, pages 366–377. Springer, 2022.

[7] C. Cinelli and C. Hazlett. Making sense of sensitivity: Extending omitted variable bias. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(1):39–67, 2020.

[8] R. K. Crump, J. V. Hotz, G. W. Imbens, and O. A. Mitnik. Dealing with limited overlap in estimation of average treatment effects. *Biometrika*, 96(1):187–199, 2009.

[9] G. Deacon and G. Chojnacki. Impacts of upchieve on-demand tutoring on students' math knowledge and perceptions. Technical report, Mathematica, 2023.

[10] J. Dietrichson, M. Bøg, T. Filges, and A. M. Klint Jørgensen. Academic interventions for elementary and middle school students with low socioeconomic status: A systematic review and meta-analysis. *Review of Educational Research*, 87(2):243–282, 2017.

[11] A. C. Eggers, G. Tuñón, and A. Dafoe. Placebo tests for causal inference. *American Journal of Political Science*, 68(3):1106–1121, 2024.

[12] A. N. Glynn and K. M. Quinn. An introduction to the augmented inverse propensity weighted estimator. *Political Analysis*, 18:36–56, 2010.

[13] A. Gurung, S. Baral, M. P. Lee, A. C. Sales, A. Haim, K. P. Vanacore, A. A. McReynolds, H. Kreisberg, C. Heffernan, and N. T. Heffernan. How common are common wrong answers? crowdsourcing remediation at scale. In *Proceedings of the Tenth ACM Conference on Learning@ Scale*, pages 70–80, 2023.

[14] A. Gurung, J. Lin, J. Gutterman, D. R. Thomas, A. Houk, S. Gupta, and K. Koedinger. Human tutoring improves the impact of ai tutor use on learning outcomes. In *International Conference on Artificial Intelligence in Education*, pages 393–407. Springer, 2025.

[15] J. Guryan, J. Ludwig, M. P. Bhatt, P. J. Cook, J. M. Davis, K. Dodge, and G. Stoddard. Not too late: Improving academic outcomes among adolescents. *American Economic Review*, 113(3):738–765, 2023.

[16] P. R. Hahn, J. S. Murray, and C. M. Carvalho. Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion). *Bayesian Analysis*, 15(3):965–1056, 2020.

[17] W. Harrison, E. Dobson, S. Higgins, G. Uwimpuhwe, and R. Khowaja. A pilot impact study to evaluate the effectiveness of eedi on raising attainment in mathematics at ks3. Technical report, What Worked Education, 2023.

[18] R. F. Kizilcec, G. M. Davis, and G. L. Cohen. Towards equal opportunities in moocs: affirmation reduces gender & social-class achievement gaps in china. In *Proceedings of the fourth (2017) ACM conference on learning@ scale*, pages 121–130, 2017.

[19] R. F. Kizilcec, A. Saltarelli, P. Bonfert-Taylor, M. Goudzwaard, E. Hamonic, and R. Sharrock. Welcome to the course: Early social cues influence women's persistence in computer science. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–13, 2020.

[20] M. A. Kraft, D. S. Edwards, and M. Cannata. The scaling dynamics and causal effects of a district-operated tutoring program. edworkingpaper no. 24-1030. *Annenberg Institute for School Reform at Brown University*, 2024.

[21] M. A. Kraft, B. E. Schueler, and G. Falken. What impacts should we expect from tutoring at scale? exploring meta-analytic generalizability, 2024.

[22] LearnLM Team, Google and Eedi, A. Wang, A. Rysbek, A. Huber, A. Nambiar, A. Kenolty, B. Caulfield, B. Lilley-Draper, B. Groot, B. Veprek, C. Burdett, C. Willis, C. Barton, D. Smith, G. Mu, H. Walters, I. Jurenka, I. Hulls, J. Stalley-Moores, J. Caton, J. Wilkowski, K. Alarakyia, K. R. McKee, L. McCafferty, L. Dalton, M. Kunesch, P. Malubay, R. Kidson, R. Wells, S. Wheeler, S. Wiltberger, S. Mohamed, S. Woodhead, and V. Brazão. Ai tutoring can safely and effectively support students: An exploratory rct in uk classrooms, 2025. arXiv preprint.

[23] A. Nickow, P. Oreopoulos, and V. Quan. The promise of tutoring for prek–12 learning: A systematic review and meta-analysis of the experimental evidence. *American Educational Research Journal*, 61(1):74–107, 2024.

[24] Z. A. Pardos and N. T. Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *International conference on user modeling, adaptation, and personalization*, pages 255–266. Springer, 2010.

[25] Y. Pei, A. Sales, and J. Gagnon-Bartsch. Boosting precision in educational a/b tests using auxiliary information and design-based estimators. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 990–993, 2024.

[26] Y. Pei, A. Sales, H.-A. Kang, and T. Whittaker. Fully latent principal stratification with misspecified measurement models in intelligent tutoring systems. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 182–191, 2025.

[27] R. Pelánek. Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User modeling and user-adapted interaction*, 27(3):313–350, 2017.

[28] D. M. Pham, K. P. Vanacore, A. C. Sales, and J. A. Gagnon-Bartsch. Lool: Towards personalization with flexible & robust estimation of heterogeneous treatment effects. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 376–384. International Educational Data Mining Society, july 2024.

[29] C. Piech et al. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, pages 505–513, 2015.

[30] E. Prihar, A. Moore, and N. Heffernan. Identifying explanations within student-tutor chat logs. In *Proceedings of the 15th International Conference on Educational Data Mining*, page 773–777, 2022.

[31] E. Prihar, T. Patikorn, A. Botelho, A. Sales, and N. Heffernan. Toward personalizing students' education with crowdsourced tutoring. In *Proceedings of the Eighth ACM Conference on Learning@ Scale*, page 37–45. Association for Computing Machinery, Inc, june 2021.

[32] E. Prihar, M. Syed, K. Ostrow, S. Shaw, A. Sales, and N. Heffernan. Exploring common trends in online educational experiments. In *Proceedings of the 15th International Conference on Educational Data Mining*, 2022.

[33] C. D. Robinson and S. Loeb. High-impact tutoring: State of the research and priorities for future learning. *National Student Support Accelerator*, 21(284):1–53, 2021.

[34] C. D. Robinson, C. Pollard, S. Novicoff, S. White, and S. Loeb. The effects of virtual tutoring on young readers: Results from a randomized controlled trial. *Educational Evaluation and Policy Analysis*, 47(4):1245–1265, 2025.

[35] D. B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, 66:688–701, 1974.

[36] A. C. Sales, B. B. Hansen, and B. Rowan. Rebar: Reinforcing a matching estimator with predictions from high-dimensional covariates. *Journal of Educational and Behavioral Statistics*, 43(1):3–31, 2018.

[37] A. C. Sales and J. F. Pane. The effect of teachers reassigning students to new cognitive tutor sections. *International Educational Data Mining Society*, 2020.

[38] D. R. Thomas, J. Lin, E. Gatz, A. Gurung, S. Gupta, K. Norberg, S. E. Fancsali, V. Aleven, L. Branstetter, E. Brunskill, and K. R. Koedinger. Improving student learning with hybrid human-ai tutoring: A three-study quasi-experimental investigation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 404–415, 2024.

[39] K. Vanacore, A. Sales, A. Liu, and E. Ottmar. Benefit of gamification for persistent learners: Propensity to replay problems moderates algebra-game effectiveness. In *Tenth ACM Conference on Learning @ Scale (L@S '23)*, Copenhagen, Denmark, 2023. ACM.

[40] K. P. Vanacore, A. Gurung, A. Sales, N. Heffernan, et al. Effect of gamification on gamers: Evaluating interventions for students who game the system. *Journal of educational data mining*, 16(1):112–140, 2024.

[41] S. Wager and S. Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018.

[42] R. E. Wang, A. T. Ribeiro, C. D. Robinson, S. Loeb, and D. Demszky. Tutor copilot: A human-ai approach for scaling real-time expertise. *arXiv preprint arXiv:2410.03017*, 2024.

[43] E. Wu and J. A. Gagnon-Bartsch. The loop estimator: Adjusting for covariates in randomized experiments. *Evaluation review*, 42(4):458–488, 2018.

APPENDIX

Digital Appendix

A New Domain-Informed Learner Model with Uncertainty-Aware Knowledge Mastery Propagation

Rania Ait Chabane
Université de Lorraine
LORIA
Nancy, France
rania.ait-chabane@loria.fr
ORCID:
0009-0002-7630-1195

Armelle Brun
Université de Lorraine
LORIA
Nancy, France
armelle.brun@loria.fr
ORCID:
0000-0002-9876-6906

Azim Roussanaly
Université de Lorraine
LORIA
Nancy, France
azim.roussanaly@loria.fr
ORCID:
0000-0002-3311-3613

ABSTRACT

Accurately tracing a learner's knowledge state is a central challenge in many applications as it directly impacts educational downstream tasks such as recommendation, dropout prediction, and next-question correctness prediction. Prior research in learner modeling and knowledge tracing has shown that incorporating domain structure leads to more accurate estimates of learner knowledge, while parallel work has demonstrated that explicitly modeling epistemic uncertainty improves performance on these tasks. However, these two research directions have largely been explored independently. In particular, existing approaches rarely propagate uncertainty beyond the level at which learner evidence is directly observed, and domain-informed propagation strategies typically ignore uncertainty altogether. In this paper, we propose KT-DUSL, an uncertainty-aware learner model that jointly represents knowledge mastery, epistemic uncertainty, and domain structure within a unified framework. Building on subjective logic, we introduce a novel propagation policy that diffuses both mastery and uncertainty across the domain structure. This policy enables the model to better handle sparse interaction data. We evaluate KT-DUSL on large-scale datasets and show that propagating uncertainty through the domain structure yields improvements in next-question correctness prediction over uncertainty-agnostic and non-propagation-based baselines.

Keywords

knowledge tracing, learner modeling, uncertainty, mastery propagation, personal knowledge graphs

1. INTRODUCTION

A learner model aims at tracing learners' knowledge and can be understood as an explicit, machine-interpretable representation of an individual learner within an online learning environment [6]. The content of a learner model consists of a structured combination of learner data and inferred characteristics. Learner data includes relatively static attributes—such as demographic and educational background—as well as dynamic data derived from interactions with learning systems. Among the various learner characteristics that can be inferred from such data, the learner's knowledge state is the most commonly modelled, typically referred to as a knowledge model [30]. Learner models can be used as an internal component of an Intelligent Tutoring System (ITS) to trace the learners' knowledge, and to support downstream tasks such as recommendation, adaptive learning, and personalised feedback [32], or as a resource that can be exposed to the learner, for example through open learner models [9].

Learner models have recently come to include information about the domain as well, referred to as a domain model [30]. Concretely, the learning domain model includes learning items (exercises), knowledge components (KCs), and the relations between them [30]. While most early learner models assumed KCs to be independent, more recent work has increasingly explored structured representations of KCs and their relations, e.g. hierarchical or prerequisite relations [26]. This evolution reflects the intrinsic nature of educational domains, in which KCs are rarely isolated but are instead meaningfully connected. Incorporating domain structure naturally raises the question of propagating knowledge estimates within this structure. Since KCs are connected, evidence observed on one KC may carry implications for related KCs [36]. Existing approaches differ in how they address this propagation problem, ranging from expert-defined propagation rules to probabilistic propagation policies and, more recently, to neural network–based latent propagation [36, 20, 23]. Overall, taking domain structure into account has significantly improved the quality of knowledge models and, consequently, learner models [11].

In educational settings, learner interaction data are often sparse and unevenly distributed across KCs, leading to increased uncertainty—specifically, epistemic uncertainty—in knowledge estimates [13]. Throughout this paper, the term *uncertainty* is used to refer to epistemic uncertainty, i.e., uncertainty arising from limited evidence about a learner's mastery of a KC. It reflects how strongly a mastery estimate is supported by observations, and is reducible as more learner–KC interactions are collected. Prior work suggests that

Rania Ait Chabane, Armelle Brun, and Azim Roussanaly. A New Domain-Informed Learner Model with Uncertainty-Aware Knowledge Mastery Propagation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 18–28. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040060

quantifying uncertainty improves next-question correctness prediction by mitigating overconfident predictions, particularly when evidence is sparse [13, 25, 3]. In learner models, this uncertainty is rarely considered. Yet, some approaches quantify uncertainty when predicting the correctness of the next learner interaction [14, 25]. However, these approaches typically provide uncertainty estimates on unstructured sets of KCs.

In parallel, research on *open learner models* emphasises the importance of transparency and explainability in learner modeling by making aspects of the learner model explicit and consultable by the learner [9, 8]. Providing uncertainty estimates alongside mastery estimates further supports the idea of opening the learner model to the learner, as it enables her to better interpret, reflect on, and trust the information presented in the learner model.

In this context, our work aims to build a unified learner model that incorporates not only the domain structure, but also the learner's knowledge and associated uncertainty throughout this structure. We hypothesize that estimating uncertainty throughout the domain structure can improve the quality of the learner model. We raise the following research question:
RQ: Does incorporating the domain structure impact the learner model?
To answer this question, we raise two sub-research questions:
S-RQ1: How to propagate uncertainty alongside knowledge mastery within the domain structure?
S-RQ2: Under which settings does the domain structure provide the greatest benefits?

The rest of the paper is organized as follows. Section 2 reviews related work, with a particular focus on learner and domain modeling, the use of knowledge graphs in education, and mastery propagation within graph-structured representations. Section 3 presents KT-DUSL, a learner model that uses subjective logic to model knowledge mastery and uncertainty in a hierarchical domain structure. Section 4 describes the experimental setup, and Section 5 discusses the results. The code is publicly available at `https://github.com/anonymousreseach/KT-DUSL.git`.

2. RELATED WORK

This section reviews prior work related to our contribution. We first discuss learner and domain modeling approaches in ITSs. We then review the use of knowledge graphs in education. Finally, we examine existing approaches to knowledge mastery propagation and the quantification of uncertainty in learner models.

2.1 Learner and Domain Modeling

Learner modeling represents learners' abilities and behavior through interaction data [30, 19]. It supports personalization and instructional decision-making in ITSs [19].

Learner models may also capture affect, motivation, metacognition, or preferences [18, 32]. However, the dominant line of work focuses on modeling learners' mastery of knowledge concepts [16, 30]. This problem is commonly framed as *knowledge tracing*, which tracks latent knowledge over time to predict future performance [16, 31, 30].

Learner modeling is related to, but distinct from, *domain modeling* [18, 30]. Learner models encode the learner's current and evolving knowledge with respect to a domain, while domain models specify the content structure (items, knowledge components and relations) [27]. Domain models are often treated as relatively stable, but they can be refined from data when the initial structure is imperfect [10, 29]. In contrast, learner models are inherently dynamic, as each interaction updates knowledge mastery estimates and can incorporate learning and forgetting processes [16, 30].

Prior work varies in how domain and learner models are represented and coupled [18, 30]. Many systems keep the domain structure fixed (e.g., a predefined KCs mapping) and infer learner mastery on top of it [16, 15]. Other approaches learn or exploit dependencies between KCs to propagate learner-specific evidence or mastery estimates across related KCs [10, 26].

Beyond predictive accuracy, learner models also play an important role in explainability. In this context, research on *open learner models* emphasizes the importance of representations that make learner knowledge states explainable to both learners and instructors [9, 8]. Building on this principle, we adopt a learner model that explicitly represents both knowledge mastery and epistemic uncertainty within the domain structure. Such structured representations can, in principle, be exposed to learners to support reflection, self-regulated learning, and trust [9, 21]. In practice, KT-DUSL supports this by maintaining explicit, interpretable KC-level estimates of both mastery and uncertainty within the learner's PKG, which can be visualized (e.g., as a concept-level dashboard) or queried to provide feedback on uncertain areas. This allows learners and instructors to inspect not only what is known, but also where knowledge estimates remain uncertain, thereby supporting informed decision-making and targeted learning.

2.2 Knowledge Graphs

Knowledge graphs (KGs) are a widely used formalism for representing structured knowledge, where concepts are modeled as nodes and semantic relations between them as edges [1]. In educational systems, KGs have been extensively used to represent domain knowledge—often referred to as domain knowledge graphs, i.e., graph-based representations of domain models—in the form of concept maps, prerequisite graphs, or hierarchical concept structures [1]. Such a representation makes explicit the relationships between knowledge components and can be used to support curriculum design, content sequencing, and learner assessment [18].

In most educational applications, domain knowledge graphs are shared across learners and capture expert-defined domain structure [17]. Learner modeling is then performed on top of this structure by associating learner-specific information -such as knowledge mastery estimates- with individual knowledge components. While effective, this separation typically treats the domain structure as static and limits the extent to which learner knowledge can be jointly reasoned about and updated across the graph, as learner evidence is incorporated locally rather than being propagated through the domain structure [34].

More recently, knowledge graphs have been extended toward learner-centered representations, giving rise to the notions of personalized and personal knowledge graphs. Personalized knowledge graphs typically adapt an existing general-purpose or domain-level KG by filtering, weighting, or annotating entities and relations according to a user's preferences or profile [5]. In contrast, personal knowledge graphs (PKGs) exist alongside other KGs and contain a disjoint set of entities that are of direct relevance to a single individual, while optionally linking to external knowledge graphs [33].

Different definitions of PKGs emphasize complementary aspects of the concept. From an ownership and access-control perspective, a PKG is a knowledge graph fully controlled by an individual, with the primary goal of supporting personalized services [33]. From a semantic perspective, PKGs represent structured knowledge about entities and relations of personal relevance, organized around a central user node [5].

In educational contexts, PKGs provide a natural foundation for learner-centered modeling, as they allow domain knowledge to be explicitly organized around the learner. Ain et al. [2] propose a PKG-based learner modeling approach in which learners construct their own PKGs by marking concepts they did not understand while interacting with learning items. More recently, Alatrash et al. [4] leverage PKGs in combination with graph neural networks to model learner knowledge mastery in a transparent manner, demonstrating the potential of graph-based representations for learner modeling.

Despite these advances, most PKG-based educational approaches attach learner state to graph nodes in a static or ad-hoc manner, typically as scalar mastery values or latent embeddings, offering limited support for principled propagation of learner knowledge and uncertainty across the graph.

2.3 Mastery Propagation through the Domain Structure

A substantial body of work goes beyond estimating mastery independently per KC and instead propagates learner evidence through a domain structure.

Early educational systems commonly relied on learner models coupled to concept networks, in which prerequisite or semantic links define hand-crafted propagation rules: evidence of mastery or failure on one KC is deterministically or heuristically propagated to related KCs. Such approaches rely on expert-defined rules and propagation strengths, resulting in transparent but largely fixed propagation behavior [7].

A second line of work embeds the domain structure directly into probabilistic graphical models, where KC mastery variables are treated as latent states and evidence is propagated through structured dependencies. Under tree-structured hierarchies, Gao et al. [20] propose KT^2, which models concept mastery using a Hidden Markov Tree to propagate evidence along parent-child relations. Beyond trees, Bayesian networks and dynamic Bayesian networks represent more general KC topologies, allowing updates on one KC to influence related prerequisites or successors [22]. Related formulations further introduce learner-specific factors or jointly infer prerequisite structure from interaction data [28, 12].

More broadly, probabilistic approaches to learner modeling have long leveraged Bayesian networks to represent dependencies between KCs and to reason under uncertainty [15]. Millán et al. [24] provide a general framework for engineering Bayesian student models, illustrating how conditional dependencies between concepts enable evidence propagation and learner knowledge mastery updates.

Finally, some deep knowledge tracing models incorporate domain structure using graph neural networks and perform inference through message passing on KC graphs or question-KC bipartite graphs [26, 34]. GIKT further leverages graph convolution to exploit higher-order question-KC relations and mitigate data sparsity [35]. While effective in terms of predictive performance, these approaches operate on latent embeddings and generally lack explicit, interpretable KC-level mastery estimates.

Uncertainty Modeling and Propagation. Although many learner models rely on probabilistic formulations, uncertainty is typically treated implicitly rather than explicitly represented in the learner model. As a result, mastery estimates are often reduced to point values, conflating low mastery due to consistent failure with low mastery arising from limited evidence. Moreover, most propagation policies focus on transferring mastery alone, without modeling how epistemic uncertainty should evolve across structured representations. These limitations are particularly problematic in sparse settings, motivating learner models that jointly represent and propagate both knowledge mastery and uncertainty.

3. INTEGRATING THE DOMAIN STRUCTURE IN A LEARNER MODEL

In this work, we introduce KT-DUSL, a new learner model that explicitly leverages the domain structure, and relies on a dedicated propagation policy to quantify both knowledge mastery and uncertainty throughout this structure. Before presenting the KT-DUSL model, we first recall propagation policies commonly used in the literature.

3.1 Propagation in a Domain Structure

Learner models that manage a domain structure rely on propagation policies. Most works in the literature aim at propagating knowledge mastery throughout the knowledge components (KCs) of the domain. Two general principles govern such a propagation: (i) information should be attenuated as it moves away from its source, and (ii) information originating from multiple sources should be combined coherently. These principles are commonly implemented through a discount operator and a consensus operator. We introduce these operators in a general setting. They will then be instantiated in KT-DUSL.

Discount Operator. The discount operator models the attenuation of information during propagation across the domain structure. A discount factor $r_{i,j} \in [0, 1]$ controls the strength of propagation from a KC c_i to another KC c_j.

Consensus Operator. When a KC receives information from multiple related KCs, these sources of information must be aggregated in a principled manner. The consensus operator

defines how multiple propagated pieces of information are combined.

3.2 The KT-DUSL Learner Model

KT-DUSL is a learner model that is designed to manage simultaneously the domain structure and the learner's mastery and associated uncertainty throughout this structure. KT-DUSL relies on two core elements: (i) a Personal Knowledge Graph (PKG), instantiated for each learner, (ii) subjective logic to estimate knowledge mastery and associated uncertainty. The originality of KT-DUSL lies in the propagation policy used to propagate these quantities throughout the domain structure. KT-DUSL also supports explainability by maintaining explicit KC-level knowledge mastery and uncertainty estimates within the PKG, whose updates can be made visible to the learner.

From now on, the general term exercise will be used to refer to a learning item.

3.2.1 Personal Knowledge Graph Structure

KT-DUSL relies on a dedicated *Personal Knowledge Graph* (PKG), constructed for each learner, that jointly represents (i) the domain structure, made up of both a hierarchy of *knowledge components* (KCs) where higher-level KCs correspond to broader concepts and leaf-level KCs capture more fine-grained concepts, and the set of exercises linked to leaf-level KCs, (ii) learner–exercise interactions, and (iii) learner-specific estimates of knowledge mastery and uncertainty on the KCs of the domain structure.

Let $PKG_l = (V, E)$ denote the personal knowledge graph of learner l. The set of nodes is defined as $V = C \cup X \cup \{l\}$, where C is the set of knowledge components, X is the set of exercises, and l denotes the node that represents the learner. E represents the set of edges. Each $(c_p, c_c) \in E$ encodes hierarchical membership relations between KCs. Each exercise $x \in X$ is associated with a set of KCs $C(x) \subseteq C$. The learner node is connected to each exercise node through the observed interaction outcome, and to each KC through the learner's current knowledge mastery and uncertainty estimates for that KC.

3.2.2 Subjective Logic to Estimate Mastery and Uncertainty of KCs

In the recent literature, the KT-USL learner model [3], has proposed to use subjective logic to trace learners' knowledge mastery and uncertainty on KCs.

At time t, the knowledge state of learner l on $c_i \in C$, is represented by $\omega_t(l, c_i)$, a subjective logic (SL) opinion:

$$\omega_t(l, c_i) = \big(b_t(l, c_i),\ d_t(l, c_i),\ u_t(l, c_i),\ a\big), \tag{1}$$

where $b_t(l, c_i)$, $d_t(l, c_i)$, and $u_t(l, c_i)$ denote belief in mastery, disbelief, and epistemic uncertainty, respectively, with $b_t(l, c_i) + d_t(l, c_i) + u_t(l, c_i) = 1$ and a is a base-rate prior.

Opinions are grounded in accumulated interaction evidence $(r_t(l, c_i), s_t(l, c_i))$, corresponding to the number of correct and incorrect responses observed for learner l on c_i. Following subjective logic, belief, disbelief, and uncertainty are computed as:

$$b = \frac{r}{r + s + W}, \qquad d = \frac{s}{r + s + W}, \qquad u = \frac{W}{r + s + W},$$

where $W > 0$ is a prior weight that controls the contribution of epistemic uncertainty under sparse evidence. The knowledge mastery estimate $m_t(l, c_i)$ is defined as the expected value of the opinion and ranges between 0 and 1.

$$m_t(l, c_i) = b_t(l, c_i) + a \cdot u_t(l, c_i), \tag{2}$$

This formulation explicitly distinguishes between low mastery supported by substantial evidence and low mastery arising from insufficient information, while remaining local to individual (learner, concept) pairs.

It is important to note that this formulation captures epistemic uncertainty arising from limited learner–KC evidence. Other sources of uncertainty may also affect learner modeling, including noisy or incomplete exercise–KC mappings, rarely accessed concepts, or inaccuracies in the domain structure. These sources are not explicitly modeled in KT-DUSL, which assumes a fixed and reliable domain structure. Therefore, the uncertainty quantified in this work should be interpreted as evidence-based epistemic uncertainty rather than a comprehensive account of all uncertainty sources.

3.2.3 Mastery and Uncertainty Propagation in KT-DUSL

KT-DUSL extends the KT-USL model, by explicitly leveraging the domain structure. KT-DUSL relies on a propagation policy for both mastery and uncertainty across KCs of the structure. More precisely, KT-DUSL propagates information between hierarchically related KCs. This propagation allows the learner model to compensate for sparse or unevenly distributed evidence.

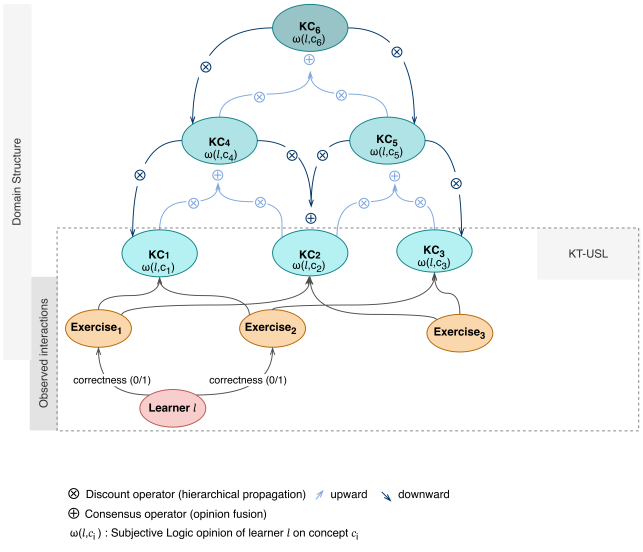


Figure 1: Overview of an example PKG and the KT-DUSL updating process.

KT-DUSL relies on PKG_l, the Personal Knowledge Graph of a learner l, where the nodes of PKG_l represent the KCs of the domain, the exercises and the learner l. Figure 1

represents an example of PKG_l, where blue nodes represent the KCs of the domain, orange nodes represent the exercises and the pink node represents the learner. The edges of the PKG encode three types of relations: hierarchical relations between KCs, relations between exercises and leaf-level KCs and relations between the learner and the exercises.

KT-DUSL operates in an online setting. First, at time t, a learner-exercise interaction generates new evidence that updates learner-exercise links (lower part of Figure 1).
Second, KT-DUSL relies on the KT-USL model [3] to update the subjective logic opinion of leaf-level KCs c_i, denoted $\omega_t(l, c_i)$, from the previous opinion $\omega_{t-1}(l, c_i)$. Recall that from this opinion, the learner's mastery and uncertainty on these leaf-level KCs can be inferred (see Equation 2). When a KC is associated with multiple exercises (in the case of batch processing of interactions), evidence is extracted from all associated exercises. The opinion on this KC is then calculated using this evidence.
Third, a specific propagation policy is performed across the hierarchy (relations between blue nodes in Figure 1). The main contribution of KT-DUSL lies in this propagation policy. It relies on the discount and consensus operators presented in Section 3.1. KT-DUSL instantiates these operators using subjective logic.

In KT-DUSL we propose to distinguish between two propagation directions: *upward* propagation, i.e., from a KC to hierarchically higher KCs, and *downward* propagation, i.e., from a KC to hierarchically lower KCs. Upward propagation is performed first, then downward propagation is performed. These two directions are governed by distinct discount rates, $r^{\uparrow}$ and $r^{\downarrow}$. In KT-DUSL an opinion on c_i can be propagated only from c_i to a directly related KC c_j. This opinion is first *discounted* as follows:

$$\omega_t(l, c_i \to c_j) = \omega_{t-1}(l, c_i) \otimes r_{i,j}. \tag{3}$$

where $r_{i,j}$ denotes the discount rate associated with the hierarchical relation from c_i to c_j ($r_{i,j}$ takes one of the two values among $r^{\uparrow}$ and $r^{\downarrow}$), and controls how much belief and disbelief are transferred, with the remaining mass assigned to epistemic uncertainty.

Considering recipient KCs, a KC c_i may receive multiple discounted opinions. In KT-DUSL, these opinions come from its related KCs in the hierarchy. Let $\Omega_t(l, c_i)$ denote the set of all opinions propagated to c_i at time t. Second, the opinion is updated using the subjective logic *consensus* operator:

$$\omega_t^{\text{prop}}(l, c_i) = \bigoplus_{s \in \Omega_t(l, c_i)} s. \tag{4}$$

The choice of this consensus operator comes from its ability to weight the contribution of different opinions according to their uncertainty. It ensures that opinions supported by stronger evidence exert greater influence on c_i.

To summarize, after each learner-exercise interaction, KT-DUSL performs: (i) a local update of the opinions of the leaf-level KCs associated with the exercise of the newly observed evidence using KT-USL; and (ii) a propagation step that diffuses mastery and uncertainty throughout the domain hierarchy (up and down). Propagation relies on both discount and consensus operators. As a result, the learner's PKG evolves over time to reflect both accumulated evidence and structural relations between concepts.

Algorithm: KT-DUSL update
Input: Learner-exercise interaction (l, x), PKG_l
Output: Updated opinions $\{\omega(l, c)\}$ in the learner's PKG

Identify the set $C(x)$ of leaf-level KCs of exercise x
Extract evidence from the interaction (l, x)

// leaf-level KC mastery and uncertainty evaluation
for each $c \in C(x)$ **do**
  Update the observed opinion ω_c^{obs} using KT-USL
end for

// discount, upward propagation
for each $c \in C(x)$ **do**
  Propagate $\omega_{c_i}^{\text{obs}}$ upward to higher-level KCs using $r^{\uparrow}$
  Update $\omega_t(l, c)$ with opinion from $t-1$ via consensus
end for

// discount, downward propagation
for each KC updated by upward propagation **do**
  Propagate opinion downward to lower-level KCs using $r^{\downarrow}$
  Update $\omega_t(l, c)$ via consensus
end for

4. EXPERIMENTAL SETUP

4.1 Datasets

To evaluate KT-DUSL, we perform experiments on two datasets from the literature. Both datasets are related to mathematics.

Junyi. The Junyi dataset is derived from an online educational platform that provides large-scale logs of student interactions with mathematical questions.[1]

EEDI (Task 1-2). EEDI is an online mathematics learning platform that provides interaction logs together with rich metadata describing questions, learners, and answers.[2]

Table 1 summarizes the main characteristics of the two datasets. A key structural difference between them has motivated their choice: in Junyi, each exercise is associated with a single KC, whereas in EEDI each exercise is associated to multiple KCs, with an average of 4.17 concepts per question. As a result, EEDI is denser than Junyi with on average about twice as many interactions per concept. We investigate if these differences affect the impact on the mastery and uncertainty, as well as the performance on downstream tasks.

4.2 Metrics

We quantify the impact of the dedicated domain-informed propagation on the learner model. Concretely, we measure

[1] `https://www.kaggle.com/datasets/junyiacademy/learning-activity-public-dataset-by-junyi-academy`
[2] `https://www.kaggle.com/datasets/alejopaullier/eedi-external-dataset`

Table 1: Characteristics of the Junyi and EEDI datasets.

Statistic	Junyi	EEDI
Interaction data		
# Learners (unique)	72,758	118,971
# Exercises	1,330	27,613
# Learner–exercise interact.	16,217,311	15,867,850
KCs		
# Number of KCs	224	388
Average # KCs per exercise	1.00	4.17
Average # interact. per KC	94.8k	200k
Hierarchy structure		
Max depth	3	3
# levels	4	4
# KCs at depth 1/2/3	10/42/171	9/69/308

the absolute change in knowledge mastery and uncertainty, induced by propagation on leaf-level KCs, defined as $|\Delta| = |x_{\text{with domain}} - x_{\text{without domain}}|$. Absolute deltas are reported for mastery (KT-A → KT-DA and KT-USL → KT-DUSL) and for uncertainty (KT-USL → KT-DUSL).

In line with the literature, we also evaluate KT-DUSL on the traditional next-question correctness prediction task, using the following standard metrics:

- **AUC** (Area Under the ROC Curve), which measures the quality of probabilistic ranking independent of a fixed decision threshold.
- **Accuracy@0.5**, computed by thresholding predicted correctness probabilities at 0.5.
- **LogLoss**, which assesses the calibration and sharpness of predicted probabilities. The lower, the better.

4.3 Baseline Models

To address the RQ raised in the introduction, we empirically evaluate KT-DUSL, and compare it against a set of baseline models. This evaluation will contribute to quantifying the effects of both uncertainty modeling and propagation policy.

KT-A is a deterministic baseline model that estimates learner mastery for leaf-level KCs by averaging observed correctness outcomes. It does not quantify uncertainty and does not exploit domain structure.

KT-DA extends **KT-A** by propagating average mastery estimates across the KC hierarchy using mean-based instantiations of the discount and consensus operators.

KT-USL (see Section 3.2.2) models learner mastery for leaf-level KCs only, and quantifies uncertainty using subjective logic. KT-USL treats each KC independently, so does not propagate information across the domain structure.

4.3.1 Parameter Initialization and Control

KT-DUSL introduces parameters governing both propagation and subjective logic. The coefficients $r^{\uparrow}$ and $r^{\downarrow}$ control the strength of upward and downward propagation in the domain hierarchy, while the subjective logic formulation relies on the base rate a and the non-informative prior weight W. Although several parameter settings were explored on a validation set, we report results for fixed values due to space constraints: $r^{\uparrow} = r^{\downarrow} = 0.8$, $a = 0.5$, and $W = 2.0$, which corresponds to the best-performing configuration.

5. EVALUATION OF THE IMPACT OF THE DOMAIN STRUCTURE

In this section we evaluate the impact of the domain structure on the learner model, and on the performance of a downstream task, including in high uncertainty settings.

5.1 Impact on the Learner Model

We start by analyzing the impact of domain-informed propagation on both knowledge mastery and uncertainty. Table 2 reports the median and quartiles of the absolute changes induced across different datasets. Recall that Junyi and EEDI mainly differ in the average number of KCs per exercise. To make the datasets more comparable, we construct a third dataset, which is a mono-concept version of the EEDI dataset, by retaining a single leaf concept per exercise. This concept is selected deterministically by first choosing the deepest associated KC in the hierarchy (i.e., the most specific concept), and mapping it to a leaf node when necessary.

Table 2: Median and quartiles of the absolute changes induced by domain-informed propagation.

Absolute Delta	Q1	Median	Q3
Junyi dataset			
$\|\Delta\|$ Mastery ($m_t(l,c)$) (KT-A → KT-DA)	**0.053**	**0.107**	**0.164**
$\|\Delta\|$ Mastery (KT-USL → KT-DUSL)	0.001	0.004	0.015
$\|\Delta\|$ Uncertainty (KT-USL → KT-DUSL)	0.002	0.008	0.034
EEDI mono-concept dataset			
$\|\Delta\|$ Mastery (KT-A → KT-DA)	**0.032**	**0.107**	**0.204**
$\|\Delta\|$ Mastery (KT-USL → KT-DUSL)	0.007	0.032	0.064
$\|\Delta\|$ Uncertainty (KT-USL → KT-DUSL)	0.036	0.095	0.192
EEDI multi-concept dataset			
$\|\Delta\|$ Mastery (KT-A → KT-DA)	**0.022**	**0.050**	**0.084**
$\|\Delta\|$ Mastery (KT-USL → KT-DUSL)	0.009	0.023	0.052
$\|\Delta\|$ Uncertainty (KT-USL → KT-DUSL)	**0.055**	**0.124**	**0.241**

Let us first consider the impact of integrating the domain structure, on the knowledge mastery. Recall that knowledge mastery ranges between 0 and 1, hence also the impact. Integrating the domain structure in a model that does not manage uncertainty (KT-DA vs. KT-A), has a significant impact on the knowledge mastery in a mono-concept setting, with a median of impact equal to 0.107. On EEDI, Q3 is

equal to 0.204, which means that the impact is larger than 20 points of percentage for 25% of the concepts. Yet, this impact is twice lower in a multi-concept setting. Managing domain structure on uncertainty-aware learner models (KT-DUSL vs. KT-USL) impacts the knowledge mastery to a lower extent (median around 0.03 on all datasets).

Focusing on the change in uncertainty on the KCs, the impact is larger on EEDI (both settings) than on Junyi. The highest impact is obtained on the multi-concept setting (median equal to 0.124 and Q3 is equal to 0.241). Recall that this large impact on uncertainty occurs in a context of small impact on knowledge mastery.

We can conclude that on a mono-concept setting, KT-DA impacts significantly more the knowledge mastery than KT-DUSL. Besides, KT-DUSL impacts more uncertainty in multi-concept settings.

Based on these conclusions we wonder if these significant changes in knowledge mastery and uncertainty translate into improved performance on a downstream task.

5.2 Impact on the Next-Question Correctness Prediction Task

In this section, we aim at evaluating the impact of the domain structure on a downstream task. In particular, we choose to evaluate on the next-question correctness prediction task.

5.2.1 Overall Impact

To evaluate the overall impact of the domain structure on the next-question correctness prediction task, we use the three metrics introduced in section 4.2, namely AUC, ACC@0.5 and LogLoss.

We first consider Junyi dataset. Evaluation metrics values are displayed in Table 3.

Table 3: AUC, ACC@0.5 and LogLoss on Junyi dataset.

Model	AUC	ACC@0.5	LogLoss
KT-A	0.702	0.728	1.178
KT-USL	0.705	0.728	0.556
KT-DA	0.702	0.728	0.575
KT-DUSL	**0.710**	**0.729**	**0.551**

The impact of the exploitation of the domain structure can be evaluated by comparing two pairs of models. First, we compare models that manage KC knowledge mastery only, concretely KT-A and KT-DA. The domain structure has no effect on AUC or accuracy but improves LogLoss significantly. The effect on AUC is thus not related to the significant impact on knowledge mastery (see Section 5.1). Second, we compare models that quantify and manage uncertainty, concretely KT-USL and KT-DUSL. KT-DUSL achieves slightly higher performance, particularly in AUC (0.705 to 0.710) and LogLoss (0.556 to 0.551). Here again, the impact on knowledge mastery was null. Overall, we can say that exploiting the domain structure and propagating mastery and uncertainty through this structure remains beneficial when used with an uncertainty-aware model, although the impact on the learner model is limited.

Let us now examine whether these conclusions hold on the EEDI dataset, in which each exercise is associated with multiple KCs. Performance is displayed in Table 4.

Table 4: AUC, ACC@0.5 and LogLoss on EEDI dataset.

Model	AUC	ACC@0.50	LogLoss
KT-A	0.723	0.693	0.620
KT-USL	**0.730**	**0.696**	0.578
KT-DA	0.722	0.692	0.587
KT-DUSL	0.729	0.695	**0.576**

On this dataset, the domain structure does not consistently improve these metrics. Indeed, KT-DUSL does not improve AUC and accuracy, and only slightly improves LogLoss. We hypothesize that this is due to the multi-concept setting, in which much of the relational information between concepts is implicitly captured through the exercise-KC associations. As a consequence, the explicit relation provided by the domain structure is already present in the data. This is in line with the limited impact on knowledge mastery from the previous section.

To test this hypothesis, we perform similar evaluation on the mono-concept version of the EEDI dataset, The associated performance is reported in Table 5.

Table 5: AUC, ACC@0.5 and LogLoss on mono-concept EEDI dataset.

Model	AUC	ACC@0.50	LogLoss
KT-A	0.667	0.676	3.043
KT-USL	0.680	0.676	0.616
KT-DA	0.686	0.680	0.620
KT-DUSL	**0.702**	**0.683**	**0.601**

We can see that, as on Junyi, considering the domain structure improves performance. KT-DUSL consistently outperforms all baseline models across AUC, ACC@0.50, and LogLoss, with particularly large gains over KT-USL (AUC 0.680 to 0.702; LogLoss 0.616 to 0.601). This tends to show that, in the context of mono-concept datasets, that can be associated to sparse data, the domain structure contributes to the quality of the learner model. More precisely, the quality of downstream next-question correctness prediction task is increased. This thus supports our hypothesis.

These experiments highlight two important conclusions: 1) quantifying uncertainty is broadly beneficial across datasets, 2) considering the domain structure has a context-dependent impact. If the relations between KCs are already implicitly encoded in the dataset (such as multiple KCs per exercise), domain-informed propagation does not provide additional information, so does not improve the performance. In contrast, in the context of sparse data (single concept per exercise), the domain structure contributes to improve the performance.

These findings lead us to investigate deeper the relation between the amount of data available, the uncertainty and performance. Concretely, we wonder if 1) the domain structure is more beneficial in the context of limited information

about the learner, 2) the domain structure is more beneficial in high uncertainty settings.

5.2.2 Limited Evidence Settings

In this section, we analyze the benefits of incorporating the domain structure under different levels of available evidence per learner-concept pair by comparing KT-USL and KT-DUSL. The amount of evidence is quantified by k, the number of past interactions a learner had with a given KC. Concretely, k denotes the interaction index for a learner-KC pair: $k = 1$ corresponds to the learner's first encounter with the KC, $k = 2$ to the second interaction, and so on.

Table 6: Performance by evidence count k on Junyi (P1, exact mode, global).

k	Method	n	AUC	ACC@0.50	LogLoss
1	Leaf-SL	484,455	0.500	**0.676**	0.693
	Prop-SL	484,455	**0.557**	**0.676**	**0.656**
3	Leaf-SL	449,388	0.655	**0.743**	0.588
	Prop-SL	449,388	**0.674**	0.741	**0.569**
5	Leaf-SL	425,942	0.689	0.759	0.543
	Prop-SL	425,942	**0.700**	**0.760**	**0.532**
10	Leaf-SL	340,884	0.707	0.740	0.538
	Prop-SL	340,884	**0.711**	**0.742**	**0.535**
20	Leaf-SL	241,172	0.716	0.737	0.539
	Prop-SL	241,172	**0.718**	**0.738**	**0.537**

Table 6 reports AUC, ACC@0.5 and LogLoss for different values of k. Higher values of k correspond to smaller dataset sizes. We can see that KT-DUSL consistently outperforms KT-USL on the three metrics, whatever is the value of k. The largest gains are observed with low values of k, i.e. with little evidence.

For $k = 1$, incorporating domain structure yields a substantial improvement in AUC (11% improvement, from 0.500 to 0.557) and a notable reduction in LogLoss (5.3% improvement, from 0.693 to 0.656). This indicates that when little information about the learner is available, better discrimination and probability calibration are reached. As evidence accumulates ($k = 3$ and $k = 5$), the performance gap narrows but remains consistent, suggesting that the domain structure continues to provide complementary information beyond local evidence. For larger values of k ($k \geq 10$), difference in performance between the two models is almost null, reflecting the fact that the structure becomes marginal as evidence increases.

Overall, these results indicate that domain-informed propagation is most beneficial in the context of low evidence, where it contributes to effectively compensate for sparse learner–KC interactions. Its impact attenuates as evidence increases.

Let us now study to what extent these conclusions hold on EEDI dataset (mono-concept and multi-concept). Table 7 shows performance on the mono-concept EEDI dataset. As on Junyi, domain-informed propagation is consistently beneficial across all evidence levels. For small values of k, KT-DUSL yields large gains over KT-USL, particularly in AUC (19% improvement, from 0.500 to 0.594 for $k = 1$) and LogLoss (4% improvement, from 0.693 to 0.666). As on Junyi, as k increases, improvements decrease for all metrics. This confirms the importance of domain structure when evidence is low and becomes useless when evidence increases.

Table 7: AUC, ACC@0.5 and LogLoss according to the amount of evidence per (learner, concept) pair on EEDI (mono-concept) dataset.

k	**Method**	n	**AUC**	**ACC@0.50**	**LogLoss**
1	KT-USL	461,224	0.500	0.619	0.693
	KT-DUSL	461,224	**0.594**	**0.630**	**0.666**
3	KT-USL	216,016	0.670	0.671	0.624
	KT-DUSL	216,016	**0.704**	**0.681**	**0.609**
5	KT-USL	131,670	0.711	0.691	0.593
	KT-DUSL	131,670	**0.730**	**0.698**	**0.582**
10	KT-USL	48,739	0.751	0.716	0.559
	KT-DUSL	48,739	**0.759**	**0.719**	**0.553**
20	KT-USL	8,037	0.757	0.725	0.538
	KT-DUSL	8,037	**0.761**	**0.727**	**0.535**

On the multi-concept EEDI dataset, we observe a markedly different behavior. The benefits of domain-informed propagation are almost null, whatever is the amount of evidence. We choose to not show values here. This confirms again the hypothesis drawn in section 5.2.1, the multi-concept setting implicitly encodes the domain structure, making the explicit domain structure useless. This is true even when evidence is low.

5.2.3 High Uncertainty Settings

We now intend to evaluate the benefits of incorporating the domain structure considering different uncertainty values. To this aim, we evaluate the impact of uncertainty on predictive performance, and in particular on AUC. Figures 2 and 3 show the AUC computed over sliding windows of 200,000 interactions, where each point corresponds to the average uncertainty of leaf KCs within a window, for the Junyi and mono-concept EEDI datasets. In both datasets, performance is consistently lower with high uncertainty values (30% decrease when uncertainty equals 1.0 compared to the case where uncertainty equals 0.0). This reflects the intrinsic difficulty of the prediction task in high uncertainty settings. However, in both datasets, KT-DUSL outperforms KT-USL when uncertainty is larger than 0.3. The higher the uncertainty, the larger the increase. When uncertainty is maximal (1.0), this increase reaches 12% and 18% on Junyi and EEDI respectively. This confirms the relevance of managing domain structure in uncertainty-aware models in the context of high uncertainty.

The previous two sections have highlighted the fact that the domain structure contributes to improve the performance on the next-question correctness prediction downstream task. This effect is particularly pronounced in low-evidence and high-uncertainty settings, where the domain structure contributes the most. The strong relation between evidence and uncertainty values is clearly stated thanks to these experiments.

These results allow us to answer the research questions in-

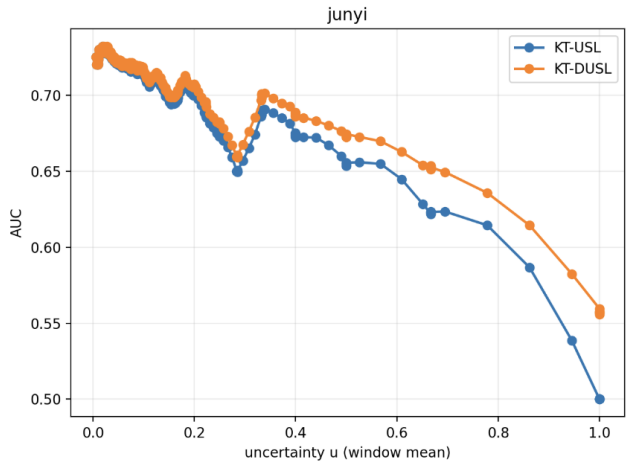


Figure 2: Sliding-window AUC of KT-USL and KT-DUSL as a function of uncertainty on the Junyi dataset.

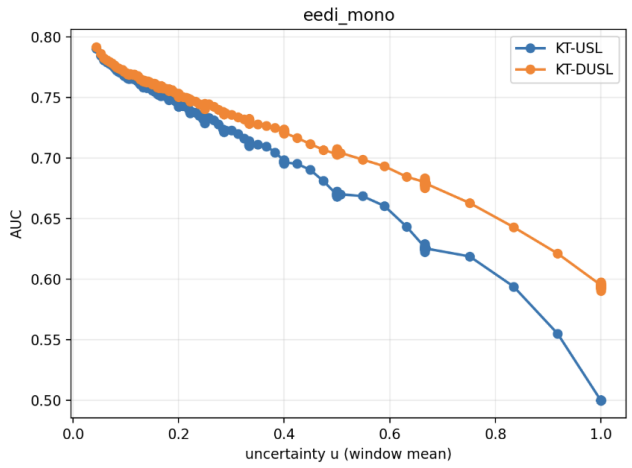


Figure 3: Sliding-window AUC of KT-USL and KT-DUSL as a function of uncertainty on the EEDI (mono-concept) dataset.

troduced in Section 1. Incorporating the domain structure actually impacts both the learner model and downstream prediction performance, thereby addressing S-RQ1. In particular, domain-informed propagation is most beneficial in settings characterized by high uncertainty or limited evidence, while its contribution becomes marginal when sufficient learner-KC information is available, thus addressing S-RQ2.

6. CONCLUSION

In this work, we studied the relevance of managing domain structure in learner models and associated downstream tasks. To this aim, we introduced KT-DUSL, a domain-informed learner model that jointly represents and propagates knowledge mastery and associated uncertainty in a learner model. Building on subjective logic, KT-DUSL performs local evidence updates and then propagates the resulting opinions through the domain structure, using discounting and consensus operators, thereby preserving uncertainty throughout the process.

Experiments on two mathematics-related datasets: Junyi and EEDI show that explicit uncertainty modeling consistently improves next-question correctness prediction, especially in AUC and LogLoss. We further find that, in the context of mono-concept exercises, domain-informed propagation provides the largest gains in high-uncertainty and low evidence per learner-concept pair settings. In contrast, in the multi-concept EEDI setting, hierarchical propagation brings limited benefit, suggesting that rich exercise-concept information encodes substantial relational information, and provides as much information as the domain structure.

Future work will leverage the joint propagation of knowledge mastery and epistemic uncertainty to develop an adaptive learning system that is guided by both estimated mastery and uncertainty within the domain structure. We also plan to investigate privacy-preserving deployments, in particular through federated learning, where learners' personal knowledge graphs are maintained locally and only a limited set of shared parameters is exchanged, thereby enabling collaborative learning.

References

[1] H. Abu-Rasheed, M. Dornhöfer, C. Weber, G. Kismihók, U. Buchmann, and M. Fathi. Building contextual knowledge graphs for personalized learning recommendations using text mining and semantic graph completion. In *2023 IEEE International Conference on Advanced Learning Technologies (ICALT)*, pages 36–40. IEEE, 2023.

[2] Q. U. Ain, M. A. Chatti, P. A. Meteng Kamdem, R. Alatrash, S. Joarder, and C. Siepmann. Learner modeling and recommendation of learning resources using personal knowledge graphs. In *Proceedings of the 14th learning analytics and knowledge conference*, pages 273–283, 2024.

[3] R. A. Ait Chabane Rania, Brun Armelle. Uncertainty-aware knowledge tracing: Towards the use of subjective logic. In *XAI-Ed 2026 Proceedings of International Conference on Learning Analytics Knowledge (LAK 2026)*, 2026. To be published.

[4] R. Alatrash, M. A. Chatti, Q. U. Ain, and S. Joarder. Transparent learner knowledge state modeling using personal knowledge graphs and graph neural networks. In *Adjunct proceedings of the 32nd ACM conference on user Modeling, adaptation and personalization*, pages 591–596, 2024.

[5] K. Balog and T. Kenter. Personal knowledge graphs: A research agenda. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 217–220, 2019.

[6] F. Böck, M. Ochs, A. Henrich, D. Landes, J. L. Leidner, and Y. Sedelmaier. Learner models: design, components, structure, and modelling; a systematic literature review. 2025.

[7] P. Brusilovsky and E. Millán. User models for adaptive hypermedia and adaptive educational systems. In *The*

adaptive web: methods and strategies of web personalization, pages 3–53. Springer, 2007.

[8] S. Bull. There are open learner models about! *IEEE Transactions on Learning Technologies*, 13(2):425–448, 2020.

[9] S. Bull and J. Kay. Open learner models. In *Advances in intelligent tutoring systems*, pages 301–322. Springer, 2010.

[10] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*, pages 164–175. Springer, 2006.

[11] P. Chen, Y. Lu, V. W. Zheng, and Y. Pian. Prerequisite-driven deep knowledge tracing. In *2018 IEEE international conference on data mining (ICDM)*, pages 39–48. IEEE, 2018.

[12] Y. Chen, J. P. González-Brenes, and J. Tian. Joint discovery of skill prerequisite graphs and student models. *International Educational Data Mining Society*, 2016.

[13] W. Cheng, H. Du, C. Li, E. Ni, L. Tan, T. Xu, and Y. Ni. Uncertainty-aware knowledge tracing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 27905–27913, 2025.

[14] S. T. Christie, C. Cook, and A. N. Rafferty. Uncertainty-preserving deep knowledge tracing with state-space models. *arXiv preprint arXiv:2407.17427*, 2024.

[15] C. Conati, A. Gertner, and K. Vanlehn. Using bayesian networks to manage uncertainty in student modeling. *User modeling and user-adapted interaction*, 12(4):371–417, 2002.

[16] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[17] F.-R. Dang, Y. Liu, W. Zhao, and W. Zhang. Constructing an educational knowledge graph with concepts linked to wikipedia. *Journal of Computer Science and Technology*, 36(5):1200–1211, 2021.

[18] M. C. Desmarais and R. S. d. Baker. A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction*, 22(1):9–38, 2012.

[19] H. Drachsler, K. Verbert, O. C. Santos, and N. Manouselis. Panorama of recommender systems to support learning. In *Recommender systems handbook*, pages 421–451. Springer, 2015.

[20] X. Gao et al. A hierarchical probabilistic framework for incremental knowledge tracing in classroom settings. *arXiv preprint arXiv:2506.09393*, 2025.

[21] D. Hooshyar, M. Pedaste, K. Saks, Ä. Leijen, E. Bardone, and M. Wang. Open learner models in supporting self-regulated learning in higher education: A systematic literature review. *Computers & education*, 154:103878, 2020.

[22] T. Käser, S. Klingler, A. G. Schwing, and M. Gross. Dynamic bayesian networks for student modeling. *IEEE Transactions on Learning Technologies*, 10(4):450–462, 2017.

[23] P. Lv, X. Wang, J. Xu, and J. Wang. Intelligent personalised exercise recommendation: A weighted knowledge graph-based approach. *Computer Applications in Engineering Education*, 29(5):1403–1419, 2021.

[24] E. Millán, T. Loboda, and J. L. Pérez-De-La-Cruz. Bayesian networks for student model engineering. *Computers & Education*, 55(4):1663–1683, 2010.

[25] J. Mitton, P. Bhattacharyya, R. Abboud, and S. Woodhead. Uncertainty-aware knowledge tracing models. *arXiv preprint arXiv:2509.21514*, 2025.

[26] H. Nakagawa, Y. Iwasawa, and Y. Matsuo. Graph-based knowledge tracing: modeling student proficiency using graph neural network. In *IEEE/WIC/aCM international conference on web intelligence*, pages 156–163, 2019.

[27] D. Oblinger and J. K. Lippincott. *Learning spaces*, volume 444. Educause Washington, DC, 2006.

[28] Z. A. Pardos and N. T. Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *International conference on user modeling, adaptation, and personalization*, pages 255–266. Springer, 2010.

[29] P. I. Pavlik Jr, H. Cen, and K. R. Koedinger. Performance factors analysis–a new alternative to knowledge tracing. *Online submission*, 2009.

[30] R. Pelánek. Bayesian knowledge tracing, logistic models, and beyond: An overview of learner modeling techniques. *User Modeling and User-Adapted Interaction*, 27(3):313–350, 2017.

[31] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[32] J. Rihák and R. Pelánek. Choosing a student model for a real world application. In *Proceedings of the Workshop on Building ITS Bridges Across Frontiers*, 2016. ITS Workshop.

[33] M. G. Skjæveland, K. Balog, N. Bernard, W. Łajewska, and T. Linjordet. An ecosystem for personal knowledge graphs: A survey and research roadmap. *AI Open*, 5:55–69, 2024.

[34] S. Tong, Q. Liu, W. Huang, Z. Hunag, E. Chen, C. Liu, H. Ma, and S. Wang. Structure-based knowledge tracing: An influence propagation view. In *2020 IEEE international conference on data mining (ICDM)*, pages 541–550. IEEE, 2020.

[35] Y. Yang, J. Shen, Y. Qu, Y. Liu, K. Wang, Y. Zhu, W. Zhang, and Y. Yu. Gikt: a graph-based interaction model for knowledge tracing. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 299–315. Springer, 2020.

[36] L.-Y. Zhou and Y.-Y. Wang. Simulation of personalized english learning path recommendation system based on knowledge graph and deep reinforcement learning. *Scientific Reports*, 15(1):34554, 2025.

A Psychometric Framework for Context-Conditional Evaluation of LLM Tutoring in Programming Education

Jinnie Shin
University of Florida
jinnie.shin@ufl.edu

Zhe Li
University of Florida
zheli@ufl.edu

Pauline Aguinalde
University of Florida
aaguinalde@ufl.edu

Laura M. Cruz Castro
University of Florida
cruzcastrol@ufl.edu

ABSTRACT

LLM-based chatbots are increasingly used as on-demand tutors in computer science education, yet evaluating pedagogical effectiveness remains difficult because rubric scoring is costly and raw aggregates confound response quality with rater severity, criterion difficulty, and student-query context. We propose a Bayesian explanatory many-facet Rasch model (MFRM) that places rubric-criterion endorsements on a common logit scale and decomposes ratings into latent response quality, criterion difficulty, and rater severity, while incorporating query-level covariates as criterion-specific effects. Applied to 249 CodeHelp exchanges from a large undergraduate C++ course, double-coded by two experts on mistake identification, mistake location, guidance, accuracy of guidance, and correctness of next steps, the model indicates minimal systematic rater differences and pronounced criterion-difficulty gaps. Diagnosis criteria are the most difficult to satisfy, whereas accuracy and next-step criteria are the easiest. Input sufficiency yields modest, criterion-dependent shifts in endorsement probability, and topic patterns are explored descriptively, underscoring that observed tutoring quality depends on what students provide and the type of help they request.

Keywords

educational chatbots; large language models; pedagogical effectiveness; Many-facet Rasch Model

1. INTRODUCTION

LLM-based chatbots, also referred to as conversational agents, are increasingly deployed as on-demand tutors across K–12 and higher education in a wide range of domains [28, 33]. In computer science education in particular, a growing body of work reports benefits for programming-focused learning tasks, including answering students' questions [36], supporting learning of core programming concepts [32], and assisting with assessment or performance feedback [12, 49]. Unlike earlier chatbots that relied on scripted rules or narrow task-specific models, LLM-based chatbots are *generative*, with an enhanced capacity to produce open-ended, context-sensitive explanations and guidance specific to a learner's prompt. Recent studies have therefore examined a range of LLM-powered tools, including general-purpose models (e.g., GPTs) and code-specialized assistants (e.g., Codex- and Copilot style systems), with emerging evidence that these tools can reshape both pedagogical interactions (e.g., debugging support, conceptual explanation) and assessment-related practices (e.g., feedback generation and evaluation) [55, 19].

At the same time, rapid adoption has intensified the need for rigorous evaluation frameworks that can determine not only whether these systems work, but also how, for whom, and under what conditions they support learning and meaningful educational interactions [21, 1, 28]. To date, evaluation evidence is often assembled by triangulating quantitative outcomes, ranging from proximal indicators such as student completion or engagement [41, 17] to more distal outcomes such as learning gains, together with qualitative analyses including error coding, interaction reviews, and feedback from students and instructors [62, 8, 53]. Complementing these outcome-focused approaches, studies increasingly emphasize pedagogically grounded indicators such as alignment with curriculum standards [3], adaptability to individual learner needs [27], and support for higher-order thinking [14, 57, 8]. Collectively, this literature underscores that chatbot effectiveness is multi-dimensional and should not be reduced to a single score or outcome [31].

However, translating this multi-dimensional view into scalable measurement at the level where tutoring actually occurs, namely, the response and interaction level, remains challenging. In practice, pedagogical effectiveness is still most often judged through expert human annotation, typically via rubric-based scoring of chatbot responses or full interaction transcripts [61, 20, 37]. To improve scalability, recent computational work has explored automated proxies, including reference-based overlap measures (e.g., BLEU, METEOR, ROUGE) and reference-free confidence and similarity scores [10]. However, there is growing concern that these metrics provide limited pedagogical insight [37]. Although they can capture surface properties such as fluency or lexical overlap, they do not directly assess whether a chatbot adapts instruction to a learner's needs, detects misconceptions, or supports conceptual understanding. This creates a practical tension: expert evaluation remains necessary for pedagogically consequential judgments, yet it is costly and difficult to interpret at scale.

Jinnie Shin, Zhe Li, Pauline Aguinalde, and Laura Cruz Castro. A Psychometric Framework for Context-Conditional Evaluation of LLM Tutoring in Programming Education. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 29–43. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039780

Recent work has also begun to apply Rasch-family models to structure LLM evaluation. Some studies treat chatbots as "examinees" and use Rasch analysis to benchmark model performance relative to item difficulty in test-like formats [7]. Other work uses many-facet Rasch models in human rating systems, often to quantify severity, consistency, or bias when LLMs act as automated scorers or raters [59, 23, 26, 34]. While valuable, these applications do not directly address the most common educational tutoring evaluation setting, in which experts apply pedagogically grounded rubrics to open-ended responses and scores must be interpreted under rater variability, uneven criterion difficulty, and systematic differences in query context.

Against this backdrop, even when pedagogically grounded rubrics are applied by experts, results are often summarized using coarse aggregates such as mean rubric scores, percent endorsement by criterion, or composite ratings [37, 46]. These summaries implicitly treat observed ratings as direct measurements of pedagogical effectiveness, which is problematic for at least three reasons. First, human judgments can vary systematically with rater severity or leniency, producing shifts in scores that are unrelated to the chatbot's underlying instructional quality [4, 54]. Second, rubric criteria are rarely equally difficult to satisfy, particularly for tutoring behaviors that require diagnosis and targeted remediation, so aggregation can mask which instructional capabilities are genuinely challenging for the system [31, 40, 38]. Third, many evaluations use cost-constrained annotation designs in which different raters score different subsets of responses, creating partially crossed designs where rater severity can become confounded with the specific responses and query contexts being scored [42]. Together, these limitations motivate a psychometric evaluation layer that can separate response quality from rater effects and criterion difficulty, and support context-conditional interpretation of rubric-based tutoring performance.

These limitations are especially consequential in programming support settings. Pedagogical effectiveness depends not only on producing a fluent response, but on accurately diagnosing student errors (e.g., identifying and localizing mistakes), providing conceptually sound guidance, and offering actionable next steps while respecting constraints that discourage directly revealing solutions. At the same time, the learner's query context strongly conditions what the chatbot can infer and how its response is judged. Students may provide complete code and error traces, partial snippets, or only a high-level description of confusion. Likewise, the requested assistance may range from debugging and error localization to conceptual explanations or strategy-level guidance. Without modeling query sufficiency and question type, it is difficult to determine whether observed rubric scores reflect the chatbot's underlying instructional capability or systematic variation in the interaction context that shapes both response generation and human judgment.

To address these gaps, we propose a Bayesian explanatory Many-Facet Rasch Model (MFRM) as a psychometric evaluation layer for rubric-based assessment of LLM tutoring responses in authentic programming help interactions. Our approach places rubric-criterion endorsements on a common logit scale and separates observed ratings into (i) latent response-level pedagogical quality, (ii) criterion difficulty, and (iii) rater severity, enabling uncertainty-aware inference even under human-annotation designs where rater severity and criterion difficulty would otherwise be confounded in simple aggregates. We further incorporate interaction-context information, most notably whether the student provided sufficient inputs (e.g., code and error traces), as criterion-specific explanatory effects, so that tutoring quality can be interpreted conditional on the help-seeking context rather than as a single global average.

This framework yields criterion-level diagnostics that are not available from raw endorsement rates or standard automated metrics and supports more actionable conclusions about where LLM tutors succeed and where they break down. In this study, we focus on mistake-remediation tutoring behaviors in a C++ debugging task and use a rubric that explicitly distinguishes *mistake location* from mistake identification and isolates *accuracy of guidance* to capture potentially harmful hallucinated or misapplied recommendations. Guided by this evaluation framework, we address the following research questions:

RQ1. Which pedagogical rubric criteria are easiest versus most difficult for an LLM-based programming tutor to satisfy, after accounting for rater severity?

RQ2. How does student-query sufficiency shift the probability that the chatbot satisfies each pedagogical criterion in a programming task?

RQ3. How does student question content relate to criterion-level pedagogical effectiveness in a programming task? *(exploratory)*

We make three contributions. First, we operationalize mistake-remediation tutoring quality for programming support with a rubric and annotation protocol that separates diagnosis (identifying and locating mistakes) from downstream guidance and correctness. Second, we introduce a Bayesian explanatory MFRM that converts rubric-based human judgments into facet-adjusted, uncertainty-aware estimates of pedagogical effectiveness and quantifies how evaluation shifts with query sufficiency. Third, we provide empirical evidence that diagnosis is the limiting capability for the studied tutor and that commonly used reference-free automated metrics capture only a weak proxy for correctness-focused outcomes.

2. RELATED WORK

2.1 LLM Chatbots as Instructional Agents

LLM-based chatbots have emerged in recent years, advancing beyond traditional rule-based algorithms, which have been limited in language understanding. Additionally, responses from rule-based algorithms are planned and predefined [48], which pose challenges regarding unexpected inputs by learners. In contrast, LLM-based chatbots directly address this constraint by training on large-scale datasets, allowing it to capture more contextual and semantic understanding [61], forgoing the need for rigid pre-planning for every potential input. Furthermore, its generative ability to produce natural language supports student learning experiences. Interactions with LLM-based instructional agents are

adaptive, provide timely feedback, and serve as an effective resource for on-demand information [2, 45, 43, 44].

LLM-based chatbots are rapidly reshaping programming education by providing on-demand, natural-language tutoring that can explain concepts, diagnose errors, and support debugging workflows in ways that earlier pre-defined rules or narrowly trained course chatbots struggled to achieve [56]. In authentic course settings, students use LLM tutors for a wide range of needs, but the dominant pattern is highly task-proximal help on assignments (e.g., fixing code, interpreting errors), with comparatively fewer interactions aimed at deeper conceptual inquiry, which raises both opportunity and risk for learning [47]. Recent systems for computing education have introduced instructional guardrails that avoid revealing complete solutions while still offering actionable, context-aware guidance, demonstrating that LLM chatbots can be intentionally shaped toward pedagogically meaningful behaviors [29, 25].

At the same time, evidence from programming courses suggests that students' use of general-purpose chatbots (e.g., ChatGPT) often concentrates on error checking and debugging and may be perceived as helpful for understanding and productivity, yet it can also introduce concerns about code quality, learning, and collaboration practices such as pair programming [18]. Alongside these classroom deployments, research prototypes increasingly combine LLM generation with program-analysis constraints to deliver finer-grained "next-step" hints, aiming to preserve learner agency while improving technical correctness and usefulness [5]. Collectively, this literature motivates treating LLM-based programming tutors not as monolithic tools but as designable instructional systems whose pedagogical value depends on their guardrails, the kinds of help students seek, and the learning context in which they are embedded [25, 47, 18].

2.2 Chatbot Pedagogical Efficacy Evaluation

Approaches to evaluating chatbots vary widely depending on the evaluation goal and the construct of interest. Prior reviews highlight that chatbot evaluation spans multiple perspectives, including user experience, information retrieval, linguistic quality, technical performance, and business outcomes, each emphasizing different attributes and metrics [39]. For instance, linguistic evaluations emphasize response correctness and coherence, whereas user-experience evaluations focus on usability and task completion. Within education, commonly reported objectives include acceptance and adoption, learning outcomes, motivation and affect, usability, technical correctness, and other psychological factors [21]. In this study, we focus specifically on *pedagogical efficacy*, which is define as the extent to which a chatbot supports learning through instructionally appropriate behaviors (e.g., promoting active learning, adapting support to learner needs, managing cognitive load) [24, 31].

Given these objectives, chatbot evaluations are typically conducted through Wizard-of-Oz studies, controlled laboratory experiments, field deployments, or technical validation studies [39]. Measurement within these designs most often relies on human judgments or user-reported instruments (e.g., rubrics, rating scales, surveys), sometimes complemented by behavioral indicators such as task success or learning gains. For example, to operationalize pedagogical ability, Maurya et al. propose eight rubric dimensions rated by human annotators on a 3-point scale [31]. More recently, automated evaluation has also been explored, particularly through lexical-overlap or classification metrics (e.g., BLEU, accuracy, F1, precision/recall) and related reference-based similarity measures (e.g., METEOR, ROUGE, CIDEr), which assume the availability of ground-truth target responses [13, 10, 35]. However, because pedagogical efficacy depends on context-sensitive instructional moves rather than surface similarity to a single reference, these automated metrics are often insufficient on their own, which helps explain why human judgment remains the dominant evidence source for pedagogical evaluation.

Consequently, the empirical literature most often reports effectiveness through aggregated human ratings or rankings. This spans the broader scope of the effectiveness of conversational agents, including educational chatbots. For example, the BEA 2023 shared task evaluated candidate *AI-teacher* responses using 298 crowd raters from Prolific [50], where each interaction was judged by multiple raters and systems were primarily compared via average ranks (1=best to 4=worst) and aggregate leaderboard scores. Similar aggregation based reporting is also common in the broader conversational-systems literature, where small panels of annotators or crowd workers rate responses on a handful of quality dimensions and results are reported as mean or composite scores [22, 16]. For instance, Jiang and Ahuja [22] collected three assessors' judgments per response on informativeness, fluency, and human-likeness and used the mean rating as the primary quality measure. Gao and Jiang [16] similarly recruited crowd workers to rate responses on multiple attributes (e.g., usefulness, accuracy, fluency, human-likeness) and reported performance via aggregated scores across conditions.

More education-focused LLM-tutor evaluations also follow this pattern, but with limited crossing and sparse overlap for reliability estimation. For instance, Daheim et al. [11] annotated a random set of 40 conversations and computed inter-rater reliability using only a small overlapping subset (9 conversations annotated by at least two raters), while still presenting results as criterion-level or overall aggregates. Even when evaluation is framed as comparative judgment, the reporting often remains rank- or mean-based. A few studies proposed to use pairwise comparative judgments and a probabilistic ranking model to infer pedagogical ability, yielding system-level scores and rankings rather than rater- and criterion-adjusted measures [51]. Collectively, this literature provides clear evidence that (a) evaluation commonly depends on human labels collected under partial crossing (different raters score different subsets), (b) outcomes are typically communicated as means, ranks, or composites, and (c) rater disagreement is acknowledged via inter-rater reliability but rarely incorporated into the final scoring model, thus, motivating measurement-model-based approaches that explicitly separate latent pedagogical quality from rater severity and criterion difficulty.

2.3 Bayesian explanatory MFRM

The LLM-based chatbot response to a student's single query is identified as a pair of *Chatbot response* and *student query*

that are rated by multiple raters. A Bayesian explanatory many-facet Rasch model (MFRM) is a Rasch-family model that decomposes the score or rating into multiple additive facets on a common logit scale. These typically involve criterion or item difficulty, response quality, and rater severity, while also allowing other covariates to explain systematic variation in ratings. This study showcases a novel application of MFRM by allowing the quality of chatbot's response as an evaluation target, and introducing other critical covariates that can systematically influence pedagogical quality.

Specifically, let $i \in \{1, \ldots, I\}$ index student–chatbot exchanges (responses), $j \in \{1, \ldots, J\}$ index rubric criteria, and $r \in \{1, \ldots, R\}$ index human raters. Let $y_{ijr} \in \{0, 1\}$ denote a binary endorsement indicating whether rater r judged response i as satisfying criterion j (1=endorsed, 0=not endorsed). We analyze the data in long format and index each observed rating by $n \in \{1, \ldots, N\}$ with mappings $i(n)$, $j(n)$, and $r(n)$. The model decomposes endorsements into (i) response-level latent pedagogical quality θ_i (higher values imply higher endorsement probability across criteria), (ii) criterion difficulty b_j (larger values indicate harder-to-satisfy criteria), and (iii) rater severity s_r (larger values indicate stricter raters, reducing endorsement probability), while allowing criterion-specific effects of query context covariates.

$$y_n \mid p_n \sim \text{Bernoulli}(p_n), \qquad \text{logit}(p_n) = \eta_n,$$
$$\eta_n \;=\; \theta_{i(n)} - b_{j(n)} - s_{r(n)} \;+\; \gamma_{j(n)} v_n \;+\; \sum_{k=1}^{K} \beta_{j(n)k}\, x_{nk}. \tag{1}$$

To characterize how the help-seeking context conditions rubric endorsement, we included two sets of query-level covariates. First, *input sufficiency* was encoded as a binary indicator $v_n \in \{0, 1\}$, where $v_n = 1$ denotes that the student query provided sufficient information (e.g., code, error traces, and/or descriptive context) for a meaningful diagnosis and response.[1] Second, we represented the *CS topic content* of each student query using K multi-label binary indicators $\{x_{nk}\}_{k=1}^{K}$, where $x_{nk} = 1$ if topic k is present in the query and $x_{nk} = 0$ otherwise (multiple topics may be active for the same query). Under this specification, γ_j captures the criterion-specific shift in endorsement associated with sufficient inputs, and β_{jk} captures the criterion-specific association between topic k and endorsement.

Because the logistic Rasch family is location-invariant, we imposed sum-to-zero constraints on additive facets via centering. Operationally, we sampled unconstrained raw parameters and applied mean-centering transformations: $\theta_i = \theta_i^{\text{raw}} - \overline{\theta^{\text{raw}}}$ and $s_r = s_r^{\text{raw}} - \overline{s^{\text{raw}}}$. For the topic coefficients, we imposed a within-criterion sum-to-zero constraint across the K topic effects, $\sum_{k=1}^{K} \beta_{jk} = 0$ for each criterion j, implemented as $\beta_{jk} = \beta_{jk}^{\text{raw}} - \overline{\beta_{j\cdot}^{\text{raw}}}$, where $\overline{\beta_{j\cdot}^{\text{raw}}} = \frac{1}{K}\sum_{k=1}^{K} \beta_{jk}^{\text{raw}}$. This constraint is compatible with multi-label topic coding because it centers the criterion-specific topic coefficients, not the observations.

[1]If input sufficiency was coded separately by each rater, then v_n is indexed by the rater in observation n; otherwise, v_n is treated as an exchange-level label shared across ratings for the exchange.

We used weakly informative Normal priors: θ_i^{raw}, b_j, $s_r^{\text{raw}} \sim \mathcal{N}(0, 1)$ and γ_j, $\beta_{jk}^{\text{raw}} \sim \mathcal{N}(0, 0.5^2)$. Posterior inference targeted $p(\boldsymbol{\Theta}, \mathbf{b}, \mathbf{s}, \boldsymbol{\gamma}, \mathbf{B} \mid \mathbf{y})$, where $\boldsymbol{\Theta} = \{\theta_i\}_{i=1}^{I}$, $\mathbf{b} = \{b_j\}_{j=1}^{J}$, $\mathbf{s} = \{s_r\}_{r=1}^{R}$, $\boldsymbol{\gamma} = \{\gamma_j\}_{j=1}^{J}$, and $\mathbf{B} = \{\beta_{jk}\}_{j=1,k=1}^{J,K}$.

$$p(\boldsymbol{\Theta}, \mathbf{b}, \mathbf{s}, \boldsymbol{\gamma}, \mathbf{B} \mid \mathbf{y}) \propto \prod_{n=1}^{N} \text{Bernoulli}\big(y_n \mid \text{logit}^{-1}(\eta_n)\big) \times p(\boldsymbol{\Theta})\, p(\mathbf{b})\, p(\mathbf{s})\, p(\boldsymbol{\gamma})\, p(\mathbf{B}). \tag{2}$$

We used Hamiltonian Monte Carlo with the No-U-Turn Sampler (NUTS), running 4 chains with 1,000 warmup iterations and 1,000 retained posterior draws per chain. Convergence was assessed using $\hat{R}$ and effective sample size.

For interpretable effect sizes, we translated selected effects into probability changes as the coefficients were estimated on the log-odds scale. For instance, for input sufficiency effect, for each rubric criteria j, we defined a baseline linear predictor at a reference point $\theta^*, s^*, \mathbf{x}^* = 0$ as follows:

$$\eta_{0j} = \theta^* - b_j - s^* \tag{3}$$

Then, the probability shift induced by input sufficiency is expressed as follows, where $\Delta P_j^{(suff)}$ was computed for each posterior draw to obtain a posterior distribution over probability-scale effects, summarized by posterior mean and 94% HDI.

$$\Delta P_j^{(suff)} = \sigma(\eta_{0j} + \gamma_j) - \sigma(\eta_{0j}), \quad \sigma(t) = \frac{1}{1 + e^{-t}} \tag{4}$$

3. METHODS

3.1 Participants and Data Collection

This study was conducted in a large-enrollment undergraduate computer science course at a public, research-intensive university in the southern United States (Spring 2024; $N = 669$)[2]. The course emphasized object-oriented programming in C++, and students could voluntarily use CodeHelp [29], which is an LLM-based programming tutor designed to provide debugging-oriented guidance while discouraging solution revealing, during a course assignment. We analyzed 249 unique, single-turn student-chatbot exchanges (one per student), each pairing a student query (including any provided code/error context captured by the platform) with the corresponding chatbot response. We excluded attempted submissions that CodeHelp flagged as insufficient and for which no response was returned.

3.2 Programming Task and Interaction Inputs

Students completed a C++ programming task targeting an intermediate object-oriented programming concept commonly referred to as the "Big Three" implementation (Fig. 4). The task required implementing a destructor, copy constructor,

[2]This study was conducted in accordance with institutional ethical guidelines for research involving human participants. The research protocol was reviewed and approved by the Institutional Review Board (IRB) RB202400344. All data involving human participants were handled in compliance with relevant privacy and ethical standards.

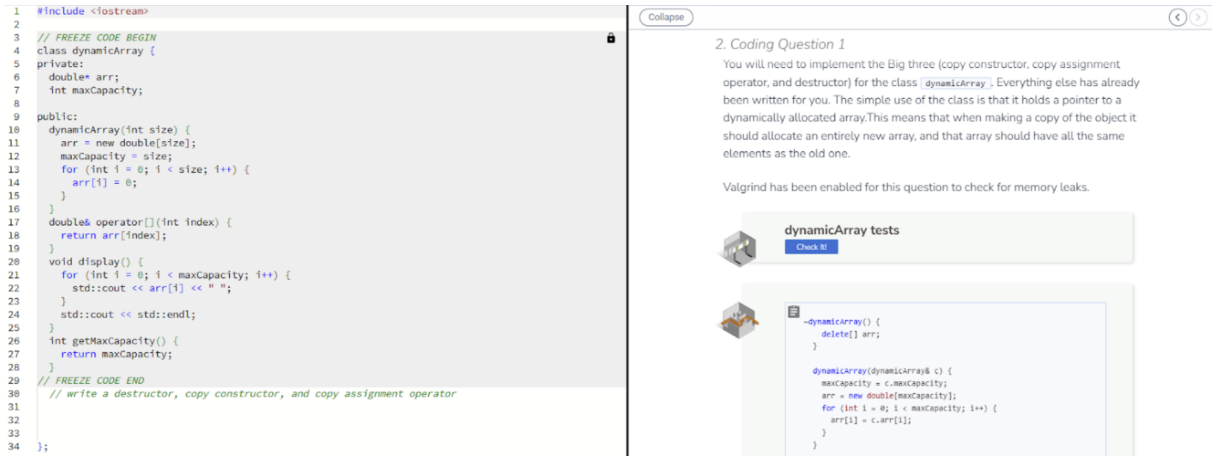


Figure 1: Programming Task Description

and copy assignment operator for a class that manages dynamically allocated memory and resources. The programming environment integrated Valgrind to detect runtime memory errors and memory leaks to support debugging. When interacting with CodeHelp, students could provide information via one or more fields: (1) *Code* (a snippet of their implementation), (2) *Error* (compiler output or Valgrind messages), and (3) *User Input* (a natural-language description of the issue). For each exchange, we logged the contents of these fields and the chatbot response.

3.3 Chatbot Evaluation

3.3.1 Rubric Development and Query Labels

To evaluate the pedagogical quality of chatbot responses in a mistake-remediation programming context, we adapted the unifying framework for mistake-remediation tasks [31, 24, 38]. The framework emphasizes core dimensions of tutoring quality (e.g., mistake remediation and guidance/actionability). Because our task required assessing whether the chatbot could pinpoint where an error occurred in student code, we introduced a dedicated *Mistake Location* dimension. To distinguish helpful guidance from potentially harmful or hallucinated recommendations, we additionally included an *Accuracy of Guidance* dimension. Table 1 summarizes all rubric dimensions. Detailed decision rules and benchmark examples used in annotator training are provided in Appendix A.

For modeling, we used a binary endorsement format. For dimensions i1–i4, raters initially assigned *yes/partial/no*, and we recoded *partial* as *no* (i.e., Yes= 1, Partial/No= 0). In this study, we treated partial as not-fully-met to target a conservative mastery-like endorsement. Dimension i5 was coded as Yes/No. In addition to rubric-based response-quality criteria, raters also coded two auxiliary query-level labels: *Input Sufficiency* (S; whether the student provided enough information for a meaningful response) and *CS topic categories* describing the content of the student query. Topic codes were multi-label (a single exchange could receive multiple topic categories). Table 2 summarizes the distribution of topic labels across exchanges.

3.3.2 Annotator Training and Coding Procedure

Table 1: Rubric Dimensions for Chatbot Evaluation

ID	Dimension	Scale	Description
Rubric dimensions (chatbot response quality)			
i1	Mistake Identification	yes partial no	Does the chatbot accurately recognize a mistake in the student's input?
i2	Mistake Location	yes partial no	Does the chatbot correctly identify the location/aspect of the student's mistake?
i3	Providing Guidance	yes partial no	Does the chatbot offer guidance relevant to the identified issue?
i4	Accuracy of Guidance	yes partial no	Is the guidance factually and conceptually correct?
i5	Correctness of Next Step	yes no	Are the suggested next steps and corrections accurate and appropriate?
Auxiliary query-level labels			
S	Input Sufficiency	yes no	Did the student provide enough information (e.g., code, error, context) for a meaningful response?

Table 2: Student Queries by Relevant CS Topics (multi-label)

Category	Description	Freq
Topic 1	Programming Fundamentals	146
Topic 2	Data Management and Processing	130
Topic 3	Object-Oriented Programming (OOP)	176
Topic 4	Code Construction	216
Topic 5	Language Unique Programming Concepts	239

Two subject matter experts independently coded all 249 student–chatbot exchanges. Before full coding, raters reviewed the task description, rubric definitions, and benchmark examples, and completed a structured training session to align decision rules. Raters then double-coded a random set of 20 exchanges and participated in a calibration meeting to resolve ambiguities and refine guidelines. A midpoint check-in was conducted to reinforce decision rules and ensure continued consistency. Disputed training cases were resolved by consensus and incorporated into an updated benchmark set used for ongoing reference during coding. Unless otherwise noted, analyses used both raters' original codes to estimate and adjust for rater severity within the many-facet Rasch framework.

Table 3: Model comparison using PSIS-LOO

Model	elpd_{loo}	SE	Δelpd	p_{loo}	Wt.	Rank
$M1_{\text{suff}}$	-1419.00	22.23	2.45	135.55	1.00	0
$M0_{\text{baseline}}$	-1421.45	22.18	0.00	135.20	0.00	1
$M3_{\text{both}}$	-1428.32	22.72	-6.87	153.87	0.00	2
$M2_{\text{topics}}$	-1430.19	22.69	-8.74	153.42	0.00	3

Note. Higher elpd_{loo} indicates better expected out-of-sample predictive fit. Δelpd is relative to $M0_{\text{baseline}}$.

Table 4: Observed endorsement rates by raters

Rater	i1	i2	i3	i4	i5
Rater 1	0.530	0.542	0.406	0.892	0.827
Rater 2	0.510	0.482	0.394	0.880	0.867

4. RESULTS

All models were fit to a long-format dataset with $N = 2{,}490$ binary observations (249 exchanges $\times$ 5 rubric criteria $\times$ 2 raters). Submissions flagged by our LLM-based chatbot, CodeHelp, as insufficient were excluded prior to annotation because no response was returned in such cases; SMEs labeled 5% as insufficient on the Input Sufficiency label (S). For modeling, *partial* ratings were recoded as 0 (not endorsed) to yield binary endorsements..

We evaluated out-of-sample predictive performance using PSIS-LOO computed from 4,000 posterior draws and the pointwise log-likelihood matrix. For the facets-only baseline model ($M0_{\text{baseline}}$), PSIS-LOO indicated stable importance sampling with all Pareto-k values ≤ 0.70 (100% "good"). We then compared four nested specifications (Table 3): a facets-only baseline ($M0_{\text{baseline}}$), a primary explanatory model adding criterion-specific input-sufficiency effects ($M1_{\text{suff}}$), and extended models that additionally included multi-label topic indicators ($M2_{\text{topics}}$) or both covariate sets ($M3_{\text{both}}$). Adding input sufficiency produced a small improvement in expected out-of-sample fit relative to the baseline ($\Delta\text{elpd}_{\text{loo}} = +2.45$), whereas including topic indicators increased effective model complexity ($p_{\text{loo}} \approx 153$) without improving PSIS-LOO, where $\Delta\text{elpd}_{\text{loo}} < 0$ for $M2_{\text{topics}}$ and $M3_{\text{both}}$. Accordingly, we use $M1_{\text{suff}}$ as the primary model for inference in RQ1–RQ2 (criterion difficulty and input sufficiency effects) and treat topic-based analyses (RQ3) as exploratory. We summarize posterior uncertainty using 94% highest-density intervals (HDIs).

Across the 249 exchanges, the two raters showed broadly similar endorsement patterns across criteria (Table 4). Here, we use *endorsement* to denote cases in which a criterion was judged as met (i.e., rated `yes`; see rubric in Table 1). Both raters endorsed the chatbot most frequently on Accuracy of Guidance (i4) and correctness of next steps (i5), and least frequently on Providing Guidance (i3), with intermediate endorsement for mistake identification and localization (i1–i2). This suggests that the chatbot more often produced factually correct and actionable guidance once it engaged with the problem, but struggled more consistently with diagnosing and responding to the student's specific error.

Table 5: Posterior mean rater severity estimates

Rater	$\bar{s}_r$
Rater 1	-0.036
Rater 2	$+0.036$

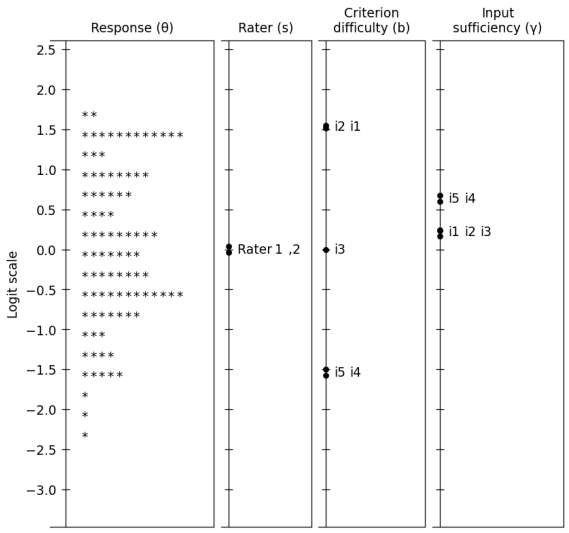


Figure 2: Posterior parameter map for the primary MFRM model ($M1_{\text{suff}}$).

We then fit a Bayesian explanatory Many-Facet Rasch Model (MFRM) to disentangle rater severity, criterion difficulty, and response-level pedagogical quality. Posterior mean rater severities were near zero and symmetric ($s_{\text{rater1}} = -0.036$, $s_{\text{rater2}} = +0.036$), indicating minimal systematic differences in endorsement tendency after accounting for criterion difficulty and response quality (Table 5). Response-level pedagogical quality estimates (θ_i; $I = 249$) were centered near zero by model identification ($\bar{\theta} \approx 0$, $SD \approx 1$) and spanned a broad range (min $= -2.45$, max $= 1.60$), indicating substantial heterogeneity in the chatbot's pedagogical effectiveness across student queries and underscoring the need to model response quality beyond raw aggregate endorsement rates.

4.1 Rubric criterion difficulty (RQ1)

Using the primary model $M1_{\text{suff}}$ (facets + input sufficiency), criterion difficulty estimates (b_j) indicated substantial differences in how challenging each pedagogical criterion was for the chatbot to satisfy. Larger b_j corresponds to a lower probability of endorsement, holding response quality and rater severity constant. The mistake diagnosis criteria were most difficult: Mistake Identification (i1) and Mistake Location (i2) had the largest positive posterior mean difficulties ($\bar{b}_{i1} = 1.517$, $\bar{b}_{i2} = 1.553$). Providing Guidance (i3) was close to the reference point ($\bar{b}_{i3} \approx 0$). In contrast, the accuracy-oriented criteria were easiest: Accuracy of Guidance (i4) and Correctness of next steps (i5) had negative difficulty estimates ($\bar{b}_{i4} = -1.577$, $\bar{b}_{i5} = -1.493$), indicating these behaviors were endorsed more readily. For interpretability, we also report baseline endorsement probabilities at a common reference setting ($\theta = 0$, $s = 0$, and $v = 0$). Baseline endorsement was lowest for i1 (0.188) and i2 (0.183), moderate for i3 (0.500), and high for i4 (0.819) and i5 (0.807) (Table 6). Together, these results suggest that the primary difficulty for the chatbot lies in *diagnosing* and *localizing* student errors, rather than in producing generally accurate or actionable guidance once an issue is addressed.

Table 6: Rubric criterion difficulty and baseline endorsement probability at the reference point

ID	Criterion	$\bar{b}_j$	94% HDI	$\bar{p}_{\text{base}}$	94% HDI
i1	Mistake identification	1.517	[0.820, 2.451]	0.188	[0.079, 0.306]
i2	Mistake location	1.553	[0.835, 2.516]	0.183	[0.075, 0.303]
i3	Providing guidance	-0.000	[-0.773, 0.768]	0.500	[0.317, 0.684]
i4	Accuracy of guidance	-1.577	[-2.666, -0.812]	0.819	[0.693, 0.935]
i5	correctness of next steps	-1.493	[-2.481, -0.724]	0.807	[0.674, 0.923]

Table 7: Input sufficiency effects by criterion.

ID	$\bar{\gamma}_j$	94% HDI	$\bar{p}_{\text{base}}$	$\overline{\Delta P}$	94% HDI
i1	0.242	[-0.384, 0.908]	0.188	0.037	[-0.067, 0.143]
i2	0.237	[-0.403, 0.871]	0.183	0.035	[-0.069, 0.135]
i3	0.169	[-0.451, 0.867]	0.500	0.041	[-0.109, 0.209]
i4	0.596	[-0.107, 1.278]	0.819	0.072	[-0.015, 0.169]
i5	0.672	[-0.007, 1.371]	0.807	0.083	[-0.008, 0.188]

Note. ΔP summarizes the probability change induced by sufficiency at the reference point ($\theta = 0$, $s = 0$, covariates $= 0$).

4.2 Effects of input sufficiency (RQ2)

Using the primary model $M1_{\text{suff}}$ (facets + input sufficiency), we examined whether rater-judged input sufficiency (i.e., whether the student query contained enough information to diagnose and address the issue) was associated with higher endorsement probabilities via criterion-specific effects (γ_j). Posterior mean sufficiency effects were positive across all criteria (Table 7), with larger mean shifts for the accuracy-oriented criteria (i4, i5). On the probability scale, the estimated mean changes were modest for the diagnosis and guidance criteria ($\Delta P \approx 0.037$–0.041 for i1–i3) and larger for the accuracy-oriented criteria ($\Delta P \approx 0.072$–0.083 for i4–i5). However, the 94% HDIs indicate substantial uncertainty, consistent with the small proportion of exchanges labeled as insufficient (5%) in the current dataset.

Figure 2 visualizes posterior parameter estimates from $M1_{\text{suff}}$, with all panels plotted on a common logit scale for comparison. The response-quality distribution (θ, left) shows substantial heterogeneity across student–chatbot exchanges. Rater severity estimates (s, second) are centered near zero for both raters, indicating minimal systematic differences in endorsement tendency. Criterion difficulty (b, third) differentiates the rubric criteria, with mistake diagnosis (i1–i2) the most difficult and the accuracy-oriented criteria (i4–i5) the easiest. Finally, the input-sufficiency effects (γ, right) are generally positive but imprecisely estimated, suggesting that more sufficient student queries may increase endorsement probabilities, with the largest expected gains for the accuracy-related criteria.

4.3 Exploratory Analysis for Effects of student query topic (RQ3)

We conducted an exploratory analysis of whether student question content was associated with criterion-level rubric outcomes. Specifically, we used multi-label indicators for five CS topic categories (Table 2) to summarize topic-conditional differences in endorsement and to estimate topic-augmented MFRM specifications (Figure 3; Table 8). Because adding topic indicators did not improve PSIS-LOO relative to the primary model and increased effective complexity (Table 3), we interpret topic patterns as descriptive and hypothesis-generating rather than confirmatory effects.

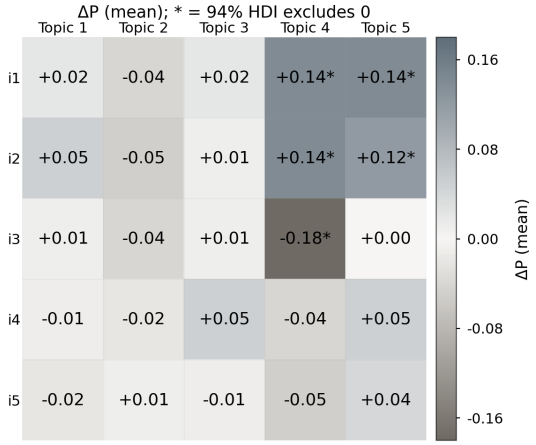


Figure 3: Topic effects by rubric criterion

Overall, topic-related variation was most apparent for the diagnosis-related criteria, Mistake Identification (i1) and Mistake Location (i2), and for Providing Guidance (i3). In contrast, topic associations for the accuracy-oriented criteria (i4–i5) were smaller and more uncertain, with 94% HDIs often spanning zero. In the topic-augmented model, topic effects are summarized as posterior mean probability shifts (ΔP) at a common reference point ($\theta = 0$, $s = 0$, input sufficiency $v = 0$, and other topic indicators set to 0), enabling criterion-by-topic comparisons on the probability scale.

For mistake diagnosis, the largest positive associations were observed for *Code Construction* (Topic 4) and *Language-Unique Programming Concepts* (Topic 5). For i1, Topic 4 and Topic 5 were associated with increases of approximately $\Delta P \approx +0.14$, with 94% HDIs excluding zero; a similar

Table 8: Exploratory topic associations with criterion endorsement (topic-augmented model).

	Topic	$\bar{\beta}$	94% HDI	$\overline{\Delta P}$	94% HDI		Topic	$\bar{\beta}$	94% HDI	$\overline{\Delta P}$	94% HDI
i1	1	0.103	[-0.390, 0.666]	0.019	[-0.057, 0.119]	**i1**	2	-0.308	[-0.827, 0.226]	-0.039	[-0.111, 0.036]
i1	3	0.113	[-0.324, 0.555]	0.019	[-0.047, 0.098]	**i1**	4	0.774	[0.209, 1.308]	0.142	[0.028, 0.268]
i1	5	0.768	[0.068, 1.450]	0.139	[0.015, 0.275]	**i2**	1	0.320	[-0.238, 0.875]	0.055	[-0.034, 0.165]
i2	2	-0.390	[-0.958, 0.131]	-0.047	[-0.128, 0.017]	**i2**	3	0.062	[-0.365, 0.486]	0.011	[-0.056, 0.079]
i2	4	0.777	[0.236, 1.333]	0.140	[0.023, 0.257]	**i2**	5	0.713	[0.022, 1.401]	0.125	[-0.005, 0.247]
i3	1	0.049	[-0.539, 0.633]	0.012	[-0.129, 0.146]	**i3**	2	-0.170	[-0.780, 0.413]	-0.040	[-0.190, 0.095]
i3	3	0.050	[-0.395, 0.506]	0.012	[-0.097, 0.117]	**i3**	4	-0.807	[-1.396, -0.247]	-0.185	[-0.316, -0.059]
i3	5	0.014	[-0.682, 0.711]	0.003	[-0.165, 0.169]	**i4**	1	-0.035	[-0.652, 0.598]	-0.010	[-0.106, 0.087]
i4	2	-0.096	[-0.783, 0.539]	-0.019	[-0.126, 0.078]	**i4**	3	0.369	[-0.145, 0.914]	0.045	[-0.028, 0.118]
i4	4	-0.274	[-0.986, 0.401]	-0.045	[-0.165, 0.064]	**i4**	5	0.376	[-0.414, 1.189]	0.047	[-0.056, 0.157]
i5	1	-0.111	[-0.753, 0.503]	-0.020	[-0.128, 0.078]	**i5**	2	0.093	[-0.562, 0.717]	0.010	[-0.091, 0.099]
i5	3	-0.060	[-0.591, 0.489]	-0.011	[-0.106, 0.067]	**i5**	4	-0.307	[-0.983, 0.344]	-0.053	[-0.180, 0.055]
i5	5	0.292	[-0.427, 1.039]	0.041	[-0.066, 0.151]						

Note. Each row reports the criterion–topic effect $\beta_{j,k}$ and the corresponding probability shift ΔP at the reference point ($\theta = 0$, $s = 0$, and other topic indicators set to 0). 94% HDIs are computed from the posterior.

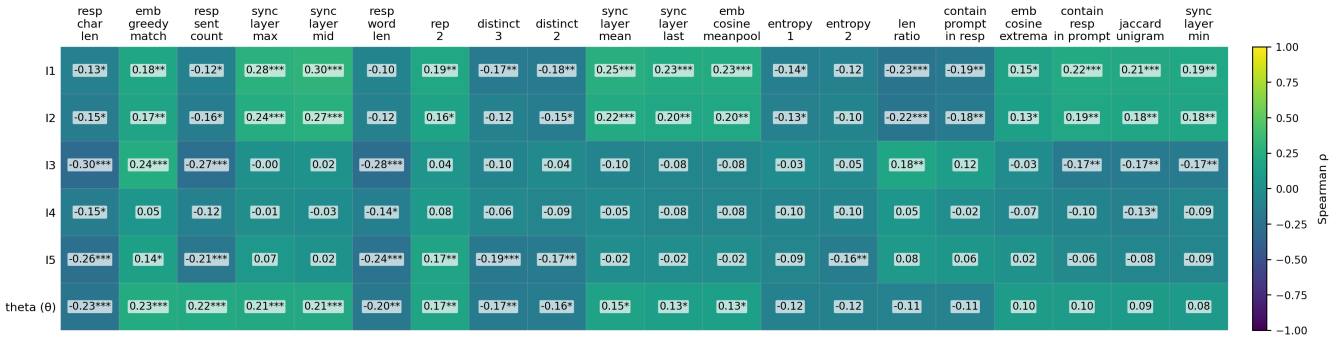


Figure 4: Spearman's correlations between the evaluation metrics and target rubric

pattern emerged for i2 (Topic 4: $\Delta P \approx +0.14$; Topic 5: $\Delta P \approx +0.12$). By comparison, *Programming Fundamentals* (Topic 1), *Data Management and Processing* (Topic 2), and *Object-Oriented Programming* (Topic 3) showed smaller shifts for i1 and i2 (typically $|\Delta P| \leq 0.05$) with uncertainty intervals overlapping zero.

Providing Guidance (i3) exhibited an asymmetric pattern: most topics were near zero, whereas *Code Construction* (Topic 4) was associated with a notable decrease in endorsement ($\Delta P \approx -0.18$; 94% HDI excluding zero). Taken together, this pattern suggests that topic differences may be most consequential for upstream diagnosis and for translating diagnosis into actionable guidance. However, given the lack of PSIS-LOO improvement for topic-augmented models, these results should be interpreted cautiously and motivate future work with larger or more balanced samples to test topic moderation more decisively.

4.4 Comparisons to Automated Metrics

To contextualize our rubric-based evaluation and provide convergent and discriminant validity evidence for the pedagogical constructs targeted by i1–i5, we compared rubric outcomes against a set of automated, reference-free metrics computed from each student–chatbot exchange. These included (a) *surface length and structure* (response character length, word length, sentence count, and the response-to-prompt length ratio), (b) *lexical diversity and redundancy* (type–token ratio; distinct-n for $n \in 1, 2, 3$; normalized n-gram entropy for $n \in 1, 2$; and repetition rates for unigram and bigram reuse), (c) *prompt–response overlap* (unigram and bigram Jaccard overlap; prompt containment in the response and vice versa), and (d) *readability* (Flesch Reading Ease and Flesch–Kincaid Grade). To approximate semantic relatedness without an external reference, we additionally computed *embedding-based similarity* between the prompt and response (mean-pooled cosine similarity, vector-extrema cosine similarity, and greedy token-level matching) and *layerwise synchrony* scores, defined as the cosine similarity between mean-pooled transformer representations across layers and summarized by the minimum, mean, maximum, mid-layer, and last-layer similarity. For each metric, we computed Spearman rank correlations (ρ) with (a) criterion-level endorsements (i1–i5) and (b) the response-level latent pedagogical quality estimate (θ) from the MFRM, and we summarize the correlation pattern in Figure 4. These correlations provide convergent evidence when rubric scores align with the signal captured by a specific class of metrics.

Overall, the correlation coefficients between automated metrics and rubric-based outcomes were generally modest (typically $|\rho| \leq .30$), indicating that commonly used reference-free metrics capture only partial information about pedagogical response quality. The MFRM latent quality estimate

θ aligned most consistently with semantic relatedness between the student query and the chatbot response, including embedding-based relevance (greedy matching; $\rho = .23^{***}$) and synchrony-based similarity (sync layer max and mid; $\rho = .21^{***}$), with smaller positive associations for overlap and reuse proxies (rep-2; $\rho = .17^{**}$). In contrast, θ was negatively associated with verbosity and lexical diversity indicators, including response length ($\rho \approx -.20$ to $-.23$) and distinctness measures (distinct-2/3; $\rho \approx -.16$ to $-.17$). Together, these findings suggest that higher-quality chatbot responses, as captured by θ, tend to be more directly aligned with the student prompt and more concise, rather than longer or more lexically diverse.

At the rubric-criterion level, the clearest alignment with automated metrics emerged for the chatbot's *diagnosis* behaviors. Mistake Identification (i1) and Mistake Location (i2) were positively associated with prompt-response semantic relatedness, including synchrony-based similarity (sync layer max and mid; i1 $\rho = .28^{***}$ and $.30^{***}$; i2 $\rho = .24^{***}$ and $.27^{***}$) and lexical overlap (unigram Jaccard; i1 $\rho = .21^{***}$; i2 $\rho = .18^{**}$). Both criteria were also negatively related to indicators of disproportionate or overly expansive responses, including the response-to-prompt length ratio (len ratio; i1 $\rho = -.23^{***}$; i2 $\rho = -.22^{***}$) and prompt-containment features (contain prompt in response; i1 $\rho = -.19^{**}$; i2 $\rho = -.18^{**}$), suggesting that higher diagnostic ratings tended to occur when responses stayed targeted to the student's query. Providing Guidance (i3) showed a different pattern that is related to a more concise and actionable help. Guidance endorsements were strongly negatively associated with verbosity (response character length $\rho = -.30^{***}$; sentence count $\rho = -.27^{***}$; word length $\rho = -.28^{***}$) while remaining positively related to embedding-based relevance (greedy matching; $\rho = .24^{***}$). This profile is consistent with raters rewarding guidance that is directly responsive and succinct rather than lengthy or diffuse.

In contrast, the correctness-focused criteria exhibited the weakest convergence with the automated metrics. Accuracy of Guidance (i4) showed near-zero correlations across most measures (with only small negative relationships with response length), and Correctness of next steps (i5) was primarily associated with surface length and diversity proxies rather than stronger positive semantic alignment signals. Overall, these findings indicate that standard reference-free features predominantly capture relevance and alignment and surface form (verbosity and lexical diversity), but provide incomplete coverage of pedagogically consequential constructs, particularly factual or conceptual correctness and the appropriateness of next-step suggestions. This limitation motivates the use of expert rubric scoring and psychometric modeling for rigorous evaluation of educational chatbot responses.

5. DISCUSSION

This study introduces a Bayesian explanatory Many-Facet Rasch Model (MFRM) as a psychometric evaluation layer for rubric-based assessment of LLM tutoring responses in authentic programming help interactions. Instead of interpreting rubric endorsements as direct measures of pedagogical effectiveness, the model disentangles latent response quality from rubric-criterion difficulty and rater severity, and incorporates interaction context through query-level covariates. The resulting facet-adjusted estimates provide a more diagnostic and context-sensitive characterization of tutoring performance than raw means or endorsement rates, clarifying when our LLM-based chatbot performs well and where it breaks down across pedagogically meaningful dimensions.

Our work builds on a small but growing set of Rasch-family applications to LLM and chatbot evaluation, which largely follow two patterns. One line of work treats a chatbot as an "examinee" and applies Rasch analysis to place chatbots and test items on a common proficiency scale, often using domain testing formats such as medical or STEM multiple-choice benchmarks [6]. While useful for benchmarking performance relative to item difficulty, this approach typically focuses on correctness in fixed-response tasks and does not directly address human rater-mediated judgments of open-ended tutoring quality. A second line of work uses many-facet Rasch models in rating systems, particularly when an LLM functions as an automated scorer or rater, with the primary goal of quantifying severity, fit, and bias relative to human raters [60, 23, 26]. Related work extends MFRM to broader human-evaluation pipelines for AI-generated conversations by jointly modeling rater effects and contextual facets [34]. Our contribution is distinct from both benchmark scaling and rater auditing alone. We operationalize MFRM for rubric-scored, open-ended tutoring behaviors under authentic help-seeking conditions and provide a principled procedure for determining which contextual extensions add predictive value. In our data, PSIS-LOO comparisons favored a parsimonious explanatory specification that adds input sufficiency to the core facets, while topic-augmented models increased effective complexity without improving predictive fit (Table 3). This positions the MFRM not only as a scoring model, but as a validity-oriented evaluation layer that supports model checking, uncertainty quantification, and context-conditional interpretation.

Addressing RQ1, criterion difficulty results indicate that the chatbot's pedagogical strengths and weaknesses were not uniformly distributed across tutoring behaviors. Mistake diagnosis criteria, including Mistake Identification (i1) and Mistake Location (i2), were substantially more difficult to satisfy than correctness-oriented criteria, Accuracy of Guidance (i4) and Correctness of next steps (i5). Pedagogically, this pattern suggests that the system more reliably produces technically plausible guidance once it commits to a direction, but is less consistent in upstream tutoring behaviors that require diagnosing what the student did wrong and where the error originates. This asymmetry matters in programming help contexts because effective support typically depends on diagnosis before explanation [58]. Without correctly recognizing and localizing the mistake, fluent and confident guidance can be misaligned with the student's actual issue. From an evaluation standpoint, these findings also clarify why global aggregates can be misleading. If criteria differ substantially in difficulty, collapsing across rubric dimensions can hide critical weaknesses. For example, high endorsement for correctness-oriented criteria can coexist with systematic fragility in diagnosis. The Rasch framing makes this visible by placing criteria on a common logit scale and treating difficulty differences as interpretable measurement parameters [30, 52].

The estimated rater severity parameters were small and symmetric around zero, indicating minimal systematic rater bias after accounting for response quality and criterion difficulty. This is encouraging for rubric-based evaluation because it suggests that conclusions about criterion ordering and response-quality ranking are not driven by one rater being consistently more lenient or severe. At the same time, Cohen's κ was low-to-moderate across dimensions while exact-match agreement was comparatively high. This divergence is consistent with the κ paradox under endorsement imbalance, where high observed agreement can coexist with depressed chance-corrected agreement [15, 9]. In this setting, the MFRM approach is particularly useful because it does not assume perfect agreement. Instead, it uses both raters' codes to estimate and adjust for severity on the same latent scale and propagates uncertainty into criterion-level and response-level estimates.

Addressing RQ2, input sufficiency effects were positive across criteria, with larger posterior mean shifts for the accuracy-oriented criteria (i4–i5). Conceptually, this direction is expected. Richer student context, such as code, error traces, and clear problem statements, should increase the probability that the chatbot can respond appropriately and that raters can evaluate the response favorably. Although uncertainty around these effects remained nontrivial, likely because only a small fraction of retained exchanges were labeled insufficient (about 5%), the PSIS-LOO comparison provided modest evidence that adding sufficiency improves predictive adequacy relative to a facets-only baseline (Table 3). This yields a practical implication for evaluation design. To estimate sufficiency effects more precisely, future datasets may require higher prevalence of insufficient queries or sampling strategies that deliberately balance sufficient and insufficient contexts. More broadly, these results support a context-conditional view of tutoring effectiveness. Observed quality reflects both system capability and what students provide, and a psychometric evaluation layer that conditions on query quality helps separate these components and makes evaluation conclusions more transportable across settings with different help-seeking behavior distributions.

Addressing RQ3, we explored whether question content was associated with criterion-level endorsements using multi-label topic indicators. Topic-conditional patterns suggested that some categories were associated with higher endorsement for diagnosis behaviors, while at least one category was associated with lower endorsement for providing guidance. However, topic-augmented specifications did not improve PSIS-LOO and increased effective complexity, so these findings are best interpreted as descriptive and hypothesis-generating rather than as confirmatory moderation effects (Table 3). Even so, they point to a practical consideration for computing education and LLM tutoring. The same system can appear effective for some help-seeking contexts and less effective for others, and reporting criterion-level outcomes helps make this heterogeneity visible without collapsing everything into a single global score. Future work with larger samples or targeted designs can more decisively test topic moderation and evaluate whether structured interfaces can shift students toward contexts where the tutor behaves more reliably.

The comparison to automated metrics is best interpreted as construct coverage evidence rather than a competition for a single "best" score. We observed only modest alignment between common reference-free metrics and rubric targets, and the correlation structure differed by rubric dimension. In particular, automated metrics captured limited signal for correctness-focused criteria, suggesting that surface form, overlap, and prompt-response relatedness measures do not reliably recover pedagogically consequential judgments about factual correctness and appropriateness of next steps. Taken together, these findings motivate the continued role of expert rubric-based evaluation in educational settings and clarify where automated evaluation may complement, but not replace, pedagogical scoring.

Overall, our results support a context-sensitive interpretation of educational LLM-based chatbot effectiveness. Pedagogical quality depends on what the system can do, what students provide, and which instructional behaviors are being evaluated. By combining expert rubric judgments with a Bayesian explanatory MFRM and explicitly validating model extensions with PSIS-LOO, the proposed approach offers a principled path toward uncertainty-aware, diagnostically meaningful evaluation of LLM-based tutors in authentic learning settings.

6. CONCLUSION AND LIMITATIONS

Several limitations should be considered when interpreting the findings. First, the study examined a single course context, a single programming task (the "Big Three"), and a single tutoring system configuration; generalization to other topics, courses, or tutor designs requires additional evidence. Second, the analysis focused on single-turn exchanges. Multi-turn tutoring introduces additional facets, such as dialogue history, repair trajectories, and evolving student understanding, which may change both rubric constructs and model structure. Future evaluations may benefit from rubric refinements or sampling strategies that ensure adequate variability across criteria. Finally, input sufficiency was relatively rare within the retained sample (5%), which limits precision for estimating sufficiency effects and suggests the need for intentional sampling or larger datasets to more sharply quantify context effects.

This work suggests several directions for improving both evaluation practice and tutoring design. For evaluation, future studies can extend the framework to (i) ordinal scoring (rather than binary endorsement), (ii) additional facets such as rubric-dimension severity differences by rater, and (iii) multilevel structures that account for student-level or prompt-level clustering. For tutor design, the strongest need indicated by criterion difficulty is improved mistake diagnosis: systems may benefit from structured prompting that elicits minimal reproducible examples, explicit error traces, or targeted clarifying questions. Finally, because effectiveness varies by question type, future work should explore adaptive interfaces that guide students toward higher information queries and monitor when the tutor is operating outside its reliable regimes.

7. REFERENCES

[1] B. Abeysinghe and R. Circi. The challenges of evaluating llm applications: An analysis of automated,

human, and llm-based approaches. *arXiv preprint arXiv:2406.03339*, 2024.

[2] Y. Albadarin, M. Saqr, N. Pope, and M. Tukiainen. A systematic literature review of empirical research on chatgpt in education. *Discover Education*, 3(1):60, 2024.

[3] B. Alsafari, E. Atwell, A. Walker, and M. Callaghan. Towards effective teaching assistants: From intent-based chatbots to llm-powered teaching assistants. *Natural Language Processing Journal*, 8:100101, 2024.

[4] Z. Ashktorab, M. Desmond, Q. Pan, J. M. Johnson, M. Santillan Cooper, E. M. Daly, R. Nair, T. Pedapati, S. Achintalwar, and W. Geyer. Aligning human and llm judgments: Insights from evalassist on task-specific evaluations and ai-assisted assessment strategy preferences, 2024.

[5] A. Birillo, E. Artser, A. Potriasaeva, I. Vlasov, K. Dzialets, Y. Golubev, I. Gerasimov, H. Keuning, and T. Bryksin. One step at a time: Combining LLMs and static analysis to generate next-step hints for programming tasks. In *Proceedings of the 24th Koli Calling International Conference on Computing Education Research*, pages 9:1–9:12. ACM, 2024.

[6] Y. Chang et al. Evaluating chatbots in psychiatry: Rasch-based insights into clinical knowledge and reasoning. *PLOS ONE*, 2025.

[7] Y. Chang, S.-S. Huang, W.-Y. Hsu, and Y.-C. Liu. Evaluating chatbots in psychiatry: Rasch-based insights into clinical knowledge and reasoning. *PLOS ONE*, 2025.

[8] G. Cheng, W. Wong, L. Luo, and M. Yu. Integrating a scaffolding-based, llm-driven chatbot into programming education: A university case study. In *2025 International Symposium on Educational Technology (ISET)*, pages 1–5. IEEE, 2025.

[9] D. V. Cicchetti and A. R. Feinstein. High agreement but low kappa: II. resolving the paradoxes. *Journal of Clinical Epidemiology*, 43(6):551–558, 1990.

[10] R. Coppola and L. Ardito. Quality assessment methods for textual conversational interfaces: a multivocal literature review. *Information*, 12(11):437, 2021.

[11] N. Daheim, J. Macina, M. Kapur, I. Gurevych, and M. Sachan. Stepwise verification and remediation of student reasoning errors with large language model tutors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8386–8411, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics.

[12] S. K. Dam, C. S. Hong, Y. Qiao, and C. Zhang. A complete survey on llm-based ai chatbots. *arXiv preprint arXiv:2406.16937*, 2024.

[13] T. Debets, S. K. Banihashem, D. Joosten-Ten Brinke, T. E. Vos, G. M. de Buy Wenniger, and G. Camp. Chatbots in education: A systematic review of objectives, underlying technology and theory, evaluation criteria, and impacts. *Computers & Education*, page 105323, 2025.

[14] L. Favero, J. A. Pérez-Ortiz, T. Käser, and N. Oliver. Enhancing critical thinking in education by means of a socratic chatbot. In *International workshop on AI in education and educational research*, pages 17–32. Springer, 2024.

[15] A. R. Feinstein and D. V. Cicchetti. High agreement but low kappa: I. the problems of two paradoxes. *Journal of Clinical Epidemiology*, 43(6):543–549, 1990.

[16] Z. Gao and J. Jiang. Evaluating human-AI hybrid conversational systems with chatbot message suggestions. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, 2021.

[17] S. Goren, O. Kalinsky, T. Stav, Y. Rapoport, Y. Fairstein, R. Yazdi, N. Cohen, A. Libov, and G. Kushilevitz. Chai-tea: A benchmark for evaluating autocompletion of interactions with llm-based chatbots. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 18–32, 2025.

[18] S. Groothuijsen, A. van den Beemt, J. C. Remmers, and L. W. van Meeuwen. Ai chatbots in programming education: Students' use in a scientific computing course and consequences for learning. *Computers and Education: Artificial Intelligence*, 7:100290, 2024.

[19] S. Groothuijsen, A. van den Beemt, J. C. Remmers, and L. W. van Meeuwen. Ai chatbots in programming education: Students' use in a scientific computing course and consequences for learning. *Computers and Education: Artificial Intelligence*, 7:100290, 2024.

[20] M. Hmoud, H. Swaity, E. Anjass, and E. M. Aguaded-Ramírez. Rubric development and validation for assessing tasks' solving via ai chatbots. *Electronic Journal of e-Learning*, 22(6):1–17, 2024.

[21] S. Hobert. How are you, chatbot? evaluating chatbots in educational settings–results of a literature review. *DELFI 2019*, pages 259–270, 2019.

[22] J. Jiang and N. Ahuja. Response quality in human-chatbot collaborative systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1545–1548, 2020.

[23] H. Jiao, D. Song, and W.-C. Lee. Comparing human and AI rater effects using the many-facet rasch model, 2025.

[24] I. Jurenka, M. Kunesch, K. R. McKee, D. Gillick, S. Zhu, S. Wiltberger, S. M. Phal, K. Hermann, D. Kasenberg, A. Bhoopchand, et al. Towards responsible development of generative ai for education: An evaluation-driven approach. *arXiv preprint arXiv:2407.12687*, 2024.

[25] M. Kazemitabaar, R. Ye, X. Wang, A. Z. Henley, P. Denny, M. Craig, and T. Grossman. Codeaid: Evaluating a classroom deployment of an LLM-based programming assistant that balances student and educator needs. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. ACM, 2024.

[26] K. Kim, J. H. Lee, and D. Shin. The potential advantages of using an llm-based chatbot for automated writing evaluation for english teaching practitioners. *Language Learning & Technology*, 29(1):1–12, 2025.

[27] C. Koyuturk, E. Theophilou, S. Patania, G. Donabauer, A. Martinenghi, C. Antico, A. Telari, A. Testa, S. Buršić, F. Garzotto, et al. Understanding learner-llm chatbot interactions and the impact of prompting guidelines. In *International Conference on Artificial Intelligence in Education*, pages 364–377. Springer, 2025.

[28] V. Kumar, P. Srivastava, A. Dwivedi, I. Budhiraja, D. Ghosh, V. Goyal, and R. Arora. Large-language-models (llm)-based ai chatbots: Architecture, in-depth analysis and their performance evaluation. In *International Conference on Recent Trends in Image Processing and Pattern Recognition*, pages 237–249. Springer, 2023.

[29] M. Liffiton, B. Sheese, J. Savelka, and P. Denny. Codehelp: Using large language models with guardrails for on-demand programming assistance. In *Proceedings of the 23rd Koli Calling International Conference on Computing Education Research*. ACM, 2023.

[30] J. M. Linacre. *Many-Facet Rasch Measurement*. MESA Press, Chicago, IL, 1989.

[31] K. K. Maurya, K. V. A. Srivatsa, K. Petukhova, and E. Kochmar. Unifying AI tutor evaluation: An evaluation taxonomy for pedagogical ability assessment of LLM-powered AI tutors. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2025)*, 2025.

[32] H. D. Nguyen, T.-V. Tran, X.-T. Pham, A. T. Huynh, V. T. Pham, and D. Nguyen. Design intelligent educational chatbot for information retrieval based on integrated knowledge bases. *IAENG International Journal of Computer Science*, 49(2), 2022.

[33] S. A. Nikou, A. Guliya, S. Van Verma, and M. Chang. A generative artificial intelligence empowered chatbot: system usability and student teachers' experience. In *International Conference on Intelligent Tutoring Systems*, pages 330–340. Springer, 2024.

[34] C. Niu, K. Bradley, B. Ma, B. Waltman, L. Cossette, and R. Jin. Bias and reliability in AI safety assessment: Multi-facet rasch analysis of human moderators. In *Proceedings of AI in Measurement and Education Conference (AIMEcon)*, pages 393–397, 2025.

[35] W. S. Nsaif, H. M. Salih, H. H. Saleh, and B. T. Al-Nuaimi. Chatbot development: Framework, platform, and assessment metrics. *The Eurasia Proceedings of Science Technology Engineering and Mathematics*, 27:50–62, 2024.

[36] C. W. Okonkwo and A. Ade-Ibijola. Chatbots applications in education: A systematic review. *Computers and Education: Artificial Intelligence*, 2:100033, 2021.

[37] H. Pauzi, S. Dodman, and M. Mavrikis. Automating pedagogical evaluation of llm-based conversational agents in educational settings. In *Proceedings of the 2nd Workshop on Automatic Evaluation of Learning and Assessment Content (EvalLAC 2025), co-located with AIED 2025*, volume 4006 of *CEUR Workshop Proceedings*, 2025.

[38] H. Pauzi, S. Dodman, and M. Mavrikis. Automating pedagogical evaluation of llm-based conversational agents in educational settings. In *Proceedings of the 2nd Workshop on Automatic Evaluation of Learning and Assessment Content (EvalLAC 2025), co-located with AIED 2025*, volume 4006 of *CEUR Workshop Proceedings*, 2025.

[39] D. Peras. Chatbot evaluation metrics. *Economic and Social Development: Book of Proceedings*, pages 89–97, 2018.

[40] K. Petukhova and E. Kochmar. Intent matters: Enhancing AI tutoring with fine-grained pedagogical intent annotation. In E. Kochmar, B. Alhafni, M. Bexte, J. Burstein, A. Horbach, R. Laarmann-Quante, A. Tack, V. Yaneva, and Z. Yuan, editors, *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 860–872, Vienna, Austria, July 2025. Association for Computational Linguistics.

[41] A. Piscitelli, G. Costagliola, M. De Rosa, and V. Fuccella. Influence of large language models on programming assignments–a user study. In *Proceedings of the 2024 the 16th International Conference on Education Technology and Computers*, pages 33–38, 2024.

[42] R. Primi, P. J. Silvia, E. Jauk, and M. Benedek. Applying many-facet rasch modeling in the assessment of creativity. *Psychology of Aesthetics, Creativity, and the Arts*, 13:176–186, 2019.

[43] M. M. Rahman and Y. Watanobe. Chatgpt for education and research: Opportunities, threats, and strategies. *Applied sciences*, 13(9):5783, 2023.

[44] S. Rohini and T. Sethukarasi. Chatbots-based systematic review for student learning and engagement with llms. In *2025 3rd International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, pages 1978–1983. IEEE, 2025.

[45] L. M. Sánchez-Ruiz, S. Moll-López, A. Nuñez-Pérez, J. A. Moraño-Fernández, and E. Vega-Fleitas. Chatgpt challenges blended learning methodologies in engineering education: A case study in mathematics. *Applied Sciences*, 13(10):6039, 2023.

[46] J. Sedoc, D. Ippolito, A. Kirubarajan, J. Thirani, L. Ungar, and C. Callison-Burch. ChatEval: A tool for chatbot evaluation. In W. Ammar, A. Louis, and N. Mostafazadeh, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 60–65, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[47] B. Sheese, M. Liffiton, J. Savelka, and P. Denny. Patterns of student help-seeking when using a large language model-powered programming assistant. In *Proceedings of the 26th Australasian Computing Education Conference (ACE '24)*, pages 49–57. ACM, 2024.

[48] D. Song, E. Y. Oh, and M. Rice. Interacting with a conversational agent system for educational purposes in online courses. In *2017 10th international conference on human system interactions (HSI)*, pages 78–82. IEEE, 2017.

[49] S. Stefanova and Y. Kalmukov. Analysis of the possibilities of using llm chatbots for solving course and exam tasks. *International Journal of Advanced Computer Science & Applications*, 16(8), 2025.

[50] A. Tack, E. Kochmar, Z. Yuan, S. Bibauw, and C. Piech. The BEA 2023 shared task on generating AI teacher responses in educational dialogues. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 785–795, Toronto, Canada, July 2023. Association for Computational Linguistics.

[51] A. Tack and C. Piech. The AI teacher test: Measuring the pedagogical ability of blender and GPT-3 in educational dialogues. 2022.

[52] M. Tavakol, G. Pinner, and R. Dennick. Using the many-facet rasch model to analyse and evaluate clinical assessment. *International Journal of Medical Education*, 10:126–131, 2019.

[53] R. Tsang and S. Y. WOOD. Help or hype? exploring llm-based chatbots in self-regulated learning. In *2025 ASEE Annual Conference & Exposition*, 2025.

[54] M. Uto, J. Tsuruta, K. Araki, and M. Ueno. Item response theory model highlighting rating scale of a rubric and rater–rubric interaction in objective structured clinical examination. *PLOS ONE*, 19(9):e0309887, 2024.

[55] M. Verleger and J. Pembridge. A pilot study integrating an ai-driven chatbot in an introductory programming course. In *2018 IEEE frontiers in education conference (FIE)*, pages 1–4. IEEE, 2018.

[56] M. Verleger and J. J. Pembridge. A pilot study integrating an AI-driven chatbot in an introductory programming course. In *Proceedings of the 2018 IEEE Frontiers in Education Conference (FIE)*, pages 1–4, San Jose, CA, USA, 2018. IEEE.

[57] P. Wang, K. Yin, M. Zhang, Y. Zheng, T. Zhang, Y. Kang, and X. Feng. The effect of incorporating large language models into the teaching on critical thinking disposition: An "ai+ constructivism learning theory" attempt. *Education and Information Technologies*, pages 1–23, 2025.

[58] T.-T. Wu, H.-Y. Lee, P.-H. Chen, W.-S. Wang, and Y.-M. Huang. Integrating computational thinking diagnostic mechanism and reflective learning: An innovative approach to enhance learning outcomes in introductory programming. *Journal of Computer Assisted Learning*, 41(5):e70121, 2025.

[59] T. Yamashita. An application of many-facet rasch measurement to evaluate automated essay scoring: A case of chatgpt-4.0. *Research Methods in Applied Linguistics*, 3(3):100133, 2024.

[60] T. Yamashita. An application of many-facet rasch measurement to evaluate automated essay scoring: A case of ChatGPT-4.0. *Assessing Writing*, 2024. Article in press/online first on ScienceDirect.

[61] D. Yigci, M. Eryilmaz, A. K. Yetisen, S. Tasoglu, and A. Ozcan. Large language model-based chatbots in higher education. *Advanced Intelligent Systems*, 7(3):2400429, 2025.

[62] Y. Zhang, E. L. Ouh, A. Ho, S. L. Lo, K. W. Tan, and F. Lin. Prompttutor: Effects of an llm-based chatbot on learning outcomes and motivation in flipped classrooms. In *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, pages 445–451, 2025.

APPENDIX
A. DETAILED EVALUATION RUBRIC

i1: Mistake Identification

Guiding Question: *Did the response correctly identify or recognize a mistake in the student's original response?*

This category assesses whether the chatbot response explicitly or implicitly acknowledges **all** errors in the student's answer. Recognition can be **direct** (e.g., pointing out what is incorrect) or **indirect** (e.g., offering a correction or prompting the student to revisit a specific part). When multiple errors are present, it is expected to recognize **each** of them.

- **Yes:** The response correctly identifies all mistakes present in the context, evaluating both the issue and the code.
- **Partially:** The response identifies some mistakes but misses others, or identifies them incompletely.
- **No:** The response fails to identify any mistakes, or incorrectly labels correct elements as mistakes.

Rater Tip: Focus on whether the response shows awareness of the student's mistake, not on how thorough the explanation is or whether the correction is accurate.

i2: Mistake Location

Guiding Question: *Did the response accurately point to a genuine mistake and its location?*

This category assesses whether the chatbot correctly indicates where each identified mistake occurs. For **each** error it recognizes, it should clearly and correctly indicate the relevant portion of the response. Line numbers are not required if the referenced location is unambiguous.

- **Yes:** The response accurately pinpoints the location of all identified mistakes.
- **Partially:** The response locates some, but not all, mistakes. If the response does not explicitly cite a line number but the intended location is clear (e.g., "in the copy assignment operator"), count this as partial.
- **No:** The response fails to locate the mistakes or points to incorrect or nonexistent locations.

Rater Tip: For each mistake identified, check whether the chatbot points to the corresponding location with enough specificity for a student to find it. If the response misses mistakes or misidentifies them, this category should generally be rated **Partially** or **No**.

i3: Providing Guidance

Guiding Question: *Does the chatbot offer guidance that is relevant and responsive to the student's issue?*

This category assesses whether the chatbot offers guidance, such as explanations, elaborations, tips, hints, and examples, that is both relevant and responsive to the issues provided and identified in the context. Fully appropriate guidance clearly addresses the issue without introducing unrelated information, omitting the student's request, or becoming overly generic, taliored to the students' needs.

- **Yes:** The response directly addresses the student's question(s) in context and provides guidance that helps remediate the identified mistake(s).
- **Partially:** The response provides incomplete guidance or includes distracting, tangential information that adds unnecessary cognitive burden.
- **No:** The response provides no guidance or guidance that is largely irrelevant to the student's issue.

Rater Tip: If the response misses part of the question or provides an irrelevant answer, rate **Partially** or **No** depending on the extent of the mismatch.

i4: Accuracy of Guidance

Guiding Question: *Is the guidance provided by the chatbot factually and conceptually correct?*

This category evaluates whether the chatbot's guidance is free from factual inaccuracies and conceptual misunderstandings. A fully appropriate response demonstrates correct understanding of the concepts, procedures, or principles and presents them without errors or misleading statements.

- **Yes:** Guidance is factually and conceptually correct.
- **Partially:** Guidance is mostly correct but contains minor inaccuracies, oversimplifications, or omissions that may reduce correctness or lead to misunderstandings.
- **No:** The guidance contains major errors, is misleading, or reflects conceptual misunderstandings.

Rater Tip: Focus on correctness of claims and reasoning. Do not penalize for brevity if the guidance is accurate.

i5: Correctness of Next Steps

Guiding Question: *Are the suggested next steps or corrections accurate and appropriate to the student's problem?*

This category evaluates whether the chatbot's proposed corrections or next steps are factually accurate, conceptually sound, and well matched to the student's specific error or question. Suggestions should be correct in principle and appropriate to the given context so that following them would resolve the problem without introducing new errors.

- **Yes:** The suggested corrections or next steps are accurate and directly appropriate to the student's problem.
- **Partially:** Suggestions are generally on the right track but are incomplete, partially misapplied, or would require important missing steps to work reliably.
- **No:** Suggestions or next steps are inaccurate, infeasible, unsafe, or not aligned with the student's problem.

Rater Tip: Evaluate whether the proposed actions would solve the stated problem in the given context.

B. EXAMPLE LLM-CHATBOT RESPONSE PATTERN AND RUBRIC TRIGGERS

Table 9: Common chatbot response pattern observed in the dataset and the rubric criteria it typically triggers.

Typical response move	Primary rubric criteria triggered
Interpret the error signal (e.g., compiler/runtime message)	i1 Mistake Identification, i4 Accuracy of Guidance
Locate the source of the error in the student code	i2 Mistake Location
Explain why the error occurs and connect it to the student's implementation	i3 Providing Guidance; i4 Accuracy of Guidance
Provide actionable remediation steps (what to change and how)	i5 Correctness of Next Steps
Address additional pitfalls or edge cases (e.g., self-assignment)	i1–i4 (as applicable); i5 if a fix is proposed
Close with general best-practice reminders (e.g., Rule of Three)	Typically supports i3; may be neutral if not tied to the specific code

Annotated example (representative response):
In the dataset, the chatbot typically provides a consolidated explanation of the student's error and offers remediation steps. Responses often follow a recurring structure, which allows the rubric to be applied consistently. Below is one representative example, with each paragraph annotated by the rubric criteria it primarily triggers.

Chatbot response (example):

[**i1: Mistake Identification**]
The error message you're seeing from Valgrind indicates that there's an "Invalid write of size 8," which means that your program is trying to write to a memory location that it shouldn't. This is often due to writing to memory that has already been freed or is out of bounds of the allocated array.

[**i2: Mistake Location; i3: Providing Guidance; i4: Accuracy of Guidance**]
In your copy constructor and copy assignment operator, you're allocating a new array and then copying the elements from the `other` array. However, after repointing `arr` to the new array, you immediately delete the array that `arr` points to. This is a mistake because you're deleting the same memory you've just allocated and copied data into. As a result, when you try to access or write to this memory later, you get the invalid write error because the memory has been freed.

[**i5: Correctness of Suggestions; Clarity of Next Steps**]
To fix this, you should only delete the old array after you've done everything else, including repointing the `arr` pointer to the new array. Also, make sure that you're not leaking the previous array that `arr` pointed to before you assign it to the new array. You need to save the old pointer, assign the new array to `arr`, and then delete the old array.

Figure 5. Annotated example illustrating a common chatbot response structure and the rubric criteria it triggers.

CantoTalk: Probing Teacher Expertise From Fine-Tuned Talk Move Representations

Mayank Sharma*
Stanford University
masharma@stanford.edu
Xinman Liu*
Stanford University
xinman@stanford.edu
Gordon Wing-Leung Yeung
Stanford University
wlyeung@stanford.edu

ABSTRACT

Classroom discourse profoundly shapes student learning, yet analyzing teacher talk at scale remains challenging in non-Western and low-resource language contexts. This paper introduces CantoTalk, a dataset of 7,518 Cantonese teacher utterances from Hong Kong mathematics classrooms annotated with ten talk-move categories. We investigate whether LLMs can reliably classify these moves and encode systematic differences in teacher expertise. Fine-tuning five open-weight LLMs yields strong performance, with the best model (Qwen3-8B) achieving micro-F1 of 0.81 and macro-F1 of 0.77. Probing utterance-level embeddings reveals that teacher expertise is linearly separable with 0.79 balanced accuracy, well above chance even after controlling for surface features. Clustering analyses uncover three coherent discourse styles differing in pedagogical authority, scaffolding, and dialogic engagement, with qualitative analysis showing systematic differences in how experienced and novice teachers execute similar talk moves. These findings demonstrate that fine-tuned LLM representations capture teacher expertise, offering a new lens for analyzing classroom discourse and informing teacher feedback tools.

Keywords

Talk Move Classification, LLM Fine-tuning, Representation Learning, Teacher Expertise Detection, Low-Resource NLP, Educational Discourse Analysis

1. INTRODUCTION

Classroom discourse profoundly shapes student learning and is influenced by pedagogical traditions and cultural norms [1, 2]. Yet in typical classrooms, students contribute far less than teachers: they speak for only 27 seconds per hour of teacher talk and ask just 2 academic questions per day, compared to teachers' 142 questions [3]. Ensuring equitable classroom talk therefore depends on teachers' ability to facilitate discourse that invites reasoning, dialogue, and participation from all learners [4]. To support these goals, researchers have developed structured "talk-move" frameworks to help teachers better engage students in class and achieve their instructional objectives [5, 6]. Traditional analysis of these talk moves, however, has relied on labor-intensive manual coding methods that cannot scale to support widespread teacher development or classroom research [4].

Recent advances in large language models (LLMs) offer the potential to automate the identification of talk moves and provide immediate actionable insights to teachers [7]. However, classroom discourse remains largely an out-of-distribution domain for foundational models, which exhibit biases and misalignment to learning outcomes [8]. While there is emerging work on enhancing talk-move analysis and aligning models with pedagogical frameworks [9, 10], most studies focus on Western, English-speaking corpora and remain limited to classification tasks and frequency counts. Such analyses overlook cross-cultural adaptations of talk moves [11], the challenges of leveraging LLMs in low-resource language settings [12], as well as other contextual factors such as how teacher expertise influence discourse patterns in classroom [13].

This study therefore aims to address these gaps by examining how LLMs can identify and interpret talk moves in the authentic, low-resource setting of Hong Kong mathematics classrooms. We investigate two key research questions with CantoTalk[1]: **(1) Can LLMs be fine-tuned to perform reliable talk move classification in Cantonese classroom discourse?** and **(2) How do the distributions and linguistic realizations of talk moves differ between novice and experienced teachers, and can these patterns be uncovered from fine-tuned representations of the LLMs?**

2. RELATED WORK

*Equal contribution.

Mayank Sharma, Xinman Liu, and Gordon Yeung. CantoTalk: Probing Teacher Expertise From Fine-Tuned Talk Move Representations. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 44–56. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039963

[1]GitHub Repository: https://github.com/matrix-mayank/canto-talk

2.1 Talk Moves in Mathematics Classrooms

"Talk moves" are utterance-level strategies used by teachers and students to elicit, clarify, or extend thinking in service of productive, dialogic learning [14]. Within mathematics instruction, these moves help navigate the tension between open-ended exploration and specific instructional objectives [5, 15]. Various frameworks have operationalized these strategies for analysis. Michaels and O'Connor [14] established a four-cluster taxonomy organized by instructional goals: encouraging students to expand their contribution, listen carefully to one another, make their reasoning explicit, and engage with others' ideas. This "Talk Moves as Tools" (TMT) framework has been widely adapted and extended. For instance, Suresh et al. [7] outline six categories: keeping everyone together, getting students to relate, restating, revoicing, pressing for accuracy, and pressing for reasoning. Research also suggests that the quality of these moves directly impacts engagement. Alic et al. [16] distinguish between "funneling" questions that guide students toward predetermined answers versus "focusing" questions that deepen mathematical thinking; while Demszky et al. [17] also highlight the importance of teacher "uptake" of student contributions, i.e., a dialogic teaching practice that acknowledges and builds upon student ideas, in ensuring students feel heard and prompted to engage more deeply with mathematical concepts.

However, these frameworks largely focus on dynamics within English-speaking classrooms; yet, the privileging of individual student talk in Western research may inadvertently undervalue other legitimate forms of mathematical communication in cultures where norms around authority, participation, and learning are different [18]. For instance, Xu and Clarke [11] demonstrate through cross-cultural analysis of Shanghai, Seoul, and Melbourne classrooms that choral responses (whole-class unison replies to a teacher's prompt) and strategic use of silence can be equally effective for developing mathematical competence, challenging assumptions that individual verbal participation is the primary indicator of engagement. Similarly, Ng et al. [19] and Ni et al. [20] highlight that in the specific context of Hong Kong mathematics classrooms, talk moves such as "inviting choral response" and "teacher evaluation" are common culturally-specific practices that put more emphasis on teacher control in advancing student's knowledge of curricular content.

2.2 Analysis of Classroom Discourse

Despite their importance, most assessments of talk moves rely on manual observation that is resource-intensive and difficult to scale [4]. Advances in computational approaches offer the potential to automate the identification and analysis of talk moves at scale [7]. With transformer-based language models, the field has progressed rapidly from early studies using small, curated datasets like TalkMoves to more robust systems trained on extensive classroom corpora such as NCTE-119 and SAGA-22 [7, 10]. The transition from Bi-LSTM architectures to BERT-based models has yielded substantial performance improvements, with reported F1-scores rising from 65% to exceeding 82% [21, 10]. More recent developments in LLMs and generative AI present new opportunities for automated feedback systems and talk move classification. While earlier studies have shown that task-specific BERT-based architectures outperform general-purpose LLMs like ChatGPT for talk move classification—a gap attributed to BERT variants learning discourse-specific cues through fine-tuning, such as turn-level patterns and move co-occurrences, that zero-shot LLMs lack [8]—the latest research suggests that fine-tuning LLMs significantly alters this landscape. For example, fine-tuned GPT-3.5-turbo models have been shown to outperform state of the art RoBERTa classifiers in identifying tutor talk moves, achieving a macro F1 score of 0.87 compared to 0.78, highlighting the critical role of fine-tuning and leveraging context (such as providing long conversation segments) to adapt LLMs effectively for the discourse analysis tasks on large-scale mathematics classroom corpora [9].

However, these advances are unevenly distributed, where developing NLP applications for low-resource languages remains challenging due to the scarcity of annotated datasets and the complexity of transferring linguistic rules from high-resource languages [22, 23]. For example, Ng et al. [24] noted that standard linguistic markers for authority (such as modal verbs) in English do not map directly to Cantonese classroom discourse, complicating the use of out-of-the-box English models. Recent benchmarks confirm that while top-tier proprietary models (e.g., GPT-4o) show emerging competence, they still exhibit significant gaps in culturally nuanced understanding and morphological generalization compared to English, with smaller open-source models lagging substantially in both fluency and accuracy when processing Cantonese colloquialisms [12]. Consequently, effective implementation requires two forms of specialization: linguistic, as Cantonese colloquialisms are under-represented in multilingual corpora [12]; and pedagogical, as culturally specific moves like *Request for Choral Response* and *Evaluation* function as authority-reinforcing practices absent from Western taxonomies [19], and thus cannot be recovered from general-purpose training alone.

2.3 Teacher Expertise and Authority

Understanding the difference between novice and experienced teachers requires examining how discourse establishes authority in the classroom: authority structures determine which talk moves are functionally available, since students are less likely to engage with reasoning demands from a teacher who has not yet established interactional authority [24]. Tong et al. [13] note that in traditional classrooms, particularly within Chinese contexts, discourse often follows an "Initiation-Response-Feedback" (IRF) pattern—where the teacher initiates a question, a student responds, and the teacher evaluates the response—which novice teachers tend to rely on to maintain control and efficiency. Experienced teachers, however, demonstrate more adaptive responsiveness, leveraging not only IRF but also "Initiation-Response-Feedback-Response" (IRFR), where the student is invited to react to the teacher's feedback, enabling a more dialogic exchange, as well as iterative initiation and response chains. Kosel et al. [25] similarly found that novice teachers tend to mostly evaluate student engagement in terms of easily observable "surface cues" (e.g., students hand-raising, digressive gaze) coupled with some cues regarding students' factual knowledge. In contrast, experienced teachers integrate a broader spectrum of not only "surface cues" but also "deep cues" (e.g., students' motivation, attentiveness, confidence, quality of verbal contributions) when diagnosing

student engagement and gauging instructional effectiveness. These findings point to the importance of accounting for not just the presence of talk moves, but how teachers configure authority and engage students within specific cultural and classroom norms.

3. DATA

The dataset used in this study is derived from classroom recordings originally collected by Ni et al. [26]. We refer to this annotated corpus as CantoTalk, which comprises 7,518 teacher utterances (M = 48.19, SD = 57.62 characters) and 7,088 student utterances (M = 7.36, SD = 10.60 characters), drawn from 64 recorded and transcribed lessons across 32 upper-primary mathematics teachers in 16 Hong Kong schools, with each teacher providing 2 lessons [26]. These utterances were labeled using a ten-category scheme designed by Ng et al. [19], including: *Ask for Expression*, *Say More*, *Revoice*, *Press for Reasoning*, *Repeat and/or Add On*, *Agree/Disagree*, *Explain Others*, *Request for Choral Response*, *Evaluation*, *Other Moves* (see Appendix A for full definition and examples).

The scheme is built upon established discourse frameworks, including Talk Moves as Tools (TMT), Classroom Discourse Analyser (CDA), authority theory, and accountability principles in productive disciplinary engagement [27, 14, 28, 29, 30], with the addition of *Request for Choral Response* and *Evaluation* to reflect the structured teacher-led nature of Hong Kong mathematics classrooms and the use of collective participation to reinforce curricular knowledge [19]. The dataset was manually annotated by the researchers of the original study, achieving inter-coder agreement ranging from 0.7 to 1.0 [26]. Appendix Table A provides a complete list of talk move labels, and their descriptions.

Teacher metadata includes years of teaching experience ranging from 1 to 23 years (see Figure 1), grade level(s) taught, and school context. We categorized teachers into novice ($\leq$ 5 years, n = 8, 2110 utterances), intermediate (6–9 years, n = 7, 1922 utterances), and experienced ($\geq$ 10 years, n = 17, 3506 utterances) based on experience thresholds established in the teacher development literature. For our expertise detection analyses, we excluded intermediate teachers to maximize contrast between groups, yielding a final dataset of 5,616 utterances with a 62.4% experienced to 37.6% novice ratio. We note that years of experience serves as a coarse proxy for instructional expertise, as experience correlates with expertise only probabilistically and nonlinearly. We adopt this operationalization pragmatically, as direct measures of instructional quality were not available in the dataset, and interpret all expertise-related findings with this caveat in mind.

This dataset, which we refer to as CantoTalk, including all teacher and student utterances with talk-move annotations, contextual windows, and teacher metadata (anonymized experience levels, grade levels taught) is hosted on the GitHub repository.

4. METHODS

We adopt a two-stage approach. First, we fine-tune a set of open-weight LLMs from different research groups and parameter scales to identify the best-performing model for talk-move classification. Second, using the top model, we extract utterance-level embeddings to compare discourse patterns between novice and experienced teachers. Teachers with 6–9 years were excluded from embedding comparison analyses. Note that we initially explored few-shot prompting using off-the-shelf LLMs, such as Deepseek-R1 and Gemini-2.5-Pro to classify teacher talk moves, but excluded this method from our final analysis as it yielded suboptimal performance ($F_1 < 0.35$) & proved computationally prohibitive, requiring over one hour per transcript.

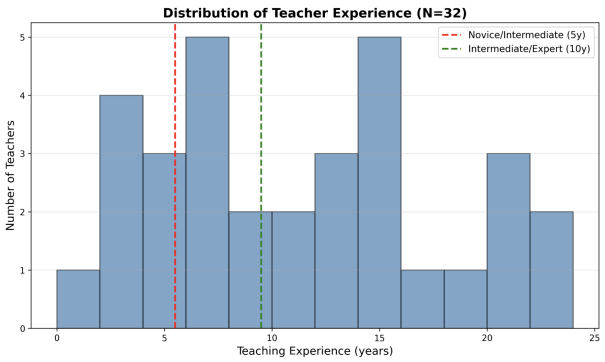


Figure 1: Distribution of teacher experience in years.

4.1 Fine-tuning

Training Data Construction. We perform a transcript-level split (70% train, 15% validation, and 15% test) to ensure that no utterance's immediate conversational context spans across splits, preventing direct context leakage between classroom sessions. Only teacher-labeled utterances are used. Because talk-move classification is strongly context dependent, each training instance is constructed using a ±6 utterance sliding window containing six preceding turns, the focal utterance, and six following turns, formatted as `<speaker>: <content>` pairs to preserve turn-taking structure. Examples without a full context window are discarded.

Prompt Format. All models receive prompts in a standardized chat format: a Cantonese (i.e., Traditional Chinese) instruction, the full set of talk-move definitions (Cantonese + English), and the contextual dialogue grouped into preceding, current, and following blocks, followed by a directive to output all applicable talk-move categories as a comma-separated list. Chat templates for the chosen models differ only in formatting tokens, and the informational content in all of them is identical. Full prompts are presented in Appendix B.

Models and Training Procedure. We fine-tune five open-weight LLMs, including: `Qwen3-4B-Instruct-2507`, `Qwen3-8B`, `Llama-3.1-8B-Instruct`, `DeepSeek-V3.1`, and `GPT-OSS-20B` using Tinker API [31]. All models are trained using LoRA adapters (rank = 32) with a batch size of 8 and a maximum input length of 4096 tokens. Adam optimization was used with a 300-step linear warm-up. The model trained for up to 3000 steps with early stopping on validation loss. Training hyperparameters are listed in Appendix C. To address severe skew in talk-move frequencies, we ap-

ply inverse-frequency loss weighting, assigning each label i weight $w_i = (1/f_i)^{0.5}$, normalized to have mean 1.0. For multi-label examples, we use the maximum weight among the associated labels. This weighting scheme amplifies the contribution of rare moves such as *Explain Others* and *Evaluation*.

Evaluation. Evaluation is conducted on held-out teacher utterances. Models generate label sequences using deterministic decoding (temperature = 0). Predictions are canonicalized by splitting on commas, trimming whitespace, and matching against the valid label inventory (see full details in Appendix D). We report: micro-F_1, micro precision, micro recall, macro-F_1 (averaged across labels) and per-label F_1 and support.

4.2 Embedding Analysis

Embedding Extraction. We analyze internal representations from our best-performing model to examine whether it encodes systematic differences between novice and experienced teachers. A well-known challenge in probing LMs is that pooling token embeddings over the entire prompt can entangle properties of the target utterance with artifacts from the surrounding instructions and definitions [32]. To avoid this, we extract target-only embeddings using character-level offset mappings.

For each example, we tokenize the full prompt (instruction + talk-move definitions + before/after context + target utterance) with `return_offsets_mapping=True`. We identify the character span of the target utterance using fixed delimiters (當前發言： ... 後文語境：) and select only the tokens whose offsets overlap this region. The embedding of the utterance is computed by mean-pooling the final-layer hidden states of these tokens. We use the last transformer layer because it typically captures task-specific refinements acquired during fine-tuning [33, 34]. This procedure yields a 4,096-dimensional embedding for all 7,538 utterances in the corpus. Extraction is performed via Colab Pro on an NVIDIA A100 using 8-bit quantization, enabling full processing in approximately 40 minutes.

Dimension Validation and Feature Selection. For expertise analyses, we retained only novice and experienced teachers, resulting in an embedding matrix of size $5{,}616 \times 4{,}096$. To identify dimensions associated with teacher expertise while avoiding overfitting, we divide them into a stratified 80/20 train-test split. On the training set, we conduct two-sample t-tests comparing experienced versus novice values across all 4,096 dimensions, applying a Bonferroni-corrected threshold ($\alpha = 1.22 \times 10^{-5}$). This yields 1,254 significant dimensions (30.6%). We re-test these dimensions on the held-out set at the same threshold; 348 dimensions (27.8%) remain significant. This two-stage procedure separates dimension validation from classifier training, enabling us to probe what the model encodes rather than just maximize classification accuracy, while guarding against dimensions that are spuriously significant in the training set alone. All downstream analyses use only these validated dimensions.

Expertise Classification. Overall, we train a linear probe implemented as an ℓ_2-regularized logistic regression classifier (`solver = lbfgs`, $C = 1.0$, `max_iter = 1000`) with balanced class weights to correct for class imbalance, using the validated 348-dimensional embeddings as features. All inputs are standardized (z-scored). We conduct 5-fold stratified cross-validation on the training set ($n = 4{,}492$) and evaluate the final model on the test set ($n = 1{,}124$). We report accuracy, balanced accuracy, precision, recall, and F_1. This analysis identifies whether the model's internal representations reliably encode distinctions between novice and experienced teachers. On a talk-move level, to test whether expertise is uniformly encoded across discourse practices, we repeat the probe analysis for each of the nine primary talk-move categories (excluding *Other Moves* as it contains varied moves). For each move, we subset the data to utterances containing that move, then train the same logistic regression classifier with 5-fold CV. We evaluate significance by one-sample t-tests against chance ($\mu = 0.5$) using balanced accuracy as the metric. This analysis identifies which pedagogical moves are most diagnostic of teacher expertise.

Clustering. To explore latent discourse patterns beyond binary expertise classification, we perform k-means clustering on the 348-dimensional validated embeddings. We compare solutions for k = 2–10 using silhouette [35], Calinski-Harabasz [36], and Davies-Bouldin [37] scores to choose k = 3. The association between cluster membership and expertise is tested with a χ^2 test, with structure visualized using UMAP [38] (`n_neighbors = 15`, `min_dist = 0.1`). To interpret the discovered clusters, we compute descriptive properties of each cluster such as proportion of experienced teachers, mean utterance length, and talk moves in it.

We further assess linguistic and pedagogical differences across clusters through several analyses. First, we compare utterance lengths using a one-way ANOVA and report η^2 effect sizes; post-hoc differences are examined with Bonferroni-corrected Mann-Whitney U tests. To evaluate whether cluster structure is driven by verbosity rather than discourse function, we train a shallow decision tree (`max_depth = 2`) to predict cluster membership using only utterance length, quantifying how much representational structure remains after accounting for this surface feature. Next, we conduct a qualitative comparison of experienced and novice utterances within each cluster to examine differences in execution quality beyond move frequency or length. These combined analyses triangulate the linguistic and pedagogical properties that define the discovered embedding clusters.

5. RESULTS

5.1 Fine-tuning

Table 1 reports test-set performance for all five fine-tuned LLMs. `Qwen3-8B` achieves the strongest results, obtaining the highest micro-F_1 (0.8145) and macro-F_1 (0.7687). Although `Qwen3-8B` outperforms `DeepSeek-V3.1` narrowly despite comparable size, we select `Qwen3-8B` for subsequent embedding analyses due to: (a) its slightly superior accuracy and (b) lower computational cost owing to its smaller size. `Qwen3-4B-Instruct-2507` and `Llama-3.1-8B-Instruct` form a middle tier, outperforming `GPT-OSS-20B` but trailing the top models by several points. `GPT-OSS-20B` underperforms across all metrics despite its larger parameter count.

Per-Label Performance. Table 2 shows per-label F_1 scores for `Qwen3-8B`. The model performs best on moves with strong

Table 1: Performance of FT models on talk-move classification

Rank	Model	F_1^{μ}	F_1^{M}	Prec.	Rec.
1	Qwen3-8B	**0.8145**	**0.7687**	0.8048	**0.8245**
2	DeepSeek-V3.1	0.8134	0.7401	0.8077	0.8193
3	Qwen3-4B-Instruct-2507	0.7959	0.7221	**0.8097**	0.7827
4	Llama-3.1-8B-Instruct	0.7927	0.7432	0.8020	0.7837
5	GPT-OSS-20B	0.5683	0.4164	0.7624	0.4529

Table 2: Per-label F_1 scores for talk-move classification by Qwen3-8B. Support indicates the number of test examples

Talk Move Label	F_1	Support
表達自己 (Ask for Expression)	0.823	620
對自己發言作補充 (Say More)	0.858	219
複述 (Revoice)	0.860	625
解釋自己 (Press for Reasoning)	0.742	353
重述及/或補充他人發言 (Repeat and/or Add On)	0.739	253
同意/否定 (Agree/Disagree)	0.788	90
解釋他人 (Explain Others)	0.545	5
集體作答 (Request for Choral Response)	0.884	444
評價 (Evaluation)	0.678	55
其他 (Other Moves)	0.768	396
Macro Average	**0.7687**	**3,060**

contextual cues and larger support, such as *Say More*, *Revoice*, and *Request for Choral Response*, each reaching F_1 scores above 0.85. In contrast, the lowest F_1 scores correspond to categories with extremely sparse support. Noticeably, *Explain Others* has $F_1 = 0.545$ (5 instances in the test set), and *Evaluation* has $F_1 = 0.678$ (55 instances in the test set), showing reduced performance. These patterns reflect the long-tail distribution of pedagogical moves as well as their inherent semantic difficulty, which can make some moves harder to distinguish than others.

5.2 Embedding Analysis

We present results in two phases: first, expertise classification demonstrates that teacher expertise is linearly recoverable from the model's internal representations; second, cluster analysis reveals the pedagogical structure underlying these representations.

Expertise Classification. For overall set, a linear probe trained on 348 validated dimensions achieved 0.733 ± 0.016 balanced accuracy (5-fold CV) and 0.790 on held-out test data, significantly above chance (0.50). The model distinguished both experienced ($F_1 = 0.83$) and novice ($F_1 = 0.74$) discourse effectively, indicating that performance was not driven by class imbalance exploitation. These results establish that teacher expertise is linearly encoded within the model's representational geometry, a key finding suggesting that teacher experience leaves distinctive signatures in fine-tuned embeddings. We repeated probe analysis separately for each talk-move category to determine whether expertise signals vary by pedagogical function. All categories showed above-chance balanced accuracy (mean = 0.754; all > 0.70), but with meaningful variation. *Revoice* (0.783), which involves deeper interpretation of student thinking to paraphrase contributions, had the highest differentiation. In contrast, *Press for Reasoning* had the weakest differentiation (0.723). This pattern suggests the model captures expertise-sensitive nuances within all talk moves.

Cluster Discovery. K-means clustering on the 348 dimensions revealed three stable discourse clusters. A χ^2 test showed strong association with expertise ($\chi^2 = 256.7$, $p < 10^{-56}$), indicating that representational regions in the embedding space differ by teacher expertise. UMAP projections (Figure 2) reveal that the three clusters occupy largely distinct regions of the embedding space, with overlap concentrated at cluster boundaries, consistent with the natural co-occurrence of talk moves across discourse styles. To ensure that clusters truly reflected pedagogical function rather than surface-level verbosity, we conducted three validation tests. First, while clusters differed significantly in average length ($p < 10^{-145}$, $\eta^2 = 0.112$), a decision tree using only length as a predictor achieved just 65% accuracy in cluster assignment, misclassifying 35% of utterances and showing particular confusion between Clusters 2 and 3 despite their 38-character mean length difference (29.1 vs. 67.3 characters). Second, within each cluster, experienced and novice utterances showed no significant length differences (Cluster 1: $p = 0.23$; Cluster 2: $p = 0.67$; Cluster 3: $p = 0.18$), yet maintained distinct embedding patterns as evidenced by the linear probe's within-cluster classification performance. Third, experienced-novice differences in pedagogical moves persisted within each cluster even when length was held constant. These findings collectively demonstrate that embedding geometry captures deeper distinctions in discourse rather than mere stylistic artifacts.

Cluster Characterization. Through qualitative analysis of utterances within clusters, we found they capture qualitatively different instructional styles. Critically, these clusters are defined not just by which moves are used, but by *how* they are executed. The embedding space appears to align with well-established pedagogical constructs, including authority and accountability roles in classroom discourse, cognitive-demand frameworks, and distinctions between monologic, authoritative, and dialogic instructional patterns.

Cluster 1: Procedural Management Integrated with Conceptual Scaffolding ($n = 118$; experienced 78%; $M = 97.8$ chars). This cluster represents long, multi-functional turns where teachers intertwine classroom management and procedural explanations with mathematical content. We note that Cluster 1 is notably smaller than the other clusters and appears less stable in the UMAP projection, which likely reflects the relative rarity of long multi-functional turns in the corpus rather than a poorly-defined cluster; internal validation metrics (silhouette, Calinski-Harabasz, Davies-Bouldin) consistently supported $k = 3$ over alternative solutions. Experienced teachers frequently turn routine instructions into opportunities for conceptual elaboration. For instance, while managing a hands-on activity in which students were cutting paper shapes to explore symmetry, one experienced teacher turned a management moment into a mathematical one: *"No, this is not a symmetrical figure... don't be like Student 4 who cut it—once it's cut, it's no longer a parallelogram."* Rather than separating the procedural warning from the mathematical concept, the teacher

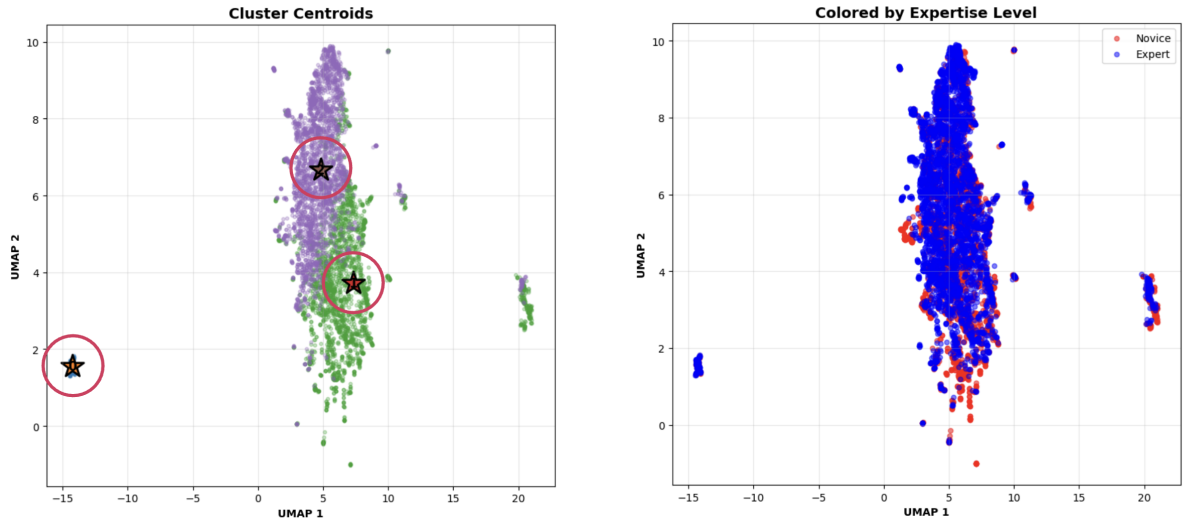


Figure 2: UMAP projection of utterance embeddings showing three discourse clusters ($k = 3$). Colors indicate cluster membership; clusters differ in expertise concentration (78%, 51%, 72%).

delivers both within a single turn. Even in explaining partitioning procedures, experts weave in conceptual reframing: *"If I split this into two equal parts, this is no longer just 'one'—what is it then?"*, inviting reconceptualization of unit size. By contrast, novice teachers issue procedural instructions as distinct steps separate from conceptual elaboration: *"Mixed numbers... I'm just going to transform one of the parts... how many equal parts should I cut it into?"* Cluster 1 represents a mode where instructional guidance and content delivery occur simultaneously, with experts more likely to weave these functions within single turns while novices sequentialize them.

Cluster 2: Rapid-Fire Elicitation and Checking (n = 2,598; experienced 51%; M = 29.1 chars). This cluster contains short prompts dominated by *Ask for Expression* and *Request for Choral Response*, aligning with IRE (Initiate-Response-Evaluate) patterns observed in the regional context [26]. Both experienced and novice groups use this mode frequently, but for different functions. Experts often reinforce structural relationships: *"The denominators are the same—meaning the addend and augend denominators are both...?"* While novices also employ short questioning sequences to elicit mathematical reasoning (*"Three over six. Why is it three over six?"*), they more often emphasize reinforcing recall or ensuring procedural correctness. Prompts like *"Say it again completely"* or *"Read it carefully. Read it carefully"* focus on the form or mechanics of participation [25]. Questions such as *"That's right—what is the question actually asking you to use?"* verify task comprehension. While both groups leverage rapid-fire turns to probe concepts, novices pay more attention to ensuring students follow procedures correctly.

Cluster 3: Dialogic Facilitation and Inquiry (n = 2,900; experienced 72%; M = 67.3 chars). This cluster represents medium-length turns where teachers orchestrate mathematical discourse by positioning students as sources of knowledge and facilitating peer reasoning. Teachers frequently employ *Press for Reasoning*, *Agree/Disagree*, and *Repeat and/or Add On* moves that redistribute mathematical authority across the classroom. While both groups encourage student expression and peer explanation, discernible differences emerge in their approaches. Experts probe underlying mathematical logic (*"Why did you do it so fast? Tell me, when you were measuring, how did you measure so quickly?"*), while novices often couple explanation-seeking with classroom management (*"Student 21 stand up, what is he saying?"*, *"When classmates are talking, you need to listen carefully"*). Experts frame inquiry to elevate student status (*"What's the secret to making you count so fast? Let me interview the fastest group first"*) and persist in understanding reasoning (*"I don't really understand what Student 22 is saying. Can you say it again?"*), while some novices redirect when students struggle (*"I don't know what you're talking about!"*). In short, Cluster 3 reflects attempts to democratize mathematical authority, however, teachers may vary in their approaches to sustain productive student-centered dialogue.

6. DISCUSSION

This study shows that fine-tuned LLMs can reliably classify talk moves in Cantonese mathematics classrooms and that their internal representations encode systematic differences in teacher expertise. A key contribution of our approach is its ability to surface pedagogical structure that is difficult to specify in advance. Traditional linguistic analyses require researchers to pre-select features such as question types, turn lengths, or lexical indicators, and may therefore miss diffuse or non-obvious cues to expertise. Embedding analysis instead reveals the latent distinctions the model uses to separate discourse from experienced and novice teachers, capturing how ideas are composed, how teachers respond to unfolding interaction, and how the same move is executed differently. Analysing representations thus provides a complementary lens on classroom instruction. We discuss three key contributions of our study and their implications below.

Fine-Tuned Models Enable Cross-Cultural Discourse Analysis. Domain-specific fine-tuning yielded strong talk-move classification performance. Qwen3-8B achieved micro-F_1 = 0.814 and macro-F_1 = 0.768, with particularly high accuracy on culturally characteristic moves such as *Request for Choral Response* (F_1 = 0.884) and *Revoice* (F_1 = 0.860). These results indicate that open-weight models, when properly adapted, can learn context-specific pedagogical practices that differ from Western instructional norms, an encouraging outcome given that Cantonese remains severely under-represented in NLP resources. Notably, models larger than 8B parameters did not yield meaningful improvements, suggesting that performance gains came primarily from adaptation rather than scale alone; this is further supported by our few-shot prompting baseline using off-the-shelf LLMs, which yielded $F_1 < 0.35$ without fine-tuning, compared to 0.81 after adaptation. We did not evaluate larger multilingual LLMs such as Qwen3-235B due to computational constraints. Nonetheless, the underperformance of `GPT-OSS-20B` relative to smaller fine-tuned models hints that scale alone may not fully compensate for domain adaptation in low-resource pedagogical settings. Evaluating larger multilingual models as zero-shot baselines remains an important direction for future work.

LLM Embeddings Encode Expertise Beyond Surface Features. Before interpreting these results, we note that our operationalization of expertise as years of experience is a known limitation. Experience correlates with instructional quality only probabilistically, and the relationship is nonlinear: some novices exhibit expert-like discourse and vice versa. The patterns we report should therefore be understood as differences between less and more experienced teachers, not as validated markers of instructional quality per se. Embedding analysis showed that teacher expertise is linearly recoverable from the model's internal representations (balanced accuracy = 0.790 on held-out data). Expertise signals were strongest for moves involving deeper reasoning, such as *Revoice* (0.783), and weaker for procedurally simpler moves. K-means clustering revealed three coherent discourse styles: (1) procedural management and scaffolding (78% expert), (2) rapid elicitation (51% expert), and (3) dialogic facilitation (72% expert). Although lengths of utterances within the clusters differed, our results show that length is not driving the structure, and deeper pedagogical signals underlie the observed geometry. Qualitative inspection supports this interpretation. Experts and novices often use similar structures, but experts integrate conceptual connections. In Cluster 2, for instance, experts extend student thinking ("The denominators are the same…"), whereas novices default to repetitive prompting ("Say it again completely"). This computational pattern mirrors educational research showing that experienced teachers exhibit stronger adaptive responsiveness and conceptual coherence [25, 13].

Implications for Automated Teacher Feedback. The linear separability of teacher expertise in embedding space provides a foundation for formative feedback tools that move beyond counting talk-move frequencies. Rather than treating discourse categories as checklists, automated systems could surface differences in how similar moves are executed, identify low-quality realizations of otherwise appropriate moves, highlight overreliance on novice-dominant discourse patterns, or suggest examples from experienced-teacher clusters as models for improvement. For instance, our Cluster 2 analysis shows that both novice and experienced teachers frequently use rapid questioning, but differ in pedagogical function: novices tend to emphasize procedural compliance, while experienced teachers use similar moves to surface underlying mathematical structure. This illustrates that the instructional value of a talk move lies in its realization, not its label.

These differences suggest concrete opportunities for targeted coaching in teacher education. Contrasting examples drawn from our discourse clusters could help novice teachers recognize how small shifts in phrasing, sequencing, or follow-up change the function of a move. Rather than generic advice such as "ask more open-ended questions," professional development can focus on specific discourse strategies, such as sustaining student reasoning through clarification and revoicing when confusion arises. Grounding coaching in authentic classroom utterances may make otherwise tacit aspects of instructional expertise more visible and learnable.

At the classroom level, these insights point toward personalized, low-stakes feedback tools that support reflection over time. Session-level summaries showing how a teacher's utterances distribute across discourse clusters, paired with illustrative examples from experienced-teacher patterns, could help teachers identify missed opportunities for conceptual scaffolding or overreliance on rapid elicitation. When framed as developmental reference points rather than evaluative benchmarks, such feedback can complement self-reflection and support growth across a semester.

Finally, by making explicit the discourse norms embedded in Hong Kong mathematics classrooms, this work opens space for cross-context professional dialogue rather than prescriptive standardization. Educators in different settings can use these representations to examine their own assumptions about participation, questioning, and authority, such as the role of choral response in collective sense-making [11, 19]. When carefully contextualized, automated discourse feedback can support reflection and learning across instructional contexts.

7. LIMITATIONS

We acknowledge several limitations of this study. First, our embedding analysis was conducted exclusively on Qwen3-8B. While this choice was methodologically justified, it means that our findings may be model dependent. Models with different capacities or architectures may encode talk-move patterns in different ways, which limits the generalizability of our results across LLM families. Additionally, our embedding extraction approach relies on delimiter-based span detection and token-to-character offset mapping, which assumes consistent tokenizer behavior and accurate recovery of span boundaries. In practice, any misalignment, particularly in non-space-delimited languages, may result in partial contamination from surrounding context. Disentangling whether low F1 on rare categories such as *Explain Others* and *Evaluation* reflects data sparsity or intrinsic semantic difficulty also remains an open question, and controlled subsampling experiments matching support across labels would be a valuable direction for future work.

Second, this study operationalizes instructional expertise as years of teaching experience, a coarse proxy whose limitations are discussed in Sections 3 and 6. A related concern involves the effective sample size of the expertise analyses: although CantoTalk contains over 7,000 utterances, these come from only 32 teachers, with expertise analyses contrasting just 25 teachers after excluding the intermediate group. The observed embedding-based distinctions may therefore partly reflect teacher-specific idiosyncrasies rather than robust markers of expertise; replication with a larger and more diverse teacher sample remains an important next step. Additionally, intermediate teachers were excluded from classification analyses to maximize contrast between novices and experts; whether embeddings order all three experience levels approximately linearly is also an open question for future work.

Finally, the teacher utterances analyzed in this work come from a narrow instructional domain, namely mathematics classrooms in Hong Kong. As a result, the discourse patterns that the model learns to associate with expertise may reflect local or context-specific norms rather than generalizable features of high-quality instruction. Findings may not generalize to other subjects, cultures, or languages. Additionally, although we identify embedding clusters that correlate with differences in teaching experience or talk-move styles, we do not establish a direct connection to student learning outcomes. While our results show that LLM embeddings encode systematic differences between novice and experienced teachers in ways that align with theoretical constructs from the teacher development literature, we do not evaluate whether identified expertise patterns actually support student learning, engagement, or equitable participation. It is possible that some of the discourse patterns we label as "expert" reflect institutional norms, curriculum alignment, or classroom management strategies rather than practices that promote deeper learning. For example, frequent use of *Request for Choral Response* may create an appearance of engagement without supporting individual sense-making. Future work should incorporate behavioral measures or expert-based validation and extend the analysis to a wider range of instructional contexts.

8. ETHICAL IMPLICATIONS

Some important ethical implications of this research include:

Privacy and Consent. This work relies on classroom discourse data, which raises important privacy concerns. Although all transcripts were anonymized and stripped of identifying information, teaching styles and discourse patterns may still be recognizable to colleagues within the same context. We mitigated this risk through anonymized analysis, and any future deployment of similar models should adopt strict data governance practices and explicitly prohibit individual teacher identification.

Potential Benefits. This work has several positive implications for education research and practice. Automated analysis of talk moves could support scalable, low-cost professional development by enabling teachers to reflect on their discourse patterns over time. By demonstrating that LLMs can be adapted to non-Western, non-English classroom contexts, this study also contributes toward more inclusive educational technologies and reduces reliance on English-centric tools. For researchers, automated classification substantially lowers the cost of large-scale discourse analysis, opening opportunities for longitudinal and cross-context studies that are infeasible with manual coding.

Risks and Misuse. At the same time, this work carries meaningful risks. Discourse patterns should not be treated as a complete or universal measure of teaching quality, yet there is a risk that such tools could be misused for high-stakes evaluation. Teaching quality is multidimensional and context-dependent, and many critical aspects of instruction are not captured in transcripts alone. There is also a risk of reinforcing narrow pedagogical norms learned from specific frameworks and contexts, which may not generalize or align with local instructional values. Additionally, because we do not validate our findings against student learning outcomes, identified expertise patterns may reflect institutional norms or surface-level engagement rather than deep learning.

Fairness and Generalization. Our use of teaching experience as a proxy for expertise introduces fairness concerns, as experience does not always align with instructional quality. The model may misclassify effective novice teachers or disadvantage teachers working in more challenging contexts. Furthermore, because the data come from Cantonese-medium mathematics classrooms in Hong Kong, the learned patterns reflect local discourse norms and linguistic features that may not transfer to other subjects, cultures, or languages.

Responsible Use. We emphasize that models like ours should be used only for low-stakes, teacher-initiated reflection, not for administrative evaluation or accountability. Future work should validate discourse signals against student outcomes, involve educators as co-designers in new contexts, and treat automated analysis as a complement to, rather than a replacement for, human professional judgment.

9. REFERENCES

[1] C. Howe, S. Hennessy, N. Mercer, M. Vrikki, and L. Wheatley. Teacher–student dialogue during classroom teaching: Does it really impact on student outcomes? *Journal of the Learning Sciences*, 28(4-5):462–512, 2019.

[2] D. Wagner and B. Herbel-Eisenmann. Identifying authority structures in mathematics classroom discourse: A case of a teacher's early experience in a new context. *ZDM*, 46:871–882, 2014.

[3] Daniel Lee and Jamie Poskin. Teachfx app (student–teacher talk measurement tool). Math Practical Measurement, WestEd, 2025. Accessed: 2025-11-06; URL: https://mpm.wested.org/measure/teachfx-app-student-teacher-talk-measurement-tool/.

[4] Bryant Jensen, Guadalupe Valdés, and Ronald Gallimore. Teachers Learning to Implement Equitable Classroom Talk. *Educational Researcher*, 50(8):546–556, November 2021. Publisher: American Educational Research Association.

[5] Y. Ni, D. Zhou, X. Li, and Q. Li. Relations of instructional tasks to teacher–student discourse in mathematics classrooms of chinese primary schools. *Cognition and Instruction*, 32(1):2–43, 2014.

[6] Klara Sedova, Martin Sedlacek, Zuzana Salamounova, Tomas Lintner, Roman Svaricek, Jakub Vlcek, Karolina Malikova, and Ivo Rozmahel. Let them all talk: equitable participation in classroom dialogue as a result of an intervention programme. *Language and Education*, 39(6):1471–1489, November 2025. Publisher: Routledge _eprint: https://doi.org/10.1080/09500782.2025.2454637.

[7] A. Suresh, T. Sumner, J. Jacobs, B. Foland, and W. Ward. Automating analysis and feedback to improve mathematics teachers' classroom discourse. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9721–9728, 2019.

[8] D. Wang, D. Shan, Y. Zheng, K. Guo, G. Chen, and Y. Lu. Can chatgpt detect student talk moves in classroom discourse? a preliminary comparison with bert. In *Proceedings of the 16th International Conference on Educational Data Mining*, pages 515–519, 2023.

[9] Baptiste Moreau-Pernet, Yu Tian, Sandra Sawaya, Peter Foltz, Jie Cao, Brent Milne, and Thomas Christie. Classifying Tutor Discursive Moves at Scale in Mathematics Classrooms with Large Language Models. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, L@S '24, pages 361–365, New York, NY, USA, July 2024. Association for Computing Machinery.

[10] J. Cao, A. Suresh, J. Jacobs, C. Clevenger, A. Howard, C. Brown, and J. H. Martin. Enhancing talk moves analysis in mathematics tutoring through classroom teaching discourse. *arXiv preprint arXiv:2412.13395*, 2024.

[11] Lihua Xu and David Clarke. Speaking or not speaking as a cultural practice: analysis of mathematics classroom discourse in Shanghai, Seoul, and Melbourne. *Educational Studies in Mathematics*, 102(1):127–146, June 2019. Publisher: Springer.

[12] Chengxuan Xia, Qianye Wu, Hongbin Guan, Sixuan Tian, Yilun Hao, and Xiaoyu Wu. Evaluating Modern Large Language Models on Low-Resource and Morphologically Rich Languages:A Cross-Lingual Benchmark Across Cantonese, Japanese, and Turkish, November 2025. arXiv:2511.10664 [cs].

[13] Zhiyue Tong, Fengcun An, and Yanji Cui. Exploring teacher discourse patterns: Comparative insights from novice and expert teachers in junior high school EFL contexts. *Heliyon*, 10(16):e36435, August 2024.

[14] S. Michaels and C. O'Connor. Conceptualizing talk moves as tools: Professional development approaches for academically productive discussion. In *Socializing Intelligence Through Talk and Dialogue*, pages 347–362. 2015.

[15] C. Howe and M. Abedin. Classroom dialogue: A systematic review across four decades of research. *Cambridge Journal of Education*, 43(3):325–356, 2013.

[16] Sterling Alic, Dorottya Demszky, Zid Mancenido, Jing Liu, Heather Hill, and Dan Jurafsky. Computationally Identifying Funneling and Focusing Questions in Classroom Discourse, July 2022. arXiv:2208.04715 [cs].

[17] Dorottya Demszky, Jing Liu, Heather C. Hill, Dan Jurafsky, and Chris Piech. Can Automated Feedback Improve Teachers' Uptake of Student Ideas? Evidence From a Randomized Controlled Trial in a Large-Scale Online Course. *Educational Evaluation and Policy Analysis*, 46(3):483–505, September 2024. Publisher: American Educational Research Association.

[18] David Clarke, Li Hua Xu, and May Ee Vivien Wan. Students Speaking Mathematics: Practices and Consequences for Mathematics Classrooms in Different Countries. In *Student Voice in Mathematics Classrooms around the World*. Brill, January 2013. Section: Student Voice in Mathematics Classrooms around the World.

[19] Oi-Lam Ng, Yujing Ni, Lian Shi, Gaowei Chen, and Zhihao Cui. Designing and Validating a Coding Scheme for Analysis of Teacher Discourse Behaviours in Mathematics Classrooms. *Journal of Education for Teaching*, 47(3):337–352, May 2021. Publisher: Routledge _eprint: https://doi.org/10.1080/02607476.2021.1896340.

[20] Yujing Ni, Lian Shi, Alan Cheung, Gaowei Chen, Oi-Lam Ng, and Jinfa Cai. Implementation and efficacy of a teacher intervention in dialogic mathematics classroom discourse in Hong Kong primary schools. *International Journal of Educational Research*, 107:101758, 2021.

[21] A. Suresh, J. Jacobs, V. Lai, C. Tan, W. Ward, J. H. Martin, and T. Sumner. Using transformers to provide teachers with personalized feedback on their classroom discourse: The talkmoves application. *arXiv preprint arXiv:2105.07949*, 2021.

[22] Juan N. Pava, Caroline Meinhardt, Haifa Badi Uz Zaman, Toni Friedman, Sang T. Truong, Daniel Zhang, Elena Cryst, Vukosi Marivate, and Sanmi Koyejo. Mind the (language) gap: Mapping the challenges of LLM development in low-resource language contexts. White paper, Stanford University Institute for Human-Centered Artificial Intelligence and The Asia Foundation, 2025.

[23] Partha Pakray, Alexander Gelbukh, and Sivaji Bandyopadhyay. Natural language processing applications for low-resource languages. *Natural Language Processing*, 31(2):183–197, March 2025.

[24] Oi-Lam Ng, Wing Kin Cheng, Yujing Ni, and Lian Shi. How linguistic features and patterns of discourse moves influence authority structures in the mathematics classroom. *Journal of Mathematics Teacher Education*, 24(6):587–612, July 2020. Publisher: Springer.

[25] Christian Kosel, Elisabeth Bauer, and Tina Seidel. Where experience makes a difference: teachers' judgment accuracy and diagnostic reasoning regarding student learning characteristics. *Frontiers in Psychology*, 15:1278472, March 2024.

[26] Y. Ni, L. Shi, A. Cheung, G. Chen, O. L. Ng, and J. Cai. Implementation and efficacy of a teacher intervention in dialogic mathematics classroom discourse in hong kong primary schools. *International Journal of Educational Research*, 107:101758, 2021.

[27] Gaowei Chen, Sherice N. Clarke, and Lauren B. Resnick. Classroom Discourse Analyzer (CDA): A Discourse Analytic Tool for Teachers. *Technology, Instruction, Cognition & Learning*, 10(2):85–105, April 2015.

[28] Randi A. Engle and Faith R. Conant. Guiding principles for fostering productive disciplinary engagement: Explaining an emergent argument in a community of learners classroom. *Cognition and Instruction*, 20(4):399–483, 2002.

[29] M. Stein, J. Remillard, and S. Smith. How curriculum influences student learning. In *Second handbook of research on mathematics teaching and learning*, volume 1, pages 319–370. 2007.

[30] E. Yackel and P. Cobb. Sociomathematical norms, argumentation, and autonomy in mathematics. *Journal for Research in Mathematics Education*, 27(4):458–477, 1996.

[31] Thinking-Machines-Lab. Tinker: a training api for large-scale model fine-tuning. https://thinkingmachines.ai/tinker/, 2025. Accessed: 2025-11-25.

[32] Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting, 2024.

[33] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics.

[34] Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. What happens to BERT embeddings during fine-tuning? In Afra Alishahi, Yonatan Belinkov, Grzegorz Chrupała, Dieuwke Hupkes, Yuval Pinter, and Hassan Sajjad, editors, *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online, November 2020. Association for Computational Linguistics.

[35] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[36] T. Caliński and J Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974.

[37] David L. Davies and Donald W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[38] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[39] Yujing Ni, Gloria Ho, Jinfa Cai, Alan Cheung, Gaowei Chen, and Oi-Lam Ng. Research protocol: Teacher interventions aimed at engaging students in dialogic mathematics classroom discourse. *International Journal of Educational Research*, 86:23–35, January 2017.

APPENDIX

A. TALK MOVES FRAMEWORK

Teacher talk-move labels, their definition, function and examples, presented in Appendix Tables A. We would like to note that teacher talk move definitions, functions, and examples are synthesized from [39], [24] and [19].

Table A.1: Teacher Talk Moves with Definitions and Illustrative Examples

Talk Move	Definition	Examples
表達自己 (Ask for Expression)	明確邀請學生把想法說出來（不限內容深度）。 *Teacher invites a student to state their thinking or idea (no reasoning required).*	"Could you tell me what features a quadrilateral has?" "If they are the same, this procedure is what we call...?"
對自己發言作補充 (Say More)	要求同一位學生把剛才的發言延伸或補充細節。 *Teacher asks the same student to elaborate on their own prior statement.*	"Talk a little bit more." "Could you Say More about the idea you've just expressed?" "Is there anything you want to add about your example?"
複述 (Revoice)	老師用自己的語言重述學生想法以確認理解或放大訊息。 *Teacher restates a student's idea in their own words to confirm understanding or clarify for others.*	"So, let me see if I've got your thinking right. Are you saying...?" "Student X told us a square should have four equal angles." [Student: "Three parts."] "Three parts—so which one is three-eighths now?"
解釋自己 (Press for Reasoning)	追問同一學生說明「為什麼／怎麼知道」。 *Teacher asks a student to explain the reasoning behind their answer.*	"Why do you think that?" "What's your evidence?" "Could you describe the procedure she used?"
重述及/或補充他人發言 (Repeat and/or Add On)	邀請學生重述同伴想法，或在其上加以補充。 *Teacher invites students to restate or add to another student's idea.*	"Could you provide an example to illustrate the definition he is giving?" "Anything wrong with that answer, Student X?"
同意/否定 (Agree/Disagree)	老師表態同意或否定學生想法。 *Teacher expresses agreement or disagreement with a student's idea.*	"Do you agree/disagree—and why?" "Do you accept his example as a quadrilateral figure?"
解釋他人 (Explain Others)	要求學生解釋「另一位」的推理或方法。 *Teacher asks a student to explain the reasoning of another student.*	"Why do you think he said that?" "Who can explain what she means when she says that?"
集體作答 (Request for Choral Response)	要求全班齊答。 *Teacher prompts the whole class to respond together.*	"Do we all agree with that?" "The first multiple of a number is... say it together!"
評價 (Evaluation)	老師給出正誤判斷、稱讚或等第式回饋。 *Teacher provides correctness judgment, praise, or evaluative feedback.*	"You're right." "Good idea." "Not exactly correct."
其他 (Other Moves)	不屬上述類別的管理、過渡、後勤或其他話語。 *Classroom management, procedural talk, transitions, etc.*	"Please do not talk to your neighbor." "Open your textbook to page 25 for homework." "Louder, I couldn't hear just now."

B. FULL PROMPT TEMPLATE

Each training example is formatted using the model's native chat template, with identical informational content across architectures. Below is the canonical prompt used for all models (Cantonese + English definitions).

B.1 User Message

分析以下香港高小數學課堂對話，識別當前發言的所有talk move類別。
Talk Move 類別定義：

- `{label_1}`: `{definition_1}`
- `{label_2}`: `{definition_2}`
- ...
- `{label_k}`: `{definition_k}`

前文語境：
`{speaker_before_1}`: `{utterance_before_1}`
`{speaker_before_2}`: `{utterance_before_2}`
`{speaker_before_3}`: `{utterance_before_3}`
`{speaker_before_4}`: `{utterance_before_4}`
`{speaker_before_5}`: `{utterance_before_5}`
`{speaker_before_6}`: `{utterance_before_6}`
當前發言：
`{current_speaker}`: `{current_utterance}`
後文語境：
`{speaker_after_1}`: `{utterance_after_1}`
`{speaker_after_2}`: `{utterance_after_2}`
`{speaker_after_3}`: `{utterance_after_3}`
`{speaker_after_4}`: `{utterance_after_4}`
`{speaker_after_5}`: `{utterance_after_5}`
`{speaker_after_6}`: `{utterance_after_6}`
任務：根據上述定義，識別當前發言的所有適用talk move類別（用逗號分隔多個類別）。

B.2 Assistant Message

Gold multi-label target:

`<label_1>, <label_2>, ...`

B.3 Model-Specific Renderers

- `Qwen3-4B-Instruct-2507, Qwen3-8B` → `qwen3_instruct`
- `Llama-3.1-8B-Instruct` → `llama3_instruct`
- `DeepSeek-V3.1` → `deepseek`
- `GPT-OSS-20B` → `gpt_oss_no_sysprompt`

Note: Renderers differ only in tokenization wrappers; content is identical.

C. TRAINING HYPERPARAMETERS

All models were fine-tuned using LoRA adapters with the following configuration:

Component	Setting
LoRA Configuration	rank = 32; α = model default; dropout = 0.0
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.95$, ϵ = 1e–8)
Learning Rate	base LR = 2e–5; LoRA LR scaled via `get_lora_lr_over_full_finetune_lr()`
Scheduling	300-step linear warm-up; constant LR thereafter
Training Steps	max 3000 steps; early stopping after 5 validation checks
Validation	every 100 steps (100-sample subset)
Batch Size	8
Sequence Length	max 4096 tokens
Checkpointing	every 50 steps

Table C.1: Hyperparameter configuration for LoRA fine-tuning across all models.

D. IMPLEMENTATION AND EVALUATION DETAILS

Tokenization and Truncation. We use each model's native tokenizer (via Tinker's `get_tokenizer()`), truncating sequences to a maximum length of 4096 tokens. Truncation is applied only to the earliest portions of the preceding context, ensuring that the focal utterance and its immediate surroundings are always preserved.

Decoding. During evaluation, models generate predictions using greedy decoding (temperature = 0) with a 50-token limit and model-specific stop sequences. Outputs are stripped of special tokens before post-processing.

Label Canonicalization. Generated text is split on commas, normalized (whitespace removed), and matched to the valid talk-move inventory. Unknown strings are discarded. Multi-label predictions are converted into binary indicator vectors for evaluation.

Characterizing Students' LLM Usage Behaviors and Their Association with Learning in Critical Thinking Tasks

Minju Park
University of British Columbia
Vancouver, BC, Canada
minju.park@ubc.ca

Ivan Orozco Vasquez
University of British Columbia
Vancouver, BC, Canada
ivanorozco9@gmail.com

Cristina Conati
University of British Columbia
Vancouver, BC, Canada
conati@cs.ubc.ca

ABSTRACT

Large language models (LLMs) are becoming increasingly embedded in students' learning practices, yet much of what is known about how students use LLMs and how this usage impacts learning comes from problem-solving domains or constrained experimental settings. We present an analysis of data on LLM usage collected during two offerings of a research-oriented course where students learn to read, reason about, and critique academic papers. Without restrictions on whether or how to use LLMs, students reported their LLM usage practices when asked to do these activities as a series of homework assignments during the course. This paper extends prior work done on data from a single offering of the same course by presenting a refined bottom-up categorization of LLM usage types, cross-labeled by the extent of student initiative these usages entail. Furthermore, we examine how LLM use impacts student learning, measured by performance on three midterms, looking at factors such as frequency and type of usage.

Keywords

Students' LLM Usage Behaviors, Bottom-up Categorization, Association with Learning

1. INTRODUCTION

As large language models (LLMs) become more powerful, sophisticated, and accessible, the educational community continues to grapple with understanding what roles these models can and should play in education [21, 13, 11]. While some argue that integrating LLMs into coursework offers benefits such as immediate and personalized feedback [21, 13], there are widespread concerns that using LLMs may undermine students' learning, prompting calls to prohibit its use until we gain a better understanding of how it can be effectively leveraged (e.g., [11]). This paper contributes to a growing body of research addressing the critical need to understand how students use LLMs in practice and how such use may support or hinder learning.

Minju Park, Ivan Orozco Vasquez, and Cristina Conati. Characterizing Students' LLM Usage Behaviors and Their Association with Learning in Critical Thinking Tasks. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 57–66. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040150

Much of the existing research on student usage of LLMs has focused on problem-solving domains, particularly programming [21, 13, 16, 2, 9, 5, 1]. These domains tend to emphasize solution-seeking and correctness, which has led prior work to frame LLM use primarily in terms of debugging or generating solutions and answers. In contrast, many general academic skills—such as reading, reasoning, and critical thinking—require interpretation, judgment, and the construction of understanding rather than arriving at a single correct solution. In such contexts, LLMs may shape how students think and engage with material in more nuanced ways that are not well captured by studies centered on problem-solving tasks.

Another limitation of prior work on students' use of LLMs is that many studies rely on controlled experiments or structured interventions, where students are instructed to use LLMs in predefined or constrained ways. Although some work has examined LLM use in authentic course assignments [7], students were still guided toward specific forms of interaction, which may not reflect how they would naturally choose to use LLMs.

In this paper, we investigate LLM use in a learning context that has received little prior attention: a research-oriented course in which students learn to read, reason about, and critique academic papers. We examine students' LLM use during two offerings of a real university course by drawing on students' self-reported usage in weekly assignments, which allow us to capture organically occurring LLM use within an in-the-wild classroom setting where students were free to decide whether to use LLMs and how to use it. The analysis presented in this paper extends prior work on data from a single offering of the same course [15]. We adopt a similar bottom-up, data-driven approach which characterizes LLM use based on students' actual reported practices rather than prescribed behaviors or predefined categories, but we present a refined bottom-up categorization of LLM usage types, cross-labeled by the extent of student initiative these usages entail. We identify patterns in usage types and frequency across assignments, and we examine how LLM use impacts student learning, measured by performance on three midterms.

Our analysis shows a wide variation in how students used LLMs—differing in frequency, timing, and the types of support they sought. Performance analyses showed that students who used LLMs in their assignments scored lower than

non-users on the first midterm, although this gap decreased as the term progressed. Within LLM users, those who used LLMs in more than half of their assignments consistently earned lower average exam scores than those who used them less often. Similarly, students who engaged in more student-driven prompting—providing meaningful input rather than relying on the model to produce most of the content—tended to score higher, though this performance gap also narrowed over the semester. Taken together, our results suggest that, although students may benefit from guidance on how to use LLMs effectively, especially earlier on in the course, given enough time and opportunities to experience the impact of LLM usage on their exam performance, many manage to decide if and how to integrate this usage in their study process without hindering learning.

In summary, our work contributes to the growing need to understand whether and how students use LLMs, and how such usage affects learning, by providing insights in a context that is unique in two ways: first, we target the previously unexplored tasks of reading, analyzing and critiquing research papers, which require interpretation and reasoning skills beyond those involved in the most commonly investigated task of programming, problem-solving and essay writing. Second, the data used for our analysis was collected in a real course setting, where students were given complete freedom on if and how to use LLMs for their assignments. Together, these contributions broaden current understanding of how students are using LLMs in their learning and highlight the importance of investigating LLM use in ecologically valid settings where students' engagement with these tools emerges organically.

2. RELATED WORK

In this section, we review prior work on students' use of LLMs in education along two dimensions: (1) educational contexts of student LLM use, with a focus on differences across domains and instructional settings, and (2) approaches to analyzing students' LLM interactions, including both top-down and bottom-up approaches.

2.1 Educational Contexts of Student LLM Use

Prior research on students' use of LLMs in educational settings has predominantly focused on problem-solving domains, particularly programming and software development. Within this line of work, many studies examine how LLMs can be intentionally integrated into learning activities through structured instructional designs or interventions. For instance, MacNeil et al. [13] examine how specific LLM-supported interventions, such as LLM-generated code explanations, influence students' learning processes and outcomes, while Rasnayaka et al. [16] investigate the incorporation of LLMs into software development projects within project-based learning contexts. Other works explore how students interact with and perceive LLM-integrated learning environments in similar problem-solving domains [9, 20]. Although these studies provide valuable insights into the pedagogical potential of LLMs, they mainly focus on LLM use within designed or structured learning settings.

Complementing this perspective of LLM integration in educational contexts, a smaller set of studies examines how students themselves use LLMs during learning, often by analyzing student prompts or interaction logs. This work has also largely centered on problem-solving domains, where LLM use is closely tied to generating correct answers and solutions [12, 18]. As a result, existing empirical characterizations of student-initiated LLM use are largely shaped by problem-solving tasks and contexts.

Beyond problem-solving domains, there is emerging but still limited work examining LLM use in other academic domains. A few studies have begun to explore contexts such as academic writing or discussion-based learning, yet these investigations typically rely on structured forms of LLM use. For example, prior work on academic writing has analyzed student prompts within an LLM-embedded system designed to scaffold writing tasks [10], while studies in ethics education have examined discussion activities in which students were guided to use LLMs in prescribed ways [7]. Taken together, these studies illustrate early efforts to extend LLM research beyond problem-solving domains, but they also show that existing evidence remains narrow in scope. Because different academic domains require and emphasize different skills, findings derived from problem-solving contexts may not generalize to learning activities centered on other skills. This observation underscores the need for further studies of students' LLM use across a broader range of domains and academic skills.

2.2 Approaches to Analyzing Students' LLM Usage

Prior studies have employed a range of analytical approaches to examine how students use LLMs, with many relying on top-down categorization schemes. A growing body of work proposes standardized taxonomies or frameworks for analyzing human-LLM interactions or prompt strategies in more general, non-educational contexts [4, 17]. Related work in prompt engineering has also developed comprehensive categorizations of prompting techniques through reviews and conceptual frameworks [19, 6]. Building on these approaches, some educational studies apply established categorizations to student data to examine patterns of LLM use in coursework, such as analyzing student prompts according to usage categories [14].

While these top-down approaches provide useful analytical structure, they rely on predefined categories and examine how student behavior fits within them. As a result, they may overlook how students actually appropriate LLMs in practice, including uses that do not align neatly with existing taxonomies. Prior work has noted that much research on prompting and LLM interaction focuses on techniques developed or analyzed by researchers rather than on students' naturally occurring behaviors in learning contexts [18]. This gap highlights the need for complementary approaches grounded in students' observed LLM use.

Beyond top-down categorization approaches, some studies have adopted inductive or bottom-up methods that ground analysis in students' actual LLM interactions. Jelson et al.[8], for example, conduct their analysis in the context of academic writing and explicitly anchor their categorization in Flower and Hayes' cognitive model of writing [3],

organizing LLM-supported activities around the phases of planning, translating, and reviewing. Within each phase, they further inductively coded students' prompts based on the specific ways learners engaged with the LLM, allowing the final scheme to emerge from observed behaviors rather than predefined functional categories. Grande et al. [7], by contrast, take a fully inductive approach, conducting qualitative thematic analysis of LLM-supported ethics discussions to surface recurring themes without constructing an explicit taxonomy. Sawalha et al. [18] similarly use thematic analysis to identify prompting strategies in problem-solving tasks. They examine how learners use LLMs to complete quiz-style programming tasks and derive a two-level structure, distinguishing among broad prompting types and the specific strategies within them. Although these strategies vary in complexity, they are all oriented toward obtaining or refining correct answers, reflecting the goal-directed nature of problem solving.

As these examples illustrate, coding schemes naturally reflect the demands of their respective learning domains. Domains that center on interpretation, critique, and reasoning require different forms of support, and in this sense, our work parallels that of Orozco et al. [15]. They developed an emergent coding scheme grounded in students' interactions from a single research-oriented course, deriving usage types directly from observed behaviors rather than predefined taxonomies. The analysis and results presented in this paper extend this previous work by expanding the dataset to include an additional course offering from a different term. Using the coding scheme from [15] as a foundation, we iteratively refined and extended the codes as we analyzed the expanded dataset to capture the full breadth of students' LLM use, yielding a more robust characterization of students' LLM usage patterns.

3. DATA OVERVIEW

This section provides an overview of the course in which the data were collected and the composition of the dataset used for analysis.

3.1 Course Structure

Our exploratory analysis was conducted in a third-year undergraduate course offered at a public North American institution. The course's objective was to teach students research methods in the areas of Cognitive Systems, covering the skills and principles common to all forms of research, such as critical thinking and communication. As part of the course, students were assessed using 14 weekly take-home assignments and 3 in-class midterm exams distributed across the term, with the take-home assignments being graded for completion while the in-class midterm exams were graded for quality. The take-home assignments required students to read an assigned research paper on the general topic of human-AI interaction and complete two tasks:

1. **Critical Summary:** Summarize the assigned paper, focusing on the motivations of the work, the proposed solution, how the solution was evaluated, the contributions of the work, and specific aspects related to the details of the human-AI interaction.
2. **Discussion Points:** Brainstorm discussion points identifying and discussing weaknesses in the presented research, relating the research to general issues in the field, making connections with other readings, and proposing alternative approaches.

Students were permitted to use LLMs for their take-home assignments, provided they submitted a brief LLM-use report documenting the model(s) used, the prompts, and the generated output, as required by institutional policy. In contrast, the in-class midterms required students to complete the same types of tasks without any assistance. Each midterm mirrored the assignment format: students read a research paper and produced a critical summary and discussion points, as described above.

In summary, the data used for our analysis was collected as part of regular course activities and institutional requirements, including mandatory LLM-use reporting. Thus, it complies with the university's policies regarding the ownership and use of course materials for pedagogical quality assurance (Policies 81 and LR11).

3.2 Dataset Description

The data used in this paper was collected across two academic terms of the course: Term 1 included 37 undergraduate students, and Term 2 included 31, for a total of 68 students. Both terms followed the same course structure, assessment design, and LLM usage policies.

The dataset includes students' take-home assignment submissions along with their LLM-use reports (for those who used LLMs). It also includes scores from the three in-class midterm exams, which serve as evidence of students' unaided performance on tasks structurally similar to the take-home assignments.

In addition, background information about participating students was collected through a brief demographic survey. The survey asked students to report their coding ability (categorized as strong, moderate, or weak), their academic major, and their prior coursework in relevant areas (Human–Computer Interaction, Artificial Intelligence, and Machine Learning). For students in the Cognitive Systems program, majors were recorded along with their specific streams, which correspond to different disciplinary specializations (e.g., psychology, linguistics, philosophy, or computer science). Students' broader disciplinary background—such as whether they belonged to Arts or Science—was then inferred from their reported major. All collected data were anonymized prior to analysis.

4. METHODS

To determine whether data from Term 1 and Term 2 could be analyzed jointly, we compared the distributions of students' backgrounds across the two terms, focusing on self-reported coding ability and disciplinary background. Although the cohorts were not identical, their distributions were similar enough for our purposes. For coding ability, strong or moderate proficiency differed by 6.22% (83.78% in Term 1 vs. 90.00% in Term 2). For disciplinary background, the proportion of Science students differed by 10.55% (62.16%

Table 1: Categories of students' LLM usage grouped by support type.

Support Type	Category	Definition	Sample Excerpt from LLM-Use Reports
Student-driven Support	**IL**: Improving Language	Students provide their own draft text and ask the LLM to improve language (e.g., clarity, grammar).	*"Prompt Used: Here is my summary for the attached paper: [student's draft] Revise it for any grammatical issues."*
	CI: Confirming Ideas	Students propose their own ideas or drafts and ask the LLM to validate the content or to check whether the draft meets the assignment requirements.	*"Prompt Used: Does this address all the required points? [student's draft]"*
LLM-driven Support	**GI**: Generating Ideas	Students ask the LLM to suggest ideas (e.g., topics, arguments, outlines) for writing.	*"I used ChatGPT 4.0 to generate ideas for discussion points beyond what I could initially come up with after reading the paper."*
	S: Summarizing	Students ask the LLM to give an overview of the paper, which is separate from the assignment's more analytic critical summary task.	*"Prompt Used: Summarize this paper."*
	GA: Generating Answers	Students ask the LLM to generate an answer or text from scratch (discussion points or critical summary).	*"In the prompt, I passed the file of the paper and the document of the discussion document [assignment format] and asked it to complete it."*
Comprehension Support	**IE**: Information Extraction	Students ask the LLM to extract specific information from a paper.	*"Asked GPT to help with finding the mitigating solution against the potential usability threats.'*
	CC: Clarifying Concepts	Students ask the LLM for further explanation to deepen understanding of certain concepts.	*"Prompt Used: What's the difference between formula 1 and formula 2 in the paper?"*

vs. 51.61%), with the remaining percentages falling into the Arts category. Given these broadly comparable patterns, we analyzed the two cohorts together in subsequent analyses.

4.1 Coding Scheme for LLM Usage

A bottom-up coding scheme was used to characterize students' LLM usage. The analysis began with the coding framework introduced in earlier work [15], which identified seven usage categories: *Improving Language*, *Paragraph from Bullets*, *Clarifying Concepts*, *Confirming Ideas*, *Generating Ideas*, *Summarizing*, and *Generating Answers*. This framework served as an initial guide and was iteratively refined while analyzing the expanded two-term dataset to better capture the full range of usage behaviors observed.

Each submission was coded individually, and multiple categories were assigned whenever a student's prompts reflected more than one form of usage, such as when they engaged in several distinct interactions within the same submission. Using the existing categorization [15] as a starting point, two authors independently coded all LLM-use reports. Inter-rater reliability, assessed with Krippendorff's alpha, which is appropriate for nominal data with multiple labels per entry, indicated low agreement beyond chance across all categories (α range = -.01 to .03). This low reliability suggested that several category definitions overlapped or required clearer differentiation. The two authors then met to review the coded data, discuss sources of disagreement, and refine the categorization by clarifying definitions and determining where categories should be merged or further distinguished. Following this refinement process, the authors iteratively discussed and reconciled remaining disagreements until consensus was reached on all coded instances, resulting in a finalized coding scheme and dataset used for analysis.

The refinement process led to modifications of the existing categories as follows: first, we consolidated "Paragraph from Bullet Points" into "Improving Language." In the initial categorization, "Paragraph from Bullet Points" captured submissions in which students provided bullet-point inputs and asked the LLM to convert them into a paragraph. However, this process inherently involved improving wording, structure, or clarity, resulting in substantial conceptual overlap with the "Improving Language" category. Moreover, a closer examination of these submissions revealed considerable variability in the level of detail provided by students, ranging from brief keyword-style bullet points to full sentences or paragraph-length inputs. This inconsistency, combined with the significant overlap in function, motivated the decision to merge "Paragraph from Bullet Points" into the broader "Improving Language" category.

Second, refinements to the "Clarifying Concepts" category led to the creation of a new category, "Information Extraction." Initially, queries that asked the LLM to locate specific information within the assigned paper were grouped under "Clarifying Concepts" since that category broadly covered student questions about course material. However, closer analysis showed that these queries were not aimed at deepening conceptual understanding but at retrieving specific information from the paper—such as checking whether particular details appeared in the text rather than interpreting the

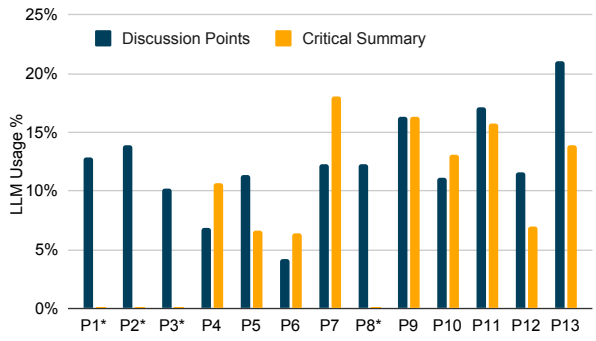


Figure 1: LLM usage rates for discussion points and critical summary tasks per paper. Asterisks indicate assignments with discussion points only.

underlying concepts. This distinction led to the separation of the two categories, where questions that sought conceptual clarification without explicitly requesting information from the paper were retained under "Clarifying Concepts," whereas queries that involved searching for text or data explicitly stated in the paper were classified as "Information Extraction."

This process resulted in our final set of seven categories—Improving Language, Confirming Ideas, Generating Ideas, Summarizing, Generating Answers, Information Extraction, and Clarifying Concepts. Definitions of each category and corresponding sample excerpts from the LLM-use reports are presented in Table 1. The table also shows that the categories are organized into three higher-level types of support types based on the role the LLM played in the student's work:

- **Student-driven Support**: cases in which students provided initial ideas or drafts and used the LLM for feedback, refinement, or confirmation (includes Improving Language and Confirming Ideas)
- **LLM-driven Support**: cases in which students used the LLM to generate content from scratch with minimal prior input (includes Generating Ideas, Summarizing, and Generating Answers)
- **Comprehension Support**: uses of LLMs to help understand the assigned paper or related concepts (includes Information Extraction and Clarifying Concepts)

4.2 Analysis Procedure

Analyses were conducted at two complementary levels. At the submission level, LLM usage behaviors were examined for each individual take-home assignment, with submissions coded into one or more usage categories based on the coding scheme described above. Category frequencies and usage patterns were then summarized across all submissions.

At the student level, LLM usage data were aggregated across each student's submissions and compared with their in-class midterm exam scores to examine how patterns of LLM use related to learning. This analysis considered both the extent of students' LLM use and the types of support they sought.

We employed descriptive statistics and group-based comparisons to examine overall usage trends, associations between LLM usage and learning, and differences across LLM support types. Analyses focused on identifying patterns of association rather than making causal claims.

5. RESULTS

In this section, we present our findings on how students used LLMs throughout the course and how these usage patterns relate to midterm exam performance, considering both the level of LLM use and differences in LLM usage types.

5.1 Students' LLM Usage Trends

We organize our analysis of students' engagement with LLM into two parts: (1) whether and how frequently students used LLMs across assignments over time, and (2) how students used LLMs in terms of the functional purposes captured in our coding scheme.

23 out of 68 students (34%) reported using LLMs in at least one submission. On average, students used LLMs in 12% of discussion point submissions and 11% of critical summary submissions. Students reported using a range of LLM tools, including ChatGPT (e.g., GPT-3.5, GPT-4, and GPT-4o), NotebookLM, and Claude models. To see how these rates varied for each assignment, Figure 1 shows the percentage of students who used LLMs for the two tasks, with papers (P1–P13) arranged chronologically from left to right on the x-axis. Note that for some papers, students had to submit only discussion points, no summaries (P1, P2, P3, and P8; indicated with an asterisk in Figure 1). The figure illustrates that LLM usage varied across assignments and tasks: for some papers, students used LLMs more for discussion points, whereas for others, usage was higher for the critical summary. A notable trend is the increase in LLM usage for later assignments, particularly for the discussion points task, suggesting that students may have grown more comfortable integrating LLMs into their workflow as the course progressed. While actual usage may be slightly higher due to potential underreporting (e.g., students neglecting to declare LLM use), it is unlikely that the true usage rate differs substantially, as students were explicitly permitted to use AI tools without penalty provided such use was disclosed.

Figure 2 provides a more detailed view of how each of the 23 students who used LLMs at least once (SN on the y-axis) engaged with LLMs for each assignment (P1 through P13 on the x-axis). The papers (P1–P13) are shown in chronological order from left to right. Submissions for each paper (PN) are separated in into discussion points (PN DP) and critical summary (PN CS). The types of LLM use categorized for each submission are indicated in each cell by the corresponding code (see Table 1). Cells contain usage codes when the student used an LLM for that task, appear in yellow when the student reported no LLM use, and are crossed out when the assignment was not submitted. Two vertical lines in the middle of the figure mark the timing of Midterm 1 and Midterm 2, and Midterm 3 occurred after the final assignment. Students are then ordered by their overall frequency

	P1 DP	P2 DP	P3 DP	P4 DP	P4 CS	P5 DP	P5 CS	P6 DP	P6 CS	P7 DP	P7 CS	P8 DP	P9 DP	P9 CS	P10 DP	P10 CS	P11 DP	P11 CS	P12 DP	P12 CS	P13 DP	P13 CS
S66			GA		X												X					
S36								X	X					S								
S65	X	X									GA						X		X			
S49	IL																X	X	X	X	X	X
S37		IL			CC												X	X				
S22		GI						X	X	GI												
S16				X	X	X	X	X	X				GI		X	X	X	X	X	X	X	X
S41			CC								S						X	X	X	X	X	X
S60					IL,CI			X	X		IL,CI							IL,CI		X		
S21			CI		IL	X	X			X	X			CC	CI							
S7	GI, CI					X	X			GI	S	GI							X	X		
S58				GI	GI				S		S		X	X		IE						
S57	GI	GA		X	X	X	X	X	X	X	X	X					X	X	X	X	X	X
S44		GA				GA							GA		X	X	GA	GA			GA	
S33		S,GI,CC				CI				X	X	GI,CI	GI,CI				X	X	X	X	IL,CI	
S23			X	X	X	GA	GA								X	X	X	X	GA		GA	GA
S62	IL	IL			GA		IL,S		IL		IL,S			IL,S	X	X	X	X	IL		X	X
S38										X	X	GA	IL	IL	IL	IL	GA	GA	GA	GA	GA	GA
S34	IL		IL	IL		IL		IL		IL		IL	CI	GA	IL			IL	X	X	CI	
S2	X					S	S	S	S	S	S	S	S	S	S	S	S	S	X	X	S	S
S61	IL,CI,GI	IL,CI,GI	IL,CI,GI	IL,CI,GI	IL,CI,GI			X	X	IL,CC,GI, S	IL,CC,S, IE	X	IL,CI,GI	IL,CI		IL,CI	IL,CI,GI	IL,CI,S,IE,CC	IL,CI,GI, IE	IL,CI,GI, S	IL,CI	IL,CI,CC
S59	GA	GA	GA	X	X			X	X	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
S63	S	S,GA	X	X	X	X	X	X	X	X	X	X	X	X	X	S	X	X	X	X	X	X

Figure 2: Temporal visualization of reported LLM use across assignments.

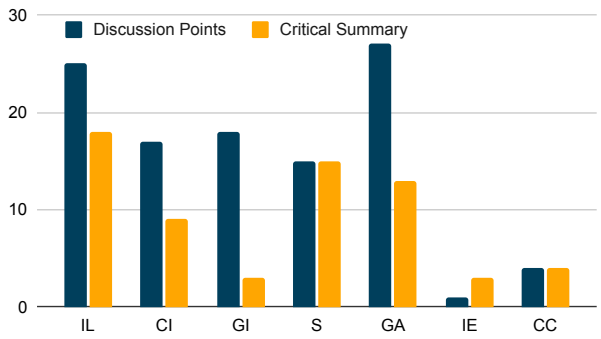


Figure 3: Counts for each LLM usage category for discussion points and critical summary.

of LLM use, with the most frequent users appearing at the bottom and the least frequent at the top. Because some students did not submit every assignment, usage frequency is computed as the number of assignments in which they used an LLM divided by the number they submitted. Notably, S63 appears at the very bottom: although their use of LLMs may look sparse visually, this is because they submitted only 3 of the 13 assignments—but used LLMs for all three—resulting in a 100% usage rate.

The visualization shows a distinct cluster of students who had a much higher usage frequency toward the bottom. Looking at the actual frequencies, a clear breakpoint emerges at S62: students above S62 used LLMs in only 5–33% of their submissions, while those from S62 downward used them in 50–100% of their work. This pattern provides a natural division into two groups, with 7 students using LLMs in at least half of their submitted assessments and 16 students whose usage remained below that threshold. In the rest of this analysis, we will indicate the two groups as *High-Reliance* and *Low-Reliance*, respectively.

These two groups showed different trend of LLM usage over time. Students in the Low-Reliance group showed more concentrated usage early in the term, with an average of three students using LLMs per assignment before Midterm 1. After Midterm 2, this average fell to one, indicating that students in this group not only rely on LLMs only lightly, but they reduced their usage as they grew more comfortable with the assignment tasks over time. In contrast, within the High-Reliance group, several students (S62, S34, S61, S59) used LLMs consistently throughout the term while others (S2, S38) shifted into heavier use partway through the term. S2 refrained from using LLMs until after Midterm 1, at which point they started using them in every subsequent submissions. Similarly, S38 began to use LLMs at the halfway point of the term, and then continues to use it for every submission until the end of the course. These varied trajectories within the High-Reliance group suggest that these two students may have adjusted their reliance on LLMs as they grew more familiar with the course structure and their workflow for using them, resulting in increased use later in the term when they felt such usage would no longer hinder their ability to perform the assignment tasks on their own during the midterm.

We next examine how LLMs were used in terms of usage category. Figure 3 presents overall counts for each LLM usage category grouped by assignment tasks (Discussion Points vs Critical Summary). As the figure shows, Summarizing (S) and Clarifying Concepts (CC) were used equally for the two tasks, which makes sense since both these usages help understand the content of the paper, which is needed for both tasks. Information Extraction (IE) is used more for critical summaries, which also makes sense since writing the critical summary is more directly related to finding specific information in the paper. All the other usages are higher for discussion points, showing that for this activity, which is more open-ended and requires going beyond the content of the paper, students needed more help on Improving Language (IL), Confirming Ideas (CI), Generating Ideas (GI) and Generating Answers (GA).

Figure 4 shows stacked bar graphs indicating, for each usage category, the proportion of individual students in the High-Reliance group who used it. Figure 5 shows the same information for students in the Low-Reliance group.

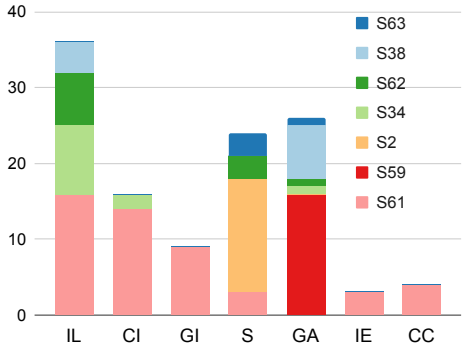


Figure 4: Counts for each LLM usage category, with stacked bars showing the proportion of individual students in the High-Reliance group.

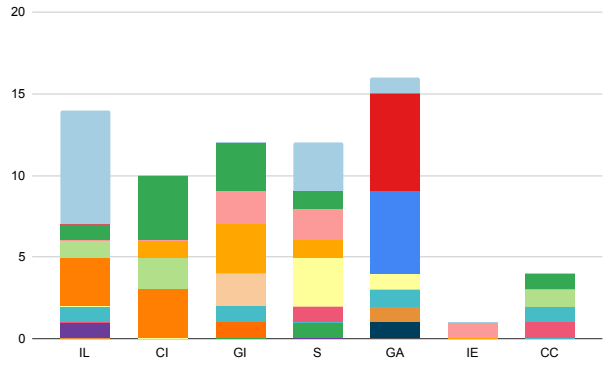


Figure 5: Counts for each LLM usage category, with stacked bars showing the proportion of individual students in the Low-Reliance group. Each color represents a different student.

As seen in Figure 4, in the High-Reliance group, three categories (Generating Ideas (GI), Information Extraction (IE), and Clarifying Concepts (CC)) were used only by one student, S61. This student used all the categories except for Generating Answers (GA), indicating that, although the student relied heavily on LLMs, they never used them to generate the assignment from scratch. Notably, S61's midterm scores steadily increased across the term (69% to 72% to 79%), illustrating that extensive LLM use—when focused on forms of support other than full answer generation—can coexist with, and may even accompany, improved performance. By comparison, S59 used LLMs solely for Generating Answers (GA), and S63 used only Generating Answers (GA) and Summarizing (S), both examples of LLM-driven support. Finally, S2 used only Summarizing (S), and their performance provides a clear illustration of the potential consequences of this pattern: their midterm scores declined consistently—from 65% to 48% to 38%. After Midterm 1, S2 appears to have increasingly relied on LLM-generated summaries rather than engaging with the full readings, which may have weakened their ability to process new material during the exams. The remaining students (S62, S38 and S34) spread their LLM usage over both student-driven and LLM-driven categories.

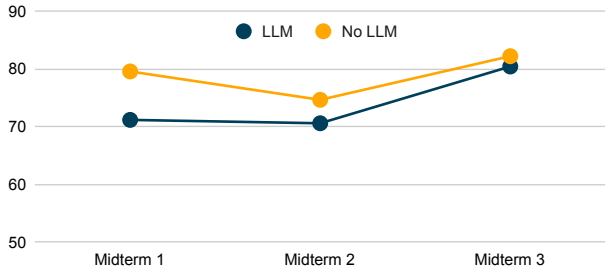


Figure 6: Comparisons of midterm exam scores between students who used LLMs in assignments (*LLM*) and those who did not (*No LLM*).

Figure 5 shows that, in the Low-Reliance group, each usage type was used by at least four different students aside from Information Extraction, indicating a relatively even distribution of usage across categories. This contrasts with the High-Reliance group, where certain categories are dominated by only a few students. Notably, even within the Low-Reliance group, Generating Answers (GA) was the most frequently used category. Two students in particular—S44 (red; second segment from the top in the GA bar) and S23 (blue; third segment from the top in the GA bar)—used Generating Answers (GA) more than any other category. As Figure 2 illustrates, both students tended to rely on Generating Answers (GA) more heavily toward the end of the term. Interestingly, S44 and S23 showed strikingly similar performance trajectories across the three midterms. Both improved substantially from Midterm 1 to Midterm 2 (S44: 54% to 82%; S23: 65% to 84%) but experienced a decline on Midterm 3 (S44: 78%; S23: 76%). This parallel trend suggests that increased reliance on Generating Answers (GA) later in the course did not necessarily prevent performance gains, but may also not have sustained them through the final assessment.

5.2 Impact of LLM Usage on Midterm Performance

5.2.1 *Performance Differences by Level of LLM Use*

We examined how students' use of LLMs in take-home assignments impacts their performance in midterm exam performance. For each midterm, we distinguished between students who had used LLMs in any prior assignment and those who had not. Because students adopted LLMs at different times during the term, group sizes varied across midterms. For example, S2 did not use LLMs before Midterm 1 but began doing so between Midterms 1 and 2, and was therefore treated as an LLM user for Midterm 2 and Midterm 3, but not Midterm 1. For Midterm 1, three students were excluded from the analysis: two were absent, and one experienced technical issues at the start of the exam that reduced their available exam time by half. Midterm 2 had one absent student, while Midterm 3 had three students absent.

We first compared midterm exam scores between students who reported using LLMs and those who did not. Figure 6 shows line graphs of the averages over the 3 midterms, for the students who did use LLMs (*LLM*) and those who did

Table 2: Comparisons of midterm exam scores between students who used LLMs in coursework (*LLM*) and those who did not (*No LLM*). Here, M denotes Midterm.

M	Condition	N	M	SD	t	df	p	d
M1	No LLM	44	79.60	13.15	-2.5	41.8	**0.016***	-0.65
	LLM	21	71.20	12.36				
M2	No LLM	44	74.68	11.52	-1.5	52.0	0.133	-0.37
	LLM	23	70.62	9.69				
M3	No LLM	42	82.24	9.04	-0.7	39.6	0.499	-0.19
	LLM	23	80.46	10.60				

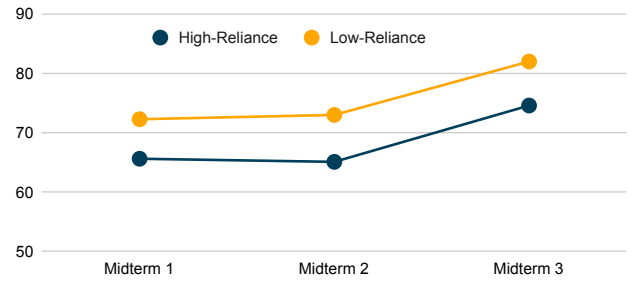


Figure 7: Comparisons of midterm exam scores between students in the High-Reliance and Low-Reliance groups.

not (*No LLM*). It shows that students in the No LLM group scored higher on average than the LLM group, although this gap narrowed as the term progressed. In Midterm 2, this is due to a decrease in performance for the No LLM group, whereas the performance of the LLM group remains similar to Midterm 1. In Midterm 3 both groups improve. A possible explanation is that, as students became more familiar with the processes entailed in the exam tasks, the impact of LLM usage on midterm performance may have diminished. Statistical tests and detailed score values are provided in Table 2, showing a statistically significant difference in performance between the LLM and No LLM groups for Midterm 1, with a high effect value.

We further investigate the impact of LLM usage on midterm exams by comparing the performance of High-Reliance and Low-Reliance students within the LLM group. Because the group sizes were highly imbalanced (seven vs sixteen), we did not conduct statistical tests; instead, we interpret the descriptive trends shown in Figure 7, which compares midterm exam scores between the two reliance groups across the three midterms. On average, Low-Reliance group consistently outperformed the High-Reliance group across all three midterms, with a stable performance gap of approximately 7-8% throughout the term. As a matter of fact, the Low-Reliance group performed similarly to the No LLM group, with a performance gap of just 3%.

5.2.2 *Performance Differences by LLM Usage Types*

Among students who reported using LLMs, we further examined whether their performances differed depending on the types of LLM support they relied on. Because individual students often engaged in multiple forms of LLM use

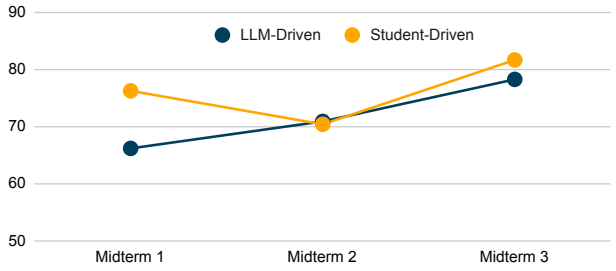


Figure 8: Comparisons of midterm exam scores between LLM-driven support group and Student-driven support group

across assignments, we conducted this analysis by classifying each student based on the three LLM support types they used most frequently among the three listed in Table 1. Those who most frequently used usage types categorized as student-driven support were assigned to the *student-driven support* group; those who most frequently used LLM usage types that are LLM-driven support were assigned to the *LLM-driven support* group; and those who most frequently used LLM usage types that are comprehension support were assigned to the *comprehension support* group. Based on this criterion, 23 students were included in the analysis: 14 were classified as LLM-driven support group and 7 as student-driven, with 2 students left unclassified due to equal frequencies across support types.

We then compared midterm exam scores between students whose LLM use was classified as LLM-driven and those classified as student-driven. Because the group sizes were highly imbalanced—similar to the comparison between the High-Reliance and Low-Reliance groups—we did not conduct statistical tests and instead observed the trends in the average midterm exam scores shown in Figure 8. As the figure illustrates, students in the student-driven group scored higher on Midterm 1, but the performance of the two groups converged in subsequent midterms. This pattern suggests that the type of support students sought—reflecting differences in how much effort they invested in contributing substantive work themselves—impacted their midterm performance early in the course, but that this effect may have diminished as students became more familiar with the course content and exam structure, similar to the comparison between LLM and No LLM groups.

6. DISCUSSION

In this section, we summarize our results regarding students' LLM usage and its impact on learning, as measured by midterm performance.

We found that only a minority of students (23 out of 68; 34%) used LLMs for the course assignments. Among these students, most (16 students) used LLMs lightly, relying on them for approximately 5–33% of their submissions (Low-Reliance group). Their LLM use also tended to be concentrated earlier in the course, primarily before the first midterm. This pattern suggests that as these students became more confident with the course material and the exam

structure, the perceived benefits of using LLMs may have decreased, reducing their incentive to rely on them. The remaining seven students used LLMs more heavily (i.e., in more than 50% of their submissions; High-Reliance group), either by using them consistently across assignments or by increasing their usage later in the course.

We observed a trend in which Low-Reliance students performed between 7% and 8% better on the three midterms than High-Reliance students, suggesting that the amount of LLM usage may be associated with differences in test performance in this context.

Our results also showed that students used LLMs in qualitatively different ways. Some forms of usage required substantial work and input from the student (student-driven support), while others relied more heavily on the LLM to generate content (LLM-driven support). We observed a trend in which students with a higher frequency of student-driven usage performed better on the midterms than students with a higher frequency of LLM-driven usage, particularly on the first midterm (a 10% difference). This finding suggests that the type of LLM usage may be another factor influencing test performance in this context.

Overall, we found a statistically significant difference in performance on the first midterm between students who reported using LLMs on the assignments preceding this exam and students who reported no LLM use, with the latter group performing better with a large effect size. This difference decreased for the subsequent two midterms and, notably, the LLM-using group improved by approximately 10% from the first midterm to the last.

Thus, although the type and amount of LLM use may have negatively affected student learning early on in the course, as the course progressed, most students appeared to calibrate their LLM use to better complement their learning processes. These findings provide encouraging evidence that the use of LLMs does not necessarily hinder learning, which is a major concern expressed within the educational community. Our results further suggest that many students are able to leverage LLMs constructively even without explicit guidance, given sufficient opportunity and time to experiment with their use. This offers initial evidence against the common concern that students will inevitably over-rely on LLMs in ways that undermine learning if left to their own devices. That said, our findings also indicate that providing guidance on effective LLM usage early in the course may help accelerate appropriate and productive adoption.

It should be noted, however, that in the course addressed in this paper, students could directly experience the impact of their LLM usage on performance through the midterms, as these assessments involved the same types of tasks as the assignments. In instructional settings where the connection between assignments and in-class exams is less explicit—and where the consequences of LLM use for exam performance are therefore less immediately observable—additional instructional scaffolding, feedback, or explicit guidance may be necessary to support effective and learning-oriented use of LLMs.

7. CONCLUSION

This work examined how students used LLMs in a research-oriented course, revealing diverse patterns of engagement and how these behaviors related to performance on an independently completed midterm exam. Our analyses showed meaningful variation in when students chose to use LLMs, the types of support they sought, and the amount of effort they contributed themselves, underscoring that LLM use is far from uniform. Overall, the results offer an encouraging view: LLM use is not inherently detrimental, and several students demonstrated that, when used constructively, these tools can support—rather than hinder—learning.

Future work can build on this study in several important ways. First, although our analysis revealed clear patterns, the dataset—particularly within certain subgroups—was relatively small, meaning that some observations should be interpreted as trends rather than definitive effects. Replicating and extending these analyses with larger dataset would help establish the robustness and generalizability of our findings. Second, future work should also explore additional factors that may shape students' engagement with LLMs. For example, prior research highlights the role of AI literacy in influencing how students use these tools [10], suggesting that students' competencies, confidence, and prior experience with AI may meaningfully interact with their usage patterns. Other individual or contextual factors—such as study strategies or prior familiarity with research-based coursework—may also play a role and warrant further investigation.

Building on our findings summarized in Section 6, we examined naturally occurring LLM usage behaviors and demonstrated how these categories relate to students' ability to perform in a new domain. By qualitatively analyzing the students' LLM-use reports to interpret their intentions and combining this with performance outcomes, we show that the manner in which students use LLMs—not merely the frequency of use—plays a meaningful role in learning. In summary, our work advances understanding of whether and how students use LLMs, and how such usage affects learning, by examining LLM engagement in a cognitively demanding, interpretation-focused context and in an authentic course setting where usage patterns emerged organically. Together, these contributions represent one step toward a more comprehensive account of student–LLM interaction across different learning domains and underscore the importance of studying LLM use in ecologically valid environments.

8. ACKNOWLEDGMENTS

This work was funded by the UBC Skylight Grant PM015111 and complies with UBC policies 81 and LR11 regarding ownership and usage of course material for pedagogical quality assurance. This work was also supported in part by the Institute for Computing, Information and Cognitive Systems (ICICS) at UBC.

9. REFERENCES

[1] J. Brender, L. El-Hamamsy, F. Mondada, and E. Bumbacher. Who's Helping Who? When Students Use ChatGPT to Engage in Practice Lab Sessions. In *International Conference on Artificial Intelligence in*

Education, pages 235–249. Springer, 2024.

[2] C. Bull and A. Kharrufa. Generative Artificial Intelligence Assistants in Software Development Education: A Vision for Integrating Generative Artificial Intelligence into Educational Practice, Not Instinctively Defending Against It. *IEEE Software*, 41(2):52–59, 2023.

[3] L. Flower and J. R. Hayes. A cognitive process theory of writing. *College Composition and Communication*, 32(4):365–387, 1981.

[4] J. Gao, S. A. Gebreegziabher, K. T. W. Choo, T. J.-J. Li, S. T. Perrault, and T. W. Malone. A Taxonomy for Human-LLM Interaction Modes: An Initial Exploration. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–11, 2024.

[5] A. Ghimire and J. Edwards. Coding with AI: How Are Tools Like ChatGPT Being Used by Students in Foundational Programming Courses. In *International Conference on Artificial Intelligence in Education*, pages 259–267. Springer, 2024.

[6] L. Giray. Prompt engineering with chatgpt: a guide for academic writers. *Annals of biomedical engineering*, 51(12):2629–2633, 2023.

[7] V. Grande, N. Kiesler, and M. A. Francisco R. Student perspectives on using a large language model (llm) for an assignment on professional ethics. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, pages 478–484. Association for Computing Machinery, 2024.

[8] A. Jelson, D. Manesh, A. Jang, D. Dunlap, Y.-H. Kim, and S. W. Lee. An empirical study to understand how students use chatgpt for writing essays. *arXiv preprint arXiv:2501.10551*, 2025.

[9] M. Kazemitabaar, X. Hou, A. Henley, B. J. Ericson, D. Weintrop, and T. Grossman. How Novices Use LLM-based Code Generators to Solve CS1 Coding Tasks in a Self-Paced Learning Environment. In *Proceedings of the 23rd Koli calling international conference on computing education research*, pages 1–12, 2023.

[10] J. Kim, S. Yu, S.-S. Lee, and R. Detrick. Students' prompt patterns and its effects in ai-assisted academic writing: Focusing on students' level of ai literacy. *Journal of Research on Technology in Education*, 58(3):638–655, 2026.

[11] S. Lau and P. Guo. From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot. In *Proceedings of the 2023 ACM Conference on International Computing Education Research-Volume 1*, pages 106–121, 2023.

[12] B. Ma, L. Chen, and S. Konomi. *Examining Student-ChatGPT Interactions in Programming Education*, pages 97–114. Springer Nature Switzerland, Cham, 2025.

[13] S. MacNeil, A. Tran, A. Hellas, J. Kim, S. Sarsa, P. Denny, S. Bernstein, and J. Leinonen. Experiences from Using Code Explanations Generated by Large Language Models in a Web Software Development E-Book. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, pages 931–937, 2023.

[14] E. Mollick and L. Mollick. Assigning AI: Seven Approaches for Students, with Prompts. *arXiv preprint arXiv:2306.10052*, 2023.

[15] I. Orozco Vasquez, R. Mahinpei, N. Elouazizi, and C. Conati. An emergent bottom-up categorization of students' llms usage in an undergraduate research course. In *International Conference on Artificial Intelligence in Education*, pages 133–139. Springer, 2025.

[16] S. Rasnayaka, G. Wang, R. Shariffdeen, and G. N. Iyer. An Empirical Study on Usage and Perceptions of LLMs in a Software Engineering Project. In *Proceedings of the 1st International Workshop on Large Language Models for Code*, pages 111–118, 2024.

[17] S. K. K. Santu and D. Feng. TELeR: A General Taxonomy of LLM Prompts for Benchmarking Complex Tasks. *arXiv preprint arXiv:2305.11430*, 2023.

[18] G. Sawalha, I. Taj, and A. Shoufan. Analyzing student prompts and their effect on ChatGPT's performance. *Cogent Education*, 11(1):2397200, 2024.

[19] S. Schulhoff, M. Ilie, N. Balepur, K. Kahadze, A. Liu, C. Si, Y. Li, A. Gupta, H. Han, S. Schulhoff, et al. The prompt report: a systematic survey of prompt engineering techniques. *arXiv preprint arXiv:2406.06608*, 2024.

[20] D. Sun, A. Boudouaia, J. Yang, and J. Xu. Investigating students' programming behaviors, interaction qualities and perceptions through prompt-based learning in chatgpt. *Humanities and Social Sciences Communications*, 11(1):1–14, 2024.

[21] A. Vadaparty, D. Zingaro, D. H. Smith IV, M. Padala, C. Alvarado, J. Gorson Benario, and L. Porter. CS1-LLM: Integrating LLMs into CS1 instruction. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, page 297–303. Association for Computing Machinery, 2024.

Confusion-Aware Rubric Optimization for LLM-based Automated Grading

Yucheng Chu
Michigan State University
chuyuch2@msu.edu

Hang Li
Michigan State University
lihang4@msu.edu

Kaiqi Yang
Michigan State University
kqyang@msu.edu

Yasemin Copur-Gencturk
University of Southern California
copurgen@usc.edu

Joseph Krajcik
Michigan State University
krajcik@msu.edu

Namsoo Shin
Michigan State University
namsoo@msu.edu

Jiliang Tang
Michigan State University
tangjili@msu.edu

ABSTRACT

Accurate and unambiguous guidelines are critical for large language model (LLM) based graders, yet manually crafting these prompts is often sub-optimal as LLMs can misinterpret expert guidelines or lack necessary domain specificity. Consequently, the field has moved toward automated prompt optimization to refine grading guidelines without the burden of manual trial and error. However, existing frameworks typically aggregate independent and unstructured error samples into a single update step, resulting in "rule dilution" where conflicting constraints weaken the model's grading logic. To address these limitations, we introduce **C**onfusion-**A**ware **R**ubric **O**ptimization (**CARO**), a novel framework that enhances accuracy and computational efficiency by structurally separating error signals. CARO leverages the confusion matrix to decompose monolithic error signals into distinct modes, allowing for the diagnosis and repair of specific misclassification patterns individually. By synthesizing targeted "fixing patches" for dominant error modes and employing a diversity-aware selection mechanism, the framework prevents guidance conflict and eliminates the need for resource-heavy nested refinement loops. Empirical evaluations on teacher education and STEM datasets demonstrate that CARO significantly outperforms existing SOTA methods. These results suggest that replacing mixed-error aggregation with surgical, mode-specific repair yields robust improvements in automated assessment scalability and precision.

Keywords

Automated Grading, Large Language Models, Automated Prompt Optimization, Confusion Matrices.

Yucheng Chu, Hang Li, Kaiqi Yang, Yasemin Copur-Gencturk, Joseph Krajcik, Namsoo Shin, and Jiliang Tang. Confusion-Aware Rubric Optimization for LLM-based Automated Grading. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 67–78. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039998

1. INTRODUCTION

The rapid shift toward personalized and competency-based learning has placed unprecedented challenges for traditional assessment frameworks. As personalized learning relies on understanding the specific needs for each learner, it necessitates high-quality formative assessment. In modern pedagogy, providing frequent and high-quality assessment feedback on students' open-ended work is essential for diagnosing misconceptions and fostering deep conceptual understanding [2, 13, 25]. Yet these benefits are hard to deliver at scale. Constructed-response tasks, such as short answers or explanations, capture students' reasoning more directly than multiple-choice formats, but they are substantially more time-consuming to assign and evaluate. Furthermore, they can exhibit nontrivial scorer disagreement even under rubric-based evaluation [19]. Notably, this disagreement often stems from the linguistic diversity and variability inherent in student answers. This "grading bottleneck" is reflected in teachers' reported workloads, where scoring and correcting student work consumes a sizable share of working time across education systems [22]. As a result, educational practice often trades off authenticity for scalability by reducing the use of open-ended items or returning feedback after long delays. These compromises weaken the instructional value of assessment and motivate automated grading as a way to preserve the pedagogical benefits of constructed responses without prohibitive cost or latency [1, 18].

Traditional methods of automated grading primarily relied on hand-crafted features, such as keyword matching, latent semantic analysis (LSA), or surface-level linguistic metrics [11]. These approaches requires task-specific feature engineering and fails to capture the semantic nuances of student reasoning, resulting in rigid evaluations that often penalize creative but correct answers [10]. The advent of deep learning introduced fine-tuning techniques using encoder-based models like BERT, which significantly improved semantic understanding and scoring accuracy [9, 20]. However, these models require extensive labeled datasets for training before applicable to the new tasks, making them impractical for dynamic classroom environments where assignments change frequently. Recent advancements demonstrate

that large language models (LLMs) overcome this data barrier. Unlike their predecessors, LLMs can leverage zero-shot and few-shot capabilities (i.e., they can operate effectively with little to no labeled data) to grade open-ended student responses with performance approaching human-to-human agreement levels [4]. Despite this promise, the transition from human grading to LLM-driven assessment is fraught with technical hurdles. In particular, reliability is heavily contingent on the quality of the grading prompt, with the instructions and rubric operationalization constraining the model's decision-making [4, 14]. Simply treating a human rubric as an instruction for an LLM is often sub-optimal; human rubrics, while intuitive for expert educators, often contain linguistic ambiguities that misalign with the causal reasoning of an LLM. This "alignment gap" necessitates extensive prompt engineering, a process that is often manual and driven by iterative trial and error, making it difficult to scale [6, 28]. Consequently, the field has moved toward *automated prompt optimization* (APO), where algorithms iteratively refine instructions based on quantitative performance feedback [29, 6].

While state-of-the-art frameworks like GradeOpt [6] leverage reflective optimization to diagnose errors, they are limited by their naive treatment of the error signal, which incurs substantial computational overhead when searching over large prompt spaces. Specifically, these methods aggregate heterogeneous error samples into a composite feedback signal, combining distinct failure types such as hallucination, excessive strictness, and rubric misunderstanding into a single optimization step without separation. This confounding of error types leads to *rule dilution*. That is, by attempting to address a mixture of conflicting signals simultaneously, the optimizer generates generalized, weak instructions that fail to enforce clear decision boundaries. Consequently, to find even a marginally improved prompt, these methods rely on resource-heavy exploration, generating and evaluating large pools of candidates.

To address this, we introduce **CARO** (**C**onfusion-**A**ware **R**ubric for **O**ptimization), a framework that achieves superior accuracy and computational efficiency by structurally separating error signals. We observe that grading errors are not random mixtures, but structured clusters. CARO leverages the confusion matrix to decompose the monolithic error signal into distinct **error modes**, eliminating the inefficiency of trying to address conflicting error types simultaneously. Specifically, CARO targets the single **dominant error mode** at each step (e.g., a specific tendency to confuse a score of 0 with 1), transforming the optimization into a structured **diagnosis-and-repair** pipeline. By isolating a single, clear failure pattern, CARO generates targeted "rule patches" that are specific and highly effective. Contributing to its precision, CARO produces robust prompts with significantly fewer iterations and lower token usages compared to the existing APO baselines. Our empirical results confirm that CARO significantly outperforms these frameworks across multiple datasets while utilizing the same cost-effective backbone (`GPT-4o-mini`[1]).

[1]`https://platform.openai.com/docs/models/gpt-4o-mini`

2. RELATED WORK

Automated short-answer grading (ASAG). Early ASAG systems relied on hand-crafted lexical features and latent semantic analysis to predict scores [21]. The field later transitioned to neural architectures, utilizing recurrent models and Transformer encoders trained on graded corpora to capture deeper semantic dependencies [26, 12]. Despite their improved accuracy, these supervised pipelines require extensive labeled data for each specific question and lack the interpretability of rubric-based assessment. This limitation motivated the shift toward Large Language Models (LLMs), which offer zero-shot flexibility and the capacity to generate human-readable feedback alongside scores.

LLM-based grading and prompt optimization. Recent studies demonstrate that LLMs can achieve high agreement with human graders, though performance remains heavily dependent on rubric specificity and prompt design [14, 3]. While frameworks like RAG can improve domain grounding [5], the challenge of manual tuning has accelerated the adoption of Automatic Prompt Optimization (APO). APO treats instructions as trainable parameters, ranging from iterative meta-optimization in OPRO and DSPy [27, 16] to the "natural-language gradients" used in ProTeGi [23]. In the educational domain, GradeOpt applies these principles to refine grading guidelines via self-reflection [6]. However, these systems generally aggregate heterogeneous errors into a single feedback signal, often obscuring the distinct failure patterns critical for precise assessment.

3. PROBLEM FORMULATION

We formalize the task of automated rubric optimization as a discrete search problem over the space of natural language instructions. Unlike fine-tuning LLM parameters, which requires extensive labeled data and computational resources, we focus on optimizing the rubric prompt in the text space. This approach leverages the in-context learning capability of pre-trained LLMs while maintaining interpretability so that graders can inspect and validate the optimized rubric before deployment. Let $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N}$ be a dataset of student responses x and ground-truth scores $y \in \mathcal{Y}$, where $\mathcal{Y} = \{0, 1, \ldots, K-1\}$ represents an ordinal grading scale. An LLM-based grader can be defined as a function f_θ that estimates the probability $P_\theta(y|x, \mathcal{P})$ of a score y given a response x and a rubric prompt $\mathcal{P}$. The predicted score is obtained as $\hat{y} = \arg\max_{y \in \mathcal{Y}} P_\theta(y|x, \mathcal{P})$. The objective of *Automated Prompt Optimization (APO)* is to identify an optimal prompt $\mathcal{P}^*$ that maximizes a performance metric $\mathcal{M}$ (e.g., classification accuracy or quadratic weighted kappa) over a validation dataset:

$$\mathcal{P}^* = \arg\max_{\mathcal{P} \in \mathcal{S}} \mathcal{M}\big(\{\hat{y}_n, y_n\}_{n=1}^{N}\big) \tag{1}$$

where $\mathcal{S}$ denotes the discrete search space of candidate prompts. Since the optimization targets the natural language prompt directly to ensure the rubric remains human-readable, standard differentiable optimization methods applicable to continuous parameters are not viable. Instead, APO methods approximate the gradient by generating textual feedback g based on the discrepancy between predicted scores $\hat{y}$ and true scores y, effectively performing a semantic descent step

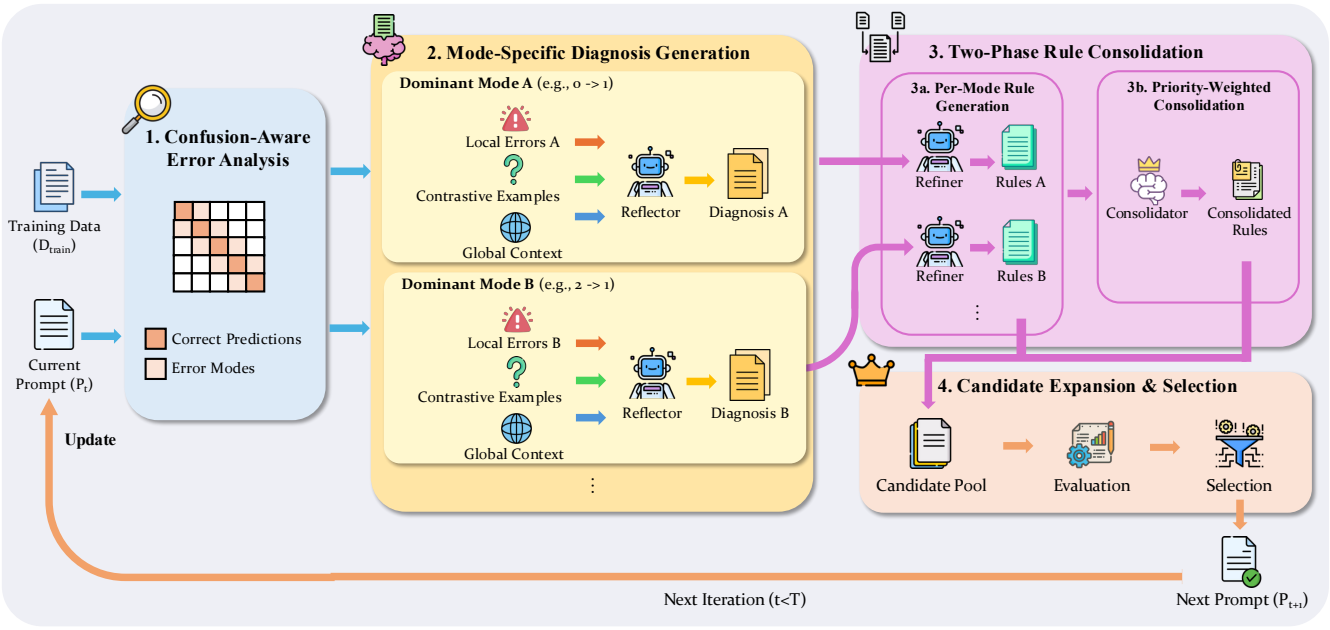


Figure 1: Framework of the CARO algorithm.

$\mathcal{P}_{t+1} \leftarrow \text{Update}(\mathcal{P}_t, g)$.

The goal of our work is to address the structural inefficiency problem of the feedback signal g when generated from aggregated error data. In the context of ordinal grading, errors are not uniform; they constitute distinct directional error modes captured by the confusion matrix $\mathbf{C}$, where an entry C_{ij} represents the frequency of true class i being misclassified as j. Existing APO frameworks typically aggregate a batch of heterogeneous error samples by mixing instances of $i \to j$ confusion with $k \to l$ confusion to generate a single, composite feedback signal. We define this aggregation as the source of *rule dilution*. By exposing multiple error modes simultaneously, such methods require the optimizer model to solve a multi-objective task that demands high reasoning capacity to disentangle competing patterns. To address this challenge, we adopt a multi-agent decomposition strategy that separates the complex task into focused sub-problems: (1) a *Mode-Specific Reflector* that analyzes each confusion cell (i, j) independently to generate targeted feedback, and (2) a *Consolidator* that synthesizes per-mode refinements into a coherent, non-redundant rule set while maintaining global consistency.

Furthermore, this optimization landscape is characterized by high inter-dependency between grading criteria, leading to *mode interference*. Addressing a specific error mode C_{ij} in isolation can act as a perturbation to the decision boundary between classes i and j. Due to the semantic continuity of the rubric, this perturbation often propagates to adjacent boundaries, inadvertently increasing the error rate in a connected mode C_{jk} or C_{ji}. This phenomenon creates a non-monotonic optimization trajectory where the reduction of one error type systematically amplifies another. Consequently, the optimization objective shifts from merely minimizing the total error count to identifying a structured sequence of rule refinements that effectively *decomposes* the confusion matrix $\mathbf{C}$ without causing regression in previously stabilized modes. Our framework addresses this via a decomposition strategy that first generates mode-specific rules with global context awareness, then consolidating them with explicit priority ordering and cross-mode safety checks.

4. METHOD

We propose CARO (**C**onfusion-**A**ware **R**ubric for **O**ptimization), a framework designed to overcome the limitations of aggregating feedback in automated rubric optimization. Existing methods often suffer from rule dilution, where conflicting error signals lead to vague instructions. To address this, CARO structurally decomposes the optimization problem using confusion matrices, treating grading errors not as random noise but as distinct, diagnosable modes. Our approach operates in three iterative phases: confusion-aware error analysis, mode-specific gradient generation, and priority-weighted rule consolidation.

4.1 Framework Overview

Figure 1 illustrates the complete CARO pipeline. To efficiently navigate the extensive search space of natural language prompts, CARO employs an iterative, minibatch-based optimization loop. In each iteration t, rather than evaluating on the entire training set, which is computationally expensive and may obscure minority error patterns, we sample a focused subset of data (a *minibatch*) to estimate the current rubric's performance. This iterative design allows the framework to progressively refine the rubric, addressing the most prominent error modes in early rounds before fine-tuning for edge cases.

The framework operates as a closed-loop optimization system with four interconnected components. **Components 1–3** form the core optimization loop within a single round:

(1) Confusion-Aware Error Analysis grades the minibatch and extracts structured error patterns; (2) Mode-Specific Gradient Generation produces targeted feedback for each confusion cell using a Reflector LLM; and (3) Two-Phase Rule Consolidation synthesizes per-mode rules into a unified rubric. These three components pass information sequentially within each optimization round. **Component 4** (Candidate Selection and Sampling) operates at a different granularity as it governs the *inter-round* dynamics. It acts as a bridge between iterations by determining which candidate prompts survive to round $t+1$ and constructing the specific training minibatch that will be used to evaluate it. This separation reflects the algorithmic structure: Components 1–3 generate candidate prompts within a single round, while Component 4 decides *which* candidates propagate across rounds and *what data* they will be evaluated on next. The dashed connection in the figure indicates this temporal decoupling.

The complete procedure is summarized in Algorithm 1. Given an initial rubric prompt $\mathcal{P}_0$, a training dataset $\mathcal{D}_{\text{train}}$, and a LLM predictor f_θ, our objective is to iteratively optimize the prompt to minimize classification errors while ensuring stability across different score categories. The optimization process functions as a directed search. At each iteration t, we first evaluate the current prompt on a minibatch to generate a confusion matrix $\mathbf{C}^{(t)}$ (Algorithm 1, lines 4–7). From this matrix, we identify the top-K error modes, denoted as $\mathcal{M} = \{(i_1, j_1), \ldots, (i_K, j_K)\}$, ranked by frequency (line 8). Unlike standard methods that aggregate these errors, CARO generates distinct gradients for each mode (lines 10–16), which are subsequently consolidated into coherent rule updates (lines 18–23). This cycle repeats until convergence or a maximum step count T is reached.

4.2 Confusion-Aware Error Analysis

The diagnostic phase begins by explicitly evaluating the current rubric $\mathcal{P}^{(t)}$ on the sampled minibatch to generate a set of predictions and their corresponding reasoning traces. By comparing these predictions $\hat{y}$ against ground truth y, we construct a confusion matrix $\mathbf{C}^{(t)}$ (lines 4–7). Based on this performance data, we construct a comprehensive context for the optimizer. For each identified confusion mode $(i, j) \in \mathcal{M}$, we assemble three distinct types of information to ground the model's reasoning. First, we extract **local error context**. This consists of specific examples $\mathcal{E}_{i\to j} = \{(x, y, \hat{y}, r)\}$ where the true label y is i, but the model predicted $\hat{y} = j$, accompanied by the model's reasoning trace r (line 12). Second, to establish clear decision boundaries, we sample *contrastive correct examples*. These are instances correctly classified as i or j but which exhibit high model uncertainty (line 13). We quantify this uncertainty using a misconfidence score [6]:

$$\text{misconf}(x) = \begin{cases} -\log P_\theta(\hat{y}|x) & \text{if } \hat{y} = y \\ \left| \frac{\log P_\theta(\hat{y}|x)}{\log P_\theta(y|x)} \right| & \text{if } \hat{y} \neq y \end{cases} \tag{2}$$

where $P_\theta(c|x)$ denotes the predicted probability for class c. High misconfidence in correct predictions signals boundary cases that provide high-value discriminative signals. Finally, we provide **global context** via the full confusion matrix summary, GlobalContext($\mathbf{C}$). This summary includes the error distribution across all classes, ensuring that the optimizer remains aware of the global performance landscape and avoids overfitting to a single error type at the expense of others.

Algorithm 1 CARO: Confusion-Aware Rubric Optimization

Require: Initial rubric prompt $\mathcal{P}_0$, training data $\mathcal{D}_{\text{train}}$, LLM predictor f_θ, rounds T, beam size B, top-K modes, diversity weight λ
Ensure: Optimized rubric prompt $\mathcal{P}^*$
1: $\mathcal{P}^{(0)} \leftarrow \{\mathcal{P}_0\}$ {Initialize candidate set}
2: **for** $t = 1$ **to** T **do**
3: `// 1. Confusion-Aware Error Analysis`
4: Minibatch$^{(t)} \leftarrow$ SBERTSample(TopMisconf$^{(t-1)}, \mathcal{D}_{\text{train}}$)
5: **for** each $\mathcal{P} \in \mathcal{P}^{(t-1)}$ **do**
6: Evaluate $\mathcal{P}$ on Minibatch$^{(t)}$ to obtain $\mathbf{C}^{(t)}_{\mathcal{P}}$
7: **end for**
8: $\mathcal{M} \leftarrow$ TopKModes($\mathbf{C}^{(t)}, K$) {Identify top-K error modes}
9: `// 2. Mode-Specific Diagnosis Generation`
10: $\mathcal{G} \leftarrow \emptyset$
11: **for** each mode $(i, j) \in \mathcal{M}$ **do**
12: $\mathcal{E}_{i\to j} \leftarrow$ Extract error examples with true label i, predicted j
13: $\mathcal{E}_i^+, \mathcal{E}_j^+ \leftarrow$ Sample high-misconfidence correct examples
14: $g_{i\to j} \leftarrow$ Reflector(Rules$^{(t)}, \mathcal{E}_{i\to j}, \mathcal{E}_i^+, \mathcal{E}_j^+, \mathbf{C}^{(t)}, \mathcal{M}$)
15: $\mathcal{G} \leftarrow \mathcal{G} \cup \{g_{i\to j}\}$
16: **end for**
17: `// 3. Two-Phase Rule Consolidation`
18: $\mathcal{R} \leftarrow \emptyset$
19: **for** each $(i, j) \in \mathcal{M}$ **do**
20: $r_{i\to j} \leftarrow$ Refiner(Rules$^{(t)}, g_{i\to j}, \mathcal{M} \setminus \{(i, j)\}, \mathbf{C}^{(t)}$)
21: $\mathcal{R} \leftarrow \mathcal{R} \cup \{(r_{i\to j}, (i, j))\}$
22: **end for**
23: $r_{\text{consolidated}} \leftarrow$ PriorityConsolidate($\mathcal{R}, \mathbf{C}^{(t)}$)
24: `// 4. Candidate Expansion and Selection`
25: $\mathcal{P}_{\text{new}} \leftarrow \{\mathcal{P} \oplus r : \mathcal{P} \in \mathcal{P}^{(t-1)}, r \in \mathcal{R} \cup \{r_{\text{consolidated}}\}\}$
26: Score each $\mathcal{P} \in \mathcal{P}_{\text{new}}$ via UCB-based evaluation
27: $\mathcal{P}^{(t)} \leftarrow$ DiverseSelect($\mathcal{P}_{\text{new}}, B, \lambda$) {Eq. 5}
28: **end for**
29: **return** $\arg\max_{\mathcal{P} \in \mathcal{P}^{(T)}} \kappa(\mathcal{P}, \mathcal{D}_{\text{val}})$

4.3 Mode-Specific Feedback Generation

To repair the identified errors, CARO avoids the "rule dilution" of baseline methods by generating separate diagnosis for each error mode (lines 10–16). We employ a Reflector LLM to analyze the confusion pattern (i, j) specifically. The gradient generation process is formalized as:

$$g_{i\to j} = \text{Reflector}(\text{Rules}^{(t)}, \mathcal{E}_{i\to j}, \mathcal{E}_i^+, \mathcal{E}_j^+, \text{GlobalContext}(\mathbf{C}), \mathcal{M}) \tag{3}$$

Here, $\mathcal{E}_i^+$ and $\mathcal{E}_j^+$ represent the contrastive correct examples. We have shown a demonstration of the Reflector's prompt in Figure 2.

As exemplified by Figure 3, the resulting gradient $g_{i\to j}$ provides a structured diagnosis, identifying the root cause of the confusion, creating discriminative criteria to distinguish score i from j, and performing a safety check to predict potential negative side effects on other score categories.

4.4 Two-Phase Rule Consolidation

Reflector Prompt for Mode $(0 \to 1)$ Confusion

[GLOBAL CONTEXT]

True\Pred	**0**	**1**	**2**
0	6	**25**	7
1	0	9	10
2	0	0	7

Error Distribution:
- $0 \to 1$: 25 errors (59.5%) ← *CURRENT FOCUS*
- $1 \to 2$: 10 errors (23.8%)
- $0 \to 2$: 7 errors (16.7%)

[LOCAL ERROR EXAMPLES] $\mathcal{E}_{0\to1}$

Example 1: "*They understand it partially because I understand what they are doing but I would show them that setting it up as a proportion makes more sense.*"
True: 0 **Pred:** 1 **Reasoning:** Teacher acknowledges some understanding but lacks specific analysis...

Example 2: "*His explanation doesn't really refer to the equivalent ratios. Although the actual work seems to show he has some understanding.*"
True: 0 **Pred:** 1 **Reasoning:** Vague acknowledgment without depth...

[CONTRASTIVE CORRECT EXAMPLES] $\mathcal{E}_0^+, \mathcal{E}_1^+$

Correctly Classified as 0	**Correctly Classified as 1**
"*Student A has a limited understanding because he based his answer on the dollar amount.*" → Incorrect/generic analysis	"*The student looked at price versus amount to find the unit rate.*" → Procedural description of student work

[TASK] Analyze WHY the $0 \to 1$ confusion occurs. Identify misleading patterns and propose rule fixes.

Figure 2: Example Reflector prompt for analyzing the dominant error mode $(0 \to 1)$. The prompt provides: (1) global context via the full confusion matrix, (2) local error examples with model reasoning traces, and (3) contrastive correct examples to establish decision boundaries.

A critical innovation of CARO is the consolidation strategy, which converts discrete error diagnostics into a unified, coherent rubric (lines 18–23). We execute this in two phases to balance specificity with logical consistency.

Phase 1: Per-Mode Rule Generation. We translate the gradients into specific rule refinements (lines 19–22). For each mode (i, j), a Refiner module generates a rule update $r_{i\to j}$. Crucially, the Refiner receives information about all other targeted error modes $\mathcal{M}\backslash\{(i, j)\}$ as additional context. This *cross-mode awareness* enables the Refiner to anticipate potential interactions. We show an example of the prompt to Refiner in Figure 4: when generating a rule to prevent $0 \to 1$ misclassification, it can avoid phrasing that would inadvertently increase $1 \to 2$ or $0 \to 2$ errors if these modes are also being addressed in the same round.

Phase 2: Priority-Weighted Consolidation. Rather than simply concatenating all new rules, which often leads to bloated or contradictory prompts, we organize rules based on the frequency of the error modes they address (line 23). The consolidated rule set for the next iteration is derived as:

$$\text{Rules}^{(t+1)} = \text{Consolidate}(\{r_{i\to j}\}_{(i,j)\in\mathcal{M}}, \mathbf{C}) \tag{4}$$

As shown in Figure 5, the consolidation assigns explicit precedence levels: the dominant error mode (highest C_{ij}) receives Priority 1 with 2–3 detailed rules; secondary modes receive Priority 2+ with single-sentence guards. When rules from different priority levels suggest different scores for an edge case, we insert explicit *tie-breaker directives*, e.g., "If Priority 1 criteria suggest score X but Priority 2 criteria suggest score Y, assign score X unless [specific exception condition]." This prioritization ensures that the optimization creates a stable "diagnosis-and-repair" pipeline, systematically eliminating the most significant failure patterns before addressing long-tail errors.

4.5 Candidate Selection and Sampling

The final phase bridges the current iteration to the next by performing two distinct operations (lines 24–26): (1) *candidate selection*, which determines which prompt variants effectively resolved errors and should survive to the next round, and (2) *minibatch sampling*, which constructs the optimal training data for the subsequent iteration to prevent overfitting and ensure robust evaluation.

Diversity-Aware Candidate Selection. CARO maintains a beam of B candidate prompts across iterations. Each candidate prompt $\mathcal{P}$ generated in "Two-Phase Rule Consolidation" is tagged with its target mode $m(\mathcal{P}) \in \mathcal{M}$, indicating which confusion cell it was designed to address. To prevent the beam from collapsing to variants of a single mode fix, we employ a greedy selection strategy that balances empirical performance with mode coverage:

$$\text{SelectScore}(\mathcal{P}) = \widetilde{\kappa}(\mathcal{P}) + \lambda \cdot \mathbb{1}[m(\mathcal{P}) \notin \mathcal{S}] \tag{5}$$

where $\widetilde{\kappa}(\mathcal{P})$ is the min-max normalized Cohen's κ score [7], $\mathcal{S}$ represents the set of error modes already covered by previously selected candidates in the current selection round, and λ is a hyperparameter controlling the diversity bonus. The indicator function $\mathbb{1}[\cdot]$ awards a bonus to candidates targeting modes not yet represented in the selected set $\mathcal{S}$, encouraging exploration of diverse rule modifications.

Reflector Output: Diagnosis $g_{0\to1}$

Root Cause:
The classifier misinterprets teachers' acknowledgment of "some understanding" as sufficient for Score 1. Phrases like "valid reasoning" or "understands partially" mislead the classifier despite lacking analytical depth.

Misleading Patterns:
- Positive language ("some understanding", "valid reasoning")
- Missing specific evidence from student work
- Vague acknowledgments without multiplicative analysis

Why These Are 0, Not 1:
Per rubric: Score 0 for responses that provide *generic* or *absent* analysis. Simply stating "valid reasoning" $\neq$ analyzing mathematical thinking.

Proposed Rule Fix:
❶ Responses acknowledging partial understanding **must** include specific examples demonstrating multiplicative relationships.
❷ Vague acknowledgments without detailed analysis $\Rightarrow$ Score 0.

Safety Check:
This fix reinforces depth requirements, which aligns with addressing $1 \to 2$ errors (also require specificity). Will not increase other error modes.

Figure 3: Structured diagnosis output from the Reflector for mode $(0 \to 1)$. The output identifies root causes, misleading patterns, and proposes targeted rule modifications with safety considerations.

Semantic Minibatch Sampling. Instead of random minibatch construction, we use an SBERT [24]-based similarity selector to focus optimization effort on challenging regions of the input space. We retrieve training examples that are semantically similar to the high-misconfidence instances identified in previous rounds:

$$\text{Minibatch}^{(t)} = \bigcup_{x \in \text{TopMisconf}^{(t-1)}} \text{kNN}_k(x, \mathcal{D}_{\text{train}}) \quad (6)$$

where kNN_k retrieves the k nearest neighbors based on SBERT embeddings. This ensures that refined rules are stress-tested against examples where the current rubric is most ambiguous, accelerating convergence on difficult boundary cases.

4.6 Inference with Optimized Prompt

Upon the termination of the optimization procedure, we select the rubric prompt $\mathcal{P}^*$ that maximizes the evaluation metric (Cohen's κ) on the validation set:

$$\mathcal{P}^* = \arg\max_{\mathcal{P} \in \mathcal{P}^{(T)}} \kappa(\mathcal{P}, \mathcal{D}_{\text{val}}) \quad (7)$$

This globally optimized prompt serves as the fixed instruction set for the final inference phase on the held-out test dataset $\mathcal{D}_{\text{test}}$. For a given student response $x \in \mathcal{D}_{\text{test}}$, the predicted score is obtained as:

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} P_\theta(y \mid x, \mathcal{P}^*) \quad (8)$$

where $P_\theta(y \mid x, \mathcal{P}^*)$ denotes the probability assigned by the LLM predictor f_θ to score y given the student response x and the optimized rubric $\mathcal{P}^*$. The model generates a completion

Refiner Prompt for Rule Generation

[INPUT]
- Current rules: $\text{Rules}^{(t)}$
- Diagnosis: $g_{0\to1}$ (from Reflector)
- Error examples: $\mathcal{E}_{0\to1}$
- Other modes: $\{(1,2),(0,2)\}$ ← *Cross-mode awareness*

[CONSTRAINTS]
❶ Fix must target $0 \to 1$ confusion specifically
❷ Must NOT break existing correct classifications
❸ Must be COMPATIBLE with fixes for $(1 \to 2)$, $(0 \to 2)$
❹ Edit budget: `medium` (2–3 new rules)

[OUTPUT FORMAT]
Generate improved rules that:
- Include clear distinguishing criteria
- Use specific patterns from error analysis
- State safety considerations for other modes

Figure 4: Refiner prompt structure showing how cross-mode awareness is incorporated. The Refiner receives information about all active error modes to generate compatible rule fixes.

that includes both a reasoning trace r and a final numerical score, extracted via regular expression parsing:

$$(\hat{y}, r) = \text{Parse}\big(f_\theta(x \oplus \mathcal{P}^*; \tau = 0)\big) \quad (9)$$

By enforcing the reasoning trace through the structure of $\mathcal{P}^*$, the inference process provides not only a quantitative grade but also an interpretable justification that reflects the resolved decision boundaries established during optimization.

5. EXPERIMENTS

5.1 Experimental Setup

Datasets. We evaluate CARO on three educational grading datasets spanning K-12 science education and teacher education:

Interaction Dataset ($\mathcal{D}_I$): This dataset contains 252 high school student responses to a physics assessment item on electrical interactions [15]. Each response is evaluated against 11 binary rubric categories (labeled 0/1), where 0 indicates the requirement is "not met" and 1 indicates it is "met." The categories assess distinct aspects of student understanding, ranging from basic concept recognition to complex reasoning about charge interactions.

Elementary Item Response Dataset ($\mathcal{D}_E$): This dataset comprises elementary student responses to five science assessment tasks graded on an ordinal scale $\{0, 1, 2, 3, 4\}$. The tasks assess foundational science concepts with varying levels of complexity.

Teacher Education Dataset ($\mathcal{D}_T$): This dataset evaluates teachers' pedagogical content knowledge in mathematics education [8]. It contains eight grading tasks organized into three knowledge dimensions:

- **KSMT** (Knowledge of Students' Mathematical Thinking): Tasks T1–T3 assess teachers' ability to interpret

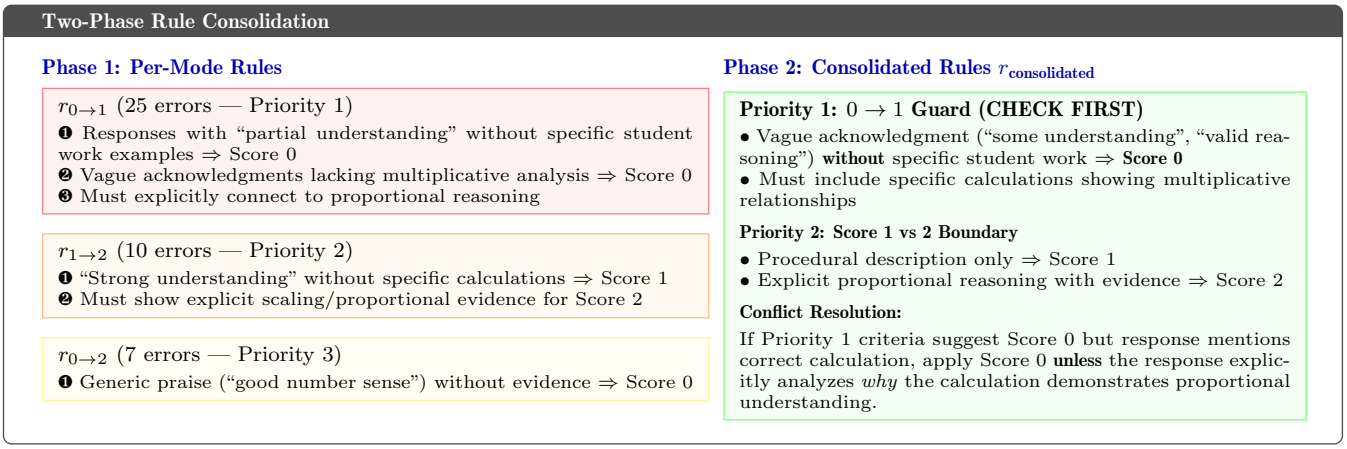


Figure 5: Two-phase rule consolidation. Phase 1 generates per-mode rules with cross-mode awareness. Phase 2 synthesizes them into a priority-weighted consolidated ruleset with explicit conflict resolution directives.

student mathematical reasoning.

- **KMT** (Knowledge of Mathematical Tasks): Tasks T4–T6 evaluate understanding of task design and cognitive demand.
- **KT** (Knowledge of Teaching): Tasks T7–T8 assess pedagogical decision-making.

All tasks in $\mathcal{D}_T$ use ordinal labels $\{0, 1, 2\}$, representing absent, partial, and complete understanding respectively.

We partition each dataset into train/validation/test splits using a 7:1:2 ratio following [6]. Dataset statistics are summarized in Table 1.

Table 1: A summary of statistics of the datasets

Dataset	Domain	#Samples	Scores
$\mathcal{D}_I$	Interaction	252	{0,1}
$\mathcal{D}_T$	Teacher Education	224∼236	{0,1,2}
$\mathcal{D}_E$	Elementary	199∼434	{0,1,2,3,4}

Baselines. We compare CARO against two baselines. First is **Naive Prompting**, which uses direct zero-shot prompting with an expert-written rubric, without any optimization. Secondly, we compare with **GradeOpt** [6], a state-of-the-art automatic prompt optimization framework that iteratively refines rubrics using aggregated error feedback.

Implementation Details. We use `GPT-4o-mini` as both the grading predictor f_θ and the optimization agents (Reflector and Refiner) across all methods for fair comparison. For CARO, we set: top-K error modes $K = 4$, beam size $B = 4$, diversity weight $\lambda = 0.3$, and maximum rounds $T = 6$. The edit budget is set to "medium" (2–3 rule modifications per mode). For GradeOpt, we use the default hyperparameters from [6] and set the maximum rounds to $T = 6$ to align with computational constraints of CARO. To mitigate the inherent stochasticity of LLM generation and ensure the reliability of our findings, we conducted 3 independent runs for all methods (CARO and baselines) across each task. The reported results represent the average performance on the held-out test set across these 3 trials.

Evaluation Metrics. We report classification accuracy (Acc) and Cohen's κ coefficient [7]. While accuracy measures overall correctness, κ accounts for class imbalance and chance agreement, making it the primary metric for grading reliability assessment. A $\kappa > 0.6$ is generally considered substantial agreement [17].

5.2 Main Results

Teacher Education Dataset ($\mathcal{D}_T$). Table 2 presents results on the teacher education dataset. CARO achieves consistent improvements across all eight tasks, with an average accuracy improvement of +11% over GradeOpt and +41% over Naive prompting.

Table 2: Performance Comparison for Teacher Education Dataset $\mathcal{D}_T$

Type	Task	Naive		GradeOpt		CARO	
		Acc	κ	Acc	κ	Acc	κ
KSMT	T1	0.40	0.20	0.61	0.41	0.67	0.47
KSMT	T2	0.58	0.42	0.74	0.61	0.79	0.69
KSMT	T3	0.39	0.09	0.51	0.25	0.55	0.30
KSMT	**Avg**	**0.46**	**0.24**	**0.62**	**0.42**	**0.67**	**0.49**
KMT	T4	0.30	0.05	0.55	0.28	0.76	0.57
KMT	T5	0.59	0.28	0.60	0.31	0.74	0.46
KMT	T6	0.68	0.44	0.76	0.58	0.76	0.59
KMT	**Avg**	**0.52**	**0.26**	**0.64**	**0.39**	**0.75**	**0.54**
KT	T7	0.45	0.29	0.74	0.60	0.77	0.65
KT	T8	0.60	0.34	0.70	0.48	0.72	0.52
KT	**Avg**	**0.52**	**0.32**	**0.72**	**0.54**	**0.75**	**0.59**
Overall	**Avg**	**0.51**	**0.27**	**0.65**	**0.44**	**0.72**	**0.53**

The most substantial gains occur on tasks with severe baseline confusion patterns. For instance, on T4 (KMT), CARO improves κ from 0.28 (GradeOpt) to 0.57, which is a relative improvement of 104%. This task exhibits significant $0 \to 1$ confusion in the baseline, which CARO directly addresses through mode-specific gradient generation. Similarly, T5 shows a κ improvement from 0.31 to 0.46, demonstrating CARO's effectiveness on tasks where GradeOpt's aggregate

feedback fails to resolve specific confusion patterns. Notably, CARO's improvements are most pronounced in the KMT dimension, where grading criteria involve subtle distinctions between "no understanding" and "partial understanding", which is precisely the type of adjacent-class confusion that CARO is designed to resolve.

Interaction Dataset ($\mathcal{D}_I$). Table 3 shows results on the binary classification tasks. Despite the simpler label space, several categories exhibit class imbalance that inflates accuracy while depressing κ. CARO achieves the highest κ on 10 of 11 categories. The improvements are particularly striking on challenging categories:

- **Category 3**: κ improves from 0.61 (GradeOpt) to 0.89, a 46% relative gain.
- **Category 4**: κ improves from 0.36 to 0.62, resolving severe confusion in this low-agreement task.
- **Category 8**: Accuracy jumps from 0.44 to 0.74 (+68%), with κ improving from 0.04 to 0.12.

Categories 6, 8, and 9 exhibit persistently low κ despite high accuracy improvements, indicating extreme class imbalance where the minority class remains difficult to identify. Nevertheless, CARO substantially outperforms baselines in accuracy, suggesting improved discrimination even when κ is constrained by distributional factors.

Table 3: Performance Comparison for Interaction Dataset $\mathcal{D}_I$

Category	Naive		GradeOpt		CARO	
	Acc	κ	Acc	κ	Acc	κ
1	0.94	0.79	0.96	0.84	0.98	0.92
2	0.86	0.71	0.88	0.76	0.93	0.87
3	0.82	0.52	0.85	0.61	0.95	0.89
4	0.52	0.19	0.64	0.36	0.81	0.62
5	0.62	0.18	0.95	0.72	0.96	0.78
6	0.42	0.03	0.73	0.09	0.87	0.21
7	0.77	0.11	0.95	0.48	0.97	0.54
8	0.22	0.03	0.44	0.04	0.74	0.12
9	0.15	-0.04	0.27	0.01	0.90	-0.02
10	0.42	0.14	0.72	0.41	0.82	0.47
11	0.60	0.14	0.90	0.50	0.94	0.64
Avg	**0.58**	**0.25**	**0.75**	**0.43**	**0.90**	**0.55**

Table 4: Performance Comparison for EIR Dataset $\mathcal{D}_E$

Task	Naive		GradeOpt		CARO	
	Acc	κ	Acc	κ	Acc	κ
T1	0.16	-0.08	0.63	0.35	0.63	0.39
T2	0.49	0.30	0.74	0.64	0.75	0.66
T3	0.57	0.25	0.79	0.67	0.81	0.70
T4	0.48	0.20	0.71	0.57	0.74	0.62
T5	0.50	0.04	0.70	0.46	0.73	0.56
Avg	**0.44**	**0.14**	**0.71**	**0.54**	**0.73**	**0.59**

Elementary Item Response Dataset ($\mathcal{D}_E$). Table 4 presents results on the elementary response dataset. CARO achieves consistent improvements across all five tasks, with an average κ of 0.59 compared to 0.54 for GradeOpt and 0.14 for Naive prompting. The most notable improvement occurs on T5, where κ increases from 0.46 (GradeOpt) to 0.56, which represents a 22% relative improvement. This task involves distinguishing between partial and complete understanding, a boundary that benefits from CARO's contrastive example selection.

Table 5: Summary of average performance across datasets.

Dataset	Naive		GradeOpt		CARO	
	Acc	κ	Acc	κ	Acc	κ
$\mathcal{D}_I$ (avg)	0.58	0.25	0.75	0.43	**0.90**	**0.55**
$\mathcal{D}_T$ (avg)	0.51	0.27	0.65	0.44	**0.72**	**0.53**
$\mathcal{D}_E$ (avg)	0.44	0.14	0.71	0.54	**0.73**	**0.59**
Overall	0.51	0.22	0.70	0.47	**0.78**	**0.56**

Summary of Improvements. Table 5 summarizes the aggregated performance across all datasets. On average, CARO improves accuracy from 0.51 (Naive prompting) to 0.78, representing a 53% increase compared to GradeOpt (37% increase from 0.51 to 0.70). CARO achieves an overall average κ of 0.56, representing a 19% relative improvement over GradeOpt (0.47) and a 155% improvement over Naive prompting (0.22).

5.3 Ablation Studies

We conduct ablation studies to analyze the key design choices and advantages of CARO.

Per-Mode vs. Consolidated Rules. As described in Section 3.4, CARO generates two types of candidate rules during the Two-Phase Rule Consolidation: (1) **per-mode rules** $r_{i \to j}$ that target individual confusion cells, and (2) **consolidated rules** $r_{\text{consolidated}}$ that synthesize all per-mode rules into a unified, priority-weighted ruleset via Eq. 4. Both types are included in the candidate pool (Algorithm 1, line 24), and the diversity-aware selection mechanism (Eq. 5) determines which candidates propagate to subsequent rounds. Table 6 reports which candidate type achieves the best validation κ across all tasks.

We observe that per-mode rules are selected as the final best prompt in over 90% of cases across all datasets. This indicates that targeted fixes for specific confusion cells typically outperform holistic rewrites that attempt to address multiple error patterns simultaneously. The consolidated rules, despite being carefully synthesized with priority weighting and conflict resolution directives, tend to dilute the specificity needed for precise decision boundaries. However, consolidated rules are occasionally preferred (6–8% of cases) in tasks with strongly interacting error modes, where coherent integration prevents the rule conflicts that can arise from independently applying multiple per-mode fixes.

Convergence Analysis. Figure 6 shows the optimization trajectory across rounds for a representative task T4 in $\mathcal{D}_I$

Table 6: **Ablation: Per-mode vs. consolidated rule selection. Values show the percentage of final best prompts using each strategy.**

Dataset	Per-Mode	Consolidated	Best κ
$\mathcal{D}_I$	93.9%	6.1%	Per-Mode
$\mathcal{D}_T$	91.7%	8.3%	Per-Mode
$\mathcal{D}_E$	93.3%	6.7%	Per-Mode
Avg	93.0%	7.0%	-

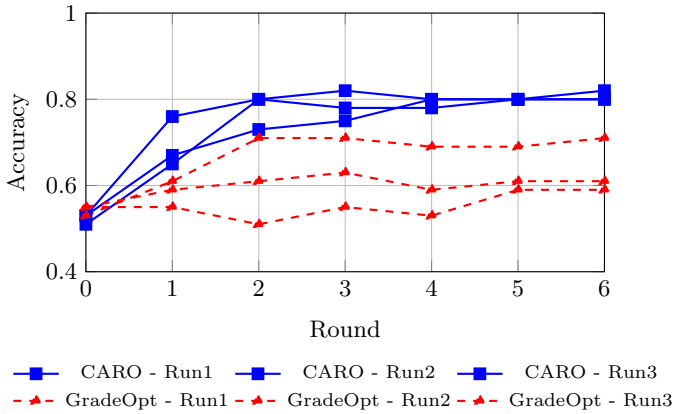


Figure 6: Convergence comparison on T4 ($\mathcal{D}_I$). CARO achieves faster convergence and higher final accuracy.

Table 7: **API cost comparison (average per task). Costs estimated using GPT-4o-mini pricing ($0.15/1M input, $0.60/1M output tokens).**

Method	Rounds	API Calls	Est. Cost
GradeOpt	6	∼40,000	∼$5
CARO	6	∼8,000	∼$2
Reduction	–	80%	60%

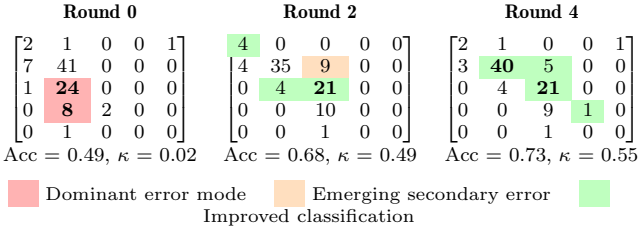


Figure 7: Confusion matrix progression on T5 ($\mathcal{D}_E$). Rows represent true labels; columns represent predictions. CARO's iterative refinement first resolves the dominant $2 \to 1$ collapse (Round 0–2), then addresses the emergent $1 \to 2$ over-correction (Round 2–4), achieving balanced diagonal improvement.

over three independent runs. Several patterns emerge from this comparison.

Rapid initial gains. CARO demonstrates substantially faster early-stage improvement, achieving an average accuracy of 0.69 after just one round (from 0.52 baseline), compared to GradeOpt's 0.58. By Round 2, all CARO runs reach or exceed 0.73 accuracy, while all GradeOpt runs remain below 0.71.

Stability vs. oscillation. CARO exhibits monotonic or near-monotonic improvement with stable convergence around 0.80 accuracy by Round 3–4. In contrast, GradeOpt displays the oscillation characteristic of aggregate feedback methods: Run 2 drops from 0.55 to 0.51 at Round 2, and Run 1 regresses from 0.63 to 0.59 at Round 4. This instability reflects mode interference, where fixing one error pattern inadvertently amplifies another.

Final performance gap. After 6 rounds, CARO achieves a final accuracy of 0.80–0.82 across all runs (avg. 0.81), while GradeOpt plateaus at 0.59–0.71 (avg. 0.64). This gap demonstrates that CARO's mode-specific updates not only converge faster but also reach superior optima by avoiding the destructive interference inherent in consolidated feedback approaches.

API Cost Analysis. Table 7 compares the computational cost of CARO and GradeOpt. Despite using the same number of optimization rounds, CARO requires 80% fewer API calls, translating to approximately 60% cost reduction. This efficiency benefits from CARO's targeted optimization strategy: rather than exhaustively sampling errors across all confusion cells to generate aggregate feedback, CARO focuses gradient generation on only the top-K error modes per round. Additionally, the mode-specific Reflector and Refiner operate on refined subsets of error examples (Eq. 3), whereas GradeOpt's aggregate approach requires broader sampling to capture diverse error patterns within a single feedback signal.

5.4 Qualitative Analysis

Case Study: Iterative Error Resolution on T5 ($\mathcal{D}_E$). We illustrate CARO's "diagnosis-and-repair" pipeline through a detailed case study on T5 ($\mathcal{D}_E$), which assesses elementary students' understanding of how a Peregrine Falcon perceives a rock pigeon through a four-step light perception process (light travels to pigeon → reflects off pigeon → enters falcon's eyes → brain processes images). The task uses a 5-point ordinal scale $\{0, 1, 2, 3, 4\}$ where Score 2 requires mentioning 1–2 steps, while Score 1 is reserved for responses containing no accurate steps. Figure 7 traces the confusion matrix evolution across optimization rounds, revealing how CARO systematically resolves cascading error patterns.

Round 0: Identifying the Dominant Mode. The initial classifier exhibits severe collapse toward Score 1, with the $2 \to 1$ cell containing 24 errors (53% of all misclassifications, highlighted in red). Analysis reveals that responses mentioning partial steps, such as *"the light source is the sun and the light travels to the eye and that's how the falcon can see the pigeon"*, were incorrectly penalized for omitting steps rather than credited for partial understanding. A secondary $3 \to 1$ pattern (8 errors) compounds this collapse, resulting

in zero correct predictions for Scores 2–4 and a near-chance $\kappa = 0.02$.

Round 2: Resolving the Primary Collapse. CARO's Reflector diagnoses the root cause: the original rubric lacks explicit guidance for recognizing partial scientific understanding. The generated rule states: *"If a response includes at least one correct element related to the four steps (e.g., light traveling, reflecting, or entering the eyes), classify as Score 2, provided it does not contain entirely irrelevant information."* As shown in the Round 2 matrix, this intervention reduces $2 \rightarrow 1$ errors from 24 to 4 (-83%, green), while correct Score-2 predictions increase from 0 to 21. Class 0 achieves perfect classification (orange). However, the fix reveals an emergent secondary pattern: $1 \rightarrow 2$ confusion (9 errors, orange) as the model now over-awards partial credit to responses lacking any valid steps.

Round 4: Addressing the Secondary Mode. CARO shifts focus to the $1 \rightarrow 2$ over-correction, introducing a discriminative rule: *"Responses must include at least one identifiable step describing light's journey; vague statements about 'seeing' without mechanistic detail remain Score 1."* The Round 4 matrix shows this refinement reduces $1 \rightarrow 2$ errors from 9 to 5 while improving Score-1 correct predictions to 40 (up from 35). Crucially, the $2 \rightarrow 1$ gains are preserved (4 errors), and Score 3 achieves its first correct prediction. The final $\kappa = 0.55$ reflects balanced diagonal improvement across all classes.

This progression exemplifies CARO's core mechanism. Instead of treating error reduction as a single, monolithic task, the framework isolates specific confusion patterns within the matrix. This decomposition turns a complex optimization challenge into a sequential and interpretable repair process. Crucially, this step-by-step approach ensures that corrections for one error type build upon previous improvements without causing regression in other classes.

6. DISCUSSION

The experimental results demonstrate that the Confusion-Aware Rubric Optimization (CARO) framework significantly advances the state of automated grading by addressing the structural inefficiencies of prior prompt optimization methods. While existing frameworks such as GradeOpt rely on aggregating heterogeneous error signals, our findings confirm that such aggregation leads to rule dilution, where the optimizer struggles to satisfy conflicting constraints simultaneously. By decomposing the optimization context into distinct error modes via the confusion matrix, CARO transforms the complex prompt refinement into a series of targeted repairs.

A primary implication of this study is the relationship between optimization specificity and efficiency. Our analysis reveals that CARO achieves superior performance with approximately 60% lower computational costs compared to baseline methods. This efficiency stems from the mode-specific gradient generation, which prevents the optimization process from cycling through redundant rule changes. Instead of requiring extensive exploration to find a marginally better prompt, the structured diagnosis allows the model to identify and resolve dominant failure patterns, such as the confusion between partial and complete understanding, with the first few iterations. This suggests that for ordinal grading tasks, structural error analysis (with confusion matrix) is a more resource-efficient method for improvement than simply increasing the volume of feedback or the size of the optimizer model.

Furthermore, the pedagogical utility of automated assessment relies heavily on interpretability and trust. This distinguishes our approach from traditional automated grading methods, such as latent semantic analysis or BERT-based fine-tuning, which process inputs through high-dimensional vectors and lack visibility into the decision-making process. Unlike these earlier supervised pipelines that function as black-box systems, CARO optimizes the natural language rubric itself. The resulting prompts contain explicit reasoning traces and priority-weighted decision rules. This transparency allows educators to validate the grading criteria before deployment, ensuring that automated system aligns with instructional goals.

Despite these strengths, there are limitations to consider. The current framework relies on the capability of the backbone LLM to self-reflect and generate diagnostic feedback. While our experiments utilized `GPT-4o-mini` effectively, the performance of CARO on smaller, open-source models remain an area for future investigation.

7. CONCLUSION

This study introduced Confusion-Aware Rubric Optimization (CARO), a novel framework designed to bridge the alignment gap between expert grading criteria and LLM logic. By identifying that grading errors are not random noise but structured clusters, we developed a method that replaces aggregate feedback with mode-specific diagnosis and repair. Through the systematic decomposition of confusion matrices, CARO isolates specific misclassification patterns and generates targeted rule patches that resolve ambiguities without degrading global performance.

Our empirical evaluation across three distinct educational datasets confirms that CARO significantly outperforms state-of-the-art prompt optimization methods in both accuracy and inter-rater reliability. Critically, the framework achieves these gains while reducing computational overhead, offering a scalable solution for high-quality formative assessment. By providing a transparent, efficient, and precise method for rubric refinement, CARO represents a substantial step reliable and scalable automated feedback in complex learning environments.

8. ACKNOWLEDGMENTS

This work was supported in part by the U.S. National Science Foundation under Grant Nos. 1813760, 2405483, 2234015, CNS-2321416, IIS-2212032, IIS-2212144, IIS-2504089, DUE-2234015, CNS-2246050, DRL-2405483, and IOS-2035472; the U.S. Department of Education under Grant No. S411B240024; the Gates Foundation; the Michigan Department of Agriculture and Rural Development; the U.S. Department of Commerce; Amazon Faculty Award; Meta; NVIDIA; Microsoft; and Snap Inc.

References

[1] Y. Attali and J. Burstein. Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, 2006.

[2] P. Black and D. Wiliam. Assessment and classroom learning. *Assessment in Education: principles, policy & practice*, 1998.

[3] I. Chamieh, T. Zesch, and K. Giebermann. Llms in short answer scoring: Limitations and promise of zero-shot and few-shot approaches. In *Proceedings of the 19th workshop on innovative use of nlp for building educational applications (bea 2024)*, pages 309–315, 2024.

[4] Z. Chen and T. Wan. Grading explanations of problem-solving process and generating feedback using large language models at human-level accuracy. *Physical Review Physics Education Research*, 21(1):010126, 2025.

[5] Y. Chu, P. He, H. Li, H. Han, K. Yang, Y. Xue, T. Li, J. Krajcik, and J. Tang. Enhancing llm-based short answer grading with retrieval-augmented generation. *arXiv preprint arXiv:2504.05276*, 2025.

[6] Y. Chu, H. Li, K. Yang, H. Shomer, Y. Copur-Gencturk, L. Kaldaras, K. Haudek, J. Krajcik, N. Shin, H. Liu, and J. Tang. A llm-powered automatic grading framework with human-level guidelines optimization. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 31–41. International Educational Data Mining Society, July 2025.

[7] J. Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[8] Y. Copur-Gencturk and T. Tolar. Mathematics teaching expertise: A study of the dimensionality of content knowledge, pedagogical content knowledge, and content-specific noticing skills. *Teaching and Teacher Education*, 114:103696, 2022.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[10] S. Dikli. An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1), 2006.

[11] P. W. Foltz, D. Laham, and T. K. Landauer. The intelligent essay assessor: Applications to educational technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 1(2):939–944, 1999.

[12] S. Haller, A. Aldea, C. Seifert, and N. Strisciuglio. Survey on automated short answer grading with deep learning: from word embeddings to transformers. *arXiv preprint arXiv:2204.03503*, 2022.

[13] J. Hattie and H. Timperley. The power of feedback. *Review of educational research*, 2007.

[14] L. Jiang and N. Bosch. Short answer scoring with gpt-4. In *Proceedings of the Eleventh ACM Conference on Learning@ Scale*, pages 438–442, 2024.

[15] L. Kaldaras, H. Akaeze, and J. Krajcik. Developing and validating next generation science standards-aligned learning progression to track three-dimensional learning of electrical interactions in high school physical science. *Journal of Research in Science Teaching*, 58(4):589–618, 2021.

[16] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, et al. Dspy: Compiling declarative language model calls into state-of-the-art pipelines. In *The Twelfth International Conference on Learning Representations*, 2024.

[17] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[18] C. Leacock and M. Chodorow. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):389–405, 2003.

[19] S. A. Livingston. Constructed-response test questions: Why we use them; how we score them. r&d connections. number 11. *Educational Testing Service*, 2009.

[20] E. Mayfield and A. W. Black. Should you fine-tune bert for automated essay scoring? In *Proceedings of the fifteenth workshop on innovative use of NLP for building educational applications*, pages 151–162, 2020.

[21] M. Mohler and R. Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575, 2009.

[22] OECD. Results from talis 2024: The state of teaching, 2025.

[23] R. Pryzant, D. Iter, J. Li, Y. T. Lee, C. Zhu, and M. Zeng. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*, 2023.

[24] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[25] V. J. Shute. Focus on formative feedback. *Review of educational research*, 2008.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[27] C. Yang, X. Wang, Y. Lu, H. Liu, Q. V. Le, D. Zhou, and X. Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*, 2023.

[28] X. Zhang, J. Cao, C. You, and D. Ding. Why prompt design matters and works: A complexity analysis of prompt search space in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32525–32555, 2025.

[29] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba. Large language models are human-level prompt engineers. In *The eleventh international conference on learning representations*, 2022.

Do interviews change participants' behaviors?: Using log data to investigate student reactivity during data-driven classroom interview

Hyeongjo Kim[1], Luc Paquette[1], Yiqiu Zhou[1], Qianhui Liu[1], Jaclyn Ocumpaugh[2], Andres Felipe Zambrano[3], Amanda Barany[3]
[1]University of Illinois at Urbana-Champaign
[2]University of Houston
[3]University of Pennsylvania
{hk61, yiqiuz3, ql29}@illinois.edu,
{luc.paquette, jlocumpaugh, afzambrano97, amanda.barany}@gmail.com

ABSTRACT

Data-Driven Classroom Interview (DDCI) uses real-time student data to identify important learning events and support the collection of rich in-situ interviews at critical moments of the learning process. Because DDCI occurs during the students' use of the learning system, they may introduce reactivity, whereby participation in the research may alter the experience of the learners. This study investigates short-term behavioral reactivity associated with DDCI and examines a potential mechanism—the attention cue effect—within What-If Hypothetical Implementations in Minecraft (WHIMC), a game-based learning environment. We used log data to compare the learning behaviors of interviewed and non-interviewed students on a set of key WHIMC behaviors, including movement, scientific observation, and measurement tool use. Results show that DDCI is associated with significant behavioral reactivity for behaviors that are more intentional and goal directed (student in-game observations and tool usage) but not for a third behavior (movement), which may function as less direct, peripheral byproduct of task engagement. Moreover, reactivity was not explained by explicit discussions of these behaviors during DDCI, which may indicate that reactivity is arising through other mechanisms. By empirically documenting short-term behavioral reactivity in DDCI and systematically examining its potential mechanisms, this study contributes to the methodological refinement of DDCI and highlights important considerations for the design and interpretation of data-driven research in open-ended learning environments.

Keywords

Data-driven classroom interview (DDCI), In-situ interviewing, Reactivity, Behavior log data analysis, Game-based learning

1. INTRODUCTION

Data-Driven Classroom Interview (DDCI) is a research method designed to deepen our understanding of learning phenomena by complementing traditional in-situ interviews using detectors based on learner modeling methods [1, 20]. While in-situ interviews have been used to capture learners' experiences and meanings within authentic learning contexts, they face two persistent methodological challenges [1]. First, researchers must determine how to identify rare yet pedagogically important events that are difficult to observe directly, often described as a "needle-in-a-haystack" " problem [1]. Second, in-situ interviews raise a sampling issue [22, 25], namely, deciding whom to interview and when, without relying solely on convenience or researcher intuition. By operationalizing theoretically and empirically meaningful learning behaviors through log data to inform interview opportunities, DDCI enables researchers to capture critical yet otherwise elusive learning events and to interview participants based on principled criteria [20].

DDCI belongs to a range of in situ methods--including more traditional interviewing techniques, think-aloud protocols [8], and ecological momentary assessment (EMA; [19])--that can improve validity by allowing the researcher access to the students thoughts about the learning experience as it unfolds. However, just like other in-situ research methods, DDCI alters the learning environment for at least the duration of that research activity, and possibly beyond. These measurement effects have a range of labels in the literature including the observer effect [30] or reactivity [4, 34]. Such effects may stem from increased awareness of being observed (the Hawthorne effect; [18]), motivational responses to perceived evaluation or comparison (the John Henry effect; [27]), or attempts to align with perceived researcher expectations (the observer-expectancy effect; [26]). Whereas earlier methodological traditions often sought to eliminate or control reactivity, recent scholarship increasingly emphasizes explicitly investigating and articulating reactivity as part of the research process [17, 34].

Several considerations are particularly important for investigating reactivity in the context of DDCI. First, reactivity research should examine behaviors that are directly logged by the system and that simultaneously relate to learning processes. Reactivity may manifest across multiple domains, including emotion, behavior, and metacognition. Prior work in educational contexts has explored some of these dimensions. For example, Liu et al. [16] found that observation influenced students' display of emotional states by increasing engaged concentration while reducing frustration and off-task behavior. Similarly, Bosch et al.'s [3] analysis of metacognitive content in DDCI found that certain forms of verbalized metacognition were associated with learning outcomes, particularly for students who engaged in metacognitive verbalization during

Hyeongjo Kim, Luc Paquette, Yiqiu Zhou, Qianhui Liu, Jaclyn Ocumpaugh, Andres Felipe Zambrano, and Amanda Barany. Do interviews change participants' behaviors?: Using log data to investigate student reactivity during data-driven classroom interview. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 79–89. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039911

interviews. However, these studies did not directly examine behavioral changes in the log data that serve as the triggering events for DDCI and may also represent important learning behaviors. This omission is consequential, as changes in logged behavior may alter detector activation, subsequent interview selection, and, ultimately, interpretations of learning.

In addition, systematic approaches to studying reactivity should be considered. Reactivity can be more rigorously examined by (1) comparing participants' behaviors before and after the measurement of interest, thereby leveraging temporal changes associated with repeated measurement [10], and (2) employing between-participant comparisons between students who are measured and those who are not, in order to distinguish reactivity from other confounding factors. In the DDCI context, prior logged behaviors may determine which students are interviewed, and reactivity may be intertwined with both interview-related influences and observer effects. Therefore, research on DDCI reactivity should consider both students' behaviors before interviews and the behaviors of non-interviewed peers to control the observer effects.

Finally, understanding the mechanisms underlying reactivity is crucial for explaining why reactivity occurs and how it can be mitigated in DDCI settings. Several mechanisms may plausibly explain DDCI-related reactivity, including observer effects, verbalization effects, and metacognitive activation. The presence of researchers in the classroom may elicit observer effects, as evidenced by prior findings [16, 30]. Additionally, DDCI interviews require students to engage in high-level verbalization about their thoughts and actions, which is likely to induce stronger reactivity than lower-level verbalization methods such as think-aloud protocols [6, 8]. Such verbalization may function as an attentional cue [4] that stimulates cognitive and metacognitive processes, which can vary depending on both individual characteristics (e.g., interest) and task features (e.g., writing versus reading). Given these multiple plausible mechanisms, researchers should design studies and analyses that aim to identify which mechanisms most plausibly account for observed reactivity effects.

The present study examines whether and how students' behaviors are influenced by DDCI in WHIMC (What-if Hypothetical Implementations in Mine-craft), a Minecraft-based learning environment [9, 14]. Specifically, we investigate reactivity in behaviors that are directly logged by the system by comparing the behaviors of interviewed students to the behavior of peers who were not interviewed during the same activity. When interview-related reactivity is identified, we further examine whether such effects are associated with direct attentional cues, through which explicit references to specific behaviors (e.g., "observation") during the interview may promote subsequent enactment of those behaviors. Based on the observed patterns of reactivity, we discuss potential distortions in DDCI-based analyses. By clarifying the nature and implications of behavioral reactivity in DDCI, this study advances the methodological development and supports the careful implementation of data-driven interviews.

2. RELATED WORK

2.1 Data-Driven Classroom Interview

Data-Driven Classroom Interview (DDCI; [1, 20]) is an emerging research method designed to deepen understanding of learning phenomena by augmenting traditional in-situ interviews with detector technologies. By operationalizing focal learning behaviors through log data, detectors can automatically identify moments when targeted behaviors occur and trigger interviews accordingly. This approach allows researchers to capture critical but otherwise elusive learning events and to interview participants based on theoretically and empirically meaningful criteria. The feasibility of such detectors has been substantially enhanced by advances in learning analytics (LA) and educational data mining (EDM), which have demonstrated that a wide range of learning phenomena can be defined, modeled, and monitored using rich streams of behavioral data, such as interaction logs [29].

The DDCI process typically proceeds as follows [20]. Based on the research questions, researchers first develop behavioral models that define interview triggers—specific patterns of activity that research hopes to investigate. Researchers can assign priorities to different triggers to detect rare or theoretically important behaviors while also distributing interview opportunities across participants. When the DDCI system detects a predefined behavior, it alerts researchers, who can then decide whether and when to conduct an interview. These interviews are typically brief (often under five minutes) to minimize disruption to learners' ongoing activity while capturing highly contextualized and ecologically valid accounts of in-situ experiences. In summary, the core stages of DDCI include trigger development, priority setting, and interviewer engagement following trigger detection.

Because DDCI relies on data-driven technologies, their systematic implementation requires technical infrastructure [20]; triggers must be developed and integrated with the learning environment's logging system to facilitate real-time detection and notification as well as the recording and storage of interview data. To date, this has been conducted with Quick Red Fox (QRF), an open-source server–client Android application that streamlines in-situ data collection by directing interviewers to learners immediately after the occurrence of research-relevant behaviors defined a priori by the research team [20].

By integrating data-driven detection with in-situ interviewing, DDCI offers methodological benefits for both qualitative and quantitative research. While in-situ interviews have been used to capture learners' experiences and meanings within authentic learning contexts [13], they face two persistent methodological challenges [1]. First, researchers must identify rare yet pedagogically significant events that are difficult to observe directly—a challenge often characterized as a "needle-in-a-haystack" problem. Second, in-situ interviews raise a sampling issue [22, 25]: deciding whom to interview and when, without relying primarily on convenience or researcher intuition. DDCI addresses these challenges by leveraging detectors derived from log data.

For quantitative research, behavioral data alone provide limited access to the contextual meanings and latent variables underlying observable actions, which raises concerns related to construct validity and inferential plausibility [32]. Qualitative data such as DDCI can complement purely quantitative approaches by addressing challenges in interpreting behavioral data, particularly when inferring learners' intentions or reasoning from logs alone [1, 11]. DDCI provides learners with opportunities to reflect on and explain their behaviors, generating insights that can inform feature interpretation, model refinement, and theory development.

However, the advantages of in-situ interviews are accompanied by an important methodological concern. Because interviews occur during ongoing learning activities in DDCI, participants may unintentionally alter their behavior in response to being interviewed or to their awareness of research participation. Understanding and accounting for such changes is therefore essential when interpreting data collected through DDCI.

2.2 Reactivity

Reactivity refers to any changes that arise from being measured or participating in research itself, with potential effects on participants' motivation, expectations, affect, and goals [4]. Empirical work has documented reactivity across a range of research methods, including ecological momentary assessment (EMA; [19]), think-aloud protocols [6, 8, 28], and interview-based approaches [3, 23, 24]. In a broad sense, all forms of research participation entail some degree of influence on participants. In any interview-based method, such as DDCI, allocating time for an interview and responding to researchers' questions are intended and necessary influence as a part of data collection process. Therefore, it is important to delimit the scope of reactivity under consideration. Accordingly, the form of reactivity most relevant to the present study is what Zahle [34] terms 'regularly unintended reactivity'—that is, routinely occurring effects of research participation that extend beyond the method's intended influence.

Traditional methodological perspectives, particularly within experimental psychology, have emphasized minimizing or eliminating reactivity [15]. From this standpoint, reactivity is treated primarily as a threat to validity. For instance, introspection-based methods, which require participants to report on their internal cognitive processes through surveys or interviews, have been especially criticized on these grounds. In response, proponents of think-aloud protocols developed theoretical frameworks distinguishing levels of verbalization, arguing that low-level verbalizations—such as articulating information already in working memory—introduce minimal reactivity and can therefore serve as valid data sources [6].

In contrast, other research traditions—particularly qualitative inquiry—have taken a more generative view of participant influence. Rather than framing interview effects as reactivity to be controlled, qualitative researchers often conceptualize them as reflexivity [23]. From this perspective, interviews are understood as opportunities for participants to reflect on, reinterpret, and reorganize their experiences in interaction with the researcher. Prior work has examined both during-interview and post-interview reflexivity, highlighting how participation in research can shape participants' self-understanding over time [23]. This orientation reflects an epistemological stance in which interviews are not treated as neutral pipelines for extracting pre-existing meanings, but as sites of meaning and knowledge construction through interviewer–interviewee interaction [13]. Within this framework, participant influence is not a methodological flaw to be eliminated but a natural and often productive feature of the research process.

More recent methodological work seeks to bridge these perspectives by advocating for a reactivity-transparent approach, in which reactivity is explicitly examined, documented, and analytically incorporated rather than treated solely as bias [17, 34]. Zahle [34] argues that researchers should supplement data with explicit assumptions about reactivity, enabling principled reasoning about how collected data may have been shaped by the research process. Depending on the assumptions adopted, researchers may (a) claim reactivity has been adequately controlled, (b) acknowledge partial reactivity effects, or (c) actively investigate the sources and forms of reactivity to infer how participants might have behaved in the absence of measurement.

Strategies have been proposed for studying reactivity itself rather than treating it exclusively as a source of contamination. Gieles et al. [10], for example, distinguish between direct and indirect approaches to assessing measurement reactivity. Direct approaches examine reactivity by explicitly asking participants about perceived changes resulting from research participation, typically through interviews or self-report questionnaires. These methods provide insight into participants' own interpretations of how measurement influenced their thinking, awareness, or behavior, but they are limited by retrospection, social desirability, and participants' ability to articulate subtle or implicit changes.

Indirect approaches [10], by contrast, examine reactivity by detecting systematic changes in responses over repeated measurement, rather than relying on participants' self-reports about influence. Such approaches are especially relevant in EMA contexts, where repeated self-monitoring may promote reflection or learning. For instance, researchers can compare whether survey-based self-perceptions align more strongly with EMA-recorded experiences after, rather than before, an EMA period to assess how EMA may enhance self-understanding (i.e., reactivity in this context). Patterns of increasing alignment or shifts across measurement phases may indicate that participation in measurement itself altered participants' awareness or behavior. These frameworks provide a foundation for examining reactivity in DDCI, where repeated interviews are conducted in real time.

2.3 Reactivity of Data-Driven Classroom Interview

Because DDCI, like other in-situ methodologies, may introduce multiple sources of reactivity, examining its reactivity is essential to understand and articulate how it affects students' learning experiences. First, DDCI requires researchers to be physically present in the classroom, which may produce observer effects [16, 30]. The visible presence of researchers can heighten students' awareness of being observed and lead them to modify their behavior relative to their typical classroom practices. For example, Liu et al. [16] compared students' emotional states in classroom settings with and without observers and found that researcher observation alone could differentially affect detected emotions—specifically increasing engaged concentration while reducing frustration or off-task behaviors. Such effects may be partly attributable to implicit power asymmetries between interviewers and students. In classroom contexts, interviewers are often perceived as authority figures or evaluators, which may inhibit students from sharing authentic accounts of their experiences [1, 31]. To alleviate the issue, DDCI prioritizes eliciting learners' accounts of their ongoing activities and experiences from their own perspectives [1, 20].

Reactivity in DDCI may also vary depending on the level of verbalization elicited during interviews and the interview strategies used to prompt it. Ericsson and Simon [5, 7] distinguished three levels of verbalization. Level 1 verbalization involves the direct vocalization of inner speech as it occurs, whereas Level 2 verbalization requires verbal encoding and organization of thoughts already present in working memory. Although both levels may slow or extend cognitive processing due to overt verbalization, they are generally assumed not to alter the underlying sequence of thought [6, 8]. In contrast, Level 3 verbalization—often referred to as introspection—requires participants to explain or justify their thinking (e.g., "Why are you solving the problem this way?"). This level introduces additional cognitive demands that can interrupt ongoing cognitive processes and resume them in a qualitatively different state. From this perspective, reactivity in DDCI is not uniform but might depend critically on interview design and the extent to which interview prompts elicit higher levels of reflective or explanatory verbalization.

Empirical evidence supports this distinction. In DDCI contexts where interviewers frequently prompted students to articulate their

problem-solving strategies, Bosch et al. [3] found that interviews were associated with increased learning outcomes, particularly for students who engaged in metacognitive verbalization. These findings suggest that reactivity in DDCI may arise not only from the act of interviewing itself but also from the specific content and cognitive demands of interview or its prompts. Consequently, both interview presence and interview design must be considered when examining DDCI reactivity.

The impact of DDCI can be further understood through metacognitive mechanisms activated by interview content. Double and Birney [4] proposed a theoretical framework that conceptualizes reactivity as arising from changes in the cognitive–metacognitive system induced by measurement. Drawing primarily on research on metacognitive measures such as think-aloud protocols, confidence judgments, and judgments of learning, their framework suggests that measurement directs learners' attention toward particular diagnostic cues—either about their ongoing experience (e.g., perceived difficulty, uncertainty, effort) or about their self-beliefs (e.g., perceived competence or memory ability). Attending these cues influences metacognitive monitoring, which in turn shapes metacognitive control decisions and ultimately affects cognitive performance [4, 12]. These cues function as inputs for diagnosing one's performance or motivation relative to task goals, thereby influencing decisions about strategy selection, effort allocation, and persistence.

Importantly, as Double & Birney [4] showed, the strength and direction of such reactivity depend on multiple interacting factors, including task characteristics (e.g., complexity, novelty, openness) and person characteristics (e.g., prior knowledge, metacognitive skill, motivation). These factors shape how measurement-induced cues are interpreted and acted upon. Accordingly, within this metacognitive framework, reactivity is not viewed as random noise, but as a systematic and theoretically explainable consequence of how measurement interacts with learners' monitoring and control processes. This perspective provides a principled basis for examining reactivity in DDCI and for interpreting its effects on learners' behavior and learning trajectories.

3. METHODS

3.1 WHIMC

WHIMC (What-If Hypothetical Implementation in Minecraft) is a suite of Minecraft-based simulations designed to support astronomy-focused inquiry in order to foster student interest [9, 14]. The WHIMC environment includes thirteen space simulation worlds: two orientation worlds (Rocket Launch and Space Station Hub), six "what-if" Earth scenarios (No Moon, Colder Sun, Tilted Earth, Two Moons, and Earth as Moon), five exoplanet worlds (Trappist, Kepler, Gliese, Cancri, and Brown Dwarf), and a Mars world centered on habitat construction.

Within WHIMC, students explore hypothetical questions by comparing normal Earth to simulated alternative conditions, such as an Earth without the Moon or with a colder Sun. As they navigate these worlds, learners freely explore the environment, identify distinctive features at points of interest (e.g., shorter vegetation or the presence of wind turbines), and use embedded scientific tools (e.g., wind-speed sensors) to make observations and collect data. These interactions encourage students to pose questions or generate hypotheses about observed phenomena (e.g., why plants are shorter in the absence of the Moon or whether stronger winds might be responsible). Figure 1 presents example screen captures of gameplay in WHIMC.



Figure 1. Example of gameplay in WHIMC. Students examine a wind turbine in the Earth without Moon scenario and learn that it serves as a primary source of power generation. They can then measure wind speed and record an observation by asking questions or making inferences (e.g., "Why is the wind so fast?").

To investigate their questions, students seek evidence through multiple forms of in-game scaffolding, including examining environmental features at points of interest, using scientific tools to gather measurements, and interacting with non-playable characters (NPCs) whose quests and floating hints guide inquiry. As students connect observations to data and explanations, their inquiry can progress toward deeper causal understanding of the underlying "what-if" scenario—for example, linking increased wind speeds to faster Earth rotation and ultimately to the Moon's gravitational influence. At the same time, the open-ended design of WHIMC means that inquiry is not strictly linear: students retain substantial autonomy and may pursue multiple exploratory paths as they navigate the simulated worlds.

3.2 Data-Driven Classroom Interviewing

3.2.1 Participants

This study examined data collected from five-day science camps for middle school students that employed WHIMC. On the first day of the camp, participants completed surveys assessing individual interest, prior knowledge, and prior game experience. During the first three days, students explored a series of "what-if" simulation worlds and investigated the distinctive characteristics of each environment. At the conclusion of Day 3, post-surveys measuring individual interest, situational interest, and knowledge were administered. During the final two days of the camp, students engaged in a Mars-based habitat construction activity. Throughout the whole gameplay, Quick Red Fox (QRF), an Android application for conducting DDCI [20], was used to detect predefined behavioral patterns in students' in-game actions (e.g., repeated use of the same scientific tool) and to identify candidates for in-situ interviews.

The dataset analyzed in this study combined data from four camps conducted in different regions of the United States, including the Midwest and Mountain West, during 2024 and 2025. The sample consisted of 98 participants, of whom 44 identified as male, 44 as female, 2 as non-binary, and 8 did not disclose their gender. Regarding racial and ethnic identity, participants included 1 Native American, 45 Black, 31 White, 11 individuals identifying with other ethnicities, and 10 participants who chose not to report their ethnicity. Research and data collection protocol was conducted under approval from the Institutional Review Board of University of

Pennsylvania and secondary analysis of the anonymized data was approved by the Institutional Review Board of University of Illinois at Urbana-Champaign. Participation was voluntary, and all students provided written assent along with parental consent.

3.2.2 Log data from WHIMC

This study used both log data and interview data to investigate interview reactivity. Log data were recorded from the Minecraft server and included students' positional information sampled every two to three seconds (x, y, and z coordinates), records of tool use, and records of observational actions, each with corresponding timestamps.

3.2.3 DDCI from Quick Red Fox (QRF)

DDCI was conducted using Quick Red Fox (QRF), an open-source Android server–client application designed to notify interviewers when participants engage in behaviors of interest specified in advance by the research team [20]. QRF integrates modeling technologies and processes real-time data to identify key moments in learners' experiences, referred to as interview triggers. Researchers can assign priorities to different triggers to detect rare or theoretically important behaviors while also distributing interview opportunities across students. When QRF detects a predefined behavioral pattern, the application notifies researchers, enabling them to decide whether and when to conduct an interview.

Trigger events and interview records are transmitted to a centralized dashboard, collecting logs of interview activity and allowing the research team to monitor interview activity in real time. The dashboard provides information such as trigger identifiers, timestamps, student and interviewer IDs, interviewer decisions to conduct interviews, and whether a second student contributed to an interview (e.g., when a neighboring student chimed in during an interview). Interviews are typically brief (generally under five minutes) to minimize disruption to students' ongoing activities while capturing highly contextualized and ecologically valid accounts of their in-situ experiences. A total of four interviewers participate across the four camps (at least two per camp).

All interviews followed established DDCI interview principles and strategies [20], which emphasize maintaining respectful, non-evaluative relationships with students and eliciting the learners' perspectives on their ongoing activities and experiences. Ocumpaugh et al. [20] describe this as the Big Sister Approach (BSA)--which they explicitly contrast with the Orwellian concept of Big Brother. Interviewers make it clear to the students before DDCI begins that interviewers are not there to enforce student behavior, and their interview approach is designed to reflect that as well. To that end, the BSA permits encouragement, but does not permit instruction or policing of off-task behaviors. An interviewer might offer a worried student gentle confirmation about whether the strategy the student wants to test might work, but students who need instruction should be referred to a teacher or peer during a DDCI.

For several reasons, DDCI blends a mix of questions about the learner and about the learning context. In part, questions about the learner allow the interviewer to build trust, however they are also part of an asset-based approach [21] where the interviewer is searching for knowledge that could be relevant to the learning context, even if the student has not yet made those connections. Although interviewers have access to the DDCI trigger information, they do not immediately ask the student about those constructs. The reason for this is two fold, since (1) it could make students feel surveilled, which could heighten self-presentation effects and (2) it limits the degree to which an interview can determine whether that construct and its associated experiences were salient to the students' game-play experience.

DDCI tends to start with questions like "how are you doing?" and "what strategies are you trying?" as they are open-ended enough to allow students to steer the topics in the initial part of each interview. As the interview progresses, questions may get more probing, and if the constructs associated with the trigger do not emerge, the interviewer may try to introduce them (e.g., asking if the student has deliberately visited any locations when determining how they came across a point of interest in the game). Throughout this process, however, the overarching goal is to encourage students to discuss their current states of metacognition, motivation, and engagement. If the constructs associated with the trigger (e.g., a particular part of the WHIMC game) are clearly not part of that metacognitive process, that is typically useful data for a DDCI study like this one, where the underlying goal is to understand student interest development. Therefore, in this study, interviewers did not always actively try to re-engage on topics directly associated with the trigger.

3.2.4 Pre-Interest Survey

The goal of QRF interviews in this research is to understand students' interest development through WHIMC. To account for individual differences, initial levels of individual interest were included in the analysis. Individual interest was measured using a survey adapted from Boeder et al. [2], which assesses context-independent interest in science across five subscales: Information Seeking, Motivation to Reengage, Persistence, Self-Regulation, and Value. Example items include "I want to know more about science" and "I seek out opportunities to engage in science."

Each subscale comprised four to five items rated on a 7-point Likert scale, ranging from Do Not Agree At All to Very Strongly Agree. The survey was administered on the first day of the camp, prior to students' engagement with the WHIMC activities.

3.3 Analysis

Among the log and interview data, only data from five What-If worlds (Lunar Crater, No Moon, Colder Sun, Tilted Earth, Two Moons) were included. Other activities (e.g., the Mars habitat activity and exoplanet exploration) were excluded due to differences in task designs.

The primary goal of the analysis was to examine whether interviews were associated with changes in students' behaviors during and after the interview. To do so, we compared interviewed and non-interviewed students while controlling for pre-interview behavior. This approach helps disentangle interview effects from other influences such as observer effects or general time-related trends (e.g., declining activity as gameplay progresses) by considering both interview and non-interview cases. To enable direct comparison within a common temporal framework, we constructed a dummy interview time frame which aligned behavioral phases across interviewed and non-interviewed students. For each non-interviewed student in a given world, we determined their dummy interview start and end time by randomly selecting one of the interviewed students in the same world and camp and assigned that student's interview start and end times to the non-interviewed student. In brief, we have a time framework composed of two parts : pre-interview (from world play start to interview start) and during/post-interview (from interview start to world play end).

This analysis focused on three core behaviors in WHIMC: tool use, observation, and movement. These behaviors represent central

learning processes in the environment. Students move through the virtual world to locate relevant information, use scientific tools to explore phenomena, and make observations to document and reflect on their findings.

The number of tool use and observation behaviors were extracted directly from server logs without additional feature transformation. In addition to the count of observation and tool uses, we computed binary variables indicating whether the behavior occurred at least once within a given time window. This decision was motivated by the highly right-skewed count distributions for these behaviors, in which most observations were zero or near zero and a small number exhibited very high counts. Under such conditions, modeling only raw counts can be unstable and disproportionately influenced by rare extreme values. Accordingly, the analysis focused on not just frequency but also behavioral occurrence.

Movement-related features were engineered from positional log data from both phase (pre-interview and during/post-interview) and included mean distance traveled, mean speed, speed standard deviation, mean absolute acceleration, acceleration standard deviation, mean direction-change speed, direction-change speed standard deviation, and the mean of direction changes within three angular ranges (45–90°, 90–150°, and 150–180°).

Player position logs were first grouped per each user, camp, and world. All movement features were derived from two-dimensional spatial coordinates (x, z). To focus spatial navigation, only x (east-west) and z (north-south) coordinate was used while y axis (vertical) was excluded [33]. For each consecutive pair of position samples, stepwise displacement was computed as the Euclidean distance:

$\text{distance}_t = \sqrt{(\Delta x_t)^2 + (\Delta z_t)^2}$ where $\Delta x_t = x_t - x_{t-1}$ and $\Delta z_t = z_t - z_{t-1}$.

Instantaneous speed was then calculated as:

$$\text{speed}_t = \frac{\text{distance}_t}{\Delta t_t}$$

Acceleration was derived as the temporal derivative of speed:

$$\text{accel}_t = \frac{\Delta \text{speed}_t}{\Delta t}$$

Movement direction was computed at each step using the arctangent of the displacement vector:

$\text{heading}_t = atan2(\Delta z_t, \Delta x_t)$, yielding angles in radians within $(-\pi, \pi]$.

Changes in heading between consecutive steps were calculated and wrapped to the principal angle range to account for circularity. The absolute turning angle (in degrees) was then defined as:

$$\text{turn_deg}_t = |\text{heading}_t - \text{heading}_{t-1}| \times \frac{180}{\pi}$$

Angular turn velocity was computed as the rate of heading change per unit time:

$$\text{ang_vel}_t = \frac{\Delta \text{heading}_t}{\Delta t_t}$$

To characterize directional changes, turning events were categorized into three angular bins based on absolute turning angle: 45°–90°, 90°–150°, and 150°–180°

The data had a nested and longitudinal structure: each student participated in one of four camps, explored multiple worlds within the camp, and could be interviewed multiple times. To account for this dependency structure, we used generalized linear mixed-effects models (GLMMs) rather than standard regression. Models were fitted in R (version 4.5.2) using the glmmTMB package (version 1.1.13).

Depending on the distributional characteristics of features, we used different GLMM families. First, time-dependent features were fit with negative binomial GLMMs using during-post interview time as offset: the number of observations, and the number of tool use. Second, binary features are fit with binomial GLMMs: binary observation and binary tool use. Lastly, continuous features, which already accounted for time as part of the feature calculations, are modeled with normal distribution: mean distance, all speed features (mean speed and speed standard deviation), all acceleration features (mean absolute acceleration and acceleration standard deviation), and all angular features (mean absolute angular velocity, standard deviation of angular velocity, and mean of direction changes within three angular ranges (45–90°, 90–150°, and 150–180°)). After fitting the models, statistical significance was determined using the Benjamini–Hochberg false discovery rate correction to control for multiple comparisons and reduce Type I error.

A total of three analyses were conducted. First, because interviews were triggered by data-based behavioral criteria, we examined whether students who were interviewed differed significantly from those who were not in their behaviors prior to the interview. Also, as the goal of the interviews were to gain insights into the students' STEM interest, their pre-interest might influence whether they are interviewed. We tested whether interview status significantly predicted students' pre-interview behaviors and pre-interview interest. In these models, camp and world were included as random effects.

Second, we compared the during-and-post features of interviewed and non-interviewed students while controlling for pre-behavior, interview, their interaction, and pre interest. Because we hypothesized that pre-interview behavior might moderate the effect of the interview, all models were initially fitted with an interaction term between pre-interview behavior and interview status. When neither the interaction nor the main effect of interview status was significant, an additional reduced model excluding the interaction term was fitted to examine potential main effects. Across all models, camp and world were included as random effects. When model convergence issues occurred, the random effect of camp was removed and the model was refitted.

Table 1. Extracted Interview Keywords.

Behaviors	Keywords
Observation	observation
Tool use	tool, measure, airflow, radiation, altitude, atmosphere, radius, tectonic, temperature, gravity, tide, humidity, tilt, magnetic, rotation, oxygen, year, pressure, composition, scale

Lastly, when interview effects were detected, we further tested whether these effects were driven by an attention-cue mechanism. Specifically, we compared students whose interviews explicitly mentioned a target behavior (e.g., observation) with those whose interviews did not mention that behavior. We determined whether an interview contained an attention cue by matching keywords associated to the behavior to the interview transcripts (Table 1). We used only 'observation' for observation as other possible keywords (e.g. observe) can extract many unrelated interviews while, for tool

Table 2. Mean and Standard Deviation of Features Before Interview for both Interviewed and Non-interviewed Students.

Feature	Interviewed			Not interviewed		
	N	Pre	During/Post	N	Pre	During/post
Absolute acceleration (mean)	79	1.31 (2.38)	1.17 (2.86)	194	1.04 (1.15)	1.19 (2.20)
Acceleration (std)	77	8.38 (24.57)	7.83 (26.15)	192	5.26 (12.36)	6.51 (19.02)
Angular absolute velocity (mean)	82	0.54 (0.33)	0.45 (0.40)	203	0.64 (0.41)	0.61 (0.43)
Angular velocity (std)	79	0.80 (0.44)	0.69 (0.51)	198	0.94 (0.51)	0.91 (0.54)
Distance (mean)	87	3.16 (6.35)	4.93 (12.13)	211	2.37 (6.61)	1.88 (5.43)
Speed (mean)	82	2.88 (2.08)	2.43 (2.16)	204	2.28 (1.61)	2.19 (1.89)
Observation (sum)	87	1.48 (2.15)	1.25 (2.09)	212	1.38 (4.10)	0.54 (1.44)
Observation (occurrence)	87	0.52 (0.50)	0.46 (0.50)	212	0.39 (0.49)	0.22 (0.41)
Speed (std)	79	9.67 (18.56)	8.86 (19.09)	198	6.19 (10.94)	6.55 (14.29)
Tool (sum)	87	1.75 (2.67)	0.77 (1.43)	212	1.58 (3.31)	0.58 (1.45)
Tool (occurrence)	87	0.59 (0.50)	0.32 (0.47)	212	0.37 (0.48)	0.23 (0.42)
Turn 150 to 180 (mean)	83	6.74 (5.37)	6.67 (8.63)	205	7.41 (6.03)	8.00 (6.65)
Turn 45 to 90 (mean)	83	9.45 (10.05)	7.09 (6.42)	205	11.42 (11.83)	9.57 (7.93)
Turn 90 to 150 (mean)	83	15.18 (12.03)	13.54 (13.00)	205	20.01 (15.29)	19.27 (15.44)

Table 3. General Linear Mixed Methods Results (Coefficients, Standard Errors, and P-values) for Models Predicting Each of the 14 Features.

Behavior	N	Pre-behavior	Interview	Pre-behavior * Interview	Pre-interest
Observation (sum)	276	β = 0.07 (0.05) p = 0.154	β = 0.53 (0.33) p = 0.108	β = -0.03 (0.12) p = 0.836	β = 0.34 (0.09) p = <.001
Tool (sum)	276	β = 0.12 (0.05) p = 0.022	β = 0.32 (0.38) p = 0.403	β = -0.12 (0.14) p = 0.380	β = 0.38 (0.11) p = <.001
Observation (occurrence)	276	β = 1.88 (0.40) p = <.001	β = 1.30 (0.49) p = 0.009	β = -0.51 (0.64) p = 0.427	β = 0.16 (0.11) p = 0.152
Tool (occurrence)	276	β = 1.09 (0.39) p = 0.005	β = 1.41 (0.48) p = 0.003	β = -1.65 (0.64) p = 0.010	β = 0.33 (0.12) p = 0.004
Absolute acceleration (mean)	231	β = 0.11 (0.19) p = 0.550	β = -0.91 (0.43) p = 0.035	β = 0.95 (0.28) p = <.001	β = -0.10 (0.10) p = 0.343
Acceleration (std)	229	β = -0.13 (0.19) p = 0.515	β = -3.98 (3.05) p = 0.191	β = 1.34 (0.29) p = <.001	β = -0.05 (0.89) p = 0.955
Angular absolute velocity (mean)	236	β = 0.61 (0.09) p = <.001	β = -0.12 (0.09) p = 0.172	β = 0.16 (0.13) p = 0.224	β = -0.01 (0.01) p = 0.500
Angular velocity (std)	231	β = 0.68 (0.11) p = <.001	β = -0.04 (0.11) p = 0.686	β = 0.02 (0.11) p = 0.868	β = -0.01 (0.02) p = 0.559
Distance (mean)	241	β = 0.30 (0.06) p = <.001	β = -0.63 (0.69) p = 0.361	β = 0.60 (0.09) p = <.001	β = -0.02 (0.20) p = 0.911
Speed (mean)	236	β = 0.54 (0.11) p = <.001	β = 0.03 (0.45) p = 0.946	β = -0.01 (0.14) p = 0.935	β = 0.01 (0.08) p = 0.921
Speed (std)	231	β = -0.00 (0.13) p = 0.997	β = -1.81 (2.41) p = 0.452	β = 0.43 (0.18) p = 0.018	β = -0.10 (0.66) p = 0.879
Turn 150 to 180 (frequency mean)	234	β = 0.39 (0.09) p = <.001	β = -2.61 (1.45) p = 0.072	β = 0.34 (0.15) p = 0.026	β = -0.26 (0.29) p = 0.359
Turn 45 to 90 (frequency mean)	234	β = 0.10 (0.05) p = 0.038	β = -1.06 (1.33) p = 0.425	β = 0.02 (0.09) p = 0.853	β = -0.09 (0.31) p = 0.760
Turn 90 to 150 (frequency mean)	234	β = 0.43 (0.08) p = <.001	β = -1.49 (2.84) p = 0.599	β = 0.02 (0.14) p = 0.866	β = -0.37 (0.54) p = 0.490

Note. β denotes fixed-effect estimates and values in parentheses are standard errors.

use, we included 'tool', 'measure', and other the name of available measurement tools. Because movement patterns cannot be reliably identified in interview transcripts, this analysis was limited to tool use and observation behaviors. After investigating direct attention cue effect, we refit the model in the second analysis (interview reactivity) while excluding interviews containing an attention cue for the target behaviors. This was done to see whether the interview effect is replicated with the condition. By reproducing interview reactivity, we can see whether direct attention cue effect can be divided with other potential mechanisms explaining the interview reactivity. Across all models, camp and world were included as random effects. When model convergence issues occurred, the random effect of camp was removed and the model was refitted.

4. RESULTS

Across the four camps, we collected 120 interviews from 60 students (mean: 2, std: 1.22) from the five target worlds. A total of 33 interviews were excluded because interview time logs (4 cases) or world playtime logs (9 cases) were unavailable; the interviews contained no meaningful content (7 cases), such as when students were chosen but they refused to participate or interviewers terminated the session within 50 seconds; or a student completed multiple interviews within the same world (20 cases), as repeated interviews within a world could distort the defined time frame. After screening the interviews, 87 interviews from 55 students remained (mean: 1.58, std: 0.85). Each camp had an average of 21.75 interviews (std: 8.06). The five worlds included in the analysis had an average of 17.4 interviews each (std: 2.30). The average interview time was 264.16 seconds (std : 117.90) while the average play time before an interview was 621.37 seconds (std : 372.32) and the average play time after an interview start was 653.14 (std : 530.16).

4.1 Behaviors Differences Before Interview

The first analysis examined whether students who were interviewed differed from those who were not interviewed in terms of their behavior or interest at a given time point. Table 2 provides descriptive statistics comparing behavior features for interviewed and non-interviewed students. GLMM models were used to test whether interview status significantly predicted students' pre-interview behaviors and pre-interview interest. In these models, camp and world were included as random effects. The results indicated that interviewed students exhibited significantly higher occurrence of tool use (binary), whereas no other significant differences were observed between the interviewed and non-interviewed groups in either behavioral or interest measure.

4.2 Interview Reactivity

Second, we examined whether interview participation significantly predicted students' behaviors after the interview. GLMM models were fit to compare during/post features while including pre-behavior, whether the student was interviewed or not, their interaction, and pre interest as fixed effect (Table 3).

Significant positive main effects of interviews were observed for several features. Specifically, interview participation significantly predicted the occurrence of observation, and occurrence of tool use. In these models, pre-interview behavior and pre-interview interest were also significant predictors, while occurrence of tool use had significant negative interaction effects. In case of observation occurrence, there is significant interview main effect without interaction effect ($\beta = 1.30$, SE = 0.49, $p = 0.009$). On the other hand, although interview was associated with an increased likelihood of tool use ($\beta = 1.41$, SE = 0.48, $p = .003$), this effect was significantly moderated by pre-interview tool use, as indicated by a negative Interview * Pre tool use interaction ($\beta = -1.65$, SE = 0.64, $p = .010$). This pattern suggests that interview reactivity may be most evident among students who had not used tools prior to the interview. As the main and interaction effects are pointed in different directions in tool use, interpreting the interview effect based solely on the main effect coefficient can be misleading. Therefore, we probed the interaction by examining the conditional pre–post behavioral association among interviewed students (i.e., the combined coefficient $\beta_int + \beta_interaction$). Simple effects analyses showed that interview participation did not significantly affect tool use among students who had already engaged in tool use before the interview ($\beta = 0.24$, SE = 0.42, $p = .57$). Together, these results highlight that students' baseline behavioral status plays an important role in shaping interview reactivity.

Some movement features—including absolute acceleration mean, acceleration standard deviation, and mean distance—showed significant positive interaction effects with interview participation. However, although the corresponding main effects were not statistically significant, they were opposite in direction to the interaction terms. This pattern suggests that the direction and magnitude of interview-related effects may vary depending on students' pre-interview behavioral levels. Overall, post-interview movement patterns remain systematically linked to baseline movement behaviors, implying that interview participation may partially shape subsequent movement dynamics.

For features that showed no interview or interaction effects in the full models, reduced models excluding the interaction term were fitted. However, no additional interview effects were detected for other features.

4.3 Direct Attention Cue Effect

Lastly, we examined whether the observed reactivity in the occurrence of observation and tool use was driven by direct attention cues during interviews. This was achieved by comparing interviews that explicitly mentioned the target behavior with those that did not. In terms of interviews mentioning observation and tool use, 31 interviews included the 'observation' keyword while 30 interviews mentioned tool use keywords (see table 1). We compared during/post features while including pre-behavior, interview mention, their interaction, and pre interest as fixed effect (Table 4). Camp and world were included as random effects across all models.

No significant direct attention cue effects or interaction effects with pre-behavior were observed for either observations or tool use. Since neither feature showed a main or an interaction effect in the full models, reduced models excluding the interaction term were fitted. However, no additional attention cue effects were detected for either observation or tool use.

After examining direct attention cue effects, we re-ran the analyses comparing the behaviors of students who were interviewed to that of students who did not get interviewed (as in section 4.2), this time excluding interviews which explicitly mentioned the target behavior (observation or tool use). These analyses were conducted to assess whether interview reactivity was still significant when only considering interviews that did not include direct attention cues. The results showed that the previously reported effect of interview on the occurrence of tool use and its interaction effect between interview and pre-behavior were fully replicated. In contrast, for the occurrence of observation, the previously identified main effect of interview participation was no longer observed.

Table 4. General Linear Mixed Methods Results (Coefficients, Standard Errors, and P-values) for Direct Attention Cue Models and Interview Effect Reproduction Models Predicting Each of Two Features: The Occurrence of Observation and Tool Use.

Behavior	N	Pre-behv	Interview	Pre-behv * Interview	Pre-interest
Direct attention cue effect					
Observation (occurrence)	47	β = 1.42 (0.83) p = 0.085	β = 1.48 (1.38) p = 0.282	β = -2.28 (1.58) p = 0.149	β = 0.14 (0.24) p = 0.564
Tool (occurrence)	47	β = -0.77 (1.18) p = 0.513	β = -0.84 (1.00) p = 0.401	β = 1.54 (1.32) p = 0.242	β = 0.04 (0.26) p = 0.879
Reproduction of interview effect without interview mention					
Observation (occurrence)	257	β = 1.88 (0.42) p = <.001	β = 1.11 (0.52) p = 0.035	β = 0.16 (0.73) p = 0.821	β = 0.15 (0.12) p = 0.187
Tool (occurrence)	246	β = 1.10 (0.39) p = 0.005	β = 1.43 (0.53) p = 0.006	β = -2.14 (0.77) p = 0.006	β = 0.33 (0.12) p = 0.007

Note. β denotes fixed-effect estimates and values in parentheses are standard errors.

5. DISCUSSION AND CONCLUSION

This study examined whether and how data-driven classroom interview (DDCI) influences students' behaviors in a game-based learning context, by comparing a range of in-game behaviors of students following their interviews to the behaviors of students who were not interviewed. Specifically, we investigated behavior differences related to the use of in game observations, scientific measurement tools and movement-related behaviors. The results indicate that DDCI does exert an influence on learners' behaviors; however, this influence is selective rather than uniform across all behavioral features.

First, the occurrences of tool use and observation, two core inquiry behaviors in WHIMC, were positively influenced by interviews. This suggests that rather than distracting students from the learning activity, students may become more engaged in inquiry processes within the game environment following interviews. This finding aligns with the metacognitive reactivity described by Bosch et al. [3], highlighting DDCI's potential to exert positive influences on students.

However, while students showed increased occurrences of tool use and observation following interviews, no significant interview effects and a few interaction effects were found for movement features. Reactivity in in-game observations and scientific measurement tool uses are particularly meaningful because they reflect intentional, goal-directed actions that learners may consciously engage in, in contrast to movement-related features that are more likely to emerge as byproducts of task engagement. This indicates that characteristics of each behavior such as intentionality or goal-direction may influence the impact of DDCI, as suggested by Double and Birney [4] that task characteristics influence how reactivity operates cognitive and metacognitive processes.

Even reactivity within a single behavior should be examined carefully, as it may involve complex underlying mechanisms. In our findings, interview reactivity predicted the occurrence of tool use and observation, but not the frequency of these behaviors. Even these occurrence-level effects were conditional: reactivity depended on students' pre-interview behavior and was most pronounced among those who had not engaged in tool use prior to the interview. Together with other interaction effects for movement-related features in this study, these results suggest that reactivity is not uniform but varies according to individuals' prior states [4]. This variability and complexity indicate that researchers should not conceptualize reactivity as a simple yes-or-no phenomenon, but rather as a multifaceted process shaped by the interplay of task characteristics and individual-level factors.

The mechanisms underlying the reactivity can be interpreted in multiple ways. In the present context, all students were aware of the presence of observers and interviewers; nevertheless, significant behavioral differences emerged between interviewed and non-interviewed students. This finding suggests that the effects observed cannot be explained solely by a general observer effect, but rather that the interview process itself is a contributing factor.

Research on verbalization reactivity supports this interpretation. Studies on think-aloud and verbal reporting have demonstrated that while spontaneous inner thought verbalization tends not to affect performance, higher-level verbalization—such as describing and explaining one's current thoughts and actions—can be reactive [5, 7, 8]. In the present study, while our interviewers attempted to minimize disruption to learners' ongoing activity by avoiding retrospective questions and instead asking students to describe what they were currently doing, responding to such prompts might still require learners to articulate and reflect on their ongoing behavior. Within Ericsson's framework, this constitutes a form of reactive high-level verbalization, which may activate metacognitive processes by encouraging learners to reflect on their actions.

However, inconsistent with this interpretation, the present study didn't identify attention cue effect described in the metacognitive reactivity framework proposed by Double and Birney [4]. No direct attention cue effect was revealed. Also, while interview reactivity was reproduced even without interviews mentioning the behaviors for the occurrence of tool use, this effect was not reproduced for the occurrence of observations. The success to reproduce reactivity of tool use occurrence suggests that the reactivity of tool use might be independent from attention cue effect. On the other hand, failure to reproduce reactivity of observation occurrence suggests either a broader form of attention cueing that is not captured by explicit behavioral mentions alone, or insufficient statistical power due to the small sample size to detect differences between cueing and non-cueing interviews.

These results may also reflect limitations in how attention cues were operationalized in this study. Rather than conducting a detailed analysis of interview content, we used a simple keyword matching approach to identify whether a given behavior was explicitly mentioned in an interview (e.g. mention of 'observation' or 'tool'). It is possible that relevant behaviors were implicitly discussed without the use of specific keywords. For example, in the case of tool use, interviews may have referenced phenomena (e.g., windy area) or interpretations derived from tools without explicitly mentioning the tools themselves (e.g., 'airflow'). Such instances were not systematically captured in the present analysis, which may have attenuated the attention cue effects. More fine-grained analyses of interview content are therefore needed to better understand the mechanisms of DDCI reactivity.

Given that both the present study and prior research [3] consistently indicated that students are likely to be reactive to DDCI, a critical methodological question concerns how such reactivity should be addressed. While traditional methodological perspectives have emphasized minimizing or eliminating reactivity [15], more recent work has advocated for a reactivity-transparent approach, in which

reactivity is explicitly examined, documented, and analytically incorporated rather than treated solely as a source of bias [17, 34]. As DDCI enables the collection of ecologically valid data within authentic learning contexts, understanding how DDCI-induced reactivity influences research findings is essential to fully utilize DDCI's power.

Researchers need to clarify what aspects of learners are influenced, how they are influenced, and when such influences occur. Beyond identifying the existence of reactivity, it is also critical to evaluate its implications for specific research questions. The same reactivity effect may have different consequences depending on the research objective. For example, when the goal is to model interest based on behavioral log, changes of their actual behavior may not substantially distort the results, and this can be tested by validating the model in another context. In contrast, when the research objective is explanatory—such as examining relationships among behaviors—asymmetric reactivity effects across behaviors may bias estimated relationships and lead to misleading conclusions.

Researchers should also consider how to mitigate or account for reactivity effects analytically. If reactivity primarily alters the overall occurrence of behaviors symmetrically across behavioral categories and individuals, normalized analytic approaches may help alleviate its impact [16]. For example, Epistemic Network Analysis (ENA), which emphasizes the relative weights of connections rather than raw behavioral frequencies, can reduce sensitivity to differences in behavioral volume across units. However, such approaches are less effective if reactivity alters the way behaviors are externally manifested, rather than merely their frequency.

Methods for the detailed analysis of students' behavioral log data, such as the one developed by the educational data mining (EDM) community, can contribute to revealing more fine-grained measurement about reactivity and suggesting good ways to address it analytically. By leveraging the richness of behavioral log data and feature engineering techniques, researchers can move beyond treating reactivity as a uniform source and instead examine how it manifests selectively across different behavioral dimensions. In the present study, the use of multiple engineered features enabled us to identify asymmetric reactivity patterns in detail. More broadly, methodological approaches developed in the EDM community may provide analytic strategies which account for the influence of students' reactivity to DDCI on the interpretation of student behavior patterns.

Although this study is among the relatively few to empirically examine reactivity associated with DDCI, it remains limited in scope, focusing on a restricted set of behaviors within a specific learning context. This limitation highlights the need for future research examining DDCI reactivity across diverse contexts and outcome measures, including behavioral, affective, and metacognitive indicators. Additionally, variations in reactivity across demographic groups should be considered, as certain groups may perceive greater levels of surveillance from interviewers. As such evidence accumulates, the role, strengths, and limitations of DDCI as a research instrument can be more clearly articulated. A similar process has unfolded in the development of think-aloud protocols. Since their introduction over three decades ago, think-aloud methods have been subject to sustained debate regarding their identity and reactivity. Through extensive empirical use, meta-analyses, and methodological debate, researchers have developed refined protocols aimed at minimizing reactivity and clarifying appropriate use cases [5–8, 28]. These findings played a central role in differentiating the method from other methods and in establishing empirically grounded guidelines for managing reactivity.

Similarly, DDCI requires sustained empirical investigation and cumulative synthesis to clarify its identity, strengths, and limitations relative to other research instruments. Beyond investigating how the interviews contribute to students' reactivity during DDCI, future work should also examine how other components of the DDCI method, such as variations in interview length, interview triggers design or the interview prioritization algorithm, may impact student reactivity. Through such efforts, DDCI can be more rigorously positioned as a research tool, with its methodological affordances and constraints clearly understood.

6. ACKNOWLEDGMENTS

This study was supported by the National Science Foundation (NSF; DRL-2301172). Any conclusions expressed in this material do not necessarily reflect the views of the NSF. We would also like to thank the entire WHIMC project team for their support throughout this project.

7. ACKNOWLEDGEMENT OF GENERATIVE AI USE

This study used generative AI to assist with data analysis programming and draft revision. No study data were directly uploaded to AI systems, and all AI-generated content was carefully reviewed by the first author.

8. REFERENCES

[1] Baker, R.S., Hutt, S., Bosch, N., Ocumpaugh, J., Biswas, G., Paquette, L., Andres, J.M.A., Nasiar, N. and Munshi, A. 2024. Detector-driven classroom interviewing: focusing qualitative researcher time by selecting cases in situ. *Educational technology research and development*. 72, 5 (Oct. 2024), 2841–2863. https://doi.org/10.1007/s11423-023-10324-y.

[2] Boeder, J.D., Postlewaite, E.L., Renninger, K.A. and Hidi, S.E. 2021. Construction and validation of the Interest Development Scale. *Motivation Science*. 7, 1 (Mar. 2021), 68–82. https://doi.org/10.1037/mot0000204.

[3] Bosch, N., Zhang, Y., Paquette, L., Baker, R., Ocumpaugh, J. and Biswas, G. 2021. Students' Verbalized Metacognition During Computerized Learning. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama Japan, May 2021), 1–12.

[4] Double, K.S. and Birney, D.P. 2019. Reactivity to Measures of Metacognition. *Frontiers in Psychology*. 10, (Dec. 2019). https://doi.org/10.3389/fpsyg.2019.02755.

[5] Ericsson, A. 2003. Valid and non-reactive verbalization of thoughts during performance of tasks towards a solution to the central problems of introspection as a source of scientific data. *Journal of Consciousness Studies*. 10, 9–10 (2003), 1–18.

[6] Ericsson, K.A. and Fox, M.C. 2011. Thinking aloud is not a form of introspection but a qualitatively different methodology: Reply to Schooler (2011). *Psychological Bulletin*. 137, 2 (2011), 351–354. https://doi.org/10.1037/a0022388.

[7] Ericsson, K.A. and Simon, H.A. 1980. Verbal reports as data. *Psychological review*. 87, 3 (1980), 215.

[8] Fox, M.C., Ericsson, K.A. and Best, R. 2011. Do procedures for verbal reporting of thinking have to be reactive? A meta-analysis and recommendations for best reporting methods. *Psychological bulletin*. 137, 2 (2011), 316.

[9] Gadbury, M. and Chad Lane, H. 2023. A Bayesian Analysis of Adolescent STEM Interest Using Minecraft. *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*. N. Wang, G. Rebolledo-Mendez, V. Dimitrova, N. Matsuda, and O.C. Santos, eds. Springer Nature Switzerland. 384–389.

[10] Gieles, E., Vogelsmeier, L., Jongerling, J., Smeets, T. and Dejonckheere, E. 2025. Participation Effects in Ecological Momentary Assessment Research: A Taxonomy and Call to Action. PsyArXiv, 4 (Nov. 2025). https://doi.org/doi:10.31234/osf.io/cj8ut_v2.

[11] Greene, J.C. 2008. Is Mixed Methods Social Inquiry a Distinctive Methodology? *Journal of Mixed Methods Research*. 2, 1 (Jan. 2008), 7–22. https://doi.org/10.1177/1558689807309969.

[12] Koriat, A., Nussinson, R., Bless, H. and Shaked, N. 2013. Information-based and experience-based metacognitive judgments: Evidence from subjective confidence. *Handbook of metamemory and memory*. Psychology Press. 117–135.

[13] Kvale, S. and Brinkmann, S. 2009. *Interviews: Learning the craft of qualitative research interviewing*. sage.

[14] Lane, H.C., Gadbury, M., Ginger, J., Yi, S., Comins, N., Henhapl, J. and Rivera-Rogers, A. 2022. Triggering STEM interest with Minecraft in a hybrid summer camp. *Technology, Mind, and Behavior*. 3, 4 (2022).

[15] Lietz, C.A. and Zayas, L.E. 2010. Evaluating qualitative research for social work practitioners. *Advances in Social work*. 11, 2 (2010), 188–202.

[16] Liu, X., Gurung, A., Baker, R.S. and Barany, A. 2024. Understanding the Impact of Observer Effects on Student Affect. *Advances in Quantitative Ethnography*. Y.J. Kim and Z. Swiecki, eds. Springer Nature Switzerland. 79–94.

[17] Maxwell, J.A. 2013. *Qualitative research design: An interactive approach: An interactive approach*. sage.

[18] McCarney, R., Warner, J., Iliffe, S., Van Haselen, R., Griffin, M. and Fisher, P. 2007. The Hawthorne Effect: a randomised, controlled trial. *BMC Medical Research Methodology*. 7, 1 (Dec. 2007), 30. https://doi.org/10.1186/1471-2288-7-30.

[19] McCarthy, D.E., Minami, H., Yeh, V.M. and Bold, K.W. 2015. An Experimental Investigation of Reactivity to Ecological Momentary Assessment Frequency among Adults Trying to Quit Smoking. *Addiction (Abingdon, England)*. 110, 10 (Oct. 2015), 1549–1560. https://doi.org/10.1111/add.12996.

[20] Ocumpaugh, J. et al. 2025. The Quick Red Fox gets the best Data Driven Classroom Interviews: A manual for an interview app and its associated methodology. arXiv.

[21] Ocumpaugh, J., Roscoe, R.D., Baker, R.S., Hutt, S. and Aguilar, S.J. 2024. Toward Asset-based Instruction and Assessment in Artificial Intelligence in Education. *International Journal of Artificial Intelligence in Education*. 34, 4 (Dec. 2024), 1559–1598. https://doi.org/10.1007/s40593-023-00382-x.

[22] Patton, M.Q. 2002. *Qualitative research and evaluation methods 3rd. ed*. Sage publications.

[23] Perera, K. 2020. The interview as an opportunity for participant reflexivity. *Qualitative Research*. 20, 2 (Apr. 2020), 143–159. https://doi.org/10.1177/1468794119830539.

[24] Procter, I. and Padfield, M. 1998. The effect of the interview on the interviewee. *International Journal of Social Research Methodology*. 1, 2 (Jan. 1998), 123–136. https://doi.org/10.1080/13645579.1998.10846868.

[25] Ravitch, S.M. and Carl, N.M. 2019. *Qualitative research: Bridging the conceptual, theoretical, and methodological*. Sage publications.

[26] Rosenthal, R. and Jacobson, L. 1966. Teachers' Expectancies: Determinants of Pupils' IQ Gains. *Psychological Reports*. 19, 1 (Aug. 1966), 115–118. https://doi.org/10.2466/pr0.1966.19.1.115.

[27] Saretsky, G. 1972. The OEO PC experiment and the John Henry effect. *The Phi Delta Kappan*. 53, 9 (1972), 579–581.

[28] Schooler, J.W. 2002. Verbalization produces a transfer inappropriate processing shift. *Applied Cognitive Psychology*. 16, 8 (Dec. 2002), 989–997. https://doi.org/10.1002/acp.930.

[29] Siemens, G. and Baker, R.S.J.D. 2012. Learning analytics and educational data mining: towards communication and collaboration. *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge* (Vancouver British Columbia Canada, Apr. 2012), 252–254.

[30] Sporrong, S.K., Kalleberg, B.G., Mathiesen, L., Andersson, Y., Rognan, S.E. and Svensberg, K. 2022. Understanding and addressing the observer effect in observation studies. *Contemporary research methods in pharmacy and health services*. Elsevier. 261–270.

[31] Wengraf, T. 2001. Qualitative research interviewing: Biographic narrative and semi-structured methods. (2001).

[32] Winne, P.H. 2020. Construct and consequential validity for learning analytics based on trace data. *Computers in Human Behavior*. 112, (Nov. 2020), 106457. https://doi.org/10.1016/j.chb.2020.106457.

[33] Zhou, Y. and Paquette, L. 2024. Investigating Student Interest in a Minecraft Game-Based Learning Environment: A Changepoint Detection Analysis. *Proceedings of the 17th International Conference on Educational Data Mining, EDM 2024* (Atlanta United States, July. 2024). https://doi.org/10.5281/ZENODO.12729844.

[34] Zahle, J. 2023. Reactivity and good data in qualitative data collection. *European Journal for Philosophy of Science*. 13, 1 (Mar. 2023), 10. https://doi.org/10.1007/s13194-023-00514-z.

Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations

E. Margaret Perkoff[1]
eperkoff@upenn.edu

Megan Lyons[2]
lyonsm3@chop.edu

Alexandria Mulqueen[2]
mulqueena@chop.edu

Ryan S. Baker[3]
ryan.baker@adelaide.edu.au

Alan Gee[4]
alan.gee@amiralearning.com

Jaclyn Ocumpaugh[5]
jocumpau@central.uh.edu

Adam Porsch[4]
adam.porsch@amiralearning.com

Neville Ryant[6]
nryant@ldc.upenn.edu

Julia Parish-Morris[2]
parishmorrisj@chop.edu

Lyle Ungar[1]
ungar@upenn.edu

[1]University of Pennsylvania [2]Children's Hospital of Philadelphia [3]Adelaide University
[4]Amira Learning [5]University of Houston [6]Linguistic Data Consortium

ABSTRACT

The prevalence of online reading platforms has surged along with improvements to automatic speech recognition (ASR) systems. These platforms facilitate personalized reading instruction, but like all learning systems, work best when students find them engaging. The Educational Data Mining (EDM) community has made great efforts to detect student engagement, but these efforts have largely been confined to school and lab environments without intentionally including neurodiverse populations. In the present study, we evaluate engagement levels in children with attention deficit hyperactivity disorder (ADHD, n=55), autism (n=38), both autism and ADHD (n=38) and with no known clinical diagnoses (n=22) as they interact with a digital literacy learning system from their homes. Engagement was measured using parent report, and acoustic features were extracted from audio recordings of the students while they interacted with the Amira literacy learning system. Our results indicate that acoustic measures related to temporal features, pitch, and vocal quality are related to engagement levels across all clinical groups. We also identify key temporal features that vary across different clinical groups. These findings demonstrate the validity of using non-invasive acoustic features to measure engagement levels during reading comprehension tasks at home. However, interaction effects also highlight the importance of taking into account children's clinical diagnosis when measuring engagement.

Keywords

Reading Engagement, Online Reading Comprehension, Neurodivergent

E. Margaret Perkoff, Megan Lyons, Alexandria Mulqueen, Ryan S. Baker, Alan Gee, Jaclyn Ocumpaugh, Adam Porsch, Neville Ryant, Julia Parish-Morris, and Lyle Ungar. Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 90–103. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039919

1. INTRODUCTION

Digital reading platforms have emerged as a promising way to provide personalized literacy instruction to students, both at home and at school. These platforms guide the student through level-appropriate passages where they can read text out loud, receive real-time interventions, and respond to reading comprehension questions[69]. Compared to traditional instruction, Intelligent tutoring systems (ITSs) for reading can even produce larger learning gains [65, 70], even among students with learning differences that sometimes interfere with reading comprehension, such as Autism Spectrum Disorder (ASD) [39]. However, meta-reviews of literacy applications have demonstrated that they are most effective when (1) the systems are highly interactive and (2) children use them with higher intensity (at least 75 minutes per week) and duration (at least 3 months) [39, 69, 70]. This leads to a direct need to understand how engaged students are with a platform in order to provide effective instruction and keep them coming back for regular sessions.

Reading engagement is often defined as the student's active involvement with reading [32], and it is sometimes distinguished from motivation by considering motivation as a mindset that is necessary in order to deeply engage with reading [2]. Although exact definitions of reading engagement vary, it is often described as including cognitive, affective, and behavioral dimensions [41]. Reading engagement is not only indicative of children's enjoyment, but is also linked to improved reading abilities and reading comprehension [6, 31, 41, 63].

Despite an abundance of research on engagement measures, there is no standardized method for measuring student's engagement with online reading platforms. Some conversational agent based literacy tools operationalize verbal engagement in terms of child response length [33] or semantic features including topic relevance and intelligibility [68]. However, these measures are proxies for the actual construct, which is likely to involve a range of cognitive, behavioral, and affective experiences [26]. Moreover, they do not account for key acoustic elements, including those that may be more in-

dicative of a students' affective experience with the reading activity [44].

In this study, we explore the use of acoustic features to detect reading engagement among neurodivergent students with different clinical diagnoses, including Autism and Attention Deficit Hyperactivity Disorder (ADHD). We examine the use of acoustic features as they can be easily extracted by voice-based reading software without the use of specialized equipment that may be difficult to acquire for use in either the home or classroom. We evaluate the interaction between clinical diagnosis and engagement levels by analyzing acoustic features that indicate voice quality, speaking rates, and prosody, and which have been associated with engagement in prior research [34, 44].

To our knowledge, this is one of the first studies to focus on understanding which acoustic features most effectively model engagement during digital reading instruction, and one of the first in the EDM community to consider how clinical diagnoses might affect those modeling efforts. Our results indicate that temporal acoustic features are robust indicators of engagement across groups in our study, but other acoustic features (e.g., variation of voiced segment length) are diagnosis-sensitive. With this work, we aim to contribute to a better understanding of how engagement manifests differently in diverse populations of learners while also putting forward tools to help detect these differences, provide personalized support, and ultimately improve outcomes.

2. BACKGROUND AND RELATED WORK

Research on student engagement defines it across many dimensions, including behavioral, emotional, and cognitive domains [26]. Real-time models of student engagement have become a common way for researchers in the Educational Data Mining community to study engagement at scale, including both affective engagement and cognitive engagement. Such models have been trained on a variety of label types, including those produced from text-based video [76], video data [10, 36], speech [71, 44] physiological signals including skin conductance data [12, 77], posture data [56], and self-report [74, 18]; and from classroom observations [5, 50].

Much of this work has focused on affective engagement [13], although behavioral and cognitive engagement [30, 42] are also modeled. Such models have been particularly important for researchers studying affect dynamics [20, 51]. They have also been useful for making short-term and long-term predictions about student learning outcomes [27]. However, given the rich history of definitions involved with studying student engagement in the broader research community [26], the EDM community continues to explore novel ways to capture this construct.

2.1 Detecting Reading Engagement in Learning Systems at Home

One particularly difficult challenge in this space is appropriately creating high quality engagement labels for students who use learning software at home. Many sensors used to detect engagement can be highly invasive (e.g., video), while others are difficult to use in the home environment (e.g., posture sensors). Meanwhile, interaction-based detectors [50] rely on classroom observation or self-report data to provide the labels. Although these detectors can certainly be applied to students in the home environment, it is difficult to verify whether detectors trained on labels from a classroom environment would generalize to students' home environments. Additionally, self-reports are susceptible to self-presentation effects and to differences in students' ability to gauge their own level of participation [21, 28].

A few studies have sought to leverage measures that could be more easily collected in a home environment. For example, models based on student response times leverage item response theory to distinguish between engaged and disengaged students in online settings [7, 35]. These could be used in home environments, even if they might need to be renormed for different clinical populations. Another approach is to measure student talk time while engaging with an online tutor[17]. Some existing reading software measures verbal engagement based on the number of turns with a conversational agent or word counts [33]. Xu et al. [68] evaluated the nuances of children's responses to comprehension conversations asked by conversational agents versus humans via five aspects of verbal engagement: lexical diversity, intelligibility, productivity, topic relevance, and accuracy [66, 64]. Similarly, Gorgun et al. leveraged linguistic features in online discussion posts to estimate engagement based on third-party manual coding [30]. Their scheme defined four attributes of cognitive engagement: interactive, constructive, active, and social.

In addition to semantic-based engagement features, some researchers have explored the use of acoustic features in the online setting. Litman et al. [44] evaluated prosodic cues of disengagement and uncertainty while students engaged with a physics tutor. Their research demonstrated that when students are disengaged, they respond significantly more slowly than engaged students. Disengaged students were also found to have lower maximum and mean pitch values than their engaged peers. Furthermore, disengaged students demonstrated more steady pitch values as well as lower levels of energy. Our study builds upon these findings, with the intention of exploring whether they apply to non-neurotypical learners.

2.2 Reading Engagement in Individuals with ASD and ADHD

As our ability to detect engagement in online settings has advanced, there have been increasing calls for better understanding for a more diverse range of learners [52], and subsequent research has shown that additional demographic information can improve models. For example, Karumbaiah et al. [38], shows that some behavioral engagement measures (i.e., help-seeking) are better understood when school-level demographics are taken into account. Likewise, other research has shown that we can improve our models of mind-wandering by accounting for neurodivergence [67].

Accounting for clinical diagnoses in engagement research is particularly important since certain conditions are characterized by developmental challenges that begin to emerge during early childhood and impact multiple life domains (e.g. school, home) across the lifespan [49]. These increasingly prevalent conditions are characterized by factors that

Table 1: Demographic and clinical characteristics of participant groups. Number of activities signifies the number of comprehension conversations that were included from each participant subset.

Clinical Group	Mean Age (SD)	Male	Female	Ethnicity	Activities
ADHD	8.527 (1.386)	35	20	Not Hispanic or Latino (48)	305
				Hispanic or Latino (7)	
ASD	8.345 (1.632)	14	15	Not Hispanic or Latino (22)	129
				Hispanic or Latino (7)	
ASD+ADHD	8.316 (0.904)	22	16	Not Hispanic or Latino (34)	169
				Hispanic or Latino (4)	
Not ASD or ADHD	8.409 (1.054)	11	11	Not Hispanic or Latino (18)	103
				Hispanic or Latino (4)	
Overall	8.417 (1.276)	82	62	Not Hispanic or Latino (122)	709
				Hispanic or Latino (22)	

could interfere with our measurement of student engagement [14]. For example, individuals with Autism Spectrum Disorder (ASD) face persistent challenges in social communication, as well as restricted or repetitive patterns of behaviors, interests, and activities [4]. Attention-deficit/hyperactivity disorder (ADHD) is a neurodevelopmental condition diagnosed in about 1 in 11 (10.5%) of U.S. school-aged children [15] and is distinguished by persistent patterns of inattention and/or hyperactivity-impulsivity. While autism and ADHD are distinct neurodevelopmental conditions, they frequently co-occur. 50-70% of people with ASD meet diagnostic criteria for ADHD [58] and 12.5% of individuals with ADHD meet diagnostic criteria for ASD [73]. This co-occurrence (colloquially referred to as AuDHD) could result in additional nuances in learning behavior that would further complicate accurate measurement of engagement in this population.

Although ASD and ADHD often co-occur with disabilities that directly impair reading skills (e.g. dyslexia; [11]), various clinical features specific to ASD and ADHD can also contribute to challenges with reading engagement and comprehension for children with either condition. Specifically, impairments in social communication, word recognition, and oral language may compromise many autistic children's reading comprehension skills [57, 46, 40]. Furthermore, some early autistic readers exhibit a striking discrepancy between their reading comprehension and word recognition skills such that their reading comprehension skills lag behind their growing proficiency in word recognition [16]. Reading comprehension may also pose difficulty for children with ADHD. Working memory impairments that are characteristic of ADHD can constrain reading comprehension and limit children's ability to recall central ideas from text directly after reading it [47]. Additionally, on average, children with ASD and/or ADHD may struggle to maintain sustained attention on tasks they are not interested in or motivated to complete. This suggests that children with ASD and/or ADHD may exhibit reduced engagement when reading texts they did not choose [37] or that is not related to their interests [22]. Thus, several clinical characteristics associated with ASD and ADHD may exert a powerful influence on reading engagement and comprehension. With the rising use of digital literacy platforms aimed at supporting children's reading skills, there is a critical need to accurately measure how children with ASD and/or ADHD engage with and comprehend the texts they read on these platforms.

3. PRESENT STUDY

3.1 Digital Literacy Platform

Amira [1] is an online, AI-powered reading platform that delivers real-time, adaptive literacy instruction. The virtual tutor,named Amira, listens to students as they read aloud, identifies reading miscues or skill gaps, and provides brief, real-time instructional supports (micro-interventions) aligned to the Science of Reading and tailored to the student's ability level and performance history. During comprehension dialogue activities, Amira prompts students with questions designed to assess and deepen skills that are aligned to grade-level Reading Comprehension standards. For students in grades 3 and above—where the cognitive and language demands are appropriate—Amira may initiate a comprehension dialogue micro-intervention rather than using the traditional multiple-choice format. These dialogues use open-ended prompts to probe the student's ability to demonstrate the skill, capturing explanation, reflection, or elaboration. Dialogues are modeled after authentic classroom questioning. The student responds either orally or by typing a response, and Amira uses advanced speech recognition and natural language processing (NLP) to capture and compare student responses against a detailed pre-defined rubric.

Each conversation can extend up to two turns of dialogue. The dialogue follows the structure of an opening prompt from the tutor, student response, a follow-up tutor prompt, student response, and final tutor response. A response timer is activated after the initial prompt is delivered, and students are given 10 seconds to begin their response to that turn. If no response is detected within this timeframe, the system will automatically deliver scaffolded support (a hint and a re-prompting of the same question). If any response attempt is detected, the student has 60 seconds in total to respond to the prompt on each turn. There is a visual timer present during the interaction to convey this per-turn time limit. The follow-up prompt is intended to scaffold the student towards further addressing rubric criteria that they did not adequately cover in their initial answer.

All response data—including ASR transcripts, timing, accuracy of the response based on the skills being taught, and behavioral markers such as hesitation or repetition—is logged for subsequent analysis, enabling researchers to investigate both dialogue engagement and comprehension skill performance within a dialogic reading context. For this study, we focus solely on the audio recordings and transcribed stu-

Table 2: Subset of acoustic features from eGeMAPS [23] use in analysis grouped by category.

Domain	Features
Temporal	Loudness Peaks Per Sec; Voiced segment length (s: mean, std. dev.); Unvoiced segment length (s: mean, std. dev.)
Energy	Harmonic-to-Noise ratio (mean, std. dev.)
Frequency	Fundamental Frequency (F0: mean, std. dev., 20th, 50th, 80th percentiles); F0 mean rising slope; F0 mean falling slope; jitter (mean, std. dev.)
Spectral	Alpha ratio, voiced (mean, std. dev.); Alpha ratio, unvoiced (mean); Hammarberg index, voiced (mean, std. dev.); Hammarberg index, unvoiced (mean, std. dev.); Spectral flux, voiced (mean, std. dev.); Spectral flux, unvoiced (mean); Spectral slope 0–500 Hz (mean, std. dev.); Spectral slope 500–1500 Hz (mean, std. dev.)

dent responses from the comprehension dialogues described above.

3.2 Methods

3.2.1 Participants and Data Collection

Participants who consented to be contacted for research studies were recruited via the Center for Autism Research (CAR) at the Children's Hospital of Philadelphia (CHOP). CHOP's Electronic Health Records (EHR) database. 1000 emails per week were sent out to CHOP patients based on certain selection criteria, including (a) diagnosis of autism spectrum and/or attention deficit/hyperactivity spectrum disorder or not, (b) age 5-12 years, and (c) in the "learning to read" phase according to parent report. 144 children aged 5-12 years with a diagnosis of autism (N=29), ADHD (N=55), autism + ADHD (N=38), or no diagnoses of autism or ADHD (N=22) consented to participate in a 7-week at-home reading intervention pilot study. Once enrolled, participants were instructed to read four to five stories per week on Amira , or for at least 30 minutes. Caregivers completed surveys providing basic demographic information and feedback on their child's experience with Amira. Full demographic details for participants can be seen in Table 1. This study was overseen by the Institutional Review Boards at CHOP and MDRC.

3.2.2 Acoustic Features and their Extraction

For each of the comprehension activity turns, we extract acoustic features based on the extended Geneva Minimalistic Acoustic Parameter set (eGeMAPS) [23] using the OpenSMILE toolkit[25]. eGeMAPS is considered a high quality acoustic feature set for use in emotion recognition tasks [19], which are closely related to the engagement identification question. eGeMAPS consists of 25 low-level acoustic features comprising four key categories related to these tasks: prosody, temporal, pitch, and spectral features. Acoustic features are extracted at regular intervals (e.g, every 10 ms) from speech samples and the resulting feature time series is mapped to fixed-dimensional feature vectors by applying one or more statistical functions (e.g., mean, standard deviation, percentiles), resulting in 88 features[1] for each student response turn. For frequency values, we focus solely on the fundamental frequency F_0 which has been used in prior studies as a measure of distinguishing pitch values between neurodivergent and neurotypical students [53]. This results in a subset of 49 acoustic features that we use for our analysis as seen in Table 2.

Table 3: Breakdown of the study dataset based on values with the parent-reported engagement labels: not engaged (NE), Somewhat Engaged (SE), and Very Engaged (VE). Only students with audio recording files are included in the count.

Clinical Group	Not Engaged	Somewhat Engaged	Very Engaged
ADHD	1	17	26
ASD	0	8	13
ASD+ADHD	1	12	8
Not ASD or ADHD	0	4	8

Prosodic features attempt to characterize the expressive qualities of speech including voice quality, rhythm, and intonation that convey emotion and emphasis; examples include jitter, shimmer, and harmonic noise ratio measures.

Frequency features attempt to characterize how high or low the voice sounds and how it varies over time, capturing both the fundamental frequency (F_0, perceived as pitch) and the resonant frequencies of the vocal tract (formants). As mentioned above, we only analyze fundamental frequency values in this work. These features were crucial in the work of Litman et al. [44] which showed that pitch was statistically significant for discriminating between engaged and disengaged students.

Temporal features attempt to characterize speech timing, including speaking rate and pause frequency. We consider the rate of speech, number of pauses based on loudness values, and variations in voiced and unvoiced segment lengths, which have been used in prior work discriminating between student engagement levels [44].

Spectral features are features that correspond to the spectral shape, such as slope, which are frequently used for affect recognition [23, 61]. These also include the first four Mel-Frequency Cepstral Coefficients (MFCCs), which have a documented association with valence and level of interest [24].

For further details regarding feature extraction (e.g., frame step, window size, MFCC filterbank/DCT parameters, functionals applied to each acoustic feature), consult [19] and [23].

3.2.3 Parent Reported Engagement (PRE) Labels

Following seven weeks of the at-home intervention, parents were given a survey to gather information about their child's

[1]Not all statistical functions are applied to all low-level acoustic features.

experience with the literacy learning system. This survey included a specific question about how engaged their child was while interacting with the tutor with three possible values: *Not Engaged*, *Somewhat Engaged*, and *Very Engaged*. This measure, which we will refer to as the parent-reported engagement (PRE) label, represents a relatively basic measure of engagement, as it does not ask parents to distinguish between the various ways in which engagement might manifest. However, it represents a valuable perspective on engagement from caregivers who are most familiar with the participants' unique focus styles and avoids concerns about the reliability of children's self-report data [21, 28].

PRE was provided by the parents of 139 participants, with 98 of those participants having audio recordings available for comprehension conversations. Labels were highly skewed towards the *Very Engaged* and *Somewhat Engaged* values with only 4 participants reported as *Not Engaged* during the sessions, only two of whom had audio recordings. As such, any patterns observed involving the Not Engaged group should be considered exploratory. The full distribution of clinical diagnosis based on engagement level can be seen in Table 3. We exclude any students who have fewer than two audio recordings present in the data. We apply the parent-reported engagement (PRE) level as a measure of overall engagement propensity to all comprehension activities with audio recording for the student. This is not intended to indicate turn-level engagement, but instead enable us to identify what acoustic features are correlated with children's general engagement statement.

3.2.4 Data Preprocessing

We conducted a series of preprocessing steps. In order to ensure that each participant had enough acoustic data for analysis, we first selected for students who had participated in at least two comprehension activities. We then aggregated all of the comprehension conversation turns for that student and aligned them with the appropriate audio file. Acoustic features were extracted using the OpenSMILE [25] toolkit for each of the student conversational turns. We collected system-generated transcriptions of student responses, which were processed through a simple word count function to generate verbal response counts for each turn. We then align the transcribed student response data, audio files, and the demographic and engagement reports from the parent surveys[2].

3.2.5 Measuring the Effect of Diagnosis x Engagement

To examine effects of clinical diagnosis and PRE levels on acoustic features while accounting for the fact that we have multiple audio recordings for each child, we employed population average models using generalized estimating equations (GEE). GEE models are well-suited for correlated data and provide robust inference without requiring explicit modeling of subject-specific random effects [43, 59, 75]. We also explored the use of mixed-effect models; however, they could not be applied across all the desired acoustic features due to limited within-child variability.

[2]The resulting dataset will be released for future work following the completion of the requisite anonymization process.

Table 4: Number of acoustic features showing significant population-level effects after FDR correction, stratified by feature category and predictor.

Feature Category	Diagnosis	Eng.	Diagnosis × Eng.
Energy	0	3	3
Frequency	0	6	8
Spectral	0	14	27
Temporal	0	2	5
Total	**0**	**25**	**33**

For each acoustic feature mentioned above, we fit a Gaussian GEE of the form:

$$\begin{aligned} Feature_{ij} = \beta_0 + \boldsymbol{\beta}_1^\top \text{Diagnosis}_i + \beta_2 \text{Engagement}_i \\ + \boldsymbol{\beta}_3^\top \left(\text{Diagnosis}_i \times \text{Engagement}_i\right) \\ + \beta_4 \text{Age}_i + \beta_4 \text{Sex}_i + \varepsilon_{ij}. \end{aligned} \quad (1)$$

where i represents children and j represents individual acoustic observations. Individual acoustic observations are represented by a vector of acoustic features from OpenSMILE that were extracted from a single turn in the comprehension dialogue. These models included clinical diagnosis, engagement, and the interaction of diagnosis and engagement as categorical predictors, with child age (mean-centered) and sex (parent-reported assigned sex at birth) as covariates. They were fit using the python *statsmodels* package [60] with the correlation structure set to exchangeable to account for the within-child correlation from the repeated observations [43]. Given the broad range of values for different acoustic features, we normalized the values using *StandardScaler* from *scikit-learn* [55]. We also use the *statsmodels* package to account for possible false-discovery rates (FDR) for diagnosis, engagement, and interaction effects with the Benjamini-Hochberg false discovery rate procedure [8] with adjusted p-values set using the methodology defined by Yekutieli and Benjamini [72].

4. RESULTS

We examine the results of fitting a GEE model across the 49 acoustic features in Table 2 as well as the verbal response count of the student responses. In Table 4, we present the acoustic features that are the most impacted by engagement, diagnosis or the combination of diagnosis and engagement. The parent-reported level of engagement acts as an independent predictor of acoustic features across all the categories. There were no statistically significant diagnosis-related effects present in the dataset. The interaction effects of diagnosis and parent-reported engagement are distributed across all of the feature categories.

4.1 Verbal Response Count

Table 5: GEE Results for Verbal Response Count. This table shows the FDR-adjusted p-values across all model terms. None of the values were considered statistically significant.

Predictor	FDR-adjusted p-value
Diagnosis	0.0627
Engagement	0.2420
Diagnosis × Engagement	0.2998
Age	0.1256
Sex	0.3986

In alignment with prior work in verbal engagement, we created a GEE model for the normalized verbal response count. This is taken as the count of words from the transcribed student response for a single turn. The results are displayed in Table 5. Our findings demonstrate that there was no statistically significant impact of engagement, diagnosis, or the interaction of the two on this metric. This suggests that this feature is not dependent on any of these categories.

4.2 Effect of Age and Sex on Results

We account for variations in age and sex in our results by adding them as intercepts to the GEE model. The results that we report for all engagement, diagnosis, and their interaction are those that did not have a statistically significant influence from these variables. Of the 49 acoustic features evaluated, only one had a statistically significant association age: the standard deviation of jitter. Mean child age had a positive coefficient of 0.101, indicating that as children get older than the mean age (in our case 8.4 years old), their variability in jitter levels increased as well. Child sex (as reported by parents in the demographic survey) did not have a statistically significant association with any of the acoustic features tested.

4.3 Engagement Effects

The GEE model shows that PRE levels have an effect on three *energy*-related features (i.e., shimmer, mean Harmonic-to-Noise Ratio and standard deviation of the Harmonic-to-Noise Ratio). PRE also impacts a number of *frequency* features (e.g., those related to the pitch based on F_0 semitone values and jitter, and changes in consecutive F_0 frames) and *spectral shape* features (e.g., Alpha Ratio, Hammarberg Index, and Spectral Slope values). In addition, PRE affects both the mean values of voiced segment length as well as loudness peaks per second - both of which are associated with speaking rates. These findings are consistent with prior work evaluating fluctuations in acoustic features related to arousal, engagement, and emotion [3, 7, 24, 44]. The majority of the contrast effects for the model appear between the *Not Engaged* and *Very Engaged* groups, however, we do see a meaningful difference between shimmer values for the *Somewhat Engaged* and *Very Engaged* groups. Students in the Somewhat Engaged groups (regardless of clinical diagnosis) exhibited higher mean values of shimmer than those in the Very Engaged group.

4.4 Diagnosis Effects

The GEE analysis of our data did not result in any statistically significant effects for diagnosis as an independent predictor. These findings are surprising given a large body of existing research that demonstrates noticeable differences in prosody, pitch, and voice quality for individuals with autism spectrum disorder vs. neurotypical peers [9, 48, 53, 54]. It is possible that no differences were detected in this study due to high variability in the speech characteristics of individuals with a given diagnosis or the fact that this sample was collected in a highly constrained (and non-social) context. There are still context-dependent diagnostic differences within the dataset that vary based on the child's parent-reported engagement level.

4.5 Diagnosis x Engagement Effects

As shown in Table 4, thirty-three acoustic features exhibited significant interactions between clinical diagnosis and PRE, indicating that diagnostic differences in vocal production depended on PRE. Multiple acoustic categories showed interaction effects, including three associated with *energy*, eight associated with *frequency*, twenty-seven involved with *spectral shape*, and five involved with *temporal values*. Of these features, 18 of them demonstrated an interaction effect without the presence of a statistically significant effect from one of the other predictors. We report the full set of the interaction effects in Table 6. Notably, features related to pitch distribution, spectral variability, and voice quality showed the strongest evidence of context-dependent diagnostic effects, underscoring the importance of considering engagement state when interpreting acoustic markers of clinical diagnosis in educational interactions. The contrast effects referencing Not Engaged in the term should be interpreted as exploratory since there were only 2 students in the analyzed data with audio recordings at this engagement propensity level. Coefficients and patterns reported for the Somewhat Engaged level are representative of a more substantial number of participants and observations in the dataset.

4.5.1 Energy

All three of the features in the energy category demonstrated interaction effects: the mean Harmonics-to-Noise (HNR) ratio and the standard deviation for the shimmer values. However, none of the individual contrast effects between groups registered as statistically significant. There are still some trends that are present in the dataset that should be mentioned for future exploration. Autistic children who were more engaged exhibited higher values of HNR whereas the neurotypical group had the opposite pattern. As seen in Figure 3, For HNR, the ADHD and ASD + ADHD groups more closely resemble the pattern exhibited by neurotypical children, with both showing lower values when Very Engaged vs. Somewhat Engaged. When children in the ADHD and ASD+ADHD groups are rated as Not Engaged, the HNR values are dramatically lower than for the mid-level of engagement, but there is high variability in these groups. This finding is harder to interpret given that there are a very limited number of samples in the dataset with the Not Engaged label. From a practical perspective, when HNR is higher it is indicative of clearer phonation [3] — which means that children with ASD are pronouncing words more clearly when more engaged whilst the other groups are actually reducing their clarity.

We also see opposing patterns between the neurotypical and ASD groups for the shimmer feature. The standard deviation for shimmer is lower for more engaged neurotypical children, whereas it is higher for more engaged ASD children. The more-engaged ADHD group shows a slight increase in shimmer values. Higher shimmer values mean that there is more intensity variability, which is frequently associated with high arousal emotions and higher levels of stress [3].

4.5.2 Frequency

Variations in the frequency characteristics of speech are highly indicative of individual's arousal levels and engagement [3, 23, 44]. In Figure 4, we can see the difference in mean values

Table 6: Acoustic features that exhibit a statistically significant contrast effect (after FDR) based on the interaction of diagnosis × engagement. The reference group for the clinical diagnosis is always *Not ASD or ADHD* and for engagement it is always *Very Engaged*, not engaged is represented as NE and somewhat engaged is represented as SE. Cells show GEE coefficient (Standard Error). * indicates $p < .05$. Dashes indicate non-significant contrasts. † indicates contrast excluded due to anomalous SE = 0. All values rounded to 4 decimals.

Feature Type	Feature	ADHD:NE	ASD:NE	ASD:SE	ASD+ADHD:NE
Spectral	alphaRatioUV sma3nz (mean)	−0.5758 (0.1514)*	—	—	0.4108 (0.1876)*
	hammarbergIndexUV sma3nz (mean)	0.8934 (0.1482)*	—	—	−0.6737 (0.1893)*
	hammarbergIndexV sma3nz (std dev Norm)	−0.1400 (0.0629)*	†	—	—
	mfcc1V sma3nz (std dev Norm)	−0.2257 (0.0614)*	†	−0.1860 (0.0932)*	—
	mfcc2 mean	0.4623 (0.1637)*	—	—	−0.4865 (0.1950)*
	mfcc3 mean	−0.8495 (0.1296)*	—	—	0.8784 (0.1643)*
	mfcc4 mean	0.7354 (0.1316)*	—	0.8278 (0.3660)*	−0.5800 (0.1899)*
	spectralFlux sma3 (std dev Norm)	0.4964 (0.1335)*	—	—	−0.4418 (0.1846)*
	slopeV0_500 sma3nz (std dev Norm)	−0.0945 (0.0404)*	—	—	—
	slopeV500_1500 sma3nz (std dev Norm)	0.0923 (0.0463)*	†	—	—
Frequency	F0semitone mean Rising Slope	−0.2054 (0.0756)*	—	—	—
	jitter Local Mean	0.2886 (0.0955)*	—	—	—
Temporal	Unvoiced Segment Length (mean)	0.5296 (0.0669)*	—	—	−0.4114 (0.1307)*
	Unvoiced Segment Length (std dev)	0.3700 (0.1083)*	—	—	−0.3406 (0.1305)*
	Voiced Segment Length (std dev)	−0.4769 (0.0996)*	†	—	0.4366 (0.1078)*

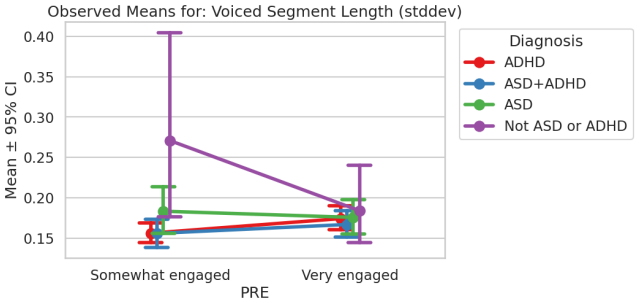


Figure 1: Observed means of the variations of standard deviation in voice segment length across different clinical diagnoses and parent reported levels of engagement.

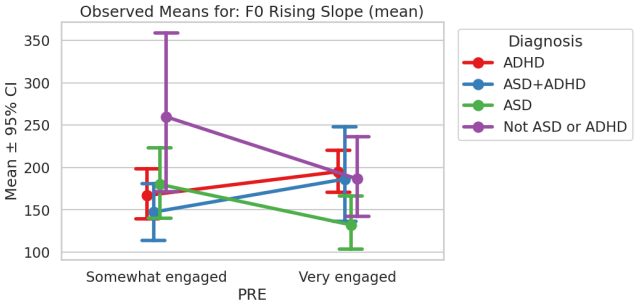


Figure 2: Diagnosis-group level means of the of F0 rising slope across different clinical diagnoses and parent reported levels of engagement.

for the rising slope for F_0 across different clinical diagnoses and engagement values. The neurotypical group exhibits a higher range of values than all of the neurodivergent groups when rated as Somewhat Engaged. These values increase between the Somewhat and Very Engaged groups for individuals with ADHD or ASD+ADHD, but they decrease for neurotypical participants. Essentially, this is indicating that when individuals with ADHD are more engaged they raise their pitch more quickly, whereas the neurotypical students will drop the rate of pitch change when more engaged.

We also notice a stark contrast between the trends of mean values for jitter between the participants with ASD when compared to all other children. Jitter values are higher for Very Engaged ASD children than the Somewhat Engaged group. The pattern is flipped for neurotypical, ADHD, and ASD+ADHD children. Jitter indicates pitch fluctuations in speech and higher values are typically associated with more tremulous or nervous speech [45]. This variation could indicate that individuals with ASD tremble more when they are highly engaged with content than their peers.

For both of these values, the only contrast effect that was statistically significant was for the Not Engaged group with ADHD when compared to the Very Engaged-Neurotypical group. However, given the extremely small sample size of

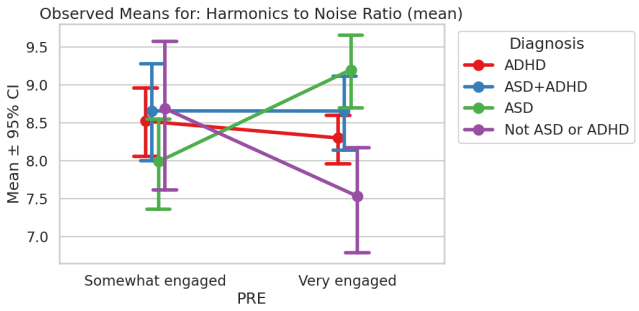


Figure 3: Diagnosis-group level means of the Harmonics-to-noise ratio across different clinical diagnoses and parent reported levels of engagement.

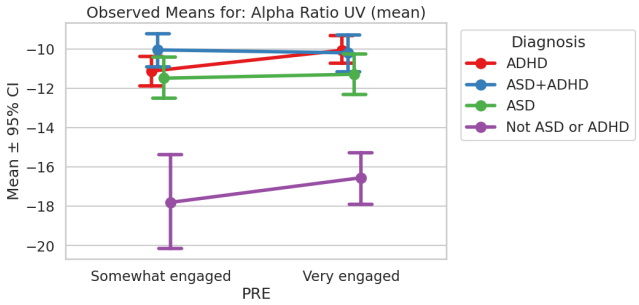


Figure 4: Diagnostic-group level means of the mean Alpha Ratio UV for each of the clinical diagnoses and parent reported levels of engagement.

the Not Engaged group, this result should be verified with a larger dataset.

4.5.3 *Spectral*

Interaction effects are most prevalent for the spectral or vocal shape related features. There are statistically significant effects for the unvoiced Alpha Ratio, voiced and unvoiced Hammarberg Index, MFCC values, spectral Flux and spectral slope values. As seen in Table 9, there are not any statistically significant contrast effects for any of the clinical diagnosis in the Very Engaged group. This means that across the clinical diagnosis, when children were Very Engaged, the spectral shape-related acoustic features appear in a uniform manner in the dataset. On the other hand, there are meaningful contrasts between different clinical diagnosis and the Not Engaged and Somewhat Engaged groups.

In our limited set of Not Engaged participants, we see a number of contrasting acoustic features from the Not ASD or ADHD:Very Engaged reference group. As demonstrated in Figure 4, the ASD+ADHD group had higher values for unvoiced Alpha Ratios than the reference, whereas the ASD group had substantially lower ones. A low Alpha ratio is usually associated with a softer voice. The coefficients for the unoviced Hammarberg Index mirror these results; high Hammarberg indices are associated with a softer voice, whereas lower values would be indicative of a sharper one [62]. This could be indicative of a trend where individuals with ADHD speak in a much softer tone when disengaged, as opposed to disengaged individuals with ASD+ADHD who are talking in sharper tones. Generally speaking, the mean values for the Alpha Ratio of the Not ASD or ADHD groups are considerably lower than they are for all other groups in the study.

We do see significant contrast effects in the Somewhat Engaged groups for some of the MFCC values. This includes lower values for MFCC1V for ADHD children and higher values for MFCC4 for Autistic Students. MFCC1V values are usually associated with variations in spectral tilt, with higher values indicating larger amounts of variation. Instead, for the ADHD children this is saying that there is a limited amount of variation in the spectral tilt when they are Somewhat Engaged.

4.5.4 *Temporal*

All of the temporal features had a statistically significant interaction effect, but only three of them had one without the presence of an Engagement effect. This includes the standard deviation of the voiced segment length as well as the mean and standard deviations of the unvoiced segment length. Lower standard deviations of voiced segment length mean that the speaker is talking with a more regular cadence, usually associated with a confident manner of speaking, whereas higher values can be indicative of hesitant speech [29]. In figure 1, we can see that the neurotypical group shows a much larger decrease in this value between the Very Engaged and Somewhat Engaged participants when compared to the neurodivergent groups. This suggests that neurotypical children will sound more confident when they have higher engagement levels, but this perceived confidence shift may not be reflected in the speech of their neurodivergent peers.

5. DISCUSSION

5.1 Implications for Automatic Detection of Reading Engagement

In highlighting the effects of parent-reported engagement, diagnosis, and their interaction on acoustic features, our ultimate goal is to better inform future models of reading engagement for diverse learner groups. Our results highlight the fact that acoustic features may be a more equitable signal than verbal response count for measuring reading engagement in the home setting. We recommend the use of acoustic features that demonstrate robust effects related to engagement propensity while taking into account interactions between clinical diagnosis and engagement levels. This includes Hammarberg index values, Alpha ratios, shimmer, as well as loudness peaks per second and mean values of voiced segment length. However, all of these values were associated with a statistically significant interaction effect between the diagnosis and PRE. Special consideration should be given when using acoustic features to predict engagement levels for children with ASD. In our data, we notice high variability for a number of acoustic features within this group. This may make it especially difficult to accurately predict engagement levels since there may not be a consistent observable speech pattern.

These findings are intended to be used to create adaptive at-home literacy learning systems. The differences presented in acoustic measures for individuals with ASD or ADHD

are not indicative of a reading comprehension deficit. Nor should they be used as a diagnostic tool. Instead, these should be interpreted as evidence that we need to develop more complex acoustic-feature based predictive models that can account for variations in the presentation of speech-related engagement signals. With effective real-time engagement predictions, it is possible to implement interventions that keep students engaged with reading materials at home, and in turns lead to more opportunities for improving their reading comprehension skills.

5.2 Limitations and Future Directions

The detectors presented here are comparable in performance to a range of others in previous research, but would still benefit from further refinement, particularly from efforts that could expand our data collection efforts. It is important to note that this a sensitivity study on the collected audio data, and we are not evaluating the impact of a particular reading intervention on features related to reading engagement. This is a logical next step in future research studies - leveraging these features as a non-invasive measure of the effect of particular reading interventions on student reading engagement in different subgroups.

Like many studies, this analysis would have benefited from the use of a larger data set. This is particularly challenging because the majority of available speech datasets that might serve as points of comparison–especially those used to identify acoustic features related to engagement–are composed of adult speech samples from extemporaneous speech. In addition to raw frequency differences between adult and child acoustic data, children often exhibit different speech behaviors in extemporaneous speech, and our samples were further constrained by the activities students were asked to perform as part of a task related to learning to reading.

In addition to benefiting generally from a larger sample, we would also benefit from diversifying our sample. Importantly, our current dataset is representative of children with several possible clinical conditions, which allowed us to present findings related to neurodiversity that have been understudied in the EDM community. However, it would still benefit from a more robust sampling of ethnic, socioeconomic, and English language learner status as variation within those diagnoses, as those factors are known to affect students' language patterns.

Future work should also look for ways to improve the granularity of the engagement labels, as the PRE labels were derived from parental evaluations of students' experience with the learning system in its entirety. However, even the most engaged child will not be completely engaged during every lesson, or every turn of that lesson. Ideally, in future experiments we will complement this measure with finer-grained labels, allowing us to develop a more nuanced understanding how students who are highly engaged overall might vary over the course of several weeks while also developing a more nuanced understanding of the acoustic indicators of this engagement. Additionally, we acknowledge that our dataset contained a very small number of students in the not engaged group. Ideally, in a larger sample we would have a less imbalanced set of engagement levels. Such efforts would also substantially improve our predictive modeling of real-time engagement.

Finally, we aim to expand our approach to comprehension conversation data. During a single comprehension activity, the microphone is open for a fixed period of time where the student responds to - at most - two conversational turns. This substantially limits the amount of data that can be collected, as students may be cut off before they are able to fully participate with the task. There may be strategic reasons for the learning system to continue such practices, but it would be good to consider ways in which we might better understand how this constraint affects student engagement, particularly as we seek to use more fine-grained levels than those employed in this study.

Still, the course-grained PRE labels deployed here–in addition to serving as an important step towards better understanding home usage–might also help us to better understand when apparent disengagement at the micro-level may be incidental. For example, they likely indicate a students' overall propensity to re-engage with the learning system, which could be used to help filter more fine-grained, moment-to-moment patterns in order to better understand which students need greater support.

6. CONCLUSION

In this paper, we have explored the use of acoustic features to measure reading engagement during online learning sessions across a diverse learner population. Acoustic features provide a non-invasive manner of capturing critical data related to student behavior and affect while they are working with a digital literacy platform at home. We present the nuances of modeling reading engagement for children with ADHD, ASD, or both based on the unique characteristics of these conditions. Our study includes audio and transcription data collected from 144 children over the course of several weeks who engaged in passage reading and reading comprehension tasks while at home.

Specifically, we use parental reports of engagement (PRE) with the system and clinical diagnosis to create generalized estimating equations (GEEs) for engagement-related acoustic features. We compare the effect of clinical diagnosis, parent report of engagement, child age and sex on a set of 49 acoustic features across 98 students from our dataset. We confirmed the results of prior work demonstrating that certain acoustic features related to frequency, temporal, spectral, and energy characteristics of speech are consistently influenced by engagement levels [44]. However, we discovered that all of the features are also impacted by the interaction between clinical diagnosis and engagement level. This suggests a critical need to develop more robust measures for measuring reading engagement that can account for the unique characteristics associated with different neurodevelopmental conditions. In future work, we hope to collect more refined turn-level engagement annotations that can be used to develop acoustic feature-driven models of reading engagement based on these findings. Furthermore, by integrating these measures of reading engagement into digital literacy systems we can create personalized adaptive learning interventions to increase individual students' reading comprehension gains.

7. ACKNOWLEDGMENTS

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305C240040. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education. This work was made possible by our collaborators at Digital Promise, the Amira Learning team, the Children's Hospital of Philadelphia, and MDRC. The authors would also like to express our gratitude to the families who participated in our study and contributed to this line of research.

APPENDIX

A. PER-FEATURE GEE RESULTS

Table 7 includes all acoustic features organized by high level category as well as the results of fitting the GEE model for that feature.

REFERENCES

[1] Amira Tutor, 2026.

[2] P. Afflerbach and C. Harrison. What Is Engagement, How Is It Different From Motivation, and How Can I Promote It? *Journal of Adolescent & Adult Literacy*, 61(2):217–220, 2017. _eprint: https://ila.onlinelibrary.wiley.com/doi/pdf/10.1002/jaal.679.

[3] F. S. Alamri, A. Rashad Mirdad, T. Saba, M. Amjad Raza, and U. Siddique. Investigating the Impact of Combined Spectral and Prosodic Features on Speech Emotion Recognition. *IEEE Access*, 14:5719–5732, 2026.

[4] A. P. Association. *Diagnostic and statistical manual of mental disorders*. American Psychiatric Publishing, Inc., Washington, DC, 5th ed., text rev. edition, 2022.

[5] R. Baker, J. Ocumpaugh, and A. Andres. BROMP Quantitative Field Observations: A Review. Dec. 2018.

[6] A. T. Barber and S. L. Klauda. How Reading Motivation and Engagement Enable Reading Achievement: Policy Implications. *Policy Insights from the Behavioral and Brain Sciences*, 7(1):27–34, 2020. _eprint: https://doi.org/10.1177/2372732219893385.

[7] J. E. Beck. Engagement tracing: using response times to model student disengagement. In *Proceedings of the 2005 conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, pages 88–95, NLD, May 2005. IOS Press.

[8] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B (Methodological)*, 57(1):289–300, 1995.

[9] Y. S. Bonneh, Y. Levanon, O. Dean-Pardo, L. Lossos, and Y. Adini. Abnormal speech spectrum and increased pitch variability in young autistic children. *Frontiers in Human Neuroscience*, 4:237, 2011.

[10] N. Bosch, S. D'Mello, R. Baker, J. Ocumpaugh, V. Shute, M. Ventura, L. Wang, and W. Zhao. Automatic Detection of Learning-Centered Affective States in the Wild. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 379–388, Atlanta Georgia USA, Mar. 2015. ACM.

[11] K. Brimo, L. Dinkler, C. Gillberg, P. Lichtenstein, S. Lundström, and J. Åsberg Johnels. The co-occurrence of neurodevelopmental problems in dyslexia. *Dyslexia (Chichester, England)*, 27(3):277–293, Aug. 2021.

[12] M. Bustos-López, N. Cruz-Ramírez, A. Guerra-Hernández, L. N. Sánchez-Morales, N. A. Cruz-Ramos, and G. Alor-Hernández. Wearables for Engagement Detection in Learning Environments: A Review. *Biosensors*, 12(7):509, July 2022.

[13] R. A. Calvo and S. D'Mello. Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications. *IEEE Transactions on Affective Computing*, 1(1):18–37, Jan. 2010.

[14] CDC. Data and Statistics on Autism Spectrum Disorder, May 2025.

[15] M. L. Danielson, A. H. Claussen, R. H. Bitsko, S. M. Katz, K. Newsome, S. J. Blumberg, M. D. Kogan, and R. Ghandour. ADHD Prevalence Among U.S. Children and Adolescents in 2022: Diagnosis, Severity, Co-Occurring Disorders, and Treatment. *Journal of Clinical Child and Adolescent Psychology: The Official Journal for the Society of Clinical Child and Adolescent Psychology, American Psychological Association, Division 53*, 53(3):343–360, 2024.

[16] M. M. Davidson and S. Ellis Weismer. Characterization and prediction of early reading abilities in children on the autism spectrum. *Journal of Autism and Developmental Disorders*, 44(4):828–845, Apr. 2014.

[17] D. Demszky, R. Wang, S. Geraghty, and C. Yu. Does Feedback on Talk Time Increase Student Engagement? Evidence from a Randomized Controlled Trial on a Math Tutoring Platform. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 632–644, Kyoto Japan, Mar. 2024. ACM.

[18] G. Dietz, Z. Pease, B. McNally, and E. Foss. Giggle gauge: a self-report instrument for evaluating children's engagement with technology. In *Proceedings of the Interaction Design and Children Conference*, IDC '20, pages 614–623, New York, NY, USA, June 2020. Association for Computing Machinery.

[19] C. Doğdu, T. Kessler, D. Schneider, M. Shadaydeh, and S. R. Schweinberger. A Comparison of Machine Learning Algorithms and Feature Sets for Automatic Vocal Emotion Recognition in Speech. *Sensors (Basel,*

Table 7: Feature-by-significance table for all acoustic features. Entries are FDR-adjusted p-values. Asterisk (*) indicates $p_{\text{FDR}} < 0.05$.

Feature	Diagnosis	Engagement	Diagnosis × Engagement
Temporal features			
loudnessPeaksPerSec	0.3927	0.0000*	0.0000*
Voiced segment length (mean)	0.9649	0.0094*	0.0000*
Voiced segment length (std. dev.)	0.9758	0.4877	0.0000*
Unvoiced segment length (mean)	0.3782	0.3930	0.0000*
Unvoiced segment length (std. dev.)	0.2547	0.8751	0.0001*
Energy features			
Harmonic-to-noise ratio (mean)	0.4363	0.0000*	0.0000*
Harmonic-to-noise ratio (std. dev.)	0.3927	0.0276*	0.0142*
Shimmer (std. dev.)	0.2547	0.0000*	0.0000*
Frequency features			
F0 mean	0.2547	0.0000*	0.0000*
F0 standard deviation	0.2547	0.0367*	0.0000*
F0 20th percentile	0.2547	0.0000*	0.0000*
F0 50th percentile	0.2547	0.0000*	0.0000*
F0 80th percentile	0.2547	0.0000*	0.0000*
F0 mean rising slope	0.2547	0.0000*	0.0000*
F0 mean falling slope	0.2547	0.0000*	0.0000*
Jitter (mean)	0.2547	0.0038*	0.0000*
Jitter (std. dev.)	0.2547	0.1076	0.0000*
Spectral features			
Alpha ratio, voiced (mean)	0.2547	0.0000*	0.0000*
Alpha ratio, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Alpha ratio, unvoiced (mean)	0.2547	0.4877	0.0000*
Hammarberg index, voiced (mean)	0.2547	0.0000*	0.0000*
Hammarberg index, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Hammarberg index, unvoiced (mean)	0.0053*	0.0000*	0.0000*
Spectral flux, voiced (mean)	0.2547	0.0000*	0.0000*
Spectral flux, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Spectral flux, unvoiced (mean)	0.2547	0.4877	0.0000*
Spectral slope 0–500 Hz (mean)	0.2547	0.0000*	0.0000*
Spectral slope 0–500 Hz (std. dev.)	0.2547	0.0000*	0.0000*
Spectral slope 500–1500 Hz (mean)	0.2547	0.0000*	0.0000*
Spectral slope 500–1500 Hz (std. dev.)	0.2547	0.0000*	0.0000*

Switzerland), 22(19):7561, Oct. 2022.

[20] S. D'Mello and A. Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, Apr. 2012.

[21] S. D'Mello, B. Lehman, R. Pekrun, and A. Graesser. Confusion can be beneficial for learning. *Learning and Instruction*, 29:153–170, Feb. 2014.

[22] F. El Zein, M. Solis, R. Lang, and M. K. Kim. Embedding perseverative interest of a child with autism in text may result in improved reading comprehension: A pilot study. *Developmental Neurorehabilitation*, 19(3):141–145, June 2016.

[23] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong. The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing. *IEEE Transactions on Affective Computing*, 7(2):190–202, Apr. 2016.

[24] F. Eyben, F. Weninger, and B. Schuller. Affect recognition in real-life acoustic conditions — a new perspective on feature selection. In *Interspeech 2013*, pages 2044–2048. ISCA, Aug. 2013.

[25] F. Eyben, M. Wöllmer, and B. Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, MM '10, pages 1459–1462, New York, NY, USA, Oct. 2010. Association for Computing Machinery.

[26] J. A. Fredricks, P. C. Blumenfeld, and A. H. Paris. School Engagement: Potential of the Concept, State of the Evidence. *Review of Educational Research*, 74(1):59–109, 2004.

[27] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, and M. P. Wenderoth. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415, June 2014.

[28] K. A. Fuller, N. S. Karunaratne, S. Naidu, B. Exintaris, J. L. Short, M. D. Wolcott, S. Singleton, and P. J. White. Development of a self-report instrument for measuring in-class student engagement reveals that pretending to engage is a significant unrecognized problem. *PLOS ONE*, 13(10):e0205828, Oct. 2018.

[29] I. Gliser, C. Mills, N. Bosch, S. Smith, D. Smilek, and J. D. Wammes. The Sound of Inattention: Predicting Mind Wandering with Automatically Derived Features of Instructor Speech. *Artificial Intelligence in Education*, 12163:204–215, June 2020.

[30] Guher Gorgun, Seyma Nur Yildirim-Erbasli, and C. D. Epp. Predicting Cognitive Engagement in Online Course Discussion Forums. July 2022.

[31] J. T. Guthrie, A. Wigfield, et al. Engagement and motivation in reading. *New York*, 2000.

[32] J. T. Guthrie, A. Wigfield, and W. You. Instructional Contexts for Engagement and Achievement in Reading. In S. L. Christenson, A. L. Reschly, and C. Wylie, editors, *Handbook of Research on Student Engagement*, pages 601–634. Springer US, Boston, MA, 2012.

[33] K. He, A. Gastón-Panthaki, D. Zhuo, J. Munsey, M. Zhang, and M. Warschauer. StoryPal: Supporting Young Children's Dialogic Reading with Large Language Models. In *Proceedings of the 24th Interaction Design and Children*, pages 494–511, Reykjavik Iceland, June 2025. ACM.

[34] A. James, Y. Chua, T. Maszczyk, A. Moreno-Núñez, R. Bull, K. Lee, and J. Dauwels. Automated Classification of Classroom Climate by Audio Analysis. pages 41–49. Sept. 2019.

[35] B. W. Jeff Johns. A Dynamic Mixture Model to Detect Student Motivation and Proficiency.

[36] S. Kai, L. Paquette, R. S. Baker, N. Bosch, S. D'Mello, V. Shute, and M. Ventura. A Comparison of Video-based and Interaction-based Affect Detectors in Physics Playground.

[37] M. Kakoulidou, F. Le Cornu Knight, R. Filippi, and J. Hurry. The Effects of Choice on the Reading Comprehension and Enjoyment of Children with Severe Inattention and no Attentional Difficulties. *Research on Child and Adolescent Psychopathology*, 49(11):1403–1417, Nov. 2021.

[38] S. Karumbaiah, J. Ocumpaugh, and R. Baker. The Influence of School Demographics on the Relationship Between Students' Help-Seeking Behavior and Performance and Motivational Measures. June 2019.

[39] K. Khowaja and S. S. Salim. A systematic review of strategies and computer-based intervention (CBI) for reading comprehension of children with autism. *Research in Autism Spectrum Disorders*, 7(9):1111–1121, Sept. 2013.

[40] S. Y. Kim, M. Rispoli, R. A. Mason, C. Lory, E. Gregori, C. A. Roberts, D. Whitford, and D. Wang. The Effects of Using Adapted Science eBooks Within Shared Reading on Comprehension and Task Engagement of Students with Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders*, 55(12):4185–4196, Dec. 2025.

[41] Y. Lee, B. G. Jang, and K. C. Smith. A Systematic Review of Reading Engagement Research: What Do We Mean, What Do We Know, and Where Do We Need to Go? *Reading Psychology*, 42(5):540–576, 2021. _eprint: https://doi.org/10.1080/02702711.2021.1888359.

[42] S. Li, S. P. Lajoie, J. Zheng, H. Wu, and H. Cheng. Automated detection of cognitive engagement to inform the art of staying engaged in problem-solving. *Computers & Education*, 163, 2021.

[43] K.-Y. LIANG and S. L. ZEGER. Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22, Apr. 1986.

[44] D. Litman, H. Friedberg, and K. Forbes-Riley. Prosodic cues to disengagement and uncertainty in physics tutorial dialogues. In *Interspeech 2012*, pages 755–758. ISCA, Sept. 2012.

[45] J. K. MacCallum, Y. Zhang, and J. J. Jiang. Acoustic analysis of the tremulous voice: assessing the utility of the correlation dimension and perturbation parameters. *Journal of communication disorders*, 43(1):35, 2010.

[46] N. S. McIntyre, E. J. Solari, J. E. Gonzales, M. Solomon, L. E. Lerro, S. Novotny, T. M. Oswald,

and P. C. Mundy. The Scope and Nature of Reading Comprehension Impairments in School-Aged Children with Higher-Functioning Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders*, 47(9):2838–2860, Sept. 2017.

[47] A. C. Miller, J. M. Keenan, R. S. Betjemann, E. G. Willcutt, B. F. Pennington, and R. K. Olson. Reading comprehension in children with ADHD: cognitive underpinnings of the centrality deficit. *Journal of Abnormal Child Psychology*, 41(3):473–483, Apr. 2013.

[48] A. Mohanta and V. Kumar Mittal. Autism Speech Analysis using Acoustic Features. In D. M. Sharma and P. Bhattacharya, editors, *Proceedings of the 16th International Conference on Natural Language Processing*, pages 85–94, International Institute of Information Technology, Hyderabad, India, Dec. 2019. NLP Association of India.

[49] K. P. Nowell, K. E. Bodner, M. D. Mohrland, and S. M. Kanne. Neurodevelopmental disorders. In *Handbook of rehabilitation psychology, 3rd ed*, pages 357–370. American Psychological Association, Washington, DC, US, 2019.

[50] J. Ocumpaugh, R. S. Baker, and M. M. T. Rodrigo. Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) 2.0 Technical and Training Manual.

[51] J. Ocumpaugh, N. Nasiar, A. F. Zambrano, A. Goslen, J. Vandenberg, J. Esiason, J. Rowe, and S. Hutt. Refocusing the lens through which we view affect dynamics: The Skills, Difficulty, Value, Efficacy and Time Model. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 192–203, Dublin Ireland, Mar. 2025. ACM.

[52] L. Paquette, Z. Li, R. Baker, J. Ocumpaugh, and A. Andres. Who's learning? Using demographics in EDM research.

[53] J. Parish-Morris, M. Liberman, N. Ryant, C. Cieri, L. Bateman, E. Ferguson, and R. Schultz. Exploring Autism Spectrum Disorders Using HLT. In K. Hollingshead and L. Ungar, editors, *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*, pages 74–84, San Diego, CA, USA, June 2016. Association for Computational Linguistics.

[54] R. Paul, L. D. Shriberg, J. McSweeny, D. Cicchetti, A. Klin, and F. Volkmar. Brief report: relations between prosodic performance and communication and socialization ratings in high functioning speakers with autism spectrum disorders. *Journal of Autism and Developmental Disorders*, 35(6):861–869, Dec. 2005.

[55] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[56] A. Ramakrishnan, E. Ottmar, J. LoCasale-Crouch, and J. Whitehill. Toward Automated Classroom Observation: Predicting Positive and Negative Climate. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8, May 2019.

[57] J. Ricketts, C. R. G. Jones, F. Happé, and T. Charman. Reading comprehension in autism spectrum disorders: the role of oral language and social functioning. *Journal of Autism and Developmental Disorders*, 43(4):807–816, Apr. 2013.

[58] Y. Rong, C.-J. Yang, Y. Jin, and Y. Wang. Prevalence of attention-deficit/hyperactivity disorder in individuals with autism spectrum disorder: A meta-analysis. *Research in Autism Spectrum Disorders*, 83:101759, May 2021.

[59] A. Rotnitzky and N. P. Jewell. Hypothesis Testing of Regression Parameters in Semiparametric Generalized Linear Models for Cluster Correlated Data. *Biometrika*, 77(3):485–497, 1990.

[60] S. Seabold and J. Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[61] L. Tamarit, M. Goudbeek, and K. Scherer. Spectral Slope Measurements in Emotionally Expressive Speech. 2008.

[62] B. Tracey, D. Volfson, J. Glass, R. Haulcy, M. Kostrzebski, J. Adams, T. Kangarloo, A. Brodtmann, E. R. Dorsey, and A. Vogel. Towards interpretable speech biomarkers: exploring MFCCs. *Scientific Reports*, 13:22787, Dec. 2023.

[63] N. J. Unrau and M. Quirk. Reading Motivation and Reading Engagement: Clarifying Commingled Conceptions. *Reading Psychology*, 35(3):260–284, 2014. _eprint: https://doi.org/10.1080/02702711.2012.684426.

[64] C. Vukelich. The development of listening comprehension through storytime. *Language Arts*, 53(8):889–891, 1976.

[65] M. Watanabe, M. Imundo, K. Christhilf, T. Arner, and D. S. Mcnamara. Building Reading Comprehension and Knowledge with iSTART: An ITS to Provide Formative Feedback in Reading Instruction at Scale. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, pages 510–513, Atlanta GA USA, July 2024. ACM.

[66] M. F. Westerveld and J. M. A. Roberts. The Oral Narrative Comprehension and Production Abilities of Verbal Preschoolers on the Autism Spectrum. *Language, Speech, and Hearing Services in Schools*, 48(4):260–272, Oct. 2017.

[67] A. Y. Wong, R. L. Bryck, R. S. Baker, S. Hutt, and C. Mills. Using a Webcam Based Eye-tracker to Understand Students' Thought Patterns and Reading Behaviors in Neurodivergent Classrooms. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, LAK2023, pages 453–463, New York, NY, USA, Mar. 2023. Association for Computing Machinery.

[68] Y. Xu, D. Wang, P. Collins, H. Lee, and M. Warschauer. Same benefits, different communication patterns: Comparing Children's reading with a conversational agent vs. a human partner. *Computers & Education*, 161:104059, Feb. 2021.

[69] Y. Xu and M. Warschauer. A content analysis of voice-based apps on the market for early literacy development. In *Proceedings of the Interaction Design and Children Conference*, pages 361–371, London United Kingdom, June 2020. ACM.

[70] Z. Xu, K. K. Wijekumar, G. Ramirez, X. Hu, and R. Irey. The effectiveness of intelligent tutoring systems on K-12 students' reading comprehension: A meta-analysis. *British Journal of Educational Technology*, 50(6):3119–3137, 2019. _eprint: https://bera-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/bjet.12758.

[71] V. P. Yache, L. Moradbakhti, I. Neuner, and T. Veselinovic. Predicting affective engagement and mental strain from prosodic speech features. *Frontiers in Psychiatry*, 16:1656292, Sept. 2025.

[72] D. Yekutieli and Y. Benjamini. Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics. *Journal of Statistical Planning and Inference*, 82(1):171–196, Dec. 1999.

[73] B. Zablotsky, M. D. Bramlett, and S. J. Blumberg. The Co-Occurrence of Autism Spectrum Disorder in Children With ADHD. *Journal of Attention Disorders*, 24(1):94–103, Jan. 2020.

[74] A. F. Zambrano, N. Nasiar, J. Ocumpaugh, A. Goslen, J. Zhang, J. Rowe, J. Esiason, J. Vandenberg, and S. Hutt. Says Who? How different ground truth measures of emotion impact student affective modeling. pages 211–223, 2024.

[75] S. L. Zeger and K.-Y. Liang. Longitudinal Data Analysis for Discrete and Continuous Outcomes. *Biometrics*, 42(1):121–130, 1986.

[76] Z. Zeng, S. Chaturvedi, and S. Bhat. Learner affect through the looking glass: Characterization and detection of confusion in online courses. 2017.

[77] J. Zhang, K. Wang, and Y. Zhang. Physiological Characterization of Student Engagement in the Naturalistic Classroom: A Mixed-Methods Approach. *Mind, Brain, and Education*, 15(4):322–343, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/mbe.12300.

Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression

Megan Caruso
University of Colorado Boulder
Boulder, CO, USA
Megan.Caruso@colorado.edu

Robert Moulder
University of Colorado Boulder
Boulder, CO, USA
Robert.Moulder@colorado.edu

Rosy Southwell
University of Colorado Boulder
Boulder, CO, USA
Rosy.Southwell@colorado.edu

Sidney D'Mello
University of Colorado Boulder
Boulder, CO, USA
Sidney.Dmello@colorado.edu

ABSTRACT

We examined dynamic relationships between eye gaze, brain activity, and text characteristics during reading comprehension, a ubiquitous educational activity. We applied multilevel vector autoregression (mlVAR) to naturalistic reading data to model across-fixation (temporal) and within-fixation (contemporaneous) relationships among eye movements, EEG, and lexical features of the word being attended. We examined different approaches to encoding the multimodal time series data and how prior knowledge, reading time, mind wandering, and comprehension outcomes were related to network dynamics. Our data were from 116 participants who engaged in a complex learning task involving five long, connected texts on scientific research methods. Across model configurations, lexical features consistently predicted fixation duration, highlighting text-driven control of reading behavior, whereas EEG-gaze interactions emerged under certain model configurations. Measures of cognitive states, mainly comprehension and reading time, predicted within-fixation network connectivity. These findings demonstrate that temporally sensitive multimodal models provide a useful framework for characterizing moment-to-moment reading dynamics.

Keywords

Multimodal modeling, eye tracking, EEG, reading comprehension, temporal modeling

1. INTRODUCTION

Reading comprehension remains a key skill in education and everyday life: whether learning from a textbook in class or reading an article at home for leisure. Decades of research have shown that comprehension emerges from dynamic interactions between text characteristics, individual differences in the reader, and the reading context [72, 42, 43]. Eye-tracking has been a key modality used for understanding encoding and processing of words, sentences, and text [27, 27, 56, 12], and even for predicting comprehension outcomes and attentional states during reading [12, 24, 5, 30]. This is due to its tight link to moment-by-moment cognitive processing, a phenomenon deemed the *Eye-Mind Link*. For example, fixations (when gaze is maintained in a single spot) and saccades (quick movements between fixations) are sensitive to textual features such as word frequency and complexity [59, 55, 61]. Consequently, a weakening of this gaze-text link can be used to predict internal mental states associated with disrupted comprehension such as mind wandering [5, 30, 62]. The eye-mind link has been applied in real-time interventions to prompt readers to re-read text if they mind wander [49], or adapt texts when comprehension difficulty is detected [67]. Yet, relying on a single modality (gaze) limits these models' ability to explain and predict the complex, temporally unfolding cognitive processes that occur during reading comprehension.

Neural measures offer a more direct window into the brain (arguably the locus of the "mind") and have been used to investigate the cognitive processes that support reading comprehension. Electroencephalography (EEG) and functional magnetic resonance imaging (fMRI) provide rich information about encoding, semantic processing, and integration during reading and language processing [20, 6, 31]. Importantly, EEG measures the brain's electrical response at the millisecond scale enabling capture of fast moving brain dynamics. Here, EEG studies have focused on capturing neural dynamics on a single word using Rapid Serial Visual Presentation (RSVP) paradigm, where words are sequentially presented one at a time. Given the lack of ecological validity of this approach, some studies have used fixation-related potentials (FRPs), where EEG activity is time-locked to fixations during more naturalistic reading [19].

However, FRP studies also generally rely on highly-controlled text stimuli, including artificially constructed sentences, often restricting analysis to a subset of experimentally manipulated target words that should elicit predicted lexical

Megan Caruso, Robert Moulder, Rosy Southwell, and Sidney D'Mello. Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 104–116. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040010

effects [20, 66, 34]. This includes removing certain short function words from analysis even though they make up a decent proportion of viewed words [10, 34], but have limited lexical information (very high frequency and low surprisal). Within these contexts, characteristic components of the FRP have been associated with linguistic variables such as surprisal (lack of predictability) and word frequency [79, 66]. However, such designs prioritize experimental control at the expense of ecological validity. They obscure how neural activity dynamically couples with eye movements to support the broader cognitive aspects of reading, such as how comprehension is constructed while reading a long connected text, and how it is affected by lapses in attention. FRP studies also tend to quantify fixations in isolation, largely ignoring the temporal dynamics across multiple fixations.

The present study focuses on modeling the temporal dynamic coupling between eye movements, neural activity (EEG), and text characteristics during natural reading of long, connected texts. We extract multivariate timeseries consisting of fixation-related EEG responses in parallel with fixation durations and linguistic features of fixated words. To capture how these features jointly evolve over time, we use multi-level vector autoregression (mlVAR) to model temporal (across fixation) and contemporaneous (within-fixation) dependencies of these timeseries across readers. This is one of the first attempts (to our knowledge) to simultaneously investigate temporal dependencies between eye, brain, and the text during relatively naturalistic reading settings.

2. BACKGROUND AND RELATED WORK

2.1 Reading Comprehension

Reading comprehension involves interacting levels of processing that unfold over time: perceptual intake and encoding of a word [36], accessing a word's literal meaning [52], constructing a representation of the text and updating it with prior knowledge from memory [40], eventually leading to a rich mental representation of what was read [42, 43]. Reading is thus a highly interactive process where readers must continually retain relevant information, reactivate earlier representations, and suppress information that becomes temporally irrelevant [80], all while continuing perceptual uptake and encoding of words as they proceed through the text.

A widely-used framework for reading comprehension distinguishes among three levels of mental representation: the surface code, textbase, and situation model [76]. The surface code reflects a literal encoding of the words and syntax of the text. The textbase captures the meaning of sentences independent of their specific wording. Finally, the situation model integrates meaning across the entire text, combining what was read with the reader's prior knowledge [35]. Cognitive theories such as Kintsch's Construction-Integration model [41, 43] explain how higher-level inferences emerge from these initial, more literal meanings of the text. This means disruptions in constructing the surface code or textbase can have downstream effects on developing the situation model [68]. For instance, if a reader mind wanders while reading a sentence containing key information, this hinders their ability to later form inferences pertaining to that information [69]. Taken together, this highlights reading comprehension as a temporally evolving process in which moment-to-moment encoding accumulates to support the construction of a coherent mental model.

2.2 Eye Movements during Reading

Linguistic processing must be tightly coordinated with the visual system to coordinate perceptual uptake [39]. As such, eye movements index and track the cognitive processes underlying reading comprehension as they unfold. Decades of research on gaze control during reading support a strong coupling between eye movements and textual features [55, 58, 27, 61]. For example, fixations occur when information is encoded, and are shorter on more frequent words [59, 58] and longer on less predictable or surprising words [55]. This relationship can even be temporally complex, as illustrated by the "spillover" effect when fixations are longer following difficult words [55, 61]. Saccades are the quick movements between fixations, and regressions are saccades to previous sections of the text and occur approximately 10%-25% of the time [57]. More difficult texts elicit more and longer fixations, as well as more regressions [14] reflecting a breakdown in encoding or integration processes [63].

This coupling between eye movements and online mental processing is deemed the Eye-Mind Link [54, 60]. Beyond indexing lower-level lexical processing, eye movements have been leveraged to predict higher-level comprehension and integration processes, such as detecting knowledge incongruence in text via characteristic regressive patterns [77]. Skim reading, typically associated with decreased comprehension [57], is characterized by fewer and shorter fixations and fewer regressions [50]. In contrast, mind wandering ("zoning out"), an attentional state also robustly linked with impaired comprehension, has a less consistent gaze signature. Mind wandering been associated with fewer fixations and regressions (similar to skim reading) and longer fixations (similiar to reading difficult texts) in some studies [4, 30, 62], but an increase in fixation count in others [75]. Other studies have explored mind wandering in regards to coupling, finding a decreased sensitivity to linguistic and lexical changes in the text[2, 11].

Given these relationships, machine learning using eye movements have been able to predict shallow [24, 73, 15, 46] and deep comprehension of texts [12], as well as mind wandering [5, 30, 62] during learning contexts. However, to address the inherent ambiguity of information in individual fixations, these models aggregate gaze features across predefined windows or entire pages of text, losing fine-grained dynamic temporal structure, which is the present focus.

2.3 EEG Correlates of Linguistic Processing

Investigating neural activity offers the potential to go beyond coarser grained eye-gaze analyses towards differentiating processing within and across fixations, enabling more precise characterization of reading dynamics. Controlled experiments using EEG have identified neural correlates of language comprehension. Event-related potentials (ERPs) measured on the scalp reflect neurons activating in synchrony [45] resulting in consistent voltage fluctuations comprising positive and negative peaks unfolding over several hundred milliseconds [45]. Linguistic ERPs are traditionally measured in tightly controlled experiments where words are presented in isolation for a brief fixed interval using RSVP

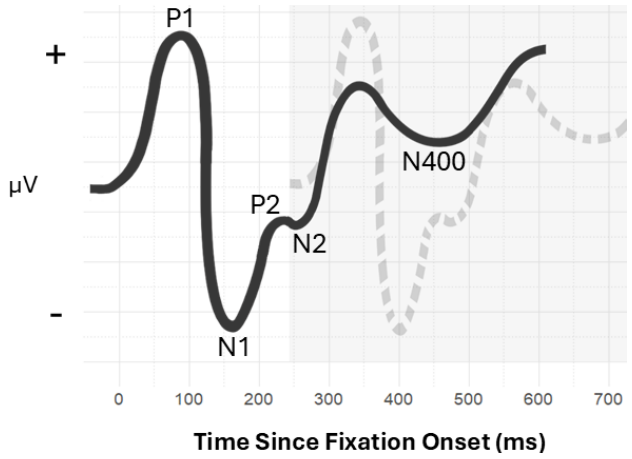


Figure 1: Characteristic FRP after the onset of a fixation with classic positive (P) and negative (N) wave components. P1 and P2 are positive peaks 100 and 200 ms after fixation onset, respectively. N1, N2, and N400 are negative wave components typically occurring in the range of 100, 200, and 400 ms after fixation onset. Grey box indicates the onset of another fixation, showing how FRPs can overlap across fixations.

[23, 20]. Whereas this approach allows for the precise delineation of the neural response to single words [21], it does not reflect naturalistic reading settings.

A fixation-related potential (FRP), on the other hand, is an ERP that is time-locked to the onset of a fixation during natural reading and allows for a richer measure of perceptual and cognitive processing that occurs during processing of a word in context [19, 20]. FRPs require more in-depth processing to remove overlapping neural responses to successive words [16, 71] and have been used during natural reading to investigate the relationship between EEG and eye movement measures [33, 23].

Early ERP components during reading are primarily associated with visual processing and encoding, and are sensitive to low-level lexical features. For instance the P1 and N1 components which occur 80-120 ms post-fixation show smaller responses for more frequent words [66]. The P2 (150-300 ms post-fixation) has been linked to early orthographic and phonological processing as well as lexical access, with amplitudes modulated by word frequency [20]. Later components, such as the well-studied N400 (300-500 ms post-fixation), reflect higher-level semantic processing and integration [79] and are sensitive to semantic predictability and contextual fit [44]. These components can overlap across successive fixations, demonstrating that visual, lexical, and semantic processes are dynamically coordinated as reading and comprehension unfold [19, 20].

The ability of FRPs to capture neural processing within a single fixation is particularly valuable in distinguishing complex cognitive states even when fixation durations are similar. In particular, longer fixation durations can arise for different reasons– either because the reader is disengaged from the text [62], or because they are allocating resources towards processing a difficult word [55, 26]. Eye movements alone cannot distinguish these cases. In contrast, EEG provides a direct window into the neural processes unfolding during a fixation, allowing covert states to be differentiated at a finer temporal scale. For example, Broadway et al. [9] showed that mind wandering during reading is associated with attenuated early visual ERP components such as P1, consistent with perceptual decoupling from external input [64]. The strength of these early neural responses predicted subsequent reading comprehension, linking momentary engagement to later comprehension outcomes. Boudewyn et al. further demonstrated that during story listening, mind wandering reduced the sensitivity of the N400 to contextual linguistic variables such as word position and surprisal [7]. More recently, Southwell et al. [74], replicated this finding involving natural reading. Using multivariate linear modeling applied to concurrent EEG and eye movement recordings, they found that mind wandering led to a decreased modulation of the N400 by surprisal, such that the amplitude of the N400 was larger (more negative voltage) during mind wandering. This further suggests a greater sensitivity to word meaning when readers are attending to the text. Together, these findings highlight the value of incorporating EEG into temporal models of reading, as neural activity captures fixation-by-fixation fluctuations in attentional and linguistic processing that are not observable from eye movements alone.

3. NOVELTY AND CONTRIBUTIONS

Reading is a temporally evolving process in which eye movements, neural activity, and textual properties continuously interact. However, most prior multimodal reading research involving the eye, brain, and text has focused on individual within-fixation associations examined under controlled experimental conditions rather than the fixation time series as a whole.

Accordingly, the goal of the current study is to provide an exploratory analysis of temporal reading dynamics across modalities and to examine how these dynamics are related to cognitive processes underlying reading. To do so, we implemented multilevel vector autoregression [29, 28], a relatively new modeling approach that enables joint estimation of across-fixation (temporal) and within-fixation (contemporaneous) relationships among eye movements, EEG activity, and lexical properties. mlVAR explicitly separates lagged dependencies across successive fixations from instantaneous coupling within a fixation, allowing these two sources of structure to be estimated simultaneously.

At the fixation level, mlVAR is implemented as a system of linear mixed-effects models in which features from the previous fixation are used to predict features at the current fixation, allowing for group-level network patterns to be estimated while capturing participant-specific deviations. By modeling multiple modalities in a unified temporal framework, mlVAR provides an exploratory lens on how gaze behavior, neural activity, and lexical properties co-vary over time. This approach enables our first research question **(RQ1): What are the within- and across-fixation relationships between EEG, gaze behavior, and text during naturalistic reading?**

The second aim of the study is to explore how cognitive processes modulate these dynamic relationships. Specifically, we ask **(RQ2): How do mind wandering, comprehension, prior knowledge, and reading time relate to variation in reading dynamics?** These factors are treated as sources of variability in the strength of dependencies among eye movements, neural activity, and text. While some of these cognitive variables have been explored at the level of single fixations, they have not been jointly modeled across modalities or across successive fixations.

Finally, because most FRP approaches do not model fixation sequences over time, they do not require preserving temporal continuity. Accordingly, it is common for these approaches to exclude stop words (short words that include articles, prepositions, conjunctions, and pronouns, e.g., "a", "the", "in", "or"), and restrict analyses to content words (e.g., nouns, words). However, stop words and their associated fixations are an integral part of the reading process and how readers move through, pause on, or transition across these words may carry meaningful information about cognitive state. For example, prolonged fixations on stop words may reflect mind wandering. We therefore introduce **(RQ3): How do different strategies for representing stop words into the fixation time series shape the temporal effects and cognitive influences observed in the model**.

To address this, we compare three modeling strategies (see Figure 4). Given that the definition of the $t-1$ fixation is the key modeling detail in mlVAR, we tried three modeling strategies that differed in how stop words were incorporated into the definition of the $t-1$ fixation. In Model 1 we used only transitions from content words to content words, making no assumptions about stop words and excluding transitions that were originally to or from stop words. This keeps the focus to content words, most clearly mirroring traditional FRP analysis. Model 2 assumes stop words are minimally processed and barely attended to, effectively skipping them by removing stop word fixations and modeling a sequence of fixations represented as between fixation content words only. In other other words, the $t-1$ fixation is defined as the previous content word attended to. Lastly, Model 3 retained stop words in the sequence by imputing them with the lowest surprisal and highest frequency values, allowing the fixation sequence to include both content words and stop words. These modeling strategies will unveil how important assumptions shape which aspects of coupling are emphasized. See Figure 2 for a visualization of each modeling strategy.

The novelty of this work lies in applying mlVAR to multimodal fixation-level, real-world reading to explore patterns that can inform future theory-driven and application-focused modeling efforts. To our knowledge, no study has investigated the temporal dynamics of eye gaze, EEG, and text characteristics during naturalistic reading of long expository educational texts. As a result, our findings can be useful in educational contexts.

4. DATASET

This study was part of a larger study and only aspects germane to the present research are discussed here.

4.1 Participants

A total of 116 participants (mean age = 22 +/- 5.7 (SD), 70% female, 1% identifying as another gender) were students from a large public university in the western United States. Participants were compensated $20 per hour via Amazon gift cards. All procedures were approved by the university's Institutional Review Board, and written informed consent was obtained from all participants prior to participation.

4.2 Data Acquisition

4.2.1 Eye Tracking

High-resolution, binocular gaze was tracked using the desktop-mounted SR Research EyeLink 1000+ with a sampling rate of 1000 Hz. Stimuli were displayed on a 23.8", 1920x1080 pixel display and viewed at a distance of 90 cm. Participants stabilized their head using a chin rest.

4.2.2 EEG

EEG was recorded at 1000 Hz using a LiveAmp wireless amplifier with 8 electrodes on an ActiCAP (Brain Products GmbH, Munich, Germany cite). A custom layout based on the 10-5 system was designed to optimize scalp coverage while also working around other sensors present on the cap. The montage included two standard mid-line locations (CPz and FCz), 3 pairs of symmetrically-located channels at frontal (AFF5h and AFF6h), centro-parietal (CCP5h and CCP6h), and parieto-occipital (PPO9h and PPO10h) locations. Reference was at Cz and Ground adjacent to FPz.

4.3 Reading Task & Measures

4.3.1 Texts and Reading Procedure

Participants read five expository texts on the computer monitor, totaling in 5306 words over 50 pages (mean 106 words per page). Each text consisted of ten pages (screens) of text and participants could advance through the pages at their own pace, but could not return to pages already read. Texts were presented in a random order, and each focused on a topic of behavioral research methods: Hypothesis, Bias, Causal Claims, Validity, and Variables. The texts had a mean Flesch-Kincaid grade level of 13.2, indicating a slightly advance reading level suitable for college students [32]. Participants spent an average time of 27.6 (8.3 SD) minutes reading all texts. On average, 57.3% $\pm$ 3.4% of words on a page were content words.

4.3.2 Reading Time

Page-level reading times were calculated from the beginning of the button press that initiates the page, to the next button press that leaves the page. Mean reading time per person was calculated by average page-level reading times across the entire text. Pages with short reading times (<10 seconds) were excluded, as they likely reflect insufficient engagement with the text rather than meaningful reading. Participants spent an average of 36 seconds (16 SD) per page.

4.3.3 Mind Wandering Probes

We used a probe-caught method previously validated to track mind wandering [65, 78]. With this method, participants indicate if they were mind wandering on the previous page via thought-probes presented throughout the study.

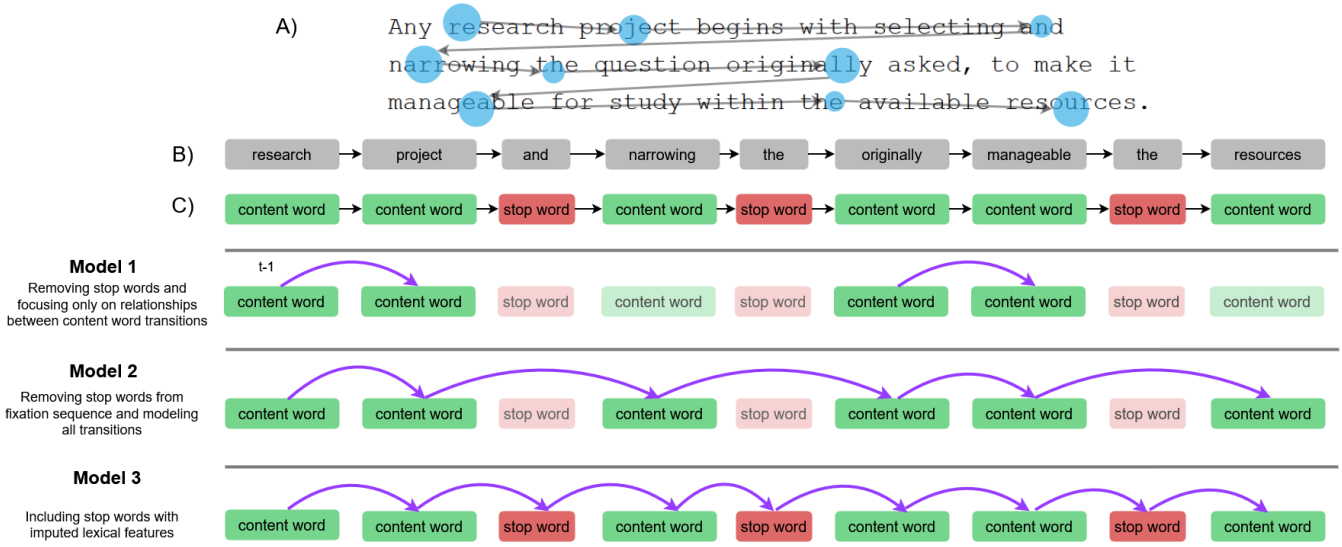


Figure 2: Illustration of a fixation sequence that consists of content words and stop words. A) Demonstrates an example sentence with eye-movements overlaid on individual words. B) The corresponding sequence of words in the order they were attended to. C) Words categorized at content words or stop words. The purple lines indicate the t-1 fixation that mlVAR models. Model 1 ignores stop words and as such, transitions from stop words, estimating temporal dependencies only between successive content words. Model 2 removes stop words from the fixation sequence, creating direct transitions between adjacent content words. Model 3 retains stop words and includes them in the temporal model with lexical features imputed.

Before beginning to read, participants are given the definition of mind wandering and instructions on how to respond to the probe. Mind wandering, or "zoning out", was defined as "not thinking about what you are actually reading, but thinking about something else altogether". Participants are asked to be as honest as possible, and informed that reporting mind wandering does not affect their progress in the study. Probes occurred twice per text on pseudo random pages, for a total of ten probes throughout the reading (see 4.3.1). When participants advanced to the next page, they would first be prompted with a yes or no question asking if they were mind wandering on the previous page. The average mind wandering rate was 47% (24% SD).

4.3.4 Comprehension Questions

Reading comprehension was measured via three different types of assessments that target different levels of comprehension. Rote questions were four-item multiple choice targeting factual knowledge explicitly presented in the text. Inference questions consisted of a statement reflecting either a true or false inference that can be made by extracting implicit information from a specific single sentence on the page. Finally, integration questions were four-item multiple-choice questions assessing deeper, integrative knowledge across multiple pages.

These comprehension questions were administered at three different time points, but not all questions were administered at each time point. Questions were administered 1) during texts, meaning the questions appeared after reading the corresponding page; 2) after an individual text was read; 3) after all five texts were administered. Rote and inference questions were administered at all time points, while integration questions were only administered during texts and after all texts were completed (time points 1 and 2). Questions always appeared in the order of Rote, Inference, Integration to reflect the increasing levels of processing, and were preceded by the mind wandering probes (see Section 4.3.3). Additionally, pages that triggered comprehension questions were selected pseudorandomly with basic constraints, such that no questions never appeared on the first page or adjacent pages of the text. The specific questions administered to each participant were selected pseudorandomly from a main question pool without repetition. Mean comprehension across all questions was 68% (10% SD).

4.3.5 Prior Knowledge Assessment

Participants completed the prior knowledge assessment prior to beginning the reading task. The participants had to match six definitions relating to scientific methods with their concept names. All definitions and names were randomly ordered (Confound, Placebo, Dependent Variable, External Validity, Control Group, and A priori). Correct answers were summed. Mean prior knowledge score were 68% (17% SD).

5. DATA PROCESSING AND MODELING

5.1 Time series measures

5.1.1 Lexical Variables

Two lexical features were extracted per-word: surprisal and the log of word frequency. Surprisal for each word was computed as the negative log probability assigned to that word after inputting its page into a pre-trained GPT-2 small language model, as this model has been shown to align closest with human reading times [51]. English word frequency was calculated using the nltk package then log-transformed. Lexical variables were only defined for content words in this

Table 1: Summary of gaze, lexical, and EEG variables included in the analysis.

Variable	Description
Word Frequency	Log-transformed word frequency, indexing how commonly a word occurs in natural language.
Surprisal	Lexical surprisal computed as the negative log probability of a word given its preceding context, reflecting contextual predictability.
Fixation Duration	Duration (in milliseconds) of a single eye fixation on a word during reading.
Mean P1	Mean EEG amplitude in the P1 time window (70-120 ms) extracted from posterior scalp regions for each fixation.
Mean P2	Mean EEG amplitude in the P2 time window (140-280 ms) extracted from central scalp regions for each fixation.
Mean N400	Mean EEG amplitude in the N400 time window (300–500 ms) extracted from central scalp regions for each fixation.

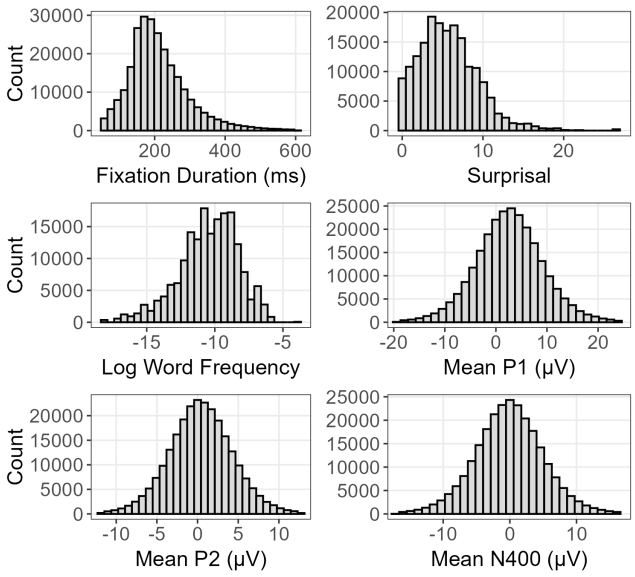


Figure 3: Histograms of gaze, lexical, and EEG variables included in the analysis. Lexical variables are for content words only. EEG variables are displayed ± 3 standard deviations. All other variables show complete range.

dataset; surprisal and word frequency values for stop words were not computed. In Model 1 lexical features for stop words were kept as missing. In Model 2, they were removed. Finally, in Model 3, lexical features were imputed to be the minimum surprisal and maximum frequency. Surprisal for content words in the text ranged between 0 and 26.5, with a mean and SD of 5.3 ± 3.5. Log word frequency ranged from -17.98 to -2.92, with a mean of -10 and SD of 2.1. Figure 3 shows distributions of lexical variables.

5.1.2 Gaze Preprocessing

Fixations, blinks, and saccades were identified from the raw time series with EyeLink's event-detection algorithm (velocity threshold of 30 degs/sec and acceleration threshold of 9500 deg/s^2). The right eye was used in the analysis if available, otherwise the left eye was used. The first and final fixations on each page were removed, as they are likely to be confounded with visual evoked potentials and motor artifacts (1.5% of fixations). Fixations with extreme durations (>600 or <50 ms) were also removed (5.1%), and pages with fewer than ten fixations were excluded (0.07%). The analysis focuses on relationship between lexical features, EEG, and gaze duration, so fixations not on words (no lexical information) or on punctuation were removed (5.12% of fixations). A total of 252795 fixations were included in the analyses, 28.37% of which were on stop words.

5.1.3 EEG Processing and FRP Extraction

EEG was time-locked to stimuli via trigger pulses generated at page transitions and other task-related events. Then EEG and eye-tracker data was aligned using a custom timestamp interpolation algorithm in Matlab R2021a [38] that accounted for timing drift between recording systems and accommodated missing triggers. For EEG preprocessing, line noise at 60 Hz was removed using the Zapline algorithm [17], and slow drifts were removed via robust detrending [18]. Bad channels were identified with the PREP pipeline [3] and interpolated from neighboring electrodes. All data was subsequently referenced to the average over all electrodes.

FRPs were estimated using a linear deconvolution approach implemented as a regression model to correct for potentially overlapping neural effects time-locked to word fixations [16, 22, 70]. For each participant and EEG channel, a linear model was fit to the continuous EEG time series. The model included categorical regressors marking the onsets of fixation events (separately for fixations on and off words), and additional regressors for blink and button-press events. This minimal model was chosen to estimate fixation-locked FRPs without imposing assumptions about cognitive or textual effects on the EEG.

Mean component amplitudes were extracted within predefined time windows and scalp regions of interest. This includes three canonical ERP components: P1 (70-120 ms) , measured over posterior visual regions; P2 (140-280 ms) measured over central and fronto-central regions; and N400 (300-500 ms), measured over central regions. These components and regions are selected based on prior literature and ensure separability given the limited spatial resolution of the EEG montage. For each fixation, component features were computed as the mean voltage within the corresponding time window and scalp region.

5.2 Multilevel Vector Autoregression

Multilevel vector autoregression (mlVAR) is a network analysis method that combines vector autoregression with linear mixed-effects modeling [8, 29]. Vector autoregression is the process of predicting a vector of variables at time t (index in sequence) using the same vector at time $t - n$, where n is lag. Combined with linear mixed effects models, this allows modeling of temporal dependencies among multiple

variables while accounting for the nested structure of the data (fixations within pages, and pages within individuals). We used the R package "mlVAR" (version 0.5.2) for all analysis [28].

5.2.1 Temporal Networks

First, mlVAR estimates temporal networks by sequentially predicting the vector of word-level EEG, gaze, and lexical variables for participant i at time t (denoted $Y_{i,t}$) from the same set of vectors at the previous time point $t-1$ using a mixed-effects formulation:

$$Y_{i,t} = Y_{i,t-1}\mathbf{B} \ + Y_{i,t-1}\mathbf{U}_i \ + \varepsilon_{i,t} \quad (1)$$

B is the fixed effects matrix that contains the average estimates of population-level lagged effects between all variables. U_i is the random-effects matrix that captures participant-specific deviations from these temporal relationship. $\epsilon_{i,t}$ is a matrix of residual error term. Each row of B corresponds to a separate regression model predicting one variable at time t from all variables at time $t-1$.

The resulting fixed-effects coefficients then form the directed temporal networks. In these networks, a significant edge between two variables indicates that the "earlier" variable $(t-1)$ significantly predicts the "later" one (t), after controlling for all other variables. Statistical significance was determine using a threshold of $p < 0.05$, consistent with prior work [25, 37].

5.2.2 Contemporaneous Networks

In addition to temporal effects, mlVAR estimates contemporaneous networks. These capture associations between variables occurring at the same time point after accounting for lagged temporal dependencies. Contemporaneous networks are computed from the residual covariance structure of the temporal models. This reflects relationships between variables that co-occur within the same fixation but are not explained by prior states of the system (temporal effects or other variables).

Let ε_{it} denote the vector of residuals obtained from the temporal model for participant i at time t. Contemporaneous associations are modeled by predicting each residual from the remaining residuals at the same time point using a mixed effects formulation:

$$\varepsilon_{it} = \varepsilon^*_{it}\mathbf{H} \ + \varepsilon^*_{it}\mathbf{G}_i \ + e_{it} \quad (2)$$

where ε^*_{it} denotes the vector of residual excluding the outcome variable currently being predicted. The fixed-effects matrix H contains population-level contemporaneous associations between variables, while G_i captures participant-specific deviations from these associations. The term e_{it} represents residual error.

Each row of H corresponds to a regression predicting one variable's residual at fixation t from the residuals of all other variables at the same time point. The resulting fixed-effects coefficients from an undirected contemporaneous network, in which an edge between two variables reflects their partial association within the same fixation, after controlling for all lagged temporal effects and the variance shared with all other modeled variables. Conceptually, whereas temporal edges represent predictive, directional associations across successive fixations, contemporaneous edges capture instantaneous co-fluctuations between gaze, EEG, and lexical variables that arise within a single fixation and are not attributable to prior dynamics.

5.2.3 Selection of Focal Network Edges

While all edges between all variables were modeled in mlVAR, we focus on a subset of cognitively-relevant edges in both networks which address our research questions. Specifically, in the temporal network, we examine directed edges from text $\rightarrow$ EEG and text $\rightarrow$ fixation duration (gaze), which capture how lexical properties (surprisal and word frequency) are related to fixation duration and FRPs on the subsequent word. We additionally consider gaze $\rightarrow$ EEG, reflecting how the amount of visual attention allocated to a word modulates subsequent neural processing, as well as EEG $\rightarrow$ gaze edges, which capture how neural indices of processing are associated with attentional allocation on the following word.

In the contemporaneous network, we included the same edges but without directional components: text $\leftrightarrow$ EEG, text $\leftrightarrow$ gaze, and gaze $\leftrightarrow$ EEG. Figure 4F shows the contemporaneous networks with these focal edges visualized. These fall into only three categories because contemporaneous edges are undirected and reflect symmetric associations between modalities that co-occur within a fixation.

Edges directed into lexical feature were excluded, as these features are properties of the text rather than of the reader and therefore do not reflect cognitive dynamics. Self-loops (edges which a variable predicts itself at the next time step) were also excluded, as they primarily reflect temporal autocorrelation rather than theoretically informative cross-modal associations. EEG self-loops in particular are difficult to interpret, as correlations across successive FRPs may arise from shared baseline activity, slow drifts, or differences in overall signal amplitude rather than meaningful cognitive processing across fixations.

By restricting analysis to focal cross-modal edges, we emphasize relationships that reflect dynamic interactions between text properties, eye movements, and neural responses during reading. Figure 4D illustrates the fourteen direct focal edges considered in the temporal network.

6. RESULTS

6.1 Temporal and Contemporaneous Effects

We first address RQ1 by examining network structure across models. Table 2 shows the significant edges and their weights. Overall, core lexical-to-gaze effects emerged within-fixations, and across fixations. These effects were consistent in direction across modeling approaches. EEG-related effects, in contrast, appeared only under certain modeling conditions.

Across both temporal and contemporaneous networks, surprisal to fixation duration had a positive effect and word frequency to fixation duration had a negative effect. These relationships are robust and were observed across all three

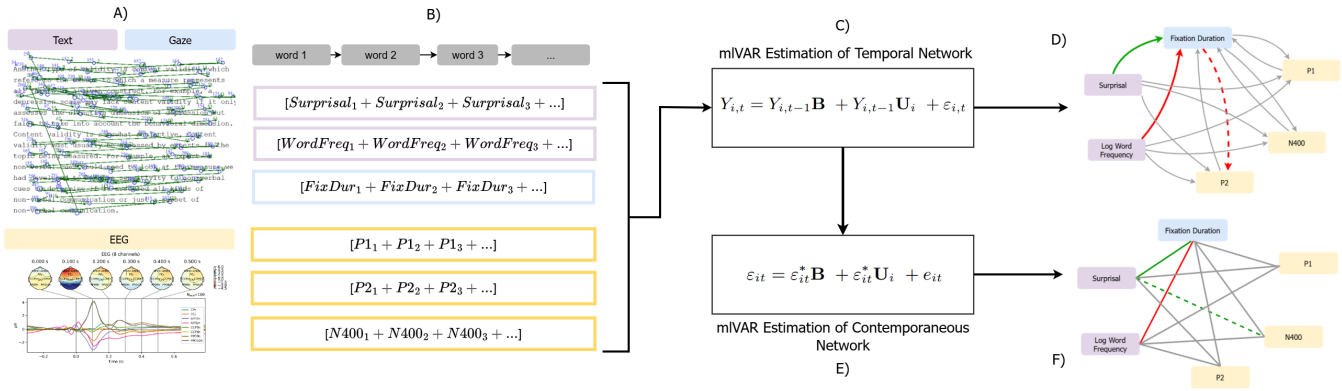


Figure 4: Study pipeline from data to final temporal and contemporaneous networks. A) Fixations and corresponding word are extracted. Fixation-related potentials (FRPs) are extracted across EEG electrodes. B) Time series is index in the order words are looked at. For each fixation, the duration is extracted along with the surprisal and word frequency of the attended word. Mean P1, P2, and N400 from relevant channels and time windows are extracted for each FRP. C) All vectors are then assessed with the mlVAR algorithm to extract temporal dependencies, which are visualized in D) where significant positive relations are in green and negative are in red. Solid lines are significant across all three models while dotted are not. E) Residuals from the temporal network are then used to estimate the contemporaneous, within-fixation dependencies using mlVAR. F) Visualization of the contemporaneous network, following the same labeling as the temporal.

Table 2: Significant temporal and contemporaneous edges across models. Temporal edges are represented as β. Contemporaneous edges are represented as ρ. $p < 0.05$ for all reported values. M1, M2, and M3 indicate Models 1, 2, and 3 respectively.

Temporal edges			
Edge	**M1**	**M2**	**M3**
Surprisal → Fixation Duration	0.012	0.010	0.011
Word Freq → Fixation Duration	-0.020	-0.012	-0.012
P2 → Fixation Duration			-0.006
Contemporaneous edges			
Edge	**M1**	**M2**	**M3**
Surprisal ↔ Fixation Duration	0.019	0.021	0.016
Word Freq ↔ Fixation Duration	-0.015	-0.019	-0.021
Surprisal ↔ N400		0.005	0.004

models. The direction of these effects align with previous research examining within-fixation patterns. Specifically, longer fixations are typically observed for words requiring deeper processing: either because they are low frequency words or are surprising, unpredictable words given the sentence context [53, 55]. The fact that these relationships were observed in the temporal models suggests a spillover effect in which word properties for fixation t influence gaze patterns for fixation $t+1$.

Two additional edges involving EEG features emerged in particular networks and model configurations. In particular, the temporal network in had a negative edge between mean P2 and subsequent fixation duration: as mean voltage within the P2 window (140-280 ms after fixation onset) became more positive, fixation duration of the next word decreases. Increase in the P2 magnitude has been previously linked to an increase in early word-level processing within-fixations [1, 20], but has been less explored temporally. Interestingly, this association is only present in Model 3, which contains stop words that usually have shorter fixation durations. Taken together, this effect may reflect either reduced attention to the upcoming word following larger P2 response or a greater likelihood that the subsequent fixation lands on a word (like a stop word) that typically requires less processing.

Surprisal also showed a positive within-fixation association with the mean N400 in the contemporaneous networks for Models 2 and 3. That is, as surprisal increased, the mean value in the N400 window (300-500 ms) became more positive (less pronounced negative amplitude). This conflicts with previous research showing higher surprisal linked to an larger negative peak in the N400 [44]. However we use a mean voltage value within the analysis window. It is possible the mean value captures too much surrounding activity in the FRP, pulling the value more positive when shifts in the timing of the N400 peak occur. Additionally, imputing stop words affects the surprisal distribution in ways not traditionally modeled. Because stop words have very low surprisal, assigning them imputed values changes which surprisal values are most common in the data, so the surprisal-N400 relationship is estimated from a different mix of words than typical ERP studies.

Although individual edge magnitudes are small, this is expected in a multivariate time series of fixation-level data where effects reflect incremental, conditional associations after accounting for strong autoregressive and cross-modal dependencies. Small estimates at the fixation level can accumulate across the thousands of fixations per participant. Importantly, these effects were highly stable as evidenced by the following permutation tests.

6.1.1 Permutation Testing

To assess the robustness of the estimated network structures, we conducted a participant-level permutation subsampling analysis. Twenty five random subsamples of fifty

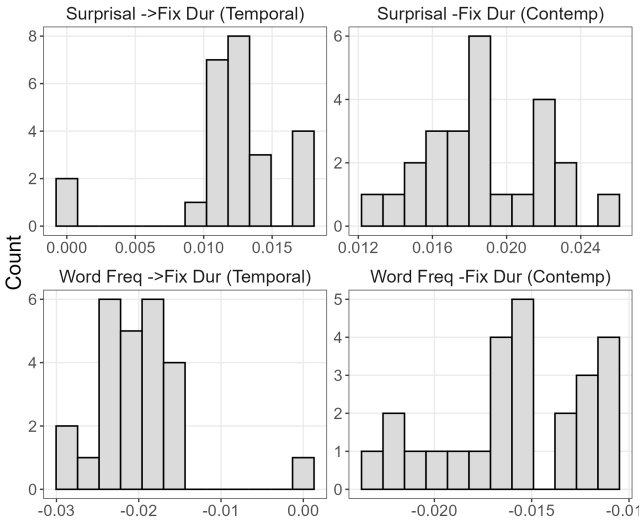


Figure 5: Histograms of edge weights from the 25 permutations for the two significant edges in the temporal and contemporaneous networks for Model 1

participants each were drawn without replacement, and the mlVAR model was fit for each. Temporal and contemporaneous networks were extracted from each fitted model, and edge weights were aggregated across permutations to form empirical distributions for each network connection. We conducted this analysis using Model 1 as an illustration. Edges identified as significant in Model 1 were verified with a 95% confidence interval of their permutation distribution excluding zero. This procedure evaluates whether observed network dependencies reflect stable population-level relationships rather than effects driven by a small subset of participants, providing a robustness of the mlVAR results.

Permutation testing on Model 1 consistently showed the same significant edges. In the temporal network, the 95% confidence interval for the significant edges were [0.010, 0.014] for Surprisal → Fixation Duration and [-0.022, -0.018] for Word Frequency → Fixation Duration. In the contemporaneous network, the corresponding intervals were [0.018, 0.020] Surprisal ↔ Fixation Duration for and [-0.017, -0.015] for Word Freq ↔ Fixation Duration. These confidence intervals that do not overlap with 0 (null effect) indicate that the observed associations are reliable and robust despite their small magnitudes.

6.2 Cognitive Variables as Predictors of Network Connectivity

To address RQ2 we examined cognitive variables as predictors of connectivity in the temporal and contemporaneous networks for each model. Specifically, we leveraged the multilevel structure of the mlVAR model to extract participant-specific edge estimates. Because temporal edges are estimated using hierarchical linear mixed effects models, we can extract participant-level deviations from the group-level network. We extracted the fourteen participant-level temporal edges for each individual, for each relevant directed connection in the network as well as the 11 focal contemporaneous

Table 3: Contemporaneous models cognitive effects on total connectivity (Mean per-person and standardized). Regression coefficients (β) and p-values are reported. Significant effects ($p < .05$) in bold.

	Model 1		Model 2		Model 3	
Predictor	β	p	β	p	β	p
Reading time	-.11	.25	.05	.59	**-.27**	**.00**
Comprehension	**.24**	**.03**	**.25**	**.02**	.08	.46
Prior knowledge	-.02	.81	-.03	.75	-.11	.28
Mind wandering	-.02	.82	-.02	.88	.03	.78
Observations	116		116		116	
R^2 / Adj. R^2	0.07 / 0.04		0.06 / 0.03		0.08 / 0.04	

edges. We then investigated how participant-level differences in comprehension outcomes, mind wandering rates, prior knowledge, and reading time predicted total connectivity in the focal edges in each network (Figure 4). We calculated total connectivity as the sum of the absolute value of each focal edge (14 edges in temporal network and 11 edges in the contemporaneous network). We used the following linear regression model:

$$\begin{aligned}\text{Connectivity} \sim \text{Mind Wandering}_{\text{mean}} + \text{Comprehension}_{\text{mean}} \\ + \text{Prior Knowledge}_{\text{mean}} + \text{Reading Time}_{\text{mean}}\end{aligned}$$

Results are shown in Table 3. There were no significant associations between the cognitive variables and total connectivity in the temporal networks. The contemporaneous networks, however, had positive effects of Comprehension in Models 1 and 2, and a negative effect of reading time in Model 3.

These results suggest that higher comprehension is associated with stronger within-fixation coupling between modalities when stop words are excluded (Models 1 and 2). One interpretation is that readers who achieve better comprehension exhibit more tightly coordinated processing across gaze, neural, and lexical signals. Greater total connectivity may reflect a more cognitively coupled state in which the modalities co-vary more strongly [48]. Importantly, this association does not imply a directional relationship between comprehension and connectivity, but instead highlights that stronger comprehension tends to co-occur with denser contemporaneous structure in the multimodal network. That absence of this pattern when stop words are retained further suggests that this coupling is more evident during fixations on content-bearing words, which entails integration of lower-level of processing into a coherent mental model of the text including making bridging inferences from previous parts of the text and also elaborative inferencing from long-term memory [47].

In contrast, when stop words are included (Model 3), longer reading times are associated with weaker contemporaneous connectivity. Reading time is a complex measure that can indicate skimming, scanning, or deep reading strategies [13], each of which is reflected in distinct gaze patterns. Increased reading time can also function as a compensatory strategy when comprehension difficulty occurs [13]. Readers

may compensate by slowing down or through regressive (i.e., re-reading) eye movements [54], which would increase reading times. Here, longer reading times were associated with weaker contemporaneous network connectivity, i.e. weaker coupling between modalities occurring within-fixation. One interpretation is an association between comprehension difficulties (longer reading time to repair difficulties) and weaker network structure.

Lastly, neither prior knowledge nor mind wandering were predictors of network connectivity. It is possible these variables show selective differences across network edges rather than changes in overall connectivity.

7. DISCUSSION

The primary aim of this work was to examine how multimodal temporal modeling can help uncover moment-to-moment eye-mind coordination between gaze, fixation-related potentials, and text characteristics during reading. Our findings illustrate how fixation-level mlVAR can serve as an exploratory tool to examine temporal and within-fixation coupling (RQ1), and examine cognitive associations on network connectivity (RQ2), and compare different approaches for representing the time series (RQ3). Although, we focused on the specific domain of reading comprehension here, our we aim for the overall modeling approach to be informative and potentially useful to related domains in educational data mining.

7.1 Main Findings

Regarding RQ1, the observed temporal and contemporaneous networks align with classic accounts of the eye-mind link that emphasize a close coupling between linguistic processing and eye-movement behavior during reading. Moment-to-moment gaze dynamics were sensitive to the lexical properties of the text, consistent with long-standing views that fixation duration reflects ongoing word-level processing. Additionally, the temporal effects across fixations extend prior static within-fixation analyses by showing that these established lexical-gaze relationships might be anticipatory or signal spillover effects. Neural associations were less stable and more sensitive to how fixation sequences were represented.

We explored RQ2 by investigating effects of participant-level cognitive variables (mind wandering, reading time, prior knowledge, comprehension) on total network connectivity. Comprehension and reading time showed effects in the contemporaneous network only: reading comprehension scores predicted connectivity whereas reading time negatively predicted it. Overall, these effects are consistent with theoretical perspectives highlighting a tight coupling between the eye, brain, and text required for successful comprehension, and that increased reading times might signal comprehension difficulties and repair strategies. Rather than treating the models as competing methods, we interpret them as complementary lenses on the same underlying reading process, each making different assumptions. Differences in cognitive effects across models reflect how these assumptions shape which aspects of multimodal coupling are emphasized, rather than indicating that one model is more correct than others. More research is needed to examine how the reported effects are sensitive to modeling choices.

Finally, to address RQ3, we compared results across three modeling strategies that differed in how function words (stop words) were treated in the determination of $t-1$ fixations for temporal modeling. Model 1 preserved only the content-to-content word transitions that were originally present in the fixation sequence, making no assumptions about stop words and excluding any transitions to or from stop words. Model 2 treated stop words as minimally processed and short in duration, effectivity skipping them by removing stop-word fixations so that all modeled transitions occurred between content words. Model 3 retained stop words in the sequence by imputing them with the lowest surprisal and highest frequency values, allowing fixation sequences to include both content and function words.

Core lexical-gaze relations were stable to different stop word treatments, while neural-gaze and lexical-neural associations were more model dependent. Given the additional P2 to fixation duration and surprisal to N400 edges in Model 3, neural coupling might be more sensitive to how stop words are handled, potentially due to prediction mechanisms that are not captured in the eye gaze sequence. Additionally, a negative effect of reading time on total contemporaneous connectivity was found only in Model 3. Because stop words are associated with minimal semantic processing and brief fixations, including them may reduce the coherence of within-fixation multimodal structure, though this needs to be examined further.

7.2 Limitations

A primary limitation of our study is the interpretation of findings given the nature of the analyses. Although we examined associations between cognitive measures and network properties, the results do not support claims about underlying mechanisms or directional relationships (e.g. whether higher comprehension stems from stronger connectivity or stronger connectivity stems from better comprehension). Addressing such questions requires analytic approaches beyond the correlational framework used here. Second, the temporal resolution of the model is constrained to the fixation-level. Cognitive processes involved in reading do not always unfold at the timescale of individual fixations, and lag-1 dependencies might fail to capture longer-range dynamics across words, sentences, or pages. Additionally, cognitive measures were aggregated at the participant-level, which may obscure moment-to-moment variability in cognitive states. Lastly, our data were collected in a specific lab setting where participants wore a neural cap, read expository texts, wore masks (due to Covid-19). Thus, it is unclear whether our findings can be generalized to more ecological reading scenarios.

7.3 Concluding Remarks

The eye-mind link is inherently dynamic, relying on moment-to-moment coordination between text, gaze, and neural activity. The results presented here suggest that temporal modeling of fixation sequences can recover well-established patterns from the reading literature while extending them to richer, multimodal, relationships and elucidating temporal effects. This approached helps assess whether relationships typically observed in isolation or within-fixations are observed in the context of full-length sequences with temporal interdependencies across fixations. While lexical properties show robust associations with subsequent fixation du-

ration, incorporating EEG alongside gaze measures offers additional information for tracking cognitive states during reading. Our findings also suggest that low-level details in how the underlying time series are encoded can have major implications on the conclusions. Lastly, while we focused on reading comprehension and a specific multimodal configuration of signals here, the overall mlVAR approach can be applied to any set of modalities collected across a range of domains, thereby providing new insights into the learning process.

8. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation (DRL 1920510). The opinions expressed are those of the authors and do not represent views of the funding agencies.

9. REFERENCES

[1] T. Baccino and Y. Manunta. Eye-fixation-related potentials: Insight into parafoveal processing. *Journal of Psychophysiology*, 19(3):204–215, 2005.

[2] R. Biedert, J. Hees, A. Dengel, and G. Buscher. A robust realtime reading-skimming classifier. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, page 123–130, New York, NY, USA, 2012. Association for Computing Machinery.

[3] N. Bigdely-Shamlo, T. Mullen, C. Kothe, K.-M. Su, and K. A. Robbins. The prep pipeline: standardized preprocessing for large-scale eeg analysis. *Frontiers in neuroinformatics*, 9:16, 2015.

[4] R. Bixler and S. D'Mello. Automatic gaze-based detection of mind wandering with metacognitive awareness. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 31–43. Springer, 2015.

[5] R. Bixler and S. D'Mello. Automatic gaze-based user-independent detection of mind wandering during computerized reading. *User Modeling and User-Adapted Interaction*, 26(1):33–68, 2016.

[6] I. Blank, Z. Balewski, K. Mahowald, and E. Fedorenko. Syntactic processing is distributed across the language system. *Neuroimage*, 127:307–323, 2016.

[7] M. A. Boudewyn. Half-listening or zoned out? it's about the same: the impact of attentional state on word processing in context. *Cognitive neuroscience*, 14(3):107–114, 2023.

[8] L. F. Bringmann, N. Vissers, M. Wichers, N. Geschwind, P. Kuppens, F. Peeters, D. Borsboom, and F. Tuerlinckx. A network approach to psychopathology: New insights into clinical longitudinal data. *PloS one*, 8(4):e60188, 2013.

[9] J. M. Broadway, M. S. Franklin, and J. W. Schooler. Early event-related brain potentials and hemispheric asymmetries reveal mind-wandering while reading and predict comprehension. *Biological Psychology*, 107:31–43, 2015.

[10] M. P. Broderick, A. J. Anderson, G. M. Di Liberto, M. J. Crosse, and E. C. Lalor. Electrophysiological correlates of semantic dissimilarity reflect the comprehension of natural, narrative speech. *Current Biology*, 28(5):803–809, 2018.

[11] G. Buscher, A. Dengel, and L. Van Elst. Query expansion using gaze-based feedback on the subdocument level. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 387–394, 2008.

[12] M. Caruso, C. E. Peacock, R. Southwell, G. Zhou, and S. K. D'Mello. Going deep and far: Gaze-based models predict multiple depths of comprehension during and one week following reading. *International Educational Data Mining Society*, 2022.

[13] R. P. Carver. Reading rate: Theory, research, and practical implications. *Journal of Reading*, 36(2):84–95, 1992.

[14] A. E. Cook and W. Wei. What can eye movements tell us about higher level comprehension? *Vision*, 3(3):45, 2019.

[15] L. Copeland, T. Gedeon, and S. Caldwell. Effects of text difficulty and readers on predicting reading comprehension from eye movements. In *2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pages 407–412, 2015.

[16] T. Cornelissen, J. Sassenhagen, and M. L.-H. Vo. Improving free-viewing fixation-related eeg potentials with continuous-time regression. *Journal of Neuroscience Methods*, 313:77–94, 2019.

[17] A. De Cheveigné. Zapline: A simple and effective method to remove power line artifacts. *NeuroImage*, 207:116356, 2020.

[18] A. De Cheveigné and D. Arzounian. Robust detrending, rereferencing, outlier detection, and inpainting for multichannel data. *NeuroImage*, 172:903–912, 2018.

[19] F. Degno and S. Liversedge. Eye movements and fixation-related potentials in reading: A review. vision, 4 (1), 11. *Vision*, 4(1), 2020.

[20] F. Degno, O. Loberg, and S. P. Liversedge. Co-registration of eye movements and fixation—related potentials in natural reading: Practical issues of experimental design and data analysis. *Collabra: Psychology*, 7(1):18032, 2021.

[21] K. A. DeLong, T. P. Urbach, and M. Kutas. Probabilistic word pre-activation during language comprehension inferred from electrical brain activity. *Nature neuroscience*, 8(8):1117–1121, 2005.

[22] O. Dimigen and B. V. Ehinger. Regression-based analysis of combined eeg and eye-tracking data: Theory and applications. *Journal of vision*, 21(1):3–3, 2021.

[23] O. Dimigen, W. Sommer, A. Hohlfeld, A. M. Jacobs, and R. Kliegl. Coregistration of eye movements and eeg in natural reading: analyses and review. *Journal of experimental psychology: General*, 140(4):552, 2011.

[24] S. K. D'Mello, R. Southwell, and J. Gregg. Machine-learned computational models can enhance the study of text and discourse: A case study using eye tracking to model reading comprehension. *Discourse processes*, 57(5-6):420–440, 2020.

[25] O. V. Ebrahimi, J. Burger, A. Hoffart, and S. U. Johnson. Within-and across-day patterns of interplay

between depressive symptoms and related psychopathological processes: a dynamic network approach during the covid-19 pandemic. *BMC medicine*, 19(1):317, 2021.

[26] S. F. Ehrlich and K. Rayner. Contextual effects on word perception and eye movements during reading. *Journal of verbal learning and verbal behavior*, 20(6):641–655, 1981.

[27] R. Engbert, A. Nuthmann, E. M. Richter, and R. Kliegl. Swift: a dynamical model of saccade generation during reading. *Psychological review*, 112(4):777, 2005.

[28] S. Epskamp, M. K. Deserno, L. F. Bringmann, and M. Veenman. Package 'mlvar', 2022.

[29] S. Epskamp, L. J. Waldorp, R. Mõttus, and D. Borsboom. The gaussian graphical model in cross-sectional and time-series data. *Multivariate behavioral research*, 53(4):453–480, 2018.

[30] M. Faber, R. Bixler, and S. K. D'Mello. An automated behavioral measure of mind wandering during computerized reading. *Behavior Research Methods*, 50(1):134–150, 2018.

[31] E. Fedorenko, J. Duncan, and N. Kanwisher. Language-selective and domain-general regions lie side by side within broca's area. *Current Biology*, 22(21):2059–2062, 2012.

[32] R. Flesch. A new readability yardstick. *Journal of applied psychology*, 32(3):221, 1948.

[33] S. L. Frank and A. Aumeistere. An eye-tracking-with-eeg coregistration corpus of narrative sentences. *Language Resources and Evaluation*, 58(2):641–657, 2024.

[34] S. L. Frank and R. M. Willems. Word predictability and semantic similarity show distinct patterns of brain activity during language comprehension. *Language, Cognition and Neuroscience*, 32(9):1192–1203, 2017.

[35] A. C. Graesser, M. Singer, and T. Trabasso. Constructing inferences during narrative text comprehension. *Psychological review*, 101(3):371, 1994.

[36] J. Grainger and P. J. Holcomb. Watching the word go by: On the time-course of component processes in visual word recognition. *Language and linguistics compass*, 3(1):128–156, 2009.

[37] L. M. Hamel, R. Moulder, F. W. Harper, L. A. Penner, T. L. Albrecht, and S. Eggly. Examining the dynamic nature of nonverbal communication between black patients with cancer and their oncologists. *Cancer*, 127(7):1080–1090, 2021.

[38] T. M. Inc. Matlab version: R2021a, 2021.

[39] M. A. Just and P. A. Carpenter. Eye fixations and cognitive processes. *Cognitive psychology*, 8(4):441–480, 1976.

[40] P. Kendeou and P. Van Den Broek. The effects of prior knowledge and text structure on comprehension processes during reading of scientific texts. *Memory & cognition*, 35(7):1567–1577, 2007.

[41] W. Kintsch. The role of knowledge in discourse comprehension: a construction-integration model. *Psychological review*, 95(2):163, 1988.

[42] W. Kintsch. Text comprehension, memory, and learning. *American psychologist*, 49(4):294, 1994.

[43] W. Kintsch. *Comprehension: A paradigm for cognition*. Cambridge university press, 1998.

[44] M. Kutas and K. D. Federmeier. Thirty years and counting: finding meaning in the n400 component of the event-related brain potential (erp). *Annual review of psychology*, 62(1):621–647, 2011.

[45] S. J. Luck. *An introduction to the event-related potential technique*. MIT press, 2014.

[46] P. Martínez-Gómez and A. Aizawa. Recognition of understanding level and language skill using measurements of reading behavior. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pages 95–104, Haifa Israel, Feb. 2014. ACM.

[47] D. S. McNamara. If integration is the keystone of comprehension: Inferencing is the key. *Discourse Processes*, 58(1):86–91, 2021.

[48] C. Mills, A. Graesser, E. F. Risko, and S. K. D'Mello. Cognitive coupling during reading. *Journal of experimental psychology: General*, 146(6):872, 2017.

[49] C. Mills, J. Gregg, R. Bixler, and S. K. D'Mello. Eye-mind reader: An intelligent reading interface that promotes long-term comprehension by detecting and responding to mind wandering. *Human–Computer Interaction*, 36(4):306–332, 2021.

[50] H. Miyata, Y. Minagawa-Kawai, S. Watanabe, T. Sasaki, and K. Ueda. Reading speed, comprehension and eye movements while reading japanese novels: Evidence from untrained readers and cases of speed-reading trainees. *PloS one*, 7(5):e36091, 2012.

[51] B.-D. Oh and W. Schuler. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350, 2023.

[52] C. J. Price, C. J. Moore, G. W. Humphreys, and R. J. Wise. Segregating semantic from phonological processes during reading. *Journal of cognitive neuroscience*, 9(6):727–733, 1997.

[53] K. Rayner, J. Ashby, A. Pollatsek, and E. D. Reichle. The effects of frequency and predictability on eye fixations in reading: implications for the ez reader model. *Journal of Experimental Psychology: Human Perception and Performance*, 30(4):720, 2004.

[54] K. Rayner, K. H. Chace, T. J. Slattery, and J. Ashby. Eye movements as reflections of comprehension processes in reading. *Scientific studies of reading*, 10(3):241–255, 2006.

[55] K. Rayner and S. A. Duffy. Lexical complexity and fixation times in reading: Effects of word frequency, verb complexity, and lexical ambiguity. *Memory & cognition*, 14(3):191–201, 1986.

[56] K. Rayner and G. W. McConkie. What guides a reader's eye movements? *Vision research*, 16(8):829–837, 1976.

[57] K. Rayner, A. Pollatsek, J. Ashby, and C. Clifton Jr. *Psychology of reading*. Psychology Press, 2012.

[58] K. Rayner, E. D. Reichle, and A. Pollatsek. Eye movementcontrol in reading: An Overview and Model. *Eye guidance in reading and scene perception*, pages 243–268, 1998.

[59] K. Rayner, T. J. Slattery, D. Drieghe, and S. P. Liversedge. Eye movements and word skipping during reading: effects of word length and predictability. *Journal of Experimental Psychology: Human Perception and Performance*, 37(2):514, 2011.

[60] E. D. Reichle. Computational models of eye-movement control during reading: Theories of the" eye-mind" link. *Cognitive Systems Research*, 2006.

[61] E. D. Reichle, K. Rayner, and A. Pollatsek. The e-z reader model of eye-movement control in reading: Comparisons to other models. *Behavioral and Brain Sciences*, 26(4):445–476, 2003.

[62] E. D. Reichle, A. E. Reineberg, and J. W. Schooler. Eye movements during mindless reading. *Psychological science*, 21(9):1300–1310, 2010.

[63] E. D. Reichle, T. Warren, and K. McConnell. Using ez reader to model the effects of higher level language processing on eye movements during reading. *Psychonomic bulletin & review*, 16(1):1–21, 2009.

[64] J. W. Schooler, J. Smallwood, K. Christoff, T. C. Handy, E. D. Reichle, and M. A. Sayette. Meta-awareness, perceptual decoupling and the wandering mind. *Trends in cognitive sciences*, 15(7):319–326, 2011.

[65] A.-L. Schubert, G. T. Frischkorn, and J. Rummel. The validity of the online thought-probing procedure of mind wandering is not threatened by variations of probe rate and probe framing. *Psychological research*, 84(7):1846–1856, 2020.

[66] S. C. Sereno, C. J. Hand, A. Shahid, I. G. Mackenzie, and H. Leuthold. Early eeg correlates of word frequency and contextual predictability in reading. *Language, cognition and neuroscience*, 35(5):625–640, 2020.

[67] V. J. Shute and D. Zapata-Rivera. Adaptive educational systems. In P. J. Durlach and A. M. Lesgold, editors, *Adaptive Technologies for Training and Education*, pages 7–27. Cambridge University Press, Cambridge, 2012.

[68] J. Smallwood, D. J. Fishman, and J. W. Schooler. Counting the cost of an absent mind: Mind wandering as an underrecognized influence on educational performance. *Psychonomic bulletin & review*, 14(2):230–236, 2007.

[69] J. Smallwood, M. McSpadden, and J. W. Schooler. When attention matters: The curious incident of the wandering mind. *Memory & cognition*, 36(6):1144–1150, 2008.

[70] N. J. Smith and M. Kutas. Regression-based estimation of erp waveforms: I. the rerp framework. *Psychophysiology*, 52(2):157–168, 2015.

[71] N. J. Smith and M. Kutas. Regression-based estimation of erp waveforms: Ii. nonlinear effects, overlap correction, and practical considerations. *Psychophysiology*, 52(2):169–181, 2015.

[72] C. Snow. *Reading for understanding: Toward an R&D program in reading comprehension*. Rand Corporation, 2002.

[73] R. Southwell, J. Gregg, R. Bixler, and S. K. D'Mello. What eye movements reveal about later comprehension of long connected texts. *Cognitive Science*, 44(10):e12905, 2020.

[74] R. Southwell, A. Kim, and S. D'Mello. (in review) eeg deconvolution modelling of naturalistic reading reveals surprisal effects modulated during mind wandering. In Review.

[75] L. Steindorf and J. Rummel. Do your eyes give you away? a validation study of eye-movement measures used as indicators for mindless reading. *Behavior research methods*, 52(1):162–176, 2020.

[76] T. A. Van Dijk, W. Kintsch, et al. Strategies of discourse comprehension. 1983.

[77] M. L. van Moort, D. D. Jolles, A. Koornneef, and P. van den Broek. What you read versus what you know: Neural correlates of accessing context information and background knowledge in constructing a mental representation during reading. *Journal of Experimental Psychology: General*, 149(11):2084, 2020.

[78] T. L. Varao-Sousa and A. Kingstone. Are mind wandering rates an artifact of the probe-caught method? using self-caught mind wandering in the classroom to test, and reject, this possibility. *Behavior Research Methods*, 51(1):235–242, 2019.

[79] R. Wong, E. D. Reichle, and A. Veldre. Prediction in reading: A review of predictability effects, their theoretical implications, and beyond. *Psychonomic Bulletin & Review*, 32(3):973–1006, 2025.

[80] M. Yeari. The role of working memory in inference generation during reading comprehension: Retention,(re) activation, or suppression of verbal information? *Learning and Individual Differences*, 56:1–12, 2017.

From Heuristics to Analytics: Forecasting Effort and Progress in Online Learning

Eric S. Qiu
Cornell University
Ithaca, NY, USA
sq225@cornell.edu

Danielle R. Thomas
Carnegie Mellon University
Pittsburgh, PA, USA
drthomas@cmu.edu

Boyuan Guo
Carnegie Mellon University
Pittsburgh, PA, USA
boyuang@andrew.cmu.edu

Vincent Aleven
Carnegie Mellon University
Pittsburgh, PA, USA
aleven@cs.cmu.edu

Conrad Borchers
Carnegie Mellon University
Pittsburgh, PA, USA
cborcher@cs.cmu.edu

ABSTRACT

Sustained effort is essential for realizing the benefits of intelligent tutoring systems (ITS), yet many learners disengage or underuse available practice time. We introduce *engagement forecasting* as a supervised prediction task based on ITS logs, targeting two outcomes central to effort and learning progress: **minutes practiced per week** and **new skills mastered per week**. Using interaction log data from 425 middle-school students over a school year, we benchmark fifteen predictors including regressions, decision trees, and neural networks. We show that these feature-based models reduce mean absolute error (MAE) by 22–33% relative to heuristic baselines, including fixed-percentile rules adapted from prior work in other behavioral domains. We find that percentile heuristics systematically overpredict, whereas feature-based models better track student practice trajectories across weeks. To support explainability, we analyze feature importance and ablations, revealing target-specific patterns: effort forecasting is driven mainly by recent activity features, while progress forecasting depends more on learner-state and content difficulty signals. Finally, in a semi-structured user interview case study with eight college tutors, we examine how tutors reasoned about system-generated predictive features when setting goals with students. We find that tutors reasoned differently about effort versus progress goals in ways that mirror our pattern analysis. Together, these results establish a reproducible benchmark for forecasting weekly effort and learning progress in ITS. By making patterns of sustained effort and progress visible at a weekly timescale, engagement forecasting offers a foundation for supporting tutor–learner goal setting and timely instructional decisions.

Keywords

Effort Regulation, Intelligent Tutoring Systems, Machine Learning, K-12

Eric S. Qiu, Danielle R Thomas, Boyuan Guo, Vincent Aleven, and Conrad Borchers. From Heuristics to Analytics: Forecasting Effort and Progress in Online Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 117–129. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039913

1. INTRODUCTION

Active learning systems that scaffold student problem solving can substantially improve learning outcomes, especially when learners engage in high volumes of practice with feedback over several months [36, 33]. Yet, persistence and effort remain bottlenecks: many students disengage after initial practice or do not utilize the class time they have at their disposal, limiting these systems' long-term impact [25, 31]. In large-scale deployments, engagement is often highly skewed—for example, only a small minority of students may reach recommended weekly usage levels [29, 24, 22]. This disparity in engagement motivates a central question: how can analytics predict and explain differences in student effort to generate actionable, learner- and teacher-facing insights that support effort regulation?

In this paper, we introduce *engagement forecasting*: a supervised prediction task that estimates a learner's next-week engagement level from prior ITS interaction traces. We focus on two engagement indicators that are both observable in ITS logs and directly relevant to effort regulation support: **minutes practiced per week** (effort) and **new skills mastered per week** (learning progress) [41]. We adopt these targets not as complete measures of motivation or learning, but as simple, observable signals that tutors and platforms already use for week-to-week planning [24], making them suitable for decision support even when imperfect.

Importantly, student engagement in ITS is constrained, but not fully fixed. Middle-school students typically engage with tutoring software within clear boundaries set by teachers and schools, such as designated class time and homework policies. Yet, students still exercise agency within these constraints: how consistently they use allotted time, whether they persist across weeks, and how they respond to prior success or difficulty. This motivates engagement forecasting at an intermediate (e.g. weekly) time scale: it can serve as actionable decision-support signals for teachers and students, helping guide agency and self-regulation as students progress through the school year [42].

A key challenge in forecasting engagement is identifying

which aspects of learner context matter. Motivational theory suggests that persistence is shaped by beliefs about competence (often discussed as self-efficacy) as well as recent success and failure experiences [40, 51]. From an analytics perspective, some of these factors are indirectly observable in ITS logs through patterns in learner interactions. For example, difficulty and learning rate—long studied in EDM through models such as the Additive Factors Model (AFM)—may influence whether continued effort translates into progress or discouragement [15]. Rather than posing a new theory of motivation, we seek empirical insights into which of these log-derived signals are most predictive of next-week engagement targets.

We compare feature-driven predictive models against strong heuristic baselines. In addition to common algorithms (e.g., mean and median), we evaluate percentile rules adapted from adjacent health behavior domains [1, 13]. These percentile heuristics often assume sustained week-over-week improvement; whether that assumption transfers to K–12 tutoring settings is an open question [25]. We further conduct feature attribution and ablation tests across four feature groups derived from ITS logs: AFM-based learner state, recent activity trends, practice gaps, and prior achievement. By evaluating these groups, we seek to identify which contextual signals are most predictive of student effort, potentially using these insights to improve future ITS adaptivity [52]. To move towards closing this loop, we conduct a semi-structured user interview to examine how tutors can use the explainable signals when setting goals with students. We ask the following research questions:

RQ1: How well can data-driven approaches improve forecasting of next-week effort and progress?
RQ2: Do percentile heuristics drawn from other contexts (e.g., health and sports) generalize to engagement forecasting in K–12 tutoring settings?
RQ3: Which features available from ITS data are most indicative of next-week effort and progress?

This paper contributes to EDM theoretically and practically. First, we introduce week-ahead engagement forecasting in ITS as a new prediction task, focusing on calendar-level effort and progress to support practical instructional decision-making, complementing prior EDM work such as affect detection, wheel-spinning, and dropout prediction. With **RQ1** and **RQ2**, we show that statistical learning is a valid approach to the task that outperforms and better tracks student trajectory compared to existing heuristics. Second, through feature analysis in **RQ3**, we provide theoretical insights into the structure of engagement, showing that effort and progress are predicted by qualitatively different features (recent activity vs. learner-centered signals). Third, we use a user interview case study to examine how these predictive signals align with tutors' reasoning about engagement and goal setting, moving towards closing the loop between engagement forecasting and improving learning benefits.

2. RELATED WORK

We review prior work in (1) modeling effort, engagement, and disengagement in ITS; (2) foundational accounts of persistence and effort regulation; and (3) prediction tasks in EDM. Across these areas, prior work motivates why effort is a distinct and consequential target, but also highlights that personalized, data-driven forecasting of week-ahead effort and progress remains underexplored.

2.1 Effort and Engagement Modeling in ITS

ITS generate fine-grained logs of student practice that enable automated feedback and data-driven analysis [3, 47]. While ITS can improve learning outcomes, their impact is often constrained by the lack of sustained student engagement over long time horizons [29, 11]. For example, large-scale deployments can exhibit substantial skew in usage, with many learners falling below recommended practice levels [24, 22]. This motivates analytics that focus not only on *knowledge progress* but also on *effort and persistence* as first-class targets.

A substantial body of ITS research has modeled engagement-related phenomena such as off-task behavior, gaming the system, and other forms of disengagement using interaction traces (e.g., response patterns, hint usage, timing signals, delayed starting), often to trigger real-time supports including adaptive prompts or teacher alerts [5, 6, 25]. These works emphasize that observed (low) performance can reflect multiple latent causes (e.g., low effort, unproductive strategies) that can be detected by various engagement signals. Studies have also explored engagement-sensitive tutoring and motivational supports within tutors. For example, Arroyo et al. [4] proposed and evaluated effort-based tutoring that infers effort from behavior and adapts activity selection accordingly. Similarly, the AnimalWatch ITS modeled student performance based on prior effort and observed learning gains from use [8].

However, much of the literature operates at the level of problem-steps, sessions, or within-day interactions, and often evaluates detectors or interventions in-situ [25]. In contrast, many educational decisions about pacing, check-ins, and expectations occur on weekly rhythms (e.g., weekly assignments, classroom schedules, weekly reflections). Our work complements fine-grained engagement modeling by introducing and benchmarking a *week-ahead forecasting* task: predicting next-week outcomes from prior logs. This framing targets longer-term engagement rather than moment-to-moment disengagement detection.

2.2 Motivation, Self-Efficacy, and Persistence

Motivational theory provides a lens for why learners persist (or fail to persist) even when instructional support is available. Self-determination theory emphasizes the roles of competence, autonomy, and relatedness in sustaining motivation [40]. Goal-setting theory further argues that specific and challenging goals can mobilize effort, but only when goals are perceived as attainable; repeated failure can undermine perceived competence and persistence [34, 35]. This also aligns with accounts of vicious cycles in learning, where recent success and failure experiences can compound over time, shaping subsequent engagement and outcomes [51].

Empirically, motivational constructs such as self-efficacy have also been studied in learning contexts, including variability over time and relations to behavior and performance. For example, Bernacki et al. assessed self-efficacy and found that efficacy varied reliably during learning and predicted subse-

quent help-seeking and performance [10]. These perspectives motivate our week-ahead targets in two ways. First, they suggest that effort and persistence are not static traits but can fluctuate with recent experiences and context, making forecasting meaningful at intermediate time scales (e.g., week-ahead). Second, they motivate evaluating forecasting not only as an error-minimization exercise, but as a foundation for decision-support while preserving learner agency. Importantly, our contribution is not to measure motivational states directly. Instead, we forecast behavioral and progress outcomes that are plausibly influenced by such dynamics, and we interpret feature patterns in light of these theories.

2.3 Student Behavior Prediction in EDM

Prior work in EDM has extensively modeled various prediction targets, but along dimensions that differ from our engagement forecasting task. For example, much of the ITS-focused literature operates at fine temporal granularity (problem-step or within-session windows), predicting affective states such as concentration, boredom, or frustration [7, 53], and detecting behaviors such as gaming the system or off-task activity. These studies typically define engagement as an affective construct (Engaged Concentration) inferred from ITS interaction traces, and the tasks are framed as supervised binary classification aimed at real-time intervention. A related line of work is interested in modeling effort and progress in terms of detecting wheel-spinning, defined as prolonged amounts of time students spend on a problem without achieving mastery [54, 49]. Wheel-spinning work examines on progress by identifying when effort is unproductive because it fails to translate into learning gains. However, such work still focuses on opportunity-to-mastery sequences rather than calendar-time horizons. On the other hand, predictions tasks are also common in MOOC and VLE settings, where models forecast dropout or stopout using aggregated clickstream features [21, 50, 23]. However, these tasks are also generally classification problems centered on retention rather than continuous behavioral effort or learning progress within an ITS. Classic learning-rate and difficulty models (e.g., AFM, BKT, and related extensions) estimate latent learner and item parameters from practice histories, but they are typically not framed for supervised forecasting of observable engagement indicators [15, 18].

At the same time, prior work has shown that effort-related measures—particularly time-on-task—can meaningfully predict learning outcomes, even when imperfect. For example, prior EDM studies have used time-on-task as measure of engagement to predict achievement and learning gains [39]. Although time-on-task has known limitations and should not be treated as a perfect proxy for learning [32], it remains practically valuable because it is simple and directly observable by tutors and students. For example, middle school students find it easier to express goals in time-on-task minutes than in progress metrics such as skill mastery [38]. Many educational platforms and vendors also recommend minimum weekly usage targets (e.g., 30 minutes) [24], and in settings without detailed skill models, time-based measures may be the only consistently available indicator of effort (and arguably more comparable than the number of completed problems, which may vary in complexity across platforms). Therefore, defining effort as minutes practiced provides a straightforward, usable signal for week-to-week decision support.

While prior work on affect detection, wheel-spinning, and weekly dropout prediction offers methodological inspiration (e.g., supervised machine learning (ML) models) [2], these approaches do not address the same prediction target, temporal granularity, or prediction environment (ITS). In contrast, we introduce engagement forecasting as a supervised, week-ahead regression task within an ITS predicting two observable outcomes: minutes practiced (effort) and new skills mastered (progress). This framing allows engagement forecasting to consider broader student context across time rather than step-level details, which practically situates predictions as week-to-week decision-support signals for tutors and students. Furthermore, through interpretability analysis, continuous prediction of weekly student trajectory provides theoretical insights into longer-term predictive factors (e.g. student behaviors over several weeks) that prior EDM works with fine-grained measures rarely look at.

Additionally, as prior prediction works are not directly applicable in our forecasting setup, we borrow work from Adams et al. [1] in the adjacent health behavior domain to provide a reasonable, non-naive baseline against our ML models. Adams et al.'s algorithm recommends a daily step target by looking at a person's last 9 days of step counts and setting today's target to the 60th percentile. However, while this heuristic algorithm is easily implementable, its transferability across domain (health to education) and time granularity (days to weeks) is an open question, and its heuristic nature may provide insight into whether or not statistical features actually bring more predictive power.

3. DATASET

We focus on ITS as a seminal paradigm of adaptive learning systems [47]. We use a longitudinal dataset available upon request [20] from students using web-based ITS for middle school mathematics. The dataset comprises activity logs collected from 425 students over 39 weeks (October 2010 to May 2012), capturing 2.9 million student-problem interactions. Students engaged with the system for an average of 39.8 ± 14.7 minutes per week, solving an average of 24.7 ± 13.1 problems weekly. Individual student-week observations ($n = 11{,}710$) showed considerable variability in engagement, with weekly practice time ranging from zero to over 6 hours (mean = 43.2 ± 25.9 minutes per student-week). The raw dataset contains 33 columns including: anonymized student identifiers, precise timestamps, interaction durations, learning outcomes, knowledge components, opportunity counts, and problem identifiers [20]. This event-level granularity enables aggregation of individual interactions into weekly summaries. While our dataset is from 2012, it remains relevant to modern ITS usage context. In particular, fundamental tutor features in ITS remain the same (adaptive step-level guidance with feedback and hints, adaptive problem selection [46]). Data format also remains the same (e.g., through DataShop log data standards [44]).

The unit of analysis for our engagement forecasting task is individual student-weeks. ITS automatically tracks two key outcomes: (1) time spent on practice problems and (2) whether related skills were mastered. We generate weekly ground-truth data for both minute and skill targets by: (1)

aggregating time and skill data by ISO weeks (Monday–Sunday), creating identifiers such as "2011-W23;" (2) summing interaction durations per student-week to create the first prediction target *minutes practiced per week*; (3) Tagging a skill (knowledge component) as mastered when the AFM predicts proficiency >0.95 for a student-skill pair [15]; (4) Counting the number of new skills mastered per student-week to create the second prediction target *new skills mastered per week*; (5) Filtering missing and invalid values and removing outliers with Tukey's Fence ($k = 1.5$).

We describe the four categories of features we created below. The detail of the entire feature set is described in Table 1.

3.1 AFM-derived Features

We adopt the AFM to derive interpretable features with respect to the learners [15]. AFM models the probability that student i answers a problem step j correctly as:

$$\Pr(Y_{ij} = 1) = \text{logit}^{-1}\left(\theta_i + \sum_{k=1}^{K} q_{jk}\,\beta_k + \sum_{k=1}^{K} q_{jk}\,\gamma_k T_{ik}\right),$$

where Y_{ij} is student i's response on step j, θ_i represents student ability, β_k is skill easiness, γ_k is the learning rate for skill k, T_{ik} is the number of prior opportunities student i has had on skill k, and $q_{jk} \in \{0, 1\}$ indicates whether step j requires skill k [16]. AFM explains performance as three additive components: (1) baseline student ability (θ_i), (2) learning gained with additional opportunities ($\gamma_k T_{ik}$), and (3) the easiness of practiced skills (β_k). Because these terms are interpretable, we map them directly into features for engagement forecasting. Specifically, we derive *student ability* from θ_i, *student learning rate* from $\gamma_k T_{ik}$, and *student week difficulty* from the negative weighted average of β_k across skills practiced in one student-week, weighted by the number of encounters T_{ik}. We recompute AFM parameters on a rolling five-week window, extracting updated student-level ability, learning rate, and difficulty values each week.

3.2 Other Feature Groups

Beyond AFM, we engineered additional feature groups to capture complementary aspects of student engagement and performance. In general, we categorize the features into four groups for reference in analysis: (1) AFM-style learner features, (2) engagement activity, (3) practice gaps, and (4) prior achievement.

Engagement Activity. We include basic engagement measures (minutes practiced, problems solved, opportunities, and skills along with their lagged values. From these sequences, we derive changes, averages, and variability statistics. These features serve as intuitive baselines, capturing both immediate performance and historical trends.

Practice Gaps. Zeros in weekly minutes are treated as informative rather than missing values. Because 98% of students exhibit at least one temporal gap, and the average student has four gaps, we encode gap patterns through features such as recent gaps, time since the last gap, and cumulative gap count. These indicators capture disengagement episodes that could influence future engagement.

Prior Achievement. Prior achievement is a strong predictor of future learning outcomes [26]. We therefore include features capturing initial performance (quartile of early skills), stability of performance (consistency score), and improvement from early to later weeks. These serve as proxies for student preparedness and trajectory.

Table 1: Overview of feature groups and individual features.

Feature	Description
AFM	
student_ability	Baseline performance
student_learning_rate	Improvement per opportunity
student_week_difficulty	Weighted average difficulty of done skills
Engagement Activity	
minutes_current	Minutes practiced this week
minutes_lagk	Minutes practiced up to k weeks back
problems_current	Problems solved this week
problems_lagk	Problems solved up to k weeks back
opportunities_curr.	Opportunities tried this week
opportunities_lagk	Opportunities tried up to k weeks back
skills_current	Skills estimate this week
skills_lagk	Skills estimates up to k weeks back
recent_change_*	Week-to-week and average changes (minutes, problems, skills)
minutes_stats	Summary statistics (mean, std, range, iqr) of minutes practiced
problems_stats	Summary statistics (mean, sum, std) of problems solved
Practice Gaps	
has_recent_gap	Indicator of zero-minute weeks recently
weeks_since_gap	Number of weeks since the last zero
gap_count	Total number of zero-minute weeks
Prior Achievement	
start_quartile	Quartile of initial skills
consistency_score	Stability of skills relative to its mean
improvement	Change in skills from early to later weeks

4. METHODS

Our methods comprise four steps. First, we describe our ML models and their training procedure. Second, we define model evaluation metrics and the model selection process for engagement forecasting that can most generalize out of sample. Third, we describe interpretability analysis conducted on the four feature groups. Finally, we describe the protocol for carrying out our exploratory case study.

4.1 Model Training

For each student i and week t, we construct a feature vector $x_{i,t}$ (described in Table 1) from the student's interaction history up to week t. We train separate models to predict next-week outcomes:

$$y_{i,t+1}^{(\text{min})} = \text{minutes practiced in week } t+1$$
$$y_{i,t+1}^{(\text{skill})} = \text{new skills mastered in week } t+1.$$

Thus, each sample is a one-step forecast, and sequential forecasts across weeks yield predicted engagement trajectories.

We benchmark fifteen predictors spanning heuristic and supervised approaches. As simple temporal baselines, we im-

plement: (1) last week's value carried forward; (2) median across all past values for the same student; (3) median across non-zero past values; (4) mean across all past values; and (5) mean across non-zero past values. The "non-zero" variants follow prior design by Adams et al. [1] and potentially avoid ambiguity: a zero entry may indicate either disengagement or simple absence. These baselines establish a lower bound for performance with naive statistical heuristics. We also adapt three percentile-based heuristics from prior work in health behavior change [1, 13]. These set the target at the (6) 50th, (7) 60th, or (8) 70th percentile of the most recent nine non-missing values, enabling us to examine whether the original 60th-percentile assumption holds when transferred from health to education. Because the original algorithm requires sufficient history, we design a staged initialization procedure. In week 1, the prediction is set to the dataset average. In week 2, the prediction equals the student's week-1 performance plus the gap between the dataset's 50th and 60th percentile in week 1. From week 3 onward, until sufficient history is available, the prediction is updated as 60% of the way toward the highest of the student's previous values. This approximates "somewhat above average" performance under limited history, aligning with engagement theory [48].

Beyond these eight baselines, we benchmark seven supervised learning methods that leverage the full engineered feature set: (9) Random Forest (RF), (10) XGBoost (XGB), (11) Ridge regression, (12) LASSO regression, (13) linear regression (LR), (14) multilayer perceptron (MLP), and (15) long short-term memory (LSTM) network. This suite of models was chosen to span a spectrum of inductive biases—linear vs. nonlinear, tree-based vs. neural, memoryless vs. sequential—thereby probing which paradigms best forecast student engagement. All implementations use standard off-the-shelf libraries to ensure reproducibility, and the full pipeline is available in a public repository[1].

4.2 Model Evaluation

We evaluate generalization to new students by using student-level splits. We allocate 70% of students for model development and hold out the remaining 30% of students for final testing, ensuring no student appears in both sets. Within the 70% development set, we conduct 5-fold time-series split for model selection. For each student, predictions are produced in chronological order, always using information available up to the current week.

We report MAE for interpretability. We also report relative error reduction:

$$\Delta_{\%} = \frac{\text{MAE}_{\text{compared model}} - \text{MAE}_{\text{model}}}{\text{MAE}_{\text{compared model}}} \times 100\%.$$

To test significance, we estimate 95% confidence intervals via paired bootstrap resampling over students ($B = 10{,}000$).

To answer **RQ1**, we evaluate models across four settings for each model hyperparameter and five history window lengths (the number of past weeks a model can see) to (1) investigate how well ML approaches forecast engagement, and (2) provide empirical analysis of what configurations and model families affect the performance gains of ML models. We then select the best-performing configurations for subsequent analysis. To answer **RQ2**, we compare weekly performance trends of our best ML model against the three Adams predictors on the remaining 30% holdout set. We examine whether the 60th percentile assumption aligns well with student behavior trends and analyze where data-driven models excel in cases where Adams do not generalize.

[1]https://github.com/EricQiu6/EngagementForecasting

4.3 Feature Importance Analysis

To address **RQ3**, we use feature importance measures to examine which information derived from ITS logs is most indicative of future engagement levels. For linear models (Ridge, LASSO, Linear Regression), importance is the absolute value of coefficients, indicating how strongly each feature relates to the target [28, 45, 43]. For tree ensembles (Random Forest, XGBoost), importance comes from impurity-based criteria, i.e., how much each feature reduces prediction error in splits [27, 17]. To make results comparable, we average importances across validation folds and normalize them to sum to one. We then group features into the four categories and compute average weights for each group. This shows whether predictive gains come mainly from AFM-style learner parameters, engagement activity, practice gaps, or prior achievement.

On top of attribution, we conduct ablation study on the models to test whether feature groups are *necessary*. For each group, we remove it and retrain the models, holding all other conditions fixed. We then compare MAE to the full model, estimating contribution of the excluded group. Bootstrap resampling across students provides confidence intervals and significance tests for these error differences.

4.4 Studying Tutors' Use Of Signals

We conducted a semi-structured qualitative user interview case study to outline how the model signals we investigated (e.g. learning rate, difficulty, consistency) may inform real-world tutoring. The study protocol was approved by our Institutional Review Board (IRB), and all participants provided informed consent and received compensation for their time. Participants (N = 8 college students doing part-time K–12 math tutoring online) interacted with a goal-setting dashboard across four scenarios. This sample size aligns with comparable educational design research [12] and was guided by qualitative saturation, as few new themes emerged in later interviews. Each scenario varied by goal type (minutes practiced vs. skills mastered) and alignment with tutor intuition (intuitive vs. counter-intuitive) to examine how tutors interpret and respond to explainable features that may conflict with the tutors' initial judgement. The dashboard displayed recent goal cycles, achieved vs. target progress, and a system recommendation panel with three settings: (1) no recommendation, (2) recommendation of a goal value only, and (3) recommendation with a justifying explanation that cites relevant features extracted from our statistical models. We designed this three-variants setup to respectively (1) capture each tutor's baseline judgment without any recommendation influence, (2) measure how a recommended value alone shifts tutor decisions, and (3) isolate the incremental effect of the explainable features beyond the system recommendation itself. Scenarios and explanation variants were presented in randomized order. See Fig. 1 for example dashboard interfaces. The study was descriptive and did not measure learning outcomes; it aimed to

Table 2: MAE summary for minutes and skills. Lower is better. Top feature-augmented model (XGBoost) vs. average of eight baselines.

Target	Baseline MAE	Top Model MAE ($\Delta_{\%}$)
Minutes/week	*10.37*	*8.05 (−22.4%)*
New skills/week	*5.21*	*3.51 (−32.6%)*

characterize tutor interpretation and decision behavior when explanations reference the interpretable features, thus clarifying whether the forecasting signals could be helpful and used in downstream tutoring scenarios. We collected qualitative think-aloud data to capture tutors' mental models and decision-making processes.

5. RESULTS

We first test how accurately feature-based models forecast next-week engagement and which settings affect these gains. Next, we check whether percentile heuristics from other domains transfer to weekly K–12 engagement by comparing Adams baselines against our best ML model. Then, we assess which feature categories drive performance by analyzing importance scores and ablation results for AFM-based features, engagement activity, practice gaps, and prior achievement. Finally, we examine the eight case study sessions to interpret our explainable features in context of downstream tutor understanding during goal setting.

5.1 RQ1: forecasting effort and progress

Performance gains over baselines. Data-driven ML models substantially outperform heuristic baselines (last value, mean/median variants, and Adams-style methods [1, 13]). On average, error falls by **22%** for minutes and **33%** for skills. Bootstrap confidence intervals on paired percentage differences exclude zero across linear, tree, and neural families, establishing that improvements are statistically significant. To better understand where these gains come from, we next compare across model family and hyperparameters.

Family-level differences. As shown in Table 3, family-level comparisons reveal only modest variation. For the **minutes** target, linear vs. tree shows a non-significant MAE gap (mean diff = 0.16; 95% CI [−0.07, 0.40]; $p = .318$; Cliff's $\delta = 0.08$). However, while linear and tree models outperform neural nets at shorter histories, this gap narrows to non-significance as history length increases past 20 weeks, consistent with neural nets' higher data requirements. For the **skills** target, all pairwise comparisons yield negligible differences (e.g., linear–tree mean diff = 0.02, CI [−0.13, 0.16], $p = .362$, $\Delta_{\%} = 0.03$), with effect sizes $|\delta| \leq 0.07$. Thus, model family matters somewhat for minutes under short histories, but practical differences vanish for skills.

Hyperparameter sensitivity. Within each family, performance varies little across hyperparameter settings. Friedman tests blocking by window×fold show low and nonsignificant concordance (Kendall's $W < .05$), with no configuration emerging as a consistent winner. Accuracy is robust to parameter choice; most reasonable configurations yield errors within a narrow range.

Table 3: Pairwise family comparisons by target (MAE and $\Delta_{\%}$ between the row model and the comparator). Lower is better. **Bold** marks significance. * $p < .05$; ** $p < .01$

(a) Skills

Family	Models	MAE	vs.Tree	vs.Linear	vs.Neural
Linear	LR,Ridge,LASSO	3.60	0.02	–	-0.11
Tree	RF, XGB	3.50	–	-0.02	-0.13
Neural	MLP, LSTM	3.68	0.13	0.11	–

(b) Minutes

Family	Models	MAE	vs.Tree	vs.Linear	vs.Neural
Linear	LR,LASSO,Ridge	9.04	0.16	–	**-1.72****
Tree	RF, XGB	8.76	–	-0.16	**-1.88****
Neural	MLP, LSTM	9.46	**1.88****	**1.72****	–

Takeaway. Data-driven models greatly outperform baselines by 22–33% (Table 2). However, the decisive factor for performance gains is not which family or hyperparameter is chosen. Linear, tree, and neural families all reach similar accuracy given sufficient history, and hyperparameters rarely alter the outcome. On practical grounds, we adopt **XGBoost** as a default model: it performs best empirically, trains efficiently, and provides directly interpretable attributions for downstream analyses.

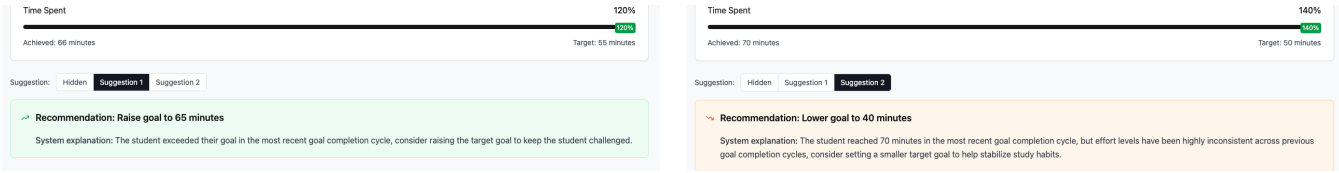
5.2 RQ2: do heuristic rules generalize

We evaluate Adams-style percentile heuristics [1, 13] against our chosen feature-based predictor XGBoost. These heuristics predict engagement through a fixed percentile (e.g., 50th, 60th, 70th) of past performance. The >50th-percentile rules assume *continuous improvement*: if learners repeatedly meet their targets, the prediction will continue to increase.

Fig. 2 shows that this assumption does not hold in our setting. Effort (minutes practiced per week) engagement stabilizes after an early surge, and progress (new skills mastered per week) rises initially but then declines mid-semester. As a result, Adams predictors systematically overshoot: P60 and P70 in particular perform 4–33% worse than P50 across targets, and averaging across percentiles further degrades accuracy. Table 4 shows full performance results.

XGBoost reduces MAE by 20–30% relative to Adams-P50, and by larger margins relative to Adams-P60 and P70 (Table 4). Still, its performance in the earliest weeks is weaker, and for the skills target, it tends to underpredict systematically. The spread bands in Fig. 2 help explain this pattern: the dataset exhibits far greater instability early in the semester. This higher variance reduces the stability of any predictor and plausibly explains why XGBoost lags in early weeks and underestimates peaks in skills. Once trajectories stabilize starting week 9, XGBoost outperforms Adams predictors by even larger margins and tracks both effort and progress levels much more closely.

Takeaway. Percentile-based heuristics from other domains do not transfer cleanly to weekly K–12 practice. Their built-in continuous improvement assumption becomes inaccurate once growth slows, leading to systematic overprediction. A feature-based model like XGBoost not only avoids this pitfall



(a) (b)

Figure 1: (a) **Intuitive goal-setting scenario (minutes practiced).** Example of an intuitive case that raises goal because student exceeded prior week's goal. (b) **Counter-intuitive goal-setting scenario (minutes practiced).** Example of a counter-intuitive case that lowers goal even though the student exceeded prior week's goal, citing the consistency feature as reason.

but adapts better to shifting regimes in the data, providing substantially more accurate predictions given sufficient data.

5.3 RQ3: which features are most indicative

The top–k feature plots (Fig. 3) show two patterns, with calibrated strength. For **effort** (minutes practiced), the top features are primarily driven by *recent engagement activity*: *minutes mean*, *current minutes per week*, and a short lag *target lag 4* take three of the top four slots; additional lag features also appear in the top–11 (*minutes per week lag 16*, *target lag 16*, *total opportunities lag 16*). However, stability across model families is low: linear and tree models rank features differently (Kendall's $\tau \approx 0.28$), and category-level agreement is nearly zero ($\tau < 0.01$). In short, effort is driven mainly by *recent engagement activity*, with smaller contributions from learner modeling.

For **progress** (skills mastered), the key features are *learner-centered* indicators: *current student ability*, *performance consistency score*, and *current student week difficulty* occupy three of the top four positions. Crucially, this ordering is more stable across model families: per–feature consistency is moderate ($\tau \approx 0.37$) and the category–level consistency is strong ($\tau \approx 0.67$). We therefore conclude that progress engagement is shaped mainly by *learner-centered signals* that are robust to choice of model family, whereas effort reflects a recent-activity-driven but family-sensitive mixture of engagement activity and learner modeling.

Ablation tests (Table 5) confirm these patterns. For **minutes**, removing AFM, practice gaps, or prior achievement features causes only small losses ($< 1\%$), while recent engagement activity captures nearly all accuracy (differences versus the full set are $< 0.3\%$). For **skills**, every group contributes: dropping engagement activity features yields the largest loss ($> 1\%$), but AFM, gaps, and prior achievement features also provide meaningful differences (0.6–0.8% each). Taken together, these results align with conclusions drawn from the feature rankings: effort forecasts are recent-activity-led, while skills forecasts rely more on learner-centered signals.

5.4 Case Study Results

We examined how eight tutors interacted with system generated recommendations and accompanying explanations that referenced distinct feature types identified in RQ3. Two independent coders conducted thematic analysis of the think-aloud data. We describe our derived patterns below.

First, tutors consistently preferred rationales backed by multi-week historical data. One participant noted, "Past data… tendency… really convinces me." This aligns with our statistically driven prediction approach, suggesting that, beyond the prediction values, the derived explainable features themselves may be valuable in downstream settings to assist in tutoring decision-making.

Second, tutors' mental models diverged by target type in a way that mirrors our prediction target split. For effort (minutes), tutors emphasized routine formation and sustainable baselines, consistent with recency-driven engagement dynamics (e.g., lags, recent change, and variability). For progress (skills), tutors more often reasoned about learner and content properties (e.g., ability, consistency, and difficulty), which matches our feature-importance pattern where AFM-derived ability and week difficulty are central.

Third, tutors expressed skepticism toward difficulty and ability claims unless the definition was clear. In addition, a few tutors raised concerns about sensitivity around student-facing explanations. For example, one participant remarked, "If the words say… 'low learning rate', I don't think that's a good thing to show kids." These critiques reinforce a key design implication for deploying model-derived signals as explainable evidence: even when predictions may be accurate, their impact depends on careful communication that preserves learner agency and avoids discouragement.

Lastly, when qualifying tutor acceptance and trust, we find that participants generally prefer to take final control of the specific goal value. However, across the 2x2 scenario split by goal type (minutes practiced vs. skills mastered) and intuition (intuitive vs. counter-intuitive), all tutors are most influenced by the third explanation variant that provided additional feature-backed evidence. In particular, one response characterized the tutors' trust: "if the system… takes the history of students, and it suggests based on that history, then I think I would be totally okay to use the suggestion of AI." Together, these findings show that while tutors preferred to retain final say, they trusted the explainable features, finding the system helpful in designing plans with their student. These feedback suggest that explainable signals may have the potential to aid decision-making in human-AI tutoring.

Overall, the tutor sessions help interpret our modeling results: the signals that are most predictive (recent history for effort; learner ability and difficulty for progress) are also the signals tutors find most helpful. This alignment between the signal's predictive power and tutor reasoning suggests that engagement forecasting offers a foundation for utilizing ex-

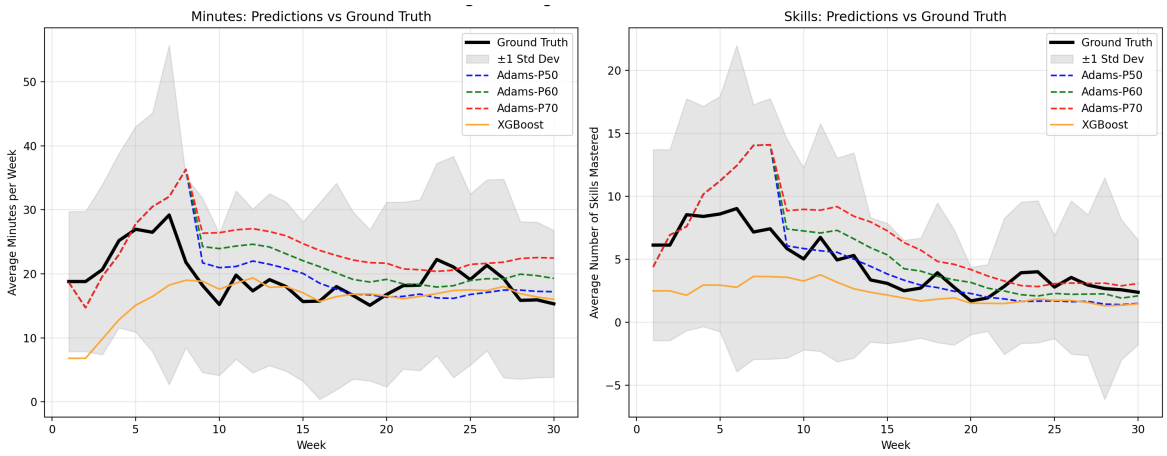


Figure 2: Average minutes and skills per week across all students plotted against the predictions of different models, also averaged across students (XGBoost, 50th percentile, 60th percentile, and 70th percentile variants of Adams predictor). Grey band is the standard deviation spread of the ground truth data.

Table 4: Performance comparison of XGBoost vs. Adams variants (P50, P60, P70) for minutes and skills prediction. MAE (lower is better). $\Delta_{\%}$ values are relative to Adams-P50 and to the average of Adams variants.

(a) Minutes prediction

Model	Final Week			Entire Sequence			Starting Week 9		
	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams
Adams-P50	10.1905	0.0%	-5.2%	10.0119	0.0%	-4.6%	9.5142	0.0%	-7.6%
Adams-P60	10.6971	+5.0%	0.0%	10.4442	+4.3%	0.0%	10.1037	+6.2%	0.0%
Adams-P70	11.8806	+16.6%	+11.1%	11.0962	+10.8%	+6.2%	10.9928	+15.5%	+8.8%
XGBoost	**7.8525**	**-22.9%**	**-26.6%**	**8.7210**	**-12.9%**	**-16.5%**	**7.2946**	**-23.3%**	**-27.8%**
Avg Adams	10.9227	+7.2%	0.0%	10.5174	+5.0%	0.0%	10.2036	+7.2%	0.0%

(b) Skills prediction

Model	Final Week			Entire Sequence			Starting Week 9		
	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams	MAE	$\Delta_{\%}$ vs P50	$\Delta_{\%}$ vs Avg Adams
Adams-P50	2.7255	0.0%	-13.2%	4.4260	0.0%	-10.3%	3.4020	0.0%	-15.0%
Adams-P60	3.1333	+15.0%	+0.5%	4.7458	+7.2%	+0.9%	3.8380	+12.8%	+0.3%
Adams-P70	3.5843	+31.6%	+15.9%	5.2541	+18.7%	+11.5%	4.5312	+33.2%	+18.3%
XGBoost	**2.2059**	**-19.1%**	**-31.5%**	**3.5099**	**-20.7%**	**-22.4%**	**2.5698**	**-24.5%**	**-32.6%**
Avg Adams	3.1477	+15.5%	0.0%	4.8086	+8.6%	0.0%	3.9237	+15.3%	0.0%

plainable contextual information to support timely instructional decisions, which may in turn lead to more calibrated and consistent engagement plans for students.

6. DISCUSSION

The present study is motivated by a persistent challenge in ITS: students often do not practice consistently enough to fully realize the system's potential learning gains, even within scheduled classroom time [25, 22]. While EDM has made major progress in modeling knowledge and performance, comparatively less work has treated effort across weeks as primary analytic targets [21, 37]. Our contribution is formalizing and benchmarking *engagement forecasting* as a novel supervised learning task over ITS log data. This empirical contribution (forecasting and feature analysis) makes patterns of sustained engagement visible and explainable, supporting its potential as a decision-support tool for teachers and students to construct more calibrated and personalized plans for effective and consistent engagement.

The remainder of this section is organized by research question. For each RQ, we recap key results, then interpret its significance in the context of prior work, draw implications for EDM and educational practice, and outline directions for future work. We conclude with analysis of our case study to connect engagement forecasting to real-world settings.

6.1 Forecasting suggests practical viability

Answering **RQ1**, feature-driven models achieve sizeable reductions in forecasting error—approximately 22% for minutes practiced and 33% for new skills mastered—relative to multiple heuristic baselines [1]. Notably, performance is similar across linear and tree-based families, and hyperparameter differences are small once sufficient history is available. This convergence suggests that, for engagement forecasting, data coverage and feature design may matter more than selecting a particular model class. Practically, this model insensitivity is encouraging: systems can deploy relatively standard models (e.g., gradient-boosted trees or regularized

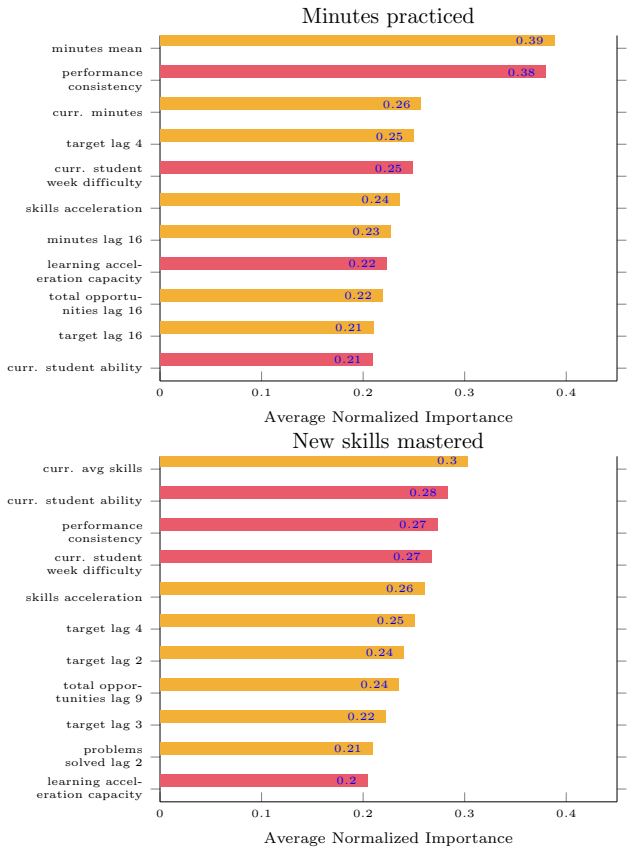


Figure 3: Feature-importance rankings. Red: learner-centered; orange: recent activity. Top: minutes; Bottom: skills.

regressions) and still obtain large improvements over heuristics, lowering barriers to adoption. Notably, although we demonstrate that our data-driven approach is successful at improving future effort and progress prediction, we do not claim that this exact model or feature stack generalizes to other contexts due to varying engagement dynamics. Instead, we offer open-source code to apply our models to local data and ensure that features are available and meaningful with respect to the class dynamics [25].

Crucially, improved forecasting accuracy alone does not imply improved learning outcomes. Rather, week-ahead predictions are best interpreted as **decision-support signals**. For example, forecasts can flag students whose engagement may deviate from recent patterns, while still leaving room for instructors and tutors to situate the forecasts using contexts the system cannot access, such as classroom constraints, instructional goals, and students' schedules. In this case, a tutor might discuss with a student any upcoming events that would get in the way of normal engagement. Such interaction helps refine engagement expectations and makes predictions more useful for week-to-week support.

6.2 Transferred heuristics do not match

Answering **RQ2**, percentile heuristics adapted from adjacent health behavior domains [1, 13] do not generalize cleanly to K–12 weekly trajectories. Percentile rules that target above-median performance systematically overshoot, with P60 and P70 performing worse than P50 across targets, and our feature-driven model outperforming P50 by 20-30%.

Table 5: Ablation results. MAE (lower is better) and relative change $\Delta_{\%}$ vs. *Full*. Bold $\Delta_{\%}$ indicates bootstrap CI on the paired difference excludes 0 (significant).

Condition	Skills		Minutes	
	MAE	$\Delta_{\%}$	MAE	$\Delta_{\%}$
Full (reference)	4.1904	0.00%	9.3067	0.00%
Base features only	4.2151	**0.59%**	9.3226	**0.17%**
Base + activity	4.2355	**1.08%**	9.3082	0.02%
Base + gaps	4.1996	**0.22%**	9.3125	0.06%
Base + prior	4.2173	**0.64%**	9.3188	0.13%
Base + AFM	4.2129	**0.54%**	9.3286	**0.24%**
All except activity	4.2300	**0.95%**	9.3088	0.02%
All except gaps	4.2141	**0.57%**	9.3155	**0.09%**
All except prior	4.2239	**0.80%**	9.3111	0.05%
All except AFM	4.2180	**0.66%**	9.3122	**0.06%**

Our analysis shows that such percentile heuristics does not reflect typical ITS usage patterns over a school year. Factors such as limited instructional time, schedule disruptions, and shifting curricular focus mean that consistently exceeding past performance is neither realistic nor always desirable [25]. This observation seems consistent with prior work [25, 22] showing large variations in how students use available time—the limited ability for a single heuristic to model all students may in part explain its lower prediction accuracy. In contrast, feature-based forecasting is more flexible: it can represent plateauing, regression, and regime shifts as a function of learner model (AFM), recent activity, inactivity patterns, and prior achievement. This flexibility is likely necessary for realistic decision support in classrooms with limited time and schedule disruptions.

6.3 Effort and progress have distinct drivers

Answering **RQ3**, our interpretability analyses reveal a consistent distinction between the two targets. **Effort** (minutes practiced) is driven by recent engagement activity, while **progress** (new skills mastered) depends more strongly on learner-centered signals such as AFM-derived ability and difficulty, as well as consistency. AFM features may not always be available in adaptive learning systems [19, 3]. Still, simplified measures of content difficulty or student ability estimated from past students' pass rates could still be used in other models. Alternatively, our ablation study suggests these features could also be removed while retaining viable model performance. In addition, data-driven modeling is weaker in the starting weeks for both targets. Nevertheless, since results show that XGBoost widens the performance gap against baselines starting week 9, one mitigation strategy may be to employ the same initialization procedure in the starting weeks as we constructed the Adams predictors (see Section 4.1) [1, 48].

The divergence between the sets of predictive signals for effort vs. progress seems theoretically counterintuitive, since learning progress ultimately arises through effort. While investigating a satisfactory explanation is beyond the scope of

this work, we hypothesize that the definition of progress in terms of weekly skills mastered may be too coarse-grained. That is, the progress-effort relationship may be obscured by discrete skill mastery, whereas classical measurements of proficiency (e.g., AFM, BKT) operate at a step-level [15, 18]. Nevertheless, defining progress as skills mastered is practically valuable as it is more aligned with this work's intention to view predictions as week-to-week decision-support signals for tutors and students. Furthermore, the separation of targets is useful for practical downstream use cases. By comparing feature groups, we aim to isolate which signals most predict student persistence and to potentially translate these insights into improved ITS adaptivity—for example, adjusting difficulty in mastery-based problem selection receptive to student motivation and effort [52]. These signals may also provide explanatory insights to learner and teacher feedback in downstream applications. We expand on this in the next case study section.

6.4 Implications for Explainable Interfaces

Across the case study, three observations were most consistent: (1) when judging recommendations, tutors placed high trust in system explanations backed by the explainable features when their definition and data source (weeks of personalized student data) were clear; (2) tutors reasoned differently about effort vs. progress in ways that mirror our target-specific feature drivers; and (3) tutors identified sensitivity concerns of learner-modeling claims (e.g., difficulty and ability) when recommendation becomes student-facing.

Together, these findings suggest that the top features extracted in RQ3 (e.g. ability, difficulty, consistency) are not only predictive inputs but also candidate explanatory evidence. We identify feature groups that map naturally onto the kinds of reasons tutors use, while also revealing that practical explainability may break down if the sources of the features are unclear (i.e. tutor confusion about definition of ability and difficulty). These findings suggest that grounded and clearly defined features may be practically compatible with existing tutor reasoning workflows, situating feature analysis as more than a post-hoc interpretability exercise.

While we find that the features themselves may provide helpful explanatory power to decision making in downstream contexts such as goal setting, the direct usage of the engagement prediction values in these applications should be further scrutinized. For example, accurate week-ahead predictions might not determine appropriate goals. In particular, while a model may predict declining engagement accurately, it is not clear whether using declining value as goal recommendation is optimal for student motivation. One promising direction of work could define and test explicit forecast-to-recommendation mappings that balance challenge and motivation while incorporating pedagogical constraints (e.g., schedule disruptions, shifting curricular focus, and teacher-configured limits). This would close the loop from prediction to intervention and enable in-situ evaluation of downstream outcomes, including tutor adoption, student acceptance, and learning gains.

6.5 Limitations and Future Work

A few limitations bound the generalizability of our findings. The study is restricted by a single ITS, subject domain, as well as platform-specific logging; some inputs underpinning our features (e.g., opportunity counters, fine-grained timestamps, skill tags for AFM) are not universal and may require proxies or simplified learner-state estimates when absent. External validation should therefore include cross-site replication and leave-site-out testing. In addition, we do not distinguish between classes that assign homework and those that do not, even though these contexts differ in how student agency and self-regulated learning operate [42, 9]. Future work could explicitly examine how forecasting performance and interpretation differ across instructional contexts. Finally, an open design question is how to communicate forecasts effectively. Prior research on open learner models and explainable learning analytics suggests that transparent representations of learner state can support self-regulation [14, 30]. Translating forecasts into actionable feedback without discouraging learners requires careful design and communication. While we motivate towards this direction with our case study, such work is beyond the scope of this paper and may be a valuable next step.

Several open questions about the prediction task itself also merit further investigation. Our contribution is the formulation and empirical validation of week-ahead engagement forecasting using standard ML methods. We leave algorithmic innovations on this prediction task for future works. In particular, further insights into which features are predictive and how accurate student engagement can be modeled may be valuable. Relatedly, both of our targets—minutes practiced and new skills mastered—are useful but imperfect proxies for effort and learning progress [32]. Alternative proxies—including problem attempts, session frequency, and active-vs.-idle time—were not directly compared in this work. As prior research show that students naturally articulate goals in both problem-count and time-duration terms [38], a comparative study of proxy choice as the forecasting target is a natural next step.

Looking forward, engagement forecasting may support a wider range of applications than goal setting alone. Potential uses include instructional pacing and estimating time-to-mastery under revised content—contexts where week-by-week forecasts could inform planning even without direct student-facing interventions. Evaluating such usages, and studying how forecasts interact with teacher-student decision-making, remains an important area for future research.

7. CONCLUSION

Prior work in EDM has introduced statistical learning methods to predict important targets such as learning, affect, momentary disengagement, performance, and dropout. However, comparatively little work has focused on explainable forecasting of student effort and progress at a weekly timescale, aligned with practical instructional decision-making related to practice goal setting. In this paper, we introduce engagement forecasting as a novel prediction task in ITS and benchmark models for forecasting two observable outcomes: minutes practiced per week (effort) and new skills mastered per week (progress).

First, as a methodological contribution to EDM, we showed that statistical learning models are well suited to this task, yielding 20–30% reductions in forecasting error relative to

heuristic baselines transferred from other behavioral domains. Our results further demonstrate that data-driven models better capture non-monotonic patterns common in real classroom distributions. Second, through interpretability analysis, we provided theoretical insight into the structure of engagement, showing that effort and progress are driven by qualitatively different features. Future effort is driven by recent engagement activity, whereas future progress depends more on learner-centered signals such as ability, difficulty, and consistency. These findings contribute insights into the features that learners need to consider to effectively calibrate their effort, a dimension rarely supported in adaptive learning systems. Third, through a semi-structured tutor interview, we examined how these predictive signals are interpreted in downstream goal-setting contexts. We found that tutors reason differently about effort vs. progress in ways that align with our findings, suggesting that the signals identified through engagement forecasting are not only predictive but also meaningful for constructing more personalized plans for effective and consistent engagement. While this study does not evaluate learning outcomes, it provides a foundation for future work linking engagement forecasts to classroom learning benefits, given prior evidence that time-on-task and related effort measures can be predictive of learning outcomes in tutoring software [39]. This foundation enables future work that explicitly links forecasts to goal-setting strategies and instructional planning, evaluated on recommendation acceptance, persistence, and learning outcomes.

8. ACKNOWLEDGMENTS

This research was primarily funded by the Institute of Education Sciences (IES) of the U.S. Department of Education under award #R305A220386 and was partially supported by the National Science Foundation under Award No. 2349558. This work was supported, in whole or in part, by the Bill & Melinda Gates Foundation INV-068909. The conclusions and opinions expressed in this work are those of the author(s) alone and shall not be attributed to the Foundation. Under the Foundation's grant conditions, a Creative Commons Attribution 4.0 Generic License has already been assigned to the Author Accepted Manuscript version that might arise from this submission. We also thank Joyce Gill for her valuable editorial insights.

References

[1] M. A. Adams, J. C. Hurley, M. Todd, N. Bhuiyan, C. L. Jarrett, W. J. Tucker, K. E. Hollingshead, and S. S. Angadi. Adaptive goal setting and financial incentives: a 2 × 2 factorial randomized controlled trial to increase adults' physical activity. *BMC Public Health*, 17(1):1–16, 2017.

[2] B. Albreiki, N. Zaki, and H. Alashwal. A systematic literature review of student' performance prediction using machine learning techniques. *Education Sciences*, 11(9):1–27, 2021.

[3] V. Aleven, C. Borchers, Y. Huang, T. Nagashima, B. McLaren, P. Carvalho, O. Popescu, J. Sewall, and K. Koedinger. An integrated platform for studying learning with intelligent tutoring systems: Ctat+tutorshop, 2025. arXiv:2502.10395.

[4] I. Arroyo, H. Meheranian, and B. P. Woolf. Effort-based tutoring: An empirical approach to intelligent tutoring. In *Proceedings of the 3rd International Conference on Educational Data Mining (EDM)*, pages 1–10, 2010.

[5] R. S. Baker, A. T. Corbett, and K. R. Koedinger. Detecting student misuse of intelligent tutoring systems. In *Proceedings of the 7th International Conference on Intelligent Tutoring Systems (ITS)*, pages 531–540, 2004.

[6] R. S. J. d. Baker. Modeling and understanding students' off-task behavior in intelligent tutoring systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1059–1068, 2007.

[7] R. S. J. d. Baker, S. M. Gowda, M. Wixon, J. Kalka, A. Z. Wagner, A. Salvi, V. Aleven, G. W. Kusbit, J. Ocumpaugh, and L. M. Rossi. Towards sensor-free affect detection in cognitive tutor algebra. In *Proceedings of the 5th International Conference on Educational Data Mining (EDM 2012)*, pages 126–133, Chania, Greece, 2012. International Educational Data Mining Society. Presented June 19–21, 2012.

[8] C. R. Beal, I. M. Arroyo, P. R. Cohen, and B. P. Woolf. Evaluation of animalwatch: An intelligent tutoring system for arithmetic and fractions. *Journal of Interactive Online Learning*, 9(1):64–77, 2010.

[9] H. Bembenutty. Meaningful and maladaptive homework practices: The role of self-efficacy and self-regulation. *Journal of Advanced Academics*, 22(3):448–473, 2011.

[10] M. L. Bernacki, T. J. Nokes-Malach, and V. Aleven. Examining self-efficacy during learning: Variability and relations to behavior, performance, and learning. *Metacognition and Learning*, 10:99–117, 2015.

[11] C. Borchers, A. Houk, V. Aleven, and K. R. Koedinger. Engagement and learning benefits of goal setting with rewards in human-ai tutoring. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education. AIED 2025. Lecture Notes in Computer Science*, volume 15880, pages 46–59. Springer, Cham, 2025.

[12] C. Borchers, J. Ooge, C. Peng, and V. Aleven. How learner control and explainable learning analytics about skill mastery shape student desires to finish and avoid loss in tutored practice. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK 2025, page 810–816. ACM, Mar. 2025.

[13] C. Borchers, C. Peng, Q. Lyu, P. F. Carvalho, K. R. Koedinger, and V. Aleven. Student perceptions of adaptive goal setting recommendations: A design prototyping study. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education*, pages 244–251, Cham, 2025. Springer Nature Switzerland.

[14] S. Bull and J. Kay. Student models that invite the learner in: The smili open learner modelling framework technical report 580. *I. J. Artificial Intelligence in Education*, 17, 01 2007.

[15] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis - a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems*, pages 164–175, 2006.

[16] H. Cen, K. Koedinger, and B. Junker. Comparing two irt models for conjunctive skills. In *International Conference on Intelligent Tutoring Systems*, pages 796–798, 2008.

[17] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[18] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, Dec. 1994.

[19] A. T. Corbett, K. R. Koedinger, and J. R. Anderson. Intelligent tutoring systems. In M. G. Helander, T. K. Landauer, and P. V. Prabhu, editors, *Handbook of Human-Computer Interaction*, chapter 37, pages 849–874. Elsevier Science B.V., Amsterdam, The Netherlands, 2 edition, 1997.

[20] S. C. Dang. *Exploring Behavioral Measurement Models of Learner Motivation*. Ph.d. thesis, Carnegie Mellon University, School of Computer Science, Pittsburgh, PA, USA, Feb. 26 2022. CMU–HCII–21–109.

[21] G. W. Dekker, M. Pechenizkiy, and J. M. Vleeshouwers. Predicting students drop out: A case study. In *Proceedings of the 2nd International Conference on Educational Data Mining*, pages 41–50, Cordoba, Spain, 2009. International Working Group on Educational Data Mining.

[22] T. Eames, E. Brunskill, B. Yamkovenko, K. Weatherholtz, and P. Oreopoulos. Computer-assisted learning in the real world: How khan academy influences student math learning. *Proceedings of the National Academy of Sciences*, 123(1):e2507708123–e2507708123, 2026.

[23] J. Gardner, Y. Yang, R. S. Baker, and C. Brooks. Modeling and experimental design for mooc dropout prediction: A replication perspective. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM 2019)*, pages 49–58, 2019.

[24] P. Grimaldi, K. Weatherholtz, and K. M. Hill. Estimating the causal effects of Khan Academy MAP Accelerator across demographic subgroups. In *Proceedings of the 15th International Conference on Educational Data Mining*, pages 839–846, Durham, United Kingdom, 2022. International Educational Data Mining Society.

[25] A. Gurung, J. Lin, Z. Huang, C. Borchers, R. Baker, V. Aleven, and K. Koedinger. Starting seatwork earlier as a valid measure of student engagement. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 303–316, Palermo, Italy, July 2025. International Educational Data Mining Society.

[26] J. Hattie. *Visible Learning: A Synthesis of Over 800 Meta-Analyses Relating to Achievement*. Routledge, London, UK, 2009.

[27] T. K. Ho. Random decision forests. *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, 1:278–282, 1995.

[28] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[29] L. Holt. The 5 percent problem: Online mathematics programs may benefit most the kids who need it least. *Education Next*, 24(4):26–31, apr 2024.

[30] D. Hooshyar, M. Pedaste, K. Saks, Äli Leijen, E. Bardone, and M. Wang. Open learner models in supporting self-regulated learning in higher education: A systematic literature review. *Computers & Education*, 154:103878–103878, 2020.

[31] K. R. Koedinger, J. Kim, J. Z. Jia, E. A. McLaughlin, and N. L. Bier. Learning is not a spectator sport: Doing is better than watching for learning from a mooc. In *Proceedings of the Second (2015) ACM Conference on Learning @ Scale*, L@S '15, page 111–120, New York, NY, USA, 2015. Association for Computing Machinery.

[32] V. Kovanovic, D. Gašević, S. Dawson, S. Joksimovic, and R. Baker. Does time-on-task estimation matter? implications on validity of learning analytics findings. *Journal of Learning Analytics*, 2(3):81–110, Feb. 2016.

[33] J. A. Kulik and J. D. Fletcher. Effectiveness of intelligent tutoring systems: A meta-analytic review. *Review of Educational Research*, 86(1):42–78, 2016.

[34] E. A. Locke and G. P. Latham. Building a practically useful theory of goal setting and task motivation: A 35-year Odyssey. *American Psychologist*, 57(9):705–717, Sept. 2002. PMID: 12237980.

[35] E. A. Locke and G. P. Latham. The development of goal setting theory: A half century retrospective. *Motivation Science*, 5(2):93–105, 2019.

[36] W. Ma, O. O. Adesope, J. C. Nesbit, and Q. Liu. Intelligent tutoring systems and learning outcomes: A meta-analysis. *Journal of Educational Psychology*, 106(4):901–918, 2014.

[37] T. Mu, A. Jetten, and E. Brunskill. Towards suggesting actionable interventions for wheel-spinning students. In *Proceedings of the 13th International Conference on Educational Data Mining*, pages 183–193, Online, 2020. International Educational Data Mining Society.

[38] C. Peng, C. Borchers, and V. Aleven. Designing homework support tools for middle school mathematics using intelligent tutoring systems. In *Proceedings of the 18th International Conference of the Learning Sciences (ICLS 2024)*, pages 1730–1733, Buffalo, NY, USA, 2024. International Society of the Learning Sciences.

[39] S. Ritter, A. Joshi, S. Fancsali, and T. Nixon. Predicting standardized test scores from cognitive tutor interactions. In S. K. D'Mello, R. A. Calvo, and A. Olney, editors, *Proceedings of the 6th International Conference on Educational Data Mining, Memphis, Tennessee, USA, July 6-9, 2013*, pages 169–176. International Educational Data Mining Society, 2013.

[40] R. M. Ryan and E. L. Deci. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55(1):68–78, 2000.

[41] P. Schaldenbrand, N. G. Lobczowski, J. E. Richey, S. Gupta, E. A. McLaughlin, A. Adeniran, and K. R. Koedinger. Computer-supported human mentoring for personalized and equitable math learning. In *Artificial Intelligence in Education: 22nd International Conference, AIED 2021, Utrecht, The Netherlands, June 14–18, 2021, Proceedings, Part II*, page 308–313, Berlin, Heidelberg, 2021. Springer-Verlag.

[42] D. H. Schunk. Goal setting and self-efficacy during self-regulated learning. *Educational Psychologist*, 25(1):71–86, 1990.

[43] G. A. Seber and A. J. Lee. *Linear Regression Analysis*. Wiley, 2003.

[44] J. Stamper, K. Koedinger, R. S. J. d. Baker, A. Skogsholm, B. Leber, J. Rankin, and S. Demi. Pslc datashop: A data analysis service for the learning science community. In V. Aleven, J. Kay, and J. Mostow, editors, *Intelligent Tutoring Systems*, pages 455–455, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[45] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[46] K. VanLehn. The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education*, 16(3):227–265, 2006.

[47] K. VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational psychologist*, 46(4):197–221, 2011.

[48] L. S. Vygotsky. Interaction between learning and development. In M. Cole, V. John-Steiner, S. Scribner, and E. Souberman, editors, *Mind in Society: Development of Higher Psychological Processes*, pages 79–91. Harvard University Press, Cambridge, MA, 1978.

[49] H. Wan and J. B. Beck. Considering the influence of prerequisite performance on wheel spinning. In *Proceedings of the 8th International Conference on Educational Data Mining*, pages 129–135, Madrid, Spain, 2015. International Educational Data Mining Society.

[50] H. Wan, J. Ding, X. Gao, and D. E. Pritchard. Dropout prediction in MOOCs using learners' study habits features. In *Proceedings of the 10th International Conference on Educational Data Mining*, pages 408–409, 2017.

[51] K. Wäschle, A. Allgaier, A. Lachner, S. Fink, and M. Nückles. Procrastination and self-efficacy: Tracing vicious and virtuous circles in self-regulated learning. *Learning and Instruction*, 29:103–114, 02 2014.

[52] M. Xia, R. Schmucker, C. Borchers, and V. Aleven. Optimizing mastery learning by fast-forwarding over-practice steps. In *Two Decades of TEL. From Lessons Learnt to Challenges Ahead: 20th European Conference on Technology Enhanced Learning, EC-TEL 2025, Newcastle upon Tyne and Durham, UK, September 15–19, 2025, Proceedings, Part I*, page 549–563, Berlin, Heidelberg, 2025. Springer-Verlag.

[53] A. F. Zambrano, R. S. Baker, S. Baral, N. T. Heffernan, and A. Lan. From reaction to anticipation: Predicting future affect. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 566–574, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[54] C. Zhang, Y. Huang, J. Wang, D. Lu, W. Fang, J. C. Stamper, S. E. Fancsali, K. Holstein, and V. Aleven. Early detection of wheel spinning: Comparison across tutors, models, features, and operationalizations. In *Proceedings of the 12th International Conference on Educational Data Mining*, pages 468–473, Montréal, QC, Canada, 2019. International Educational Data Mining Society.

Improving Hybrid Human-AI Tutoring by Differentiating Human Tutor Roles Based on Student Needs

Ashish Gurung
Carnegie Mellon University
agurung@cmu.edu

Ge Gao
Stanford University
gegao@stanford.edu

Jordan Gutterman
Carnegie Mellon University
jgutterm@andrew.cmu.edu

Danielle R. Thomas
Carnegie Mellon University
drthomas@cmu.edu

Shivang Gupta
Carnegie Mellon University
shivang@cmu.edu

Lee Branstetter
Carnegie Mellon University
branstet@andrew.cmu.edu

Emma Brunskill
Stanford University
ebrun@cs.stanford.edu

Vincent Aleven
Carnegie Mellon University
aleven@cs.cmu.edu

Kenneth R. Koedinger
Carnegie Mellon University
koedinger@cmu.edu

ABSTRACT

Hybrid human-AI tutoring, where technology and humans jointly facilitate student learning, can be more beneficial than AI-only tutoring. However, preliminary evidence suggests that lower-performing students derive greater benefit from human-AI tutoring than higher-performing students. As such, this study evaluates whether a differentiated tutoring policy can effectively support both groups: human tutors initiate support for lower-performing students, while higher-performing students receive reactive, on-demand support. Using their within-grade median state test scores, we assigned 635 students (grades $5-8$) to receive proactive ($< median$) or reactive ($\geq median$) tutoring. Using a difference-in-discontinuities (DiDC) design, we compare outcomes across two time periods: fall (AI-only tutoring for all) and spring (proactive-reactive human-AI tutoring). This quasi-experimental design isolates the effects of proactive-reactive tutoring approaches by comparing the discontinuity in spring outcomes to the fall baseline, where no such discontinuity existed. Using data around the cutoff (Imbens-Kalyanaraman criterion), we find significant overall improvements from human-AI tutoring compared to AI-only baseline: 25% increase in time on task, 36% in skill proficiency, and 61% in academic growth (standardized MAP test). Between proactive and reactive tutoring, we find comparable improvements in time-on-task and skill proficiency (i.e., no significant difference). We also did not find a significant difference in MAP growth at the cutoff itself, however, proactive tutoring, on average, showed marginally higher MAP growth (75%, $p = .065$) than reactive tutoring, i.e., proactive tutoring was more beneficial to students farther below the cutoff and helped narrow achievement gaps. Our findings provide evidence that differentiated human-AI tutoring addresses the needs of both groups, offering a practical and cost-effective strategy for scaling hybrid instruction.

Ashish Gurung, Ge Gao, Boyuan Guo, Alex Houk, Erin Gatz, Danielle R Thomas, Shivang Gupta, Lee Branstetter, Emma Brunskill, Vincent Aleven, Kenneth Koedinger, and Jordan Gutterman. Improving Hybrid Human-AI Tutoring by Differentiating Human Tutor Roles Based on Student Needs. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 130–142. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039782

Keywords

Human-AI Tutoring, Academic Growth, Remote Tutoring, Difference-in-Discontinuities

1. INTRODUCTION

The efficacy of human tutoring is well established in educational research [43], with meta-analyses showing it to be particularly beneficial for lower-performing and socioeconomically disadvantaged students [19, 44]. However, scaling human tutoring remains a persistent challenge due to cost and logistical constraints. Automated alternatives that replicate key elements of human tutoring, such as Intelligent Tutoring Systems (ITS) and Adaptive Tutoring Systems (ATS), have also proven effective [20, 36, 56]. At the same time, Steenbergen-Hu and Cooper's [54] meta-analysis found that automated systems were less effective for lower-performing students than for higher-performing peers, suggesting that some learner needs go beyond the affordances of automated systems (e.g., accountability and motivation).

Recently, hybrid human-AI tutoring models have emerged as a potential cost-effective alternative to high-impact human tutoring [1, 11, 14, 26, 55]. This hybrid approach combines the continuous adaptive support of AI tutors with the supplemental motivational and conceptual support from human tutors [1]. Initial evaluations by Chine et al. [14] reported nearly double the learning gains among students receiving human-AI tutoring compared to a business-as-usual control group. Similar evaluations by Gurung et al. [26] found that human-AI tutoring was more beneficial for lower-performing students than their higher-performing peers, suggesting that the benefits of human support vary by performance level. One possible explanation for this differential benefit is that higher-performing students are better equipped to overcome challenges and utilize the support provided by tutoring systems [36], whereas lower-performing students may benefit from additional motivational or conceptual support beyond the affordances of a tutoring system. Findings from Holstein et al. [31] support this interpretation: strategically directing

teacher time and effort to struggling students increases overall learning gains for the entire group compared to students in a business-as-usual control.

If tutor attention in hybrid settings is especially effective for lower-performing students, then decision policies to optimize tutor time and effort are critical. Yet there is limited evidence on how effective such policies are in real classrooms or what forms they should take (e.g., should tutors proactively initiate support for struggling students or provide reactive, on-demand assistance?). Developing effective policies is particularly important in hybrid settings where a single tutor supports multiple students during a session [14, 26]. Educational data mining is well-positioned to address this gap, as it has demonstrated success with similar decision policies in peer tutoring [59, 58].

In short, there exists a research gap in studying and evaluating effective and adaptive forms of tutor-student pairing in hybrid tutoring. We address this gap by examining a proactive-reactive tutoring policy in which tutors initiate support for lower-performing students, while higher-performing students work independently and receive reactive, on-demand support. To evaluate the policy's effects on learning, we implement a Difference-in-Discontinuities design (DiDC [46]) using prior-year state test performance as the running variable.

2. BACKGROUND

This section reviews the current evidence on the effectiveness of human tutoring, AI tutors, and human-AI tutoring. We also examine how researchers and practitioners have sought to scale these interventions.

2.1 Human Tutoring

Bloom's seminal 2-sigma effect [9] highlighted the impressive learning benefits of one-on-one human tutoring (under ideal conditions) compared to traditional classroom instruction. While such strong gains remain difficult to replicate in real-world settings, efforts to implement human tutoring at scale have consistently shown positive results. For instance, several meta-analyses have reported an effect size of 0.36 SD for human tutoring [19, 44], with Nickow et al. [44] finding tutoring most effective when delivered by certified teachers (0.59 SD). Encouragingly, support from non-professional tutors also resulted in meaningful learning gains (0.21 SD).

The benefits of human tutoring are generally attributed to the personalized motivational and conceptual support provided by tutors [19, 44, 56, 9, 28]. VanLehn's meta-analysis (see [56]) provides a comprehensive review of human tutoring. He attributes tutor success to their ability to closely monitor learning processes, identify misunderstandings, and offer immediate feedback and scaffolding support. Immediate feedback helps tutors address uncertainty in student responses, correct errors, and provide clarification [23, 22]. Scaffolding, in turn, enables tutors to deliver structured support that strengthens reasoning and helps learners grasp complex concepts [17, 13]. Together, these strategies create a highly personalized learning experience tailored to the student's task-specific needs.

2.2 AI Tutors

Current human-AI tutoring research refers to automated tutoring systems as AI tutors [26, 55] to highlight their AI use while contrasting them with their human counterparts. For instance, platforms such as MATHia (formerly Cognitive Tutor [5]) and Lynnette [57] incorporate Symbolic AI and Bayesian Knowledge Tracing (BKT [16]) to estimate student mastery and personalize their math practice. Others, such as W-Pal [50] and ASSISTments [30], utilize natural language processing (NLP) to automate the grading and feedback generation processes. W-Pal uses NLP to help students improve their writing skills (essays), while ASSISTments uses NLP to recommend grades and provide feedback to student work (open-ended problems).

Relatively larger commercial platforms such as IXL [15], i-Ready [25], and ALEKS [21] also leverage AI and psychometric methods to deliver personalized, adaptive support. For example, ALEKS utilizes Knowledge Space Theory [18] to assess student ability and personalize their math practice. IXL, on the other hand, uses learning analytics methods and dashboards to provide insights into student progress [26].

Similar to human tutoring, the effectiveness of AI tutors (particularly, ITS) is well established, with several meta-analyses reporting on their effectiveness [20, 36, 56]. AI tutors have been effectively deployed across a diverse range of educational domains, including mathematics [5, 57], physics [35, 53], writing [40, 50], and programming [41, 47]. Beyond meta-analyses and randomized studies, the effectiveness of AI tutors has also been established through external evaluations. For instance, the Center for Research and Reform in Education (CRRE[1]) has certified platforms such as MATHia, ASSISTments, and IXL as demonstrating strong evidence of improving learning outcomes.

2.3 Human-AI tutoring

During human-AI tutoring, students work independently with AI tutors, while human tutors monitor progress in real time and provide appropriate support [56]. Broadly, these supports are either instructional or motivational. Instructional support addresses gaps in knowledge or misconceptions, typically through hints, explanations, or scaffolding. For example, if a student applies the formula for volume incorrectly, the tutor might ask, "When calculating the volume of a cylinder, do we use the area or the perimeter of the base?" (procedural error). Similarly, to address motivational needs and encourage productive engagement and effort, tutors may affirm student progress by celebrating patterns of success (e.g., individual or streaks).

The quality of human support can be strengthened through learning analytics and teaching-augmentation tools [4]. By surfacing insights into learner needs, these tools can support instructional judgment and pedagogical decisions. For example, Holstein et al. [31], in a teacher-AI context, demonstrated how teaching augmentation tools can effectively guide teacher attention toward students most in need of support. This targeted redirection of teacher attention led to significant improvements in post-test performance compared to business-as-usual control groups.

[1]List of math interventions with evidence of effectiveness.

2.3.1 In-person and Virtual human-AI tutoring

The human-AI tutoring model has been explored in both in-person and virtual settings [14, 7]. For example, Bhatt et al. [7] explored an in-person human-AI model with a 4:1 student-to-tutor ratio. The tutor provided high-dosage tutoring to 2 of the 4 students on alternative days, while the other 2 worked with an AI tutor. This alternating approach was explored as a more cost-effective alternative (30% cost reduction) to the continuous human tutoring model explored by Guryan et al. [26]. Encouragingly, the benefit of an alternating human-AI tutoring model on state-test performance (0.23 SD) was comparable to the high-dosage human tutoring reported by Guryan et al. (0.26 SD). Both studies were year-long randomized controlled trials that used business-as-usual conditions as the control group.

Other researchers have explored the feasibility of virtual human–AI tutoring with remote human tutors, primarily through small-scale randomized trials [48, 49] and quasi-experimental evaluations [11, 26, 55]. Early evaluations report smaller effect sizes for virtual tutoring (0.05 SD; Ready et al. [48]) compared to in-person models (0.23 SD; Bhatt et al. [7]). However, Ready et al. [48] report high variance in AI tutor use, likely due to reduced accountability and disengagement. Importantly, among students who met the recommended dosage, the effect size was comparable to in-person tutoring (0.26 SD). Provided implementation fidelity, virtual tutoring can be as effective as in-person alternatives. Quasi-experimental evaluations [11, 14, 26, 55] provide converging evidence that human-AI tutoring improves both AI tutor progress and standardized test performance.

2.3.2 Heterogeneous Treatment Effect of human-AI tutoring

In a year-long study comparing virtual human-AI tutoring with AI-only tutoring using propensity score matching, Gurung et al. [26] found that access to human tutors was associated with significant improvement in students' grade-level progress, equivalent to 0.36 grade levels by year's end. While these in-platform gains did not translate to significant aggregate effects on state test performance, lower-performing students showed meaningful improvements: students 1 SD below the mean experienced an additional 0.15 SD gain ($p < .001$) on state tests compared to AI-only tutoring. Gurung et al. also observed lower AI tutor use among lower-performing students than their higher-performing peers. Consistent with Ready et al. [48], these patterns suggest the need for additional accountability support for lower-performing students.

3. PRESENT STUDY

Building on evidence that lower-performing students benefit more from additional human tutor support (Section 2.3.2), we propose a proactive-reactive tutoring policy that matches the intensity of human support to student needs. Human tutors initiate additional motivational and conceptual support to lower-performing students and reactive, on-demand support to higher-performing students. We classify students below the within-grade median as lower-performing and those above the median as higher-performing. This approach maximizes human support for students who need it most while maintaining the autonomy of those who benefit from working independently. Throughout the paper, "proactive" and "reactive" refer to how tutors deliver support, not how students seek it. This needs-based tutoring design motivates the following research questions:

RQ1. What are the benefits of human-AI tutoring compared to AI-only tutoring?

- **1.1** To what extent does access to human-AI tutoring improve students' academic growth on standardized tests?
- **1.2** To what extent does access to human-AI tutoring improve students' engagement (time on task) and performance (skills proficiency) in the AI tutor?

For RQ 1.1 and RQ 1.2, we employ a difference-in-differences (DiD [24]) design. Using an AI-only condition as the baseline, we first estimate the overall benefit of hybrid human-AI tutoring, then separately estimate the average effects of proactive and reactive tutoring.

RQ2. Do all students benefit from a policy to assign proactive tutoring to lower-performing students and reactive tutoring to higher-performing students?

For RQ2, we compare the impact of proactive and reactive tutoring on academic growth by estimating the average treatment effect at the discontinuity and testing whether the relative benefits of proactive-reactive tutoring vary by student performance level.

4. METHOD

We use a DiDC design to estimate the effects of introducing human-AI tutoring (RQ1; DiD) and the causal effects of proactive versus reactive tutoring (RQ2; estimated at the median cutoff).

4.1 Dataset

The dataset includes assignment logs from IXL (AI tutor) and standardized test scores for 635 students in grades 5–8 at a middle school in a Mid-Atlantic U.S. state during the 2024–2025 school year. Of approximately 1,000 eligible students, 635 provided informed consent to participate. The IXL logs capture students' platform use throughout the school year, with each record detailing the practiced skill, time-on-task, and skill proficiency.

Standardized assessments include the prior year's state test (Spring 2024) and Measures of Academic Progress (MAP [29]), administered in fall (Sept 2024), winter (Dec 2024), and spring (Apr 2025). Developed by the Northwest Evaluation Association (NWEA), MAP uses a continuous vertical scale to track academic growth. Of the 635 students, prior-year state test scores were available for 557; the remaining 78 were newly enrolled and lacked prior-year records. Data collection followed IRB protocols, and participation was based on opt-in informed consent.

4.2 Adaptive Tutoring System (IXL)

IXL is an adaptive tutor that dynamically adapts to student performance at two levels: item selection (based on performance within a skill) and skill selection (based on overall skill proficiency). This approach to adaptive support

is characterized by micro-adaptive and macro-adaptive support [52]. Student proficiency is estimated using SmartScore (0—100), IXL's proprietary measure, where scores above 80 indicate proficiency and a score of 100 indicates mastery.

In terms of effectiveness, IXL is classified as a Tier 1 platform[2] ("strong evidence") [15], showing an average effect size of 0.08 SD compared to business-as-usual controls on standardized tests. State-level evaluations comparing IXL and non-IXL schools have replicated these findings with effect sizes of 0.10–0.13 SD [34, 51].

4.3 Virtual Human-AI Tutoring

Remote tutors provided tutoring support via Zoom during math practice on IXL. Tutors were graduate and undergraduate students from a U.S.-based university who received tutor training [37, 38]. Tutoring sessions were conducted twice a week during regular school hours.

At the beginning of each classwork session, students were assigned to individual Zoom breakout rooms by remote tutors, where they independently practiced math on IXL. After setup and wrap-up time, each session typically provided approximately 30 minutes of practice. On average, proactive tutors supported 4 students per session (∼7.5 minutes per student), while 1 tutor provided reactive support to all students in the reactive group (∼ 10 students; ∼1–3 minutes per student). Proactive tutors actively checked in on student progress by visiting breakout rooms, whereas reactive tutors provided on-demand support when students requested help via Zoom chat or the hand-raise feature.

Implementation fidelity was monitored at two levels: session leads monitored protocol adherence and tutoring quality in real time (session leads often served concurrently as reactive tutors), while senior tutor supervisors observed multiple sessions to ensure consistency across tutors.

[2] https://evidenceforessa.org/program/ixl-math/

4.4 Experimental Design

Traditionally, regression discontinuity (RD [33]) designs leverage cutoffs to estimate the causal effects of policies or interventions. Common examples include age thresholds for legal drinking, test-score cutoffs for scholarship eligibility, and income thresholds for access to social programs. These cutoffs are often constrained by exogenous factors such as budgetary constraints or legal precedent, rather than by theoretical or empirical evidence about where the intervention would be most beneficial.

Unlike conventional cutoffs, we use an evidence-based cutoff intended to benefit both proactive and reactive groups. Building on Gurung et al.'s [26] finding that the benefits of human-AI tutoring vary by prior performance (interaction: -0.13 SD, p = 0.03), we chose median prior performance as the cutoff. Given the expected heterogeneous effects, the choice of cutoff requires further elaboration. As illustrated in Figure 1, if the median closely approximates the optimal cutoff, we could observe an equivalence between conditions at the cutoff (i.e., no significant difference) and an interaction indicating students farther below the median experience greater benefits from proactive tutoring (Figure 1 b).

However, equivalence at the cutoff does not, by itself, indicate whether the differentiated policy produced meaningful benefits. Given these considerations, we use MAP growth measures and DiDC design [46] that combines RD [33] with difference-in-differences (DiD)[24]. By leveraging fall baseline outcomes (IXL use and MAP performance), DiDC pro-

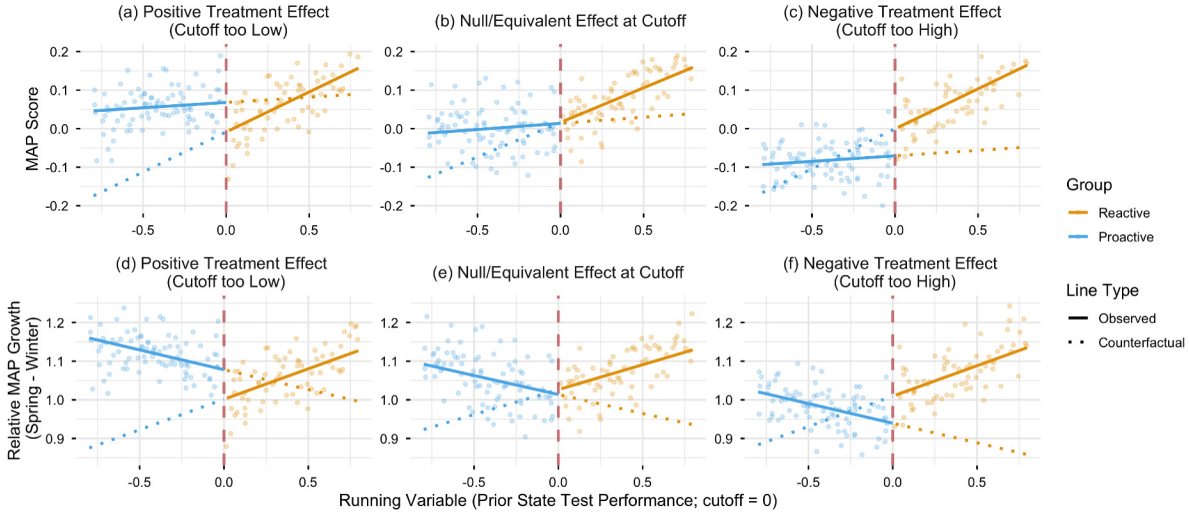


Figure 1: Simulated scenarios for RD (top panel) and DiDC (bottom panel). The top panel shows heterogeneous treatment effects in an RD design, with proactive tutoring assigned below the cutoff and reactive tutoring above it. Depending on cutoff placement, proactive tutoring may appear (a) more beneficial, (b) equivalent, or (c) less beneficial than reactive tutoring at the cutoff, with effects varying by prior performance. The bottom panel shows the corresponding DiDC estimates using the MAP growth ratio (observed vs. expected) between periods: (d) positive growth ratio when the cutoff is too low, (e) null growth ratio when it closely approximates the optimal cutoff, and (f) negative growth ratio when it is too high.

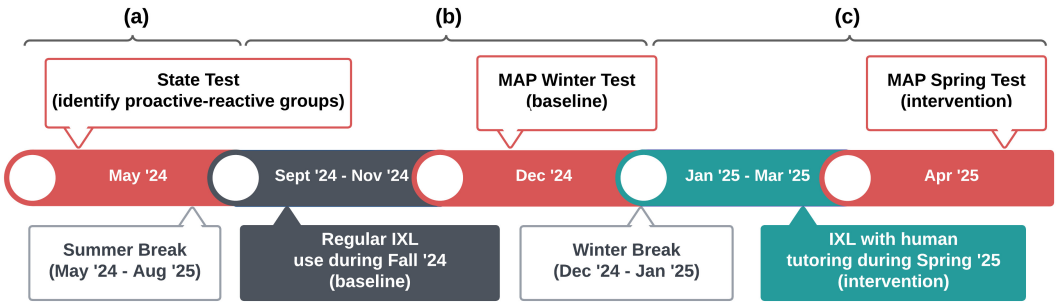


Figure 2: Experimental timeline for evaluating a proactive-reactive tutoring policy. Fall IXL data and winter MAP scores serve as the baseline, while the spring IXL data and spring MAP scores capture the effect of the intervention.

vides two complementary estimates: First, a DiD analysis compares spring human-AI tutoring to the fall AI-only baseline, establishing whether hybrid tutoring improved outcomes overall. Second, the DiDC estimator isolates the causal effect of proactive versus reactive tutoring at the cutoff by differencing out any baseline discontinuity. Figure 1 (d—f) illustrates how gains in MAP scores from winter to spring are used to estimate the impact of introducing human-AI tutoring in the spring.

4.4.1 *Experiment Timeline*

Figure 2 illustrates the experimental timeline. Students were assigned to proactive or reactive tutoring using a within-grade median cutoff in prior-year state test scores (Figure 2a). By introducing human tutoring in the spring (Figure 2 c), we can use the Fall IXL measures and winter MAP scores as pre-intervention baselines (Figure 2 b).

For RQ1, we use a within-subjects DiD [24] comparison between the baseline and intervention periods to estimate the overall and the differential benefits of proactive and reactive tutoring. We use Imbens and Kalyanaraman (IK) bandwidth [32] near the cutoff to strengthen the parallel time trends assumption by restricting the analysis to locally comparable students, making it plausible that their growth trajectories would have been similar absent the introduction of human-AI tutoring.

For RQ2, we use DiDC to estimate the treatment effect at the cutoff and examine variation in the benefits of proactive versus reactive tutoring across student performance levels, measured by MAP gain (winter to spring). By differencing out the fall baseline discontinuity, DiDC isolates the causal effect of the differentiated policy on spring academic growth.

4.5 Analysis Plan

To address our research questions, we first discuss MAP growth norms and the local comparability assumption required for RD design. We then discuss our analysis plan to examine the impact of the human-AI tutoring policy on student IXL use and MAP performance (RQ1 and RQ2). All analyses are conducted in R using *lme4*, *rdd*, and *lavaan*.

4.5.1 *Expected MAP Growth*

The MAP [29] assessment reports scores on a continuous vertical scale designed to measure academic growth over time in Rasch Units (RIT). The vertical scale allows comparisons both within and across school years. To contextualize observed gains, we use NWEA's expected growth norms, which summarize typical score changes between consecutive test administrations for the national MAP testing population. The expected growth norms presented in Table 1 are from the NWEA 2025 release (based on 13.8 million students).[3]

Table 1: NWEA 2025 expected MAP growth norms by grade. The higher fall-to-winter growth also reflects the longer interval between fall and winter ($\sim$ 3 months) than between winter and spring ($\sim$ 2 months), as spring MAP tests are generally administered early in April.

	Expected MAP Growth (RIT)	
Grade	Fall to Winter	Winter to Spring
5	6	4
6	6	4
7	4	3
8	4	3

To account for grade-level differences in typical MAP growth between consecutive test administrations, we normalize each student's observed winter to spring MAP gain by the expected gain for their grade using Equation 1. For example, a 4-point increase from winter to spring corresponds to 100% of expected growth for grades 5 and 6, but 133% of expected growth (i.e., 33% above expected) for grades 7 and 8.

$$\text{Relative MAP Growth} = \frac{\text{Observed Growth}}{\text{Expected Growth}} \quad (1)$$

4.5.2 *Bandwidths and Local Comparison*

We use the IK bandwidth selector [32] to support credible estimation of treatment effect at the cutoff. IK bandwidth helps identify the analytic sample around the cutoff that minimizes mean squared error for a local linear RD estimate. Limiting comparisons to the local neighborhood strengthens the credibility of the continuity assumption by enhancing comparability across observed characteristics and reducing differences in unobserved determinants of outcomes. The IK procedure balances the bias–variance tradeoff inherent in

[3]NWEA 2025 Norms Quick Reference (resource page)

RD designs: narrower windows reduce bias from functional-form misspecification and distant comparisons, while wider windows improve precision.

4.5.3 RQ1: Effects of Introducing Human Tutors

Impact of Human-AI tutoring on Academic Growth (RQ1.1). We estimate the effect of introducing human-AI tutoring using a within-subject comparison of the students' relative MAP growth from winter to spring (human-AI tutoring) with fall to winter (AI-only baseline). We use a multilevel mixed-effects model with random effects for grade and student within grades, and period indicators (AI-only tutoring, human-AI tutoring), as outlined in Equation 2.

$$\mathrm{Y}_i = \beta_0 + \beta_1 \,\text{human-AI tutoring}_{ig} + u_{ig} + v_g + \varepsilon, \quad (2)$$

Here, β_1 estimates the added benefit of human-AI tutoring on relative MAP growth compared to the AI-only tutoring, and β_0 represents the expected outcome under AI-only tutoring. We account for between-grade variance using random intercepts v_g, between-student variance within grades using random intercept u_{ig} and assume normality for $v_g \sim \mathcal{N}(0, \sigma_v^2)$, $u_{ig} \sim \mathcal{N}(0, \sigma_u^2)$ and $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, with v_g and u_{ig} independent of ε_i (and of each other).

We extend Equation 2 to evaluate the difference in outcomes between proactive and reactive tutoring, as outlined in Equation 3.

$$\begin{aligned}\mathrm{Y}_i = \beta_0 &+ \beta_1 \,\text{human-AI tutoring}_{ig} + \beta_2 \,\text{proactive}_{ig} \\ &+ \beta_3 \,\text{human-AI tutoring}_{ig} \times \text{proactive}_{ig} \\ &+ u_{ig} + v_g + \varepsilon,\end{aligned} \quad (3)$$

Here, β_2 estimates the baseline difference between groups in the fall (AI-only), whereas β_3 estimates the additional benefit of access to proactive human-AI tutoring.

Impact of Human-AI tutoring on IXL Engagement and Performance (RQ1.2). Beyond MAP performance, we also estimate the effect of introducing human-AI tutoring on IXL engagement (time-on-task) and performance (skill proficiency). We limit the analysis to the 10 normal school weeks preceding each MAP test, as the fall semester was longer than the spring (the spring MAP was administered early in April). We use the same model specification outlined in Equations 2 and 3 to estimate the overall impact of human-AI tutoring and compare the benefits of proactive and reactive tutoring on student engagement and performance.

4.5.4 RQ2: Do All Students Benefit From the Differentiated Tutoring Policy

We use DiDC to estimate the treatment effects of proactive and reactive tutoring at the cutoff and to examine treatment-effect heterogeneity across performance levels.

$$\begin{aligned}\mathrm{Y}_{ig} = \beta_0 &+ \beta_1 \,\text{Proactive}_{ig} + \beta_2 \,\text{PriorPerformance}_{ig} \\ &+ \beta_3 \left(\text{Proactive}_{ig} \times \text{PriorPerformance}_{ig}\right) \\ &+ v_g + \varepsilon,\end{aligned} \quad (4)$$

Here, Y_{ig} is the relative MAP growth (Equation 1) from winter to spring for student i in grade g. β_1 estimates the relative MAP growth of receiving proactive versus reactive tutoring at the cutoff, β_2 estimates the association between prior performance and relative MAP growth, and β_3 estimates the heterogeneity in the treatment effect of proactive tutoring with prior performance. We include a grade-level random intercept v_g to account for between-grade variability, assuming $v_g \sim \mathcal{N}(0, \sigma_v^2)$ and $\varepsilon_{ig} \sim \mathcal{N}(0, \sigma^2)$, with v_g independent of ε_{ig}.

5. RESULTS

The code used for the analysis is available in the anonymous GitHub repository[4].

5.1 Descriptive Statistics

The descriptive statistics for student performance are reported in Table 2. These include prior year's state test scores and the winter and spring MAP results.

Table 2: The mean and SD of student performance on the state and MAP tests.

Grade	N	State Test	Winter MAP	Spring MAP
5	167	1046 (108)	218 (13)	231 (14)
6	154	1014 (110)	226 (17)	229 (17)
7	114	984 (148)	229 (18)	232 (18)
8	122	955 (110)	231 (18)	238 (18)

5.2 Validity Tests and Bandwidth Selection

To ensure validity, we first conducted a McCrary manipulation test [39] to assess whether the running variable (state test scores) exhibited strategic sorting around the cutoff. We found no statistically significant density discontinuities at the cutoff across all four grades (all $p > .05$), providing no evidence of manipulation. We then used the IK bandwidth procedure [32] to identify the optimal range around the cutoff for credible estimation of the treatment effect at the cutoff.Table 3 reports the optimal bandwidth ranges for each grade, estimated using the running variable (prior state test performance) and winter-to-spring MAP gains. Finally, we evaluated the local comparability assumption by assessing covariate balance (ethnicity, gender, socioeconomic status) and checking continuity in baseline measures (winter MAP, fall IXL time on task, and fall IXL skill proficiency) within the selected bandwidths. Overall, the diagnostics are consistent with the assumptions required for the DiDC design.

5.3 RQ1: Human-AI vs. AI-only Tutoring

We estimate the benefit of human-AI tutoring by comparing spring MAP test performance, IXL time on task, and skill proficiency to the AI-only baseline period. We first estimate the overall effect using a within-subject comparison between

[4]http://tiny.cc/EDM26

Table 3: Sample sizes determined using the IK bandwidth selection procedure [32], with bandwidth estimated in SD.

Grade	IK Bandwidth	n	Human-AI Tutoring	
			proactive	reactive
5	0.77	78	36	42
6	0.71	75	39	36
7	0.71	57	32	25
8	0.89	77	38	39
	Total:	287 (52% of N)		

human-AI tutoring and AI-only baseline (Equation 2). We then use a DiD approach (Equation 3) to compare the differential effects of proactive and reactive tutoring. We report results for both the full sample and the subsample within the IK bandwidth. Given that the parallel trends assumption required for DiD analysis is more plausible within the IK bandwidth due to greater local comparability, we prioritize these estimates for causal interpretation.

5.3.1 *RQ1.1: Human-AI Tutoring and Academic Growth*

We fit multilevel regression models (Equation 2) with both the IK bandwidth subsample and the full sample. Student-level random intercepts produced singular fits (i.e., student-level variance estimated at 0) and were omitted. The final models include grade-level random effects and a fixed effect for human-AI tutoring (with AI-only tutoring as reference).

As illustrated in Figure 3, within the IK bandwidth, human-AI tutoring was associated with significant improvements in relative MAP growth by an additional 61% (p = .003). The full sample showed a similar but not significant trend of 26% improvement (p = .086). Additionally, both models had an interclass correlation between 0.03–0.04, indicating that only 3–4% of the variance is attributable to grade-level differences, i.e., the intervention's effect was consistent across grades.

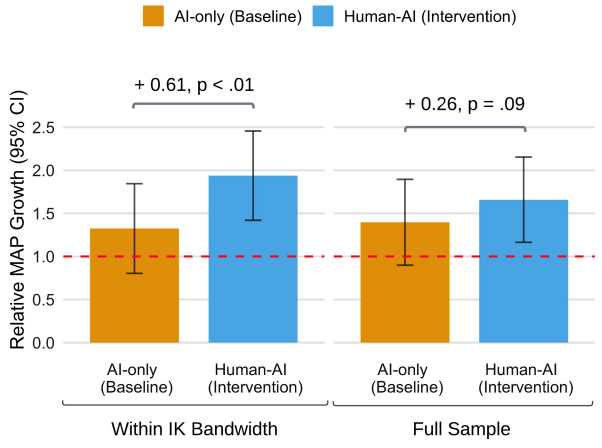


Figure 3: Within-subject improvement of relative academic gain from the introduction of human-AI tutoring. The dotted red line represents students meeting 100% of their expected growth norm on the MAP assessment.

Table 4 compares relative MAP growth between proactive and reactive tutoring groups (Equation 3), using AI-only tutoring as the baseline. Within the IK bandwidth subsample, proactive tutoring produced larger relative growth than reactive tutoring (0.75, p = .065), with proactive students achieving 2.18× expected growth compared to 1.70× for reactive students in the spring. A similar pattern was observed in the full sample, where the proactive tutoring produced larger relative growth (0.78, p = .010) than reactive tutoring.

Table 4: Comparing the relative growth between the proactive and reactive tutoring from human-AI tutoring compared to AI-only baselines.

	IK Bandwidth	Full Sample
	β	β
(Intercept)	1.46 ***	1.66 ***
Human-AI	0.24	-0.12
proactive	-0.27	-0.53 *
Human-AI×proactive	0.75$^{\bullet}$	0.78 *
ICC	0.03	0.04
N_{grade}	4	4

$^{\bullet}p < 0.1$, $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Overall, human-AI tutoring was associated with improved relative academic growth for both proactive and reactive groups (Figure 3), with the proactively tutored students experiencing larger relative growth than their reactively tutored peers (Table 4).

5.3.2 *RQ1.2: Human-AI Tutoring and IXL Use*

Impact of human-AI tutoring on IXL engagement: As illustrated in Figure 4, we find that human-AI tutoring associated with significant improvement in student time on task in both the IK bandwidth subsample (25%; +1.32 hours, p < .001) and the full sample (31%; +1.61 hours, p < .001). The models had an intraclass correlation between 0.51–0.58, reflecting strong student-level variation in engagement within grades.

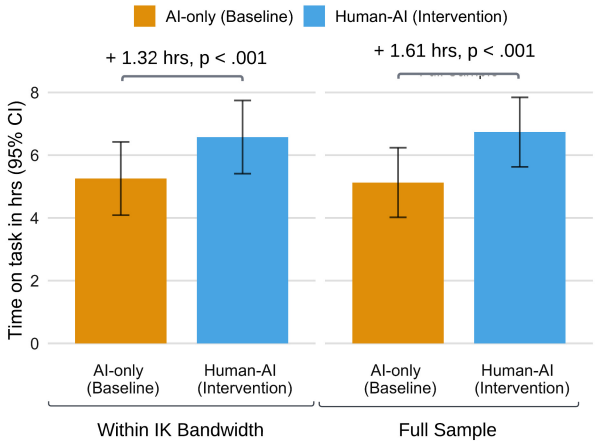


Figure 4: Within subject improvement of time on task in IXL from the introduction of human-AI tutoring.

Additional analysis revealed a significant increase in student

time on task from human-AI tutoring compared to AI-only tutoring (Table 5). However, we did not find a significant difference in time on task between proactive and reactive tutoring in either the IK bandwidth subsample or the full sample, i.e., time on task did not differ reliably between the two tutoring approaches.

Table 5: Comparing the impact of proactive and reactive human-AI tutoring on student engagement in IXL.

	IK Bandwidth	Full Sample
	β	β
(Intercept)	5.40 ***	5.37 ***
Human-AI	1.38 ***	1.68 ***
proactive	-0.29	-0.51$^{\bullet}$
Human-AI×proactive	-0.11	-0.14
ICC	0.51	0.57
$N_{student:grade}$	287	491
N_{grade}	4	4
$R^2/adj.R^2$	0.041/0.527	0.063/0.599

$^{\bullet}p < 0.1$, $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Impact of human-AI tutoring on IXL performance: As illustrated in Figure 5, we find that human-AI tutoring was associated with significant improvement in student skill proficiency in both the IK bandwidth subsample (36%; +6.10 skills, p < .001) and the full sample (42%; +6.75 skills, p < .001). The models had an intraclass correlation between 0.52–0.60, reflecting strong student-level variation in engagement within grades.

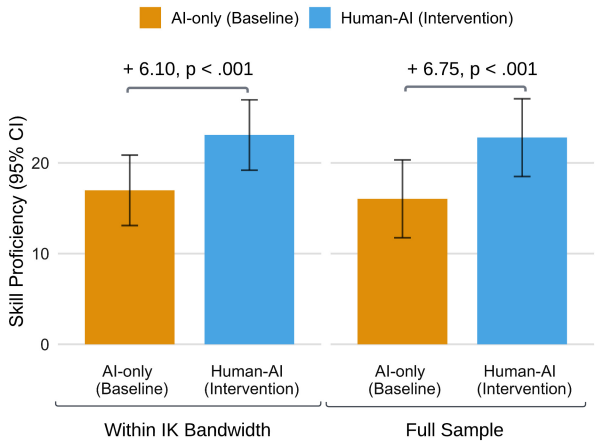


Figure 5: Within subject improvement of skill proficiency in IXL from the introduction of human-AI tutoring.

Further analysis revealed a significant increase in skill proficiency from human-AI tutoring for both groups compared to the AI-only tutoring (Table 6). The benefit was significant for both the IK bandwidth subsample (6.87, p<.001) and the full sample (7.97, p<.001). Within the IK bandwidth, results suggest that the proactive group, on average, had lower skill proficiency than the reactive group (-2.17, p=.067). Following the introduction of human support, despite improvements for both groups, the proficiency gap widened by 3.69 skills (14.79%), with the reactive group demonstrating greater proficiency than their proactive peers.

Table 6: Comparing the impact of proactive and reactive human-AI tutoring on student skill proficiency in IXL.

	IK Bandwidth	Full Sample
	β	β
(Intercept)	18.08 ***	17.49 ***
Human-AI	6.87 ***	7.97 ***
proactive	-2.17$^{\bullet}$	-3.06 **
Human-AI×proactive	-1.52	-2.56**
ICC	0.51	0.59
$N_{student:grade}$	287	491
N_{grade}	4	4
$R^2/adj.R^2$	0.094/0.554	0.115/0.634

$^{\bullet}p < 0.1$, $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

In IXL, both proactive and reactive groups were associated with significant improvement in their time on task and skill proficiency from human-AI tutoring compared to the AI-only baseline. While there was no significant difference in time on task between the two groups, we find suggestive evidence that the reactively tutored students demonstrated higher skill proficiency than their proactively tutored peers.

5.4 RQ2: Proactive vs. Reactive Tutoring

Building on RQ1, we estimate the differential effect of proactive versus reactive tutoring at the discontinuity using (Equation 4). Student prior performance is standardized within grades and adjusted to have the cutoff (median) be 0 for each grade. Under the local comparability assumption, we use the IK bandwidth subsample to estimate the treatment effect at the cutoff. To assess the robustness of our findings, we conducted several additional analyses: sensitivity analyses using alternative bandwidth selections, robust bias-corrected estimates using CCT bandwidth [12], and donut checks excluding observations immediately adjacent to the cutoff. The results of the additional analysis are available in the linked anonymous GitHub repository[5].

Figure 6 illustrates the differential effects of proactive and reactive tutoring across prior performance. At the cutoff, students in the reactive group achieved 1.74× expected MAP growth (+74%, p = .007). Although proactive students at the cutoff showed numerically lower growth, this difference was not statistically significant (-0.33, p = .579). The interaction between proactive tutoring and prior performance was also not statistically significant (-1.61, p = .247); however, the magnitude of this estimate suggests a meaningful pattern. This point estimate indicates that for each SD below the cutoff, the benefit of proactive tutoring increased by approximately 1.6×—a magnitude nearly equal to the baseline effect at the cutoff (1.74×). This suggests that the higher overall relative MAP gains for the proactive group (RQ1) compared to reactive tutoring are driven by greater benefits among students farther below the cutoff.

6. DISCUSSION

Given the heterogeneous treatment effects of human-AI tutoring, we study a differentiated proactive–reactive tutoring policy in which lower-performing students receive tutor-initiated support, whereas higher-performing students re-

[5]http://tiny.cc/EDM26

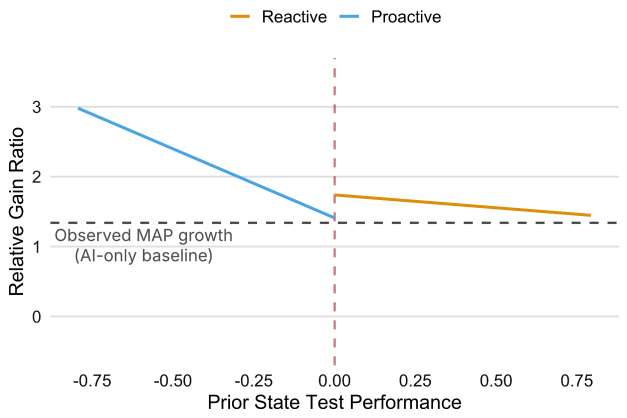


Figure 6: Examining the impact of proactive and reactive tutoring on students' relative MAP growth. The gray line represents the students' relative MAP growth (1.3 ×) during the AI-only baseline period. Additionally, Relative Gain

ceive reactive, on-demand support. Using a quasi-experimental DiDC design, we show that this policy improves engagement and overall learning while delivering larger gains for lower-performing students, offering a practical and scalable strategy for human–AI tutoring. As we discuss these findings, it is important to note that the estimates likely reflect the benefits of the implemented policy as a bundle, rather than the isolated effect of the assignment policy itself. As with most real-wold interventions, the observed gains likely encapsulate the combined effects of student needs, teacher context, tutor behavior, AI tutor use, and the proactive–reactive assignment rule. Next, we review the findings and their broader implications for educational research and policy.

6.1 Evaluating the Impact of Human-AI Tutoring on Standardized Test Performance

We find that access to human-AI tutoring significantly improved student performance on standardized tests (Figure 3), compared to AI-only tutoring. Within the IK bandwidth, we observe an average increase of 61% in MAP performance. Overall, students receiving human-AI tutoring demonstrated 2× the expected MAP growth during this period. While access to human-AI tutoring was beneficial to both groups, we find that it was particularly beneficial for the proactively tutored students, with the students experiencing an additional 75% improvement in performance—proactive tutoring helped narrow the achievement gap between the lower and higher performing students.

Our findings that students achieved approximately 2 × the expected national NWEA growth norms align with findings from Chine et al. [14]. As the Chine et al. study was conducted using in-person human-AI tutoring, it is promising that we are observing similar results with remote human tutors. Additionally, a key methodological distinction is that Chine et al. used a propensity-matched control group, whereas we compare against expected norms from the national sample of all MAP assessment takers.

6.2 Evaluating the Impact of Human-AI Tutoring on AI tutor use

Our evaluation of human-AI tutoring on student engagement and proficiency in IXL was consistent with the impact on MAP scores. Within the IK bandwidth, access to human-AI tutoring increased student time on task by 25% and skill proficiency by 36% across both groups. Notably, both groups achieved similar gains through different support mechanisms. Proactive tutoring helped lower-performing students sustain engagement beyond what AI-only support could provide, whereas reactive tutoring allowed higher-performing students to work independently at their own pace while retaining access to additional human support. Together, these results suggest that aligning tutoring intensity with student needs benefits both groups.

Our findings on the benefits of human-AI tutoring on IXL measures are consistent with recent findings from Borchers et al. [11], who observed increases of 25% in student time on task and 38% in skill proficiency with access to human-AI tutoring. An important distinction is that Borchers et al. used an additional rewards-based goal-setting intervention to motivate IXL use, whereas we explored the impacts of a differentiated tutoring policy that allocates tutor time and attention based on prior performance. These complementary approaches suggest promising potential for combining differentiated tutoring with goal-setting interventions to further improve the effectiveness of human-AI tutoring.

6.3 Comparing Proactive and Reactive Human-AI Tutoring

We find no significant difference between proactive and reactive tutoring at the cutoff. Given that human-AI tutoring overall improved student MAP performance by 61% and that proactive tutoring produced 75% greater gains than reactive tutoring on average, this null result at the cutoff suggests that the median represents an appropriate threshold where both groups benefit from differentiated support. However, as illustrated in Figure 6, the benefits of proactive tutoring were larger for students farther below the median, indicating that the observed improvements in MAP performance are driven by improvement in MAP performance of lower-performing students and thus contribute to the narrowing of achievement gaps.

6.4 Tutoring and EDM

Tutoring has long been a core focus of EDM, and research on effective human tutoring has shaped the design of tutoring systems [13, 56]. Over time, these systems have scaled access to tutoring support [5, 30, 3]. While AI tutors can provide reliable, continuous support at scale, they often fall short in delivering the motivational and conceptual support that human tutors excel at, particularly when student needs extend beyond current AI capabilities [54, 1]. Recent work on human-AI tutoring explores how targeted human support can complement AI tutors [48, 26, 14]. Our study advances this line of research by providing empirical evidence supporting the learning benefits of a differentiated tutoring strategy based on student needs and readiness. Beyond demonstrating the benefits of human-AI tutoring, these findings highlight opportunities to draw on established EDM methods for optimizing support allocation [59, 58], learner modeling

for adaptive interventions [2], and behavioral analytics that detect and respond to engagement states in real time [6, 27].

Taken together, our findings provide converging evidence with prior human-AI tutoring research. We find that our differentiated policy using remote tutors can achieve learning gains comparable to in-person human-AI tutoring [14]. Our approach also extends prior work on AI tutor engagement by achieving engagement and learning outcomes comparable to those reported by Borchers et al. [11], which used reward-based goal-setting interventions. These results underscore the value of adaptive support tailored to student needs and readiness over a one-size-fits-all approach.

7. LIMITATIONS

Our quasi-experimental DiDC design [46] provides evidence for the benefits of personalizing tutoring support, though stronger causal evidence could be achieved from randomized controlled trials. Similarly, our reliance on national growth norms as a reference point, rather than a matched control group, limits causal inference. However, ethical considerations complicate randomizing students to different support conditions within the intervention, given our findings that proactive support substantially benefits lower-performing students. A more appropriate RCT design would be to compare the differentiated human-AI tutoring policy as a whole with a business-as-usual control condition, providing a more rigorous baseline for evaluating intervention effects.

It is also important to acknowledge the limitations of our dichotomous allocation of support. Dividing students into proactive and reactive groups at the median may oversimplify the relationship between student need and optimal tutoring intensity. For instance, students much farther below the median (the bottom 10%) may require more intensive support than the proactive tutoring used in our study (1 tutor for 4 students). Additionally, our treatment allocation relied on a single baseline performance measure, leading to a static allocation of students to groups, whereas dynamically determining student need based on ongoing performance and engagement could further enhance the benefits of differentiated support. Past EDM research has developed algorithms for dynamic pairing policies in peer tutoring, which future work could repurpose for the study of optimal tutor-to-student pairing policies [59].

8. FUTURE WORK

The absence of Zoom interaction data (actual student–tutor interactions) represents a missed opportunity for deeper analysis, especially for understanding which elements of proactive tutoring were effective for student learning. We primarily relied on lead tutors and tutor supervisors to ensure implementation fidelity; better Zoom integration would provide additional data to monitor program implementation and identify areas for improvement. For instance, Borchers et al. [10] found that brief tutor visits during a remote tutoring session can significantly improve student performance. These analyses could better explain the success of our intervention. Access to multi-modal Zoom sessions data could provide a more comprehensive view of the dynamics of human-AI tutoring, presenting opportunities to build on current research [13, 23, 22, 17]. These insights would also enhance our understanding of how differentiated support models function in practice and inform their ongoing refinement.

An important improvement for future implementations is to facilitate consistent student–tutor pairing to support relationship building [45, 42]. In our study, students were randomly assigned to Zoom breakout rooms at the start of each session, and tutors were allocated a fixed set of rooms (e.g., a tutor might oversee rooms 1–4, though the students in those rooms varied across sessions). While this approach simplified logistical constraints, it limited opportunities for relationship-building and continuity of support across sessions. Prior work underscores the value of such consistency: Bleiberg et al. [8] found that female students experienced greater benefits when paired with female tutors, highlighting how relational dynamics and familiarity can enhance tutoring effectiveness. Related analyses of the differential impact of tutoring across student demographic groups could be subject to future work.

9. CONCLUSION

This study explored a key challenge in scaling human-AI tutoring: how to allocate limited human tutor support to effectively meet diverse student learning needs. Motivated by preliminary evidence that lower-performing students derive greater benefits from human-AI tutoring than higher-performing peers, we evaluated a differentiated proactive-reactive tutoring policy. Results from our quasi-experimental study provide evidence of a differentiated policy benefiting both lower-performing and higher-performing students. Compared to AI-only baselines, human-AI tutoring significantly improved students' time on task (+25%) and skill proficiency (+36%) in IXL, as well as their relative academic growth (+61%) on standardized MAP tests. Notably, the differentiated policy helped narrow achievement gaps, with lower-performing students experiencing 75% greater improvement in academic growth from proactive support compared to reactive tutoring. Relative to expected norms from the national sample of all MAP assessment takers, differentiated human-AI tutoring resulted in students achieving approximately 2× the expected growth. These findings provide converging evidence with ongoing evaluations of human-AI tutoring, demonstrating that differentiated allocation policies can significantly improve both learning outcomes and student engagement while optimizing limited human tutor resources.

These findings have important implications for scaling personalized learning and for EDM research. By differentiating support according to student needs (1:4 tutor-to-student ratio for proactive tutoring, 1:10 for reactive), human-AI tutoring can balance effectiveness with resource constraints. Students requiring greater assistance benefit from proactive support, while those demonstrating readiness for independent work benefit from reactive, on-demand support. This work demonstrates how readily available performance data can inform resource allocation decisions. Future EDM research could enhance this approach through more sophisticated learner modeling to dynamically adjust support allocation, behavioral indicators to identify when students need intervention, and adaptive algorithms that refine allo-

cation thresholds based on ongoing performance. This differentiated approach offers a sustainable pathway for scaling human-AI tutoring while advancing EDM's capacity to close the loop between learner diagnosis and instructional action.

Acknowledgments

The authors would like to thank the schools and IXL for their partnership and collaborative efforts in this research. This work was supported by the Learning Engineering Virtual Institute. The opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the Institute or IXL.

10. ADDITIONAL AUTHORS

Additional authors: Alex Houk (ahouk@andrew.cmu.edu), Erin Gatz (egatz@andrew.cmu.edu), and Boyuan (Bill) Guo (boyuang@cmu.edu).

11. REFERENCES

[1] V. Aleven, R. Baraniuk, E. Brunskill, S. Crossley, D. Demszky, S. Fancsali, S. Gupta, K. Koedinger, C. Piech, S. Ritter, et al. Towards the future of ai-augmented human tutoring in math learning. In *International Conference on Artificial Intelligence in Education*, pages 26–31. Springer, 2023.

[2] V. Aleven, B. Mclaren, I. Roll, and K. Koedinger. Toward meta-cognitive tutoring: A model of help seeking with a cognitive tutor. *International Journal of Artificial Intelligence in Education*, 16(2):101–128, 2006.

[3] V. Aleven, E. A. McLaughlin, R. A. Glenn, and K. R. Koedinger. Instruction based on adaptive learning technologies. *Handbook of research on learning and instruction*, 2:522–560, 2016.

[4] P. An, K. Holstein, B. d'Anjou, B. Eggen, and S. Bakker. The ta framework: Designing real-time teaching augmentation for k-12 classrooms. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–17, 2020.

[5] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *The journal of the learning sciences*, 4(2):167–207, 1995.

[6] R. S. Baker. Modeling and understanding students' off-task behavior in intelligent tutoring systems. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1059–1068, 2007.

[7] M. P. Bhatt, J. Guryan, S. A. Khan, M. LaForest-Tucker, and B. Mishra. Can technology facilitate scale? evidence from a randomized evaluation of high dosage tutoring. Technical report, National Bureau of Economic Research, 2024.

[8] J. Bleiberg, C. D. Robinson, E. Bennett, and S. Loeb. The impact of tutor gender match on girls' stem interest, engagement, and performance. 2025.

[9] B. S. Bloom. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational researcher*, 13(6):4–16, 1984.

[10] C. Borchers, A. Gurung, Q. Liu, D. R. Thomas, M. Khalil, and K. R. Koedinger. Brief but impactful: How human tutoring interactions shape engagement in online learning. *arXiv preprint arXiv:2601.09994*, 2026.

[11] C. Borchers, A. Houk, V. Aleven, and K. R. Koedinger. Engagement and learning benefits of goal setting with rewards in human-ai tutoring. In *International Conference on Artificial Intelligence in Education*, pages 46–59. Springer, 2025.

[12] S. Calonico, M. D. Cattaneo, and R. Titiunik. Robust nonparametric confidence intervals for regression-discontinuity designs. *Econometrica*, 82(6):2295–2326, 2014.

[13] M. T. Chi, S. A. Siler, H. Jeong, T. Yamauchi, and R. G. Hausmann. Learning from human tutoring. *Cognitive science*, 25(4):471–533, 2001.

[14] D. R. Chine, C. Brentley, C. Thomas-Browne, J. E. Richey, A. Gul, P. F. Carvalho, L. Branstetter, and K. R. Koedinger. Educational equity through combined human-ai personalization: A propensity matching evaluation. In *International conference on artificial intelligence in education*, pages 366–377. Springer, 2022.

[15] S. Copeland, M. A. Cook, A. A. Grant, and S. M. Ross. Randomized-control efficacy study of ixl math in holland public schools. *Center for Research and Reform in Education*, 2023.

[16] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

[17] M. G. Core, J. D. Moore, and C. Zinn. The role of initiative in tutorial dialogue. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, 2003.

[18] E. Cosyn, H. Uzun, C. Doble, and J. Matayoshi. A practical perspective on knowledge space theory: Aleks and its data. *Journal of Mathematical Psychology*, 101:102512, 2021.

[19] J. Dietrichson, M. Bøg, T. Filges, and A.-M. Klint Jørgensen. Academic interventions for elementary and middle school students with low socioeconomic status: A systematic review and meta-analysis. *Review of educational research*, 87(2):243–282, 2017.

[20] B. du Boulay. Recent meta-reviews and meta–analyses of aied systems. *International Journal of Artificial Intelligence in Education*, 26(1):536–537, 2016.

[21] Y. Fang, Z. Ren, X. Hu, and A. C. Graesser. A meta-analysis of the effectiveness of aleks on learning. *Educational Psychology*, 39(10):1278–1292, 2019.

[22] K. Forbes-Riley and D. J. Litman. Analyzing dependencies between student certainness states and tutor responses in a spoken dialogue corpus. In *Recent Trends in Discourse and Dialogue*, pages 275–304. Springer, 2008.

[23] B. A. Fox. *The Human Tutorial Dialogue Project: Issues in the design of instructional systems*. CRC Press, 2020.

[24] A. Goodman-Bacon. Difference-in-differences with variation in treatment timing. *Journal of econometrics*, 225(2):254–277, 2021.

[25] A. A. Grant, M. A. Cook, and S. M. Ross. The impacts of i-ready personalized instruction on student math achievement. *Center for Research and Reform in Education*, 2023.

[26] A. Gurung, J. Lin, J. Gutterman, D. R. Thomas, A. Houk, S. Gupta, E. Brunskill, L. Branstetter, V. Aleven, and K. Koedinger. Human tutoring improves the impact of ai tutor use on learning outcomes. In *International Conference on Artificial Intelligence in Education*, pages 393–407. Springer, 2025.

[27] A. Gurung, J. Lin, Z. Huang, C. Borchers, R. S. Baker, V. Aleven, and K. R. Koedinger. Starting seatwork earlier as a valid measure of student engagement. *arXiv preprint arXiv:2505.13341*, 2025.

[28] J. Guryan, J. Ludwig, M. P. Bhatt, P. J. Cook, J. M. Davis, K. Dodge, G. Farkas, R. G. Fryer Jr, S. Mayer, H. Pollack, et al. Not too late: Improving academic outcomes among adolescents. *American Economic Review*, 113(3):738–765, 2023.

[29] W. He and J. Meyer. Map growth universal screening benchmarks: Establishing map growth as an effective universal screener. *Northwest Evaluation Association*, 2021.

[30] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[31] K. Holstein, B. M. McLaren, and V. Aleven. Student learning benefits of a mixed-reality teacher awareness tool in ai-enhanced classrooms. In *International conference on artificial intelligence in education*, pages 154–168. Springer, 2018.

[32] G. Imbens and K. Kalyanaraman. Optimal bandwidth choice for the regression discontinuity estimator. *The Review of economic studies*, 79(3):933–959, 2012.

[33] G. W. Imbens and T. Lemieux. Regression discontinuity designs: A guide to practice. *Journal of econometrics*, 142(2):615–635, 2008.

[34] IXL Learning. Measuring the impact of ixl math and ixl language arts in pennsylvania schools. Technical report, IXL Learning, 2020. Retrieved July 6, 2025.

[35] S. Katz, P. Albacete, I.-A. Chounta, P. Jordan, B. M. McLaren, and D. Zapata-Rivera. Linking dialogue with student modelling to create an adaptive tutoring system for conceptual physics. *International journal of artificial intelligence in education*, 31(3):397–445, 2021.

[36] J. A. Kulik and J. D. Fletcher. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of educational research*, 86(1):42–78, 2016.

[37] J. Lin, E. Chen, Z. Han, A. Gurung, D. R. Thomas, W. Tan, N. D. Nguyen, and K. R. Koedinger. How can i improve? using gpt to highlight the desired and undesired parts of open-ended responses. *arXiv preprint arXiv:2405.00291*, 2024.

[38] J. Lin, Z. Han, D. R. Thomas, A. Gurung, S. Gupta, V. Aleven, and K. R. Koedinger. How can i get it right? using gpt to rephrase incorrect trainee responses. *International journal of artificial intelligence in education*, 35(2):482–508, 2025.

[39] J. McCrary. Manipulation of the running variable in the regression discontinuity design: A density test. *Journal of econometrics*, 142(2):698–714, 2008.

[40] D. S. McNamara, I. B. Levinstein, and C. Boonthum. istart: Interactive strategy training for active reading and thinking. *Behavior Research Methods, Instruments, & Computers*, 36(2):222–233, 2004.

[41] A. Mitrovic. An intelligent sql tutor on the web. *International Journal of Artificial Intelligence in Education*, 13(2-4):173–197, 2003.

[42] J. Morrow and M. Ackermann. Intention to persist and retention of first-year students: The importance of motivation and sense of belonging. *College student journal*, 46(3):483–491, 2012.

[43] K. Muldner, R. Lam, and M. T. Chi. Comparing learning from observing and from human tutoring. *Journal of Educational Psychology*, 106(1):69, 2014.

[44] A. Nickow, P. Oreopoulos, and V. Quan. The impressive effects of tutoring on prek-12 learning: A systematic review and meta-analysis of the experimental evidence. 2020.

[45] P. O'Keeffe. A sense of belonging: Improving student retention. *College student journal*, 47(4):605–613, 2013.

[46] P. Picchetti, C. C. Pinto, and S. T. Shinoki. Difference-in-discontinuities: estimation, inference and validity tests. *arXiv preprint arXiv:2405.18531*, 2024.

[47] T. W. Price, Y. Dong, and D. Lipovac. isnap: towards intelligent tutoring in novice programming environments. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on computer science education*, pages 483–488, 2017.

[48] D. D. Ready, S. G. McCormick, and R. J. Shmoys. The effects of in-school virtual tutoring on student reading development: Evidence from a short-cycle randomized controlled trial. *Journal of Education for Students Placed at Risk (JESPAR)*, pages 1–21, 2026.

[49] C. D. Robinson, C. Pollard, S. Novicoff, S. White, and S. Loeb. The effects of virtual tutoring on young readers: Results from a randomized controlled trial. *Educational Evaluation and Policy Analysis*, 47(4):1245–1265, 2025.

[50] R. D. Roscoe and D. S. McNamara. Writing pal: Feasibility of an intelligent writing strategy tutor in the high school classroom. *Journal of Educational Psychology*, 105(4):1010, 2013.

[51] C. Schonberg. The impact of ixl on math and ela learning in pennsylvania. Technical report, 2025. Retrieved July 6, 2025.

[52] V. Shute, S. Lajoie, and K. Gluck. Individualized and group approaches to training. *Training and retraining: A handbook for business, industry, government, and the military*, pages 171–207, 2000.

[53] V. J. Shute, G. Smith, R. Kuba, C.-P. Dai, S. Rahimi, Z. Liu, and R. Almond. The design, development, and testing of learning supports for the physics playground game. *International Journal of Artificial Intelligence in Education*, 31(3):357–379, 2021.

[54] S. Steenbergen-Hu and H. Cooper. A meta-analysis of the effectiveness of intelligent tutoring systems on k–12 students' mathematical learning. *Journal of educational psychology*, 105(4):970, 2013.

[55] D. R. Thomas, J. Lin, E. Gatz, A. Gurung, S. Gupta, K. Norberg, S. E. Fancsali, V. Aleven, L. Branstetter, E. Brunskill, et al. Improving student learning with

hybrid human-ai tutoring: A three-study quasi-experimental investigation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 404–415, 2024.

[56] K. VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational psychologist*, 46(4):197–221, 2011.

[57] M. Waalkens, V. Aleven, and N. Taatgen. Does supporting multiple student strategies lead to greater learning and motivation? investigating a source of complexity in the architecture of intelligent tutoring systems. *Computers & Education*, 60(1):159–171, 2013.

[58] K. B. Yang, V. Echeverria, Z. Lu, H. Mao, K. Holstein, N. Rummel, and V. Aleven. Pair-up: prototyping human-ai co-orchestration of dynamic transitions between individual and collaborative learning in the classroom. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pages 1–17, 2023.

[59] K. B. Yang, V. Echeverria, X. Wang, L. Lawrence, K. Holstein, N. Rummel, and V. Aleven. Exploring policies for dynamically teaming up students through log data simulation. *International Educational Data Mining Society*, 2021.

LLM Reasoning Predicts When Models Are Right: Evidence from Coding Classroom Discourse

Bakhtawar Ahtisham
Cornell University
ba453@cornell.edu

Kirk Vanacore
Cornell University
kpv27@cornell.edu

Zhuqian Zhou
Cornell University
zz968@cornell.edu

Jinsook Lee
Cornell University
jl3369@cornell.edu

Rene F. Kizilcec
Cornell University
kizilcec@cornell.edu

ABSTRACT

Large Language Models (LLMs) are increasingly deployed to automatically label and analyze educational dialogue at scale, yet current pipelines lack reliable ways to detect when the models are wrong. We investigate whether reasoning generated by LLMs can be used to predict the correctness of a model's own predictions. We analyzed 30,300 teacher utterances from classroom dialogue, each labeled by multiple state-of-the-art LLMs with an instructional move construct and an accompanying reasoning. Using human-verified ground-truth labels, we framed the task as predicting whether a model's assigned label for a given utterance was correct. We encoded the LLM reasoning using Term Frequency-Inverse Document Frequency (TF-IDF) and evaluated five supervised classifiers on their ability to predict label correctness. Among the evaluated classifiers, Random Forest achieved the strongest performance with an F1 score of 0.83 (Recall = 0.854), successfully identifying the majority of incorrect predictions and significantly outperforming baselines. Training specialist detectors for specific instructional move constructs further improved performance on difficult constructs, suggesting that error detection benefits from construct-specific linguistic cues. Using the Linguistic Inquiry and Word Count (LIWC) framework, we examined four linguistic constructs as markers of correctness: Causation (logical reasoning), Differentiation (complexity), Tentativeness (uncertainty), and Insight (metacognition). Correct predictions were characterized by grounded causal logic (e.g., 'because', 'therefore'), while faulty reasoning was five times more likely to rely on epistemic hedging (e.g., 'might', 'could') and twice as likely to employ performative metacognition (e.g., 'think', 'realize'). Syntactic complexity did not distinguish correct from incorrect reasoning, indicating that evidential grounding — not structural sophistication — drives validity. Our findings also challenge the "longer reasoning is better" heuristic; for most constructs, concise reasoning is a stronger predictor of correctness. These findings suggest that reasoning-based error detection provides a practical and scalable approach to quality control in automated educational dialogue analysis.

Keywords

Large Language Models, Natural Language Processing, Automated Coding, Classroom Discourse

1. INTRODUCTION

The analysis of qualitative educational data is increasingly central to Educational Data Mining (EDM) and related fields [11, 46]. Researchers routinely annotate discussion posts, assignments, or dialogue with specific constructs to support downstream analyses, such as Epistemic Network Analysis (ENA) [4] or sequential pattern mining [39]. These annotations enable large-scale investigations of how students and educators engage in learning tasks in classrooms [29], tutoring sessions [8], and online chat environments. Historically, this work has relied on supervised classification methods [5, 9], and more recently, Large Language Models (LLMs) to annotate data at scale [45, 21]. However, automated annotation pipelines often exhibit low to moderate reliability and inconsistent generalization across constructs [24, 18]. In particular, false positives remain a persistent challenge because they can systematically bias downstream structural and sequential modeling [40].

This paper addresses this central reliability gap: current pipelines lack robust mechanisms for detecting when automated annotations are wrong. Existing work primarily focuses on improving model accuracy, yet even high-performing models produce errors that propagate into subsequent analyses. We propose a complementary approach: using the reasoning generated by LLMs as a diagnostic signal for identifying misclassifications. Rather than treating LLMs as black-box classifiers, we examine whether their reasoning can serve as a secondary layer of quality control.

We operationalize this idea by training models to predict whether an LLM-generated annotation is correct or a false positive based solely on the accompanying reasoning text– asking the LLM to provide reasoning along with the annotation. Using traditional natural language processing (NLP) and machine learning (ML) methods, we show that correct and incorrect annotations can be distinguished with high accuracy ($F_1 = 0.830$ to 0.854), with further improvements when models are trained for specific instructional constructs

Bakhtawar Ahtisham, Kirk Vanacore, Zhuqian Zhou, Jinsook Lee, and Rene F. Kizilcec. LLM Reasoning Predicts When Models Are Right: Evidence from Coding Classroom Discourse. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 143–154. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039952

(e.g., separate models for classifying "Press for Accuracy" and "Press for Reasoning"). We additionally analyze linguistic markers associated with misclassification, finding that reasoning supporting false positives is longer and contains substantially more epistemic hedging, suggesting that models often verbalize their own uncertainty. Together, these findings introduce reasoning-based error detection as a scalable strategy for improving the robustness of automated discourse analysis in education.

This paper makes three main contributions: First, we introduce reasoning-based error detection as a new reliability layer for automated educational discourse annotation. Second, we demonstrate that traditional NLP and ML methods can accurately predict LLM annotation misclassification using only the LLM reasoning. And third, we identify linguistic markers of correctness and error in LLM reasoning, providing new insight into how and when automated annotations fail.

2. RELATED WORK

2.1 The Shift to Large Language Models in Classification

The EDM community has a long history of using machine learning and AI methods for classifying complex data at scale, ranging from students' knowledge states [7, 30] to epistemic affect [10, 27]. While much of this classification has dealt with structured log-file data using supervised machine learning models, recent work has shifted towards classification using Large Language Models (LLMs). This shift has been particularly beneficial for processing unstructured data, such as open-ended math problems [31], dialogues in tutoring sessions [1, 47], classroom discourse [37, 43], and writing tasks [26, 13].

Unlike many classification methods that require extensive feature engineering or large labeled datasets, LLMs offer the potential for "zero-shot" or "few-shot" classification, allowing researchers to scale qualitative coding to datasets previously considered too costly to annotate manually [18, 12, 49]. To improve classification accuracy, researchers have employed a variety of methods beyond simple prompting. For instance, recent studies have demonstrated that including diverse few-shot examples significantly boosts performance in identifying collaborative argumentation moves [36] and assessing instructional quality [29]. Furthermore, the integration of reasoning—asking the model to "think" before it classifies—has been shown to improve the detection of latent constructs like self-regulated learning processes [47]. By explicitly modeling the inference steps, these methods aim to bridge the gap between surface-level keywords and the deep semantic understanding required for qualitative coding.

2.2 Challenges with Misclassifications, Faithfulness

Despite these methodological advances, the reliability of LLM-based annotation remains inconsistent. While some studies report high agreement between LLMs and human experts for broad categories [26], others highlight significant declines in performance when models encounter ambiguous or high-inference constructs [18]. Crucially, [24] found that LLM precision in deductive coding tasks can vary drastically (from 0.10 to 0.87) depending on the specific prompt and construct. Similarly, [40] observed that LLMs consistently exhibit a high false positive rate, even when achieving moderate reliability overall.

This phenomenon is often described in NLP literature as a "faithfulness failure"—where the model generates a label that is plausible on the surface but contradicts the deeper semantic meaning of the input text [15, 2, 14, 22]. In educational contexts, this often manifests as the model over-relying on specific keywords (e.g., classifying any question starting with "Why" as a *Press for Reasoning*) rather than analyzing the true pedagogical intent [40].

2.3 LLM Reasoning as Diagnostic Signals

Recent literature suggests that LLMs possess an internal awareness of their own correctness, which can be elicited through probability estimation [17] or verbalized confidence scores [35, 16, 19]. However, explicit confidence prompts do not always capture the nuance of model failure, and existing mitigation strategies often rely on sampling consistency. Methods such as *SelfCheckGPT* [23] or *Self-Consistency* [41] generate multiple responses for the same input and aggregate them to filter out outliers. While effective, these methods increase computational cost linearly with the number of samples (N), making them prohibitively expensive for real-time educational feedback.

Consequently, a growing body of work suggests that the structure and content of model-generated explanations can serve as diagnostic signals for identifying incorrect predictions. In prior literature, this signal is often studied through reasoning traces—step-by-step intermediate reasoning sequences produced during inference, commonly in chain-of-thought or process supervision settings. While [38] caution that such traces may function as unfaithful post-hoc rationalizations, recent work in process supervision has nevertheless shown that properties of these traces, such as length and linguistic hedging, correlate with model uncertainty and error [44, 34, 32]. In contrast, the explanations analyzed in this study are elicited rationales: short, post-hoc justifications generated alongside a predicted label rather than procedural traces of internal inference. We do not assume these rationales faithfully reflect latent reasoning processes. Instead, we treat them as observable linguistic artifacts whose structure may still provide diagnostic information. Building on insights from trace-based work, we operationalize these linguistic markers within the educational domain to filter false positives in classroom discourse analysis, without the computational overhead of multiple sampling or process-level supervision.

2.4 Linguistic Analysis of Generated Text

To understand the nature of these failures, researchers have turned to linguistic analysis tools like Linguistic Inquiry and Word Count (LIWC) [28]. LIWC counts words in psychologically meaningful categories (e.g., affect, cognition, certainty) and has been widely used to analyze student essays, forum posts, and increasingly, AI-generated text. Prior work evaluating human-AI differences suggests that AI-generated text often exhibits distinct linguistic profiles, such as lower variance in vocabulary and specific patterns of hedging when the model is "confabulating" or hallucinating [48]. Similarly, [44] identified that uncertainty in reasoning is often marked by increased verbosity and "hedging" language. Our work extends

this methodology to the validation of automated coding. By applying LIWC analysis to LLM reasoning, we isolate the specific linguistic markers—such as high "insight" and low "causation"—that differentiate valid, grounded classification from misclassified outputs.

3. THE CURRENT STUDY

The increasing use of Large Language Models (LLMs) for automated discourse annotation has enabled educational researchers to analyze classroom dialogue at unprecedented scale. However, despite improvements in classification accuracy, LLM-based annotation pipelines continue to suffer from persistent misclassifications—particularly false positives—that can systematically bias downstream analyses such as sequential modeling and epistemic network analysis. Existing work has largely focused on improving model performance, with comparatively less attention paid to mechanisms for detecting when an automated annotation is likely incorrect.

The current study investigates whether model-generated explanations—elicited alongside predicted instructional move labels—can serve as a diagnostic signal for identifying annotation errors. Rather than treating LLMs as black-box classifiers, we examine whether the linguistic structure of the explanations they produce when justifying a label provides useful information about the correctness of that label. This framing positions explanation-based verification as a complementary reliability layer that operates independently of the original classification model and does not require additional sampling or access to model internals. We evaluate this approach in the context of automated coding of classroom discourse using a well-established Talk Moves framework. Specifically, we treat misclassification detection as a secondary supervised learning task: given an LLM's predicted label and its accompanying explanation, can we predict whether the label is correct relative to expert human annotations? In addition to assessing whether such a signal exists, we examine how robust it is across different state-of-the-art LLM architectures, whether it can be explained in terms of interpretable linguistic features, and whether performance improves when verification models are specialized for individual instructional constructs.

Guided by this framing, the study addresses the following research questions:

RQ1: Can the reasoning generated by LLMs serve as a reliable diagnostic signal for detecting misclassifications in educational discourse analysis?
RQ2: How robust is this reasoning-based verification signal across different state-of-the-art LLM architectures?
RQ3: What specific linguistic features within LLM reasoning are most predictive of classification correctness?
RQ4: Does training specialized models for specific constructs outperform a single generalist model in detecting misclassifications?

4. METHODOLOGY

The current study relies on a novel two-step method to explore improving annotations through classifying misclassifications. The first step, described in Section 4.2 uses a typical LLM qualitative coding scheme in which LLMs are prompted to annotate a data set, in this case classroom transcripts with *Talk Moves* derived from accountable talk theory. As part of this step, the LLMs are also prompted to provide reasoning for their classification. In the second step, described in Section 4.3, we trained machine learning models to detect misclassifications (false positives) using the reasoning output.

4.1 Talk Moves Data Set

We drew on the previously published *TalkMoves Dataset*, which contains classroom transcripts annotated and validated by expert educators and is described in detail in prior work [33]. For the present study, we used a stratified sample of 63 transcripts covering multiple grade bands (sampling procedure described below). The corpus consists of mathematics instruction transcripts from authentic K–12 settings, including whole-class discussions, small-group work, and online lesson environments. Transcripts were human-transcribed, speaker-segmented, and labeled at the utterance level using a six-construct *Talk Moves* scheme aligned with the Accountable Talk framework. Constructs associated with accountability to the learning community include *Keeping Everyone Together*, *Getting Students to Relate to Another's Ideas*, and *Restating*. Constructs associated with accountability to content knowledge include *Pressing for Accuracy*. Constructs associated with accountability to rigorous thinking include *Revoicing* and *Pressing for Reasoning*. Utterances that did not exhibit any of these discourse moves were assigned a *None* label.

The human-produced labels were treated as the reference standard for all performance comparisons. These labels were generated by trained mathematics educators using an established Accountable Talk coding protocol, with prior reporting showing very high inter-rater agreement (Cohen's $\kappa > 0.90$) [33]. For model evaluation, we reorganized transcripts into context windows of up to 20 utterances, a preprocessing step designed to fit model context limits while maintaining local discourse continuity.

To manage computational and API cost, we conducted the study on a structured subsample rather than the full corpus. We applied proportional stratified random sampling to select 800 focal utterances so that the class distribution matched the original Talk Move label frequencies. Each focal utterance was then augmented with its surrounding dialogue (up to 20 preceding turns and one following turn), and overlapping windows were consolidated into continuous segments. Because context windows were included by design, the final set exceeded the initial focal count: about 79.5% (10,522) were unique teacher utterances with valid Talk Move codes, and the remaining 20.5% were additional turns retained to preserve conversational context.

4.2 Annotation Procedure

We annotated the data sample for six talk moves constructs using four state-of-the-art Large Language Models (LLMs): **GPT-5** (OpenAI), **Claude 4.5 Sonnet** (Anthropic), **Gemini 2.5 Pro** (Google), **o3** (OpenAI). We employed a simple **zero-shot prompting strategy** with definitions of TalkMoves taken from

the Talk Moves coding schema[1]. The prompt included a request to "provide a concise explanation in the 'Reasoning' field only when a Talk Move is present." This generated a diverse dataset of LLM reasoning, ensuring the classifier was trained on a rich distribution of valid and invalid arguments across different generation attempts. This process generated an initial pool of over 30,817 LLM-predicted annotations. For 518 annotations (1.8%), the models did not produce reasoning, resulting in a total of 30,300 annotations used in the analysis.

4.3 Misconception Classification

4.3.1 Study Context & Data Partitioning

The dataset consists of $N = 30,300$ instances of teacher-student interactions, derived from a stratified sample of classroom logs. Each instance includes a target variable, `IsCorrect` (binary: 0 or 1), establishing ground truth for the correctness of the LLM's classification. To ensure robust evaluation, we employed a stratified random sampling technique to partition the data into a Training Set (D_{train}): 80% of the corpus ($n = 24,190$), used for model fitting and hyperparameter tuning; and a Test Set (D_{test}): 20% of the corpus ($n = 6,110$), held out strictly for final performance evaluation. Stratification was performed based on the `IsCorrect` label to preserve the class distribution (Correct vs. Incorrect) inherent in natural model processing. Transcript IDs were used as grouping constraints during partitioning so that utterances derived from the same transcript were not distributed across both training and test sets.

4.3.2 Feature Representation Details

The primary input variable X is the unstructured text of the reasoning for the annotation generated by the LLM. We transformed this raw text into a numerical feature space using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. The feature space dimension V had a maximum of 3,227 unigrams ($|V| = 3,227$), based on the max unigrams in the corpus. Stop-word removal (English) was applied to reduce feature noise. Preliminary experiments comparing Unigram vs. Bigram/Trigram architectures yielded negligible performance improvements ($< 0.2\%$ $\Delta F1$), justifying the parsimonious choice of a unigram-only model.

4.3.3 Classification Algorithms

We implemented and compared five supervised classification algorithms to predict misclassification. All models incorporated class weighting (`class_weight = 'balanced'`) to penalize misclassification of the minority class proportionally to its frequency.

Logistic Regression (Baseline): Selected for its interpretation capabilities (coefficient analysis) and efficiency in high-dimensional sparse spaces.
Random Forest (Ensemble): A bagging ensemble of 100 decision trees. This model was utilized to capture non-linear feature interactions (e.g., the conjunction of "not" and "verbatim") that linear models might miss.
Support Vector Machine (Linear SVM): A max-margin classifier. We utilized a linear kernel to handle the high dimensionality of the TF-IDF space efficiently.

[1]`https://github.com/SumnerLab/TalkMoves/blob/main/Coding\%20Manual.pdf`

Gradient Boosting (GBM): A boosting ensemble that builds simplified trees sequentially to minimize residual error, included to test robustness against hard-to-classify examples.
Multinomial Naive Bayes: A probabilistic baseline relying on Bayes' theorem with an assumption of feature independence, traditionally strong for text classification.

4.4 Per-Construct Model

To address the heterogeneity of linguistic correctness markers across different TalkMoves, we additionally trained one model per construct. This approach decomposes the global classification problem into a set of distinct sub-problems. Let C be the set of all possible TalkMoves Constructs (e.g., "Pressing for Reasoning", "Revoicing"). For each construct $c \in C$, we defined a specialist subset $D_c \subset D_{train}$ such that:

$$D_c = \{(x_i, y_i) \mid \text{PredictedLabel}(x_i) = c\} \tag{1}$$

For the top $k = 6$ most frequent constructs, we trained an independent Random Forest classifier M_c exclusively on D_c. At inference time, the system routes a new instance x_{new} to the appropriate model M_c based on its predicted label. This allows M_c to learn a localized vocabulary of correctness—for instance, weighting the term "classmate" heavily for *Getting Students to Relate*, a signal that might be irrelevant or noisy for *Pressing for Accuracy*.

4.5 Linguistic Features (LIWC)

To systematically analyze the cognitive and rhetorical structure of LLM reasoning, we processed all reasoning outputs using the Linguistic Inquiry and Word Count (LIWC) framework [28]. We focused on three theoretically relevant constructs: (a) logical grounding, (b) epistemic calibration, and (c) cognitive self-referential framing We selected these linguistic feature constructs to operationalize dimensions of reasoning quality and epistemic stance in LLM-generated explanations. Prior work on model explanations and uncertainty suggests that incorrect or weakly grounded reasoning is often characterized not by syntactic simplicity, but by epistemic hedging, metacognitive narration, and reduced evidential linkage between claims and conclusions [6, 15].

Causation markers (e.g., *because, therefore, implies*) were included to measure explicit premise–conclusion linking. In instructional move classification, correct labels require mapping observable discourse evidence to construct definitions. Reasoning that explicitly encode causal or justificatory structure may be more likely to reflect evidence-based classification rather than keyword matching or surface heuristics [3].

Differentiation markers (contrastives, discrepancies, and exclusions; e.g., *but, however, except, should*) were included to measure contrastive and error-checking structure. Many construct definitions involve ruling out alternative interpretations (e.g., distinguishing revoicing from restating). We therefore tested whether valid reasoning exhibit greater contrastive structure. This composite category captures logical branching and boundary-setting language.

Tentativeness markers (e.g., *might, could, possibly*) were selected to capture epistemic hedging and uncertainty expression. Work on verbalized model uncertainty shows that LLMs frequently externalize low confidence through modal and hedge language [15]. We therefore treat tentativeness

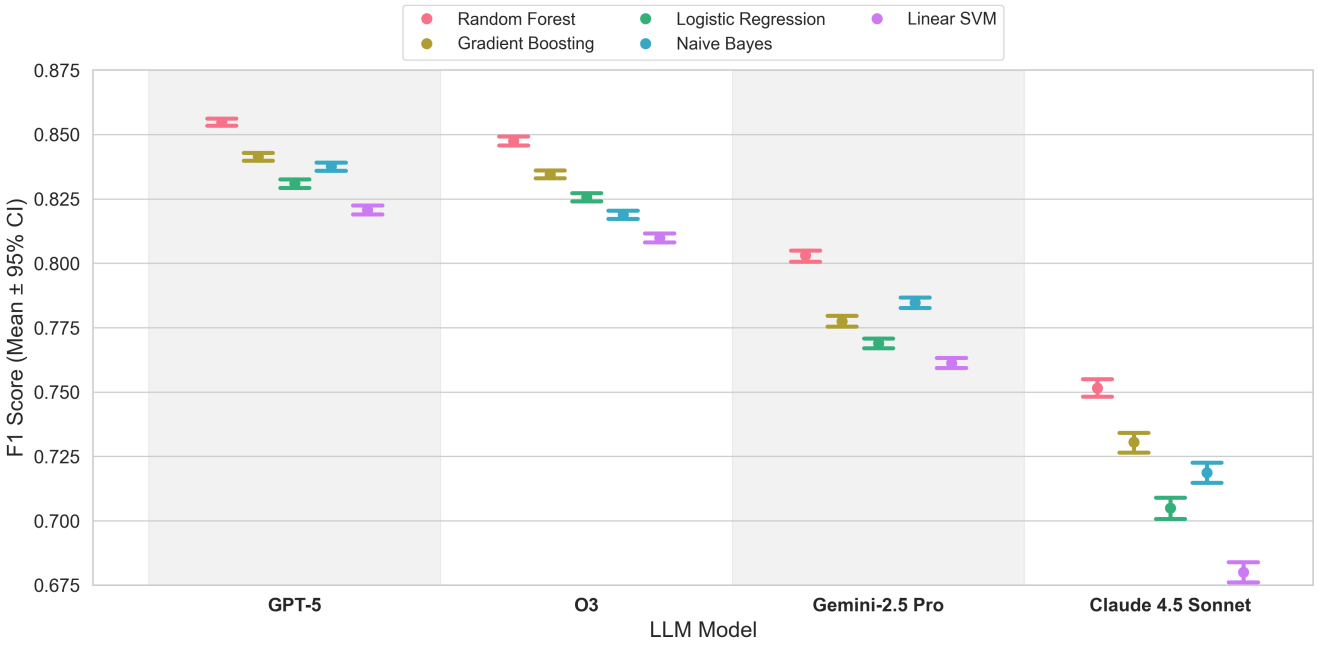


Figure 1: Cross-Model Detection Performance. A comparison of F1 scores for five supervised classification algorithms (Logistic Regression, Random Forest, SVM, Gradient Boosting, Naive Bayes) across the four target LLMs (Claude 4.5 Sonnet, Gemini 2.5 Pro, GPT-5, o3). The Random Forest classifier (pink points) consistently outperforms linear baselines across all generative models. Error bars represent 95% confidence intervals.

density as a proxy for internal uncertainty signals that may correlate with misclassification risk [6].

Insight markers (e.g., *think, realize, know*) were included to detect metacognitive and self-referential reasoning frames. For our proposal, these marks might be indicative of stances rather than empirically grounded evidence. Prior work on stance and epistemic modality shows that verbs such as "think" and "know" are linguistic markers of subjective commitment or metacognitive framing rather than text-based evidential support, and have been used as features in uncertainty/stance tasks in NLP [25].

Together, these feature groups map onto complementary dimensions for exploratory analysis of LLM reasoning and its association with misclassification. This design enables direct testing of whether correctness is primarily associated with logical grounding versus rhetorical or metacognitive elaboration. Finally we examine the differences in reasoning length between correct and incorrect annotations. A common heuristic in prompt engineering is that "more reasoning equals better performance" [42]. However, this is often used in the context of 'chain-of-thought' reasoning. Less is known about how explanation length behaves when models are prompted to justify a label they have already produced. Accordingly, we analyze the distribution of rationale lengths for correct and incorrect annotations to assess whether explanation length provides a useful signal for verification.

5. RESULTS

5.1 Can LLM Reasoning Predict Correctness? (RQ1)

First, we evaluate whether the LLM reasoning text contains sufficient signal to verify the correctness of an LLM's label by benchmarking classifier performance on the holdout sample. As shown in Table 1, all models achieved high predictive accuracy, with F_1 scores ranging from 0.79 to 0.83. The Random Forest classifier yielded the top performance, achieving a Precision of 0.807 and a Recall of 0.854. This high recall is particularly critical in an educational context, as it minimizes the risk of inadvertently discarding valid Talk Moves during the verification process. Overall, these results provide strong evidence that LLM reasoning contains latent linguistic markers that effectively signal classification accuracy.

Table 1: Performance of Verification Classifiers. Evaluation on the held-out test set ($N = 6,110$). Random Forest yields the highest F1 score.

Model	Precision	Recall	F1	Accuracy
Random Forest	**0.807**	0.854	**0.830** ($\pm$ 0.009)	**81.1%**
Gradient Boosting	0.764	**0.867**	0.812 ($\pm$ 0.010)	78.3%
Naive Bayes	0.783	0.832	0.807 ($\pm$ 0.010)	78.5%
Logistic Regression	0.806	0.797	0.802 ($\pm$ 0.010)	78.7%
Linear SVM	0.798	0.780	0.789 ($\pm$ 0.011)	77.4%

To contextualize this prediction task, consider the following two examples of model-generated reasoning drawn from our dataset, only one of which is correct.

"The teacher rephrases the student's question, clar-

ifying the idea that the measurement might not be the whole volume because the object is floating." — **Reasoning A** (Gemini 2.5 Pro)

"The teacher explicitly asks the student to provide the reasoning ('why') for their answer." — **Reasoning B** (Gemini 2.5 Pro)

Recognizing that Reasoning A accompanied a misclassification, while Reasoning B presents a correct justification, is not obvious. Notice that the incorrect reasoning leaks uncertainty through hedging or speculative language ("might not be"), whereas the correct reasoning is more direct and grounded in observable instructional actions ("explicitly asks"). We formally analyze linguistic markers in Section 5.3.

5.2 Robustness Across LLMs (RQ2)

We evaluated classifier performance across the four distinct LLMs used for annotation: GPT-5, o3, Gemini 2.5 Pro, and Claude 4.5 Sonnet. As illustrated in Figure 1, all misconception classification models performed well ($F1 > 0.67$) on classifications from all LLMs. The Random Forest classifier consistently outperformed the other models across all four LLMs. This dominance was particularly pronounced for annotations generated by Claude 4.5 Sonnet, where other models dropped to $F1$ scores as low as 0.70, while the Random Forest classifier maintained a significantly higher performance of 0.75. This performance gap indicates that the linguistic features distinguishing valid from invalid reasoning may be inherently nonlinear, necessitating the complex decision boundaries of ensemble methods regardless of the source LLM.

While the detection method remained effective globally, the absolute detectability of misclassifications varied by source model. Misclassifications generated by reasoning models like o3 and GPT-5 were the easiest to detect, with the Random Forest achieving $F1$ scores exceeding 0.85. The strong agreement between linear and nonlinear models in these conditions suggests that errors in these models may be marked by distinct, easily separable lexical artifacts. In contrast, misclassifications from Claude 4.5 Sonnet proved more challenging to verify. In these cases, the generated reasoning may be more linguistically homogenous with valid reasoning, reducing the effectiveness of simple keyword-based detection. Overall, while the approach performs moderately well across the board, performance is maximized when applied to models that exhibit distinct and detectable misclassifications.

5.3 Linguistic Markers of Misclassification (RQ3)

Our multidimensional linguistic analysis reveals distinct cognitive profiles for valid versus invalid reasoning. As illustrated in Figure 2, correctness in mathematical reasoning is not signaled by complexity or "insight," but by the prosaic density of causal logic.

5.3.1 Metacognition as a Negative Signal

Insight was a significant *negative* signal ($p < 0.001$). Reasoning that contained high densities of metacognitive terms like "I think," "I realize," and "understand" were overwhelmingly more likely to be incorrect. This finding identifies a specific

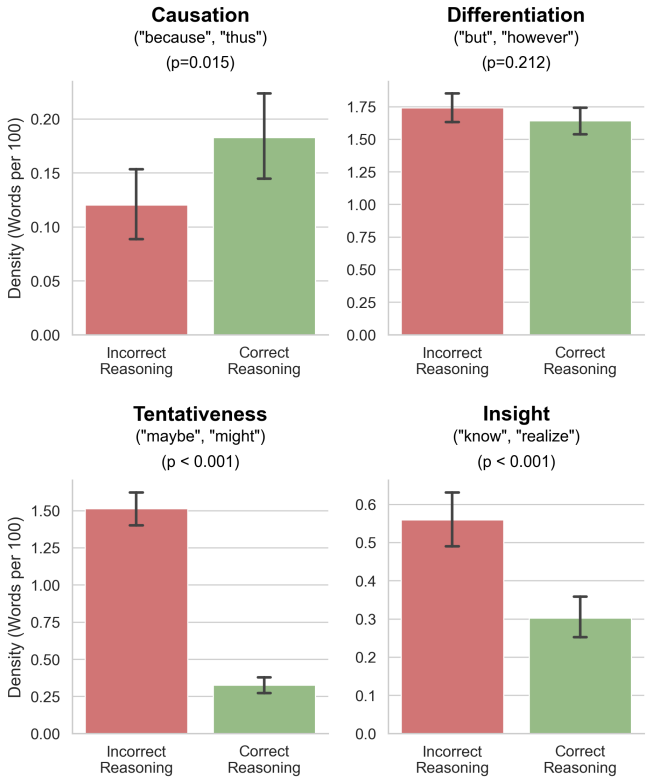


Figure 2: Linguistic Markers of Correct vs. Incorrect Reasoning. Average density of linguistic features (words per 100 words) in correct (orange) vs. incorrect (blue) instructional move reasoning. Causation ("therefore", "because") is the primary predictor of correctness ($p = 0.015$), indicating that explicit logical structure aligns with accurate problem solving. Conversely, Tentativeness ("might", "could") and Insight ("believe", "think") are more prevalent in incorrect responses ($p < 0.001$), suggesting that models use hedging and metacognitive language as "filler" when reasoning is flawed or uncertain. Differentiation ("but", "however") showed no significant difference ($p = 0.212$), appearing equally in both valid and invalid arguments. Error bars represent 95% confidence intervals.

mode of failure: mere "performative reasoning." When the model lacks a clear logical path, it compensates by narrating its own cognitive state, filling the reasoning with vacuous self-reflection instead of derivation.

5.3.2 Verbalized Uncertainty and Calibration

We examined *Tentativeness* (e.g., "might," "could," "possibly") to determine if models effectively signaled their uncertainty and found a statistically significant trend where incorrect answers contained a higher density of tentative language. This indicates a degree of successful calibration: when the model is less certain of its reasoning, it employs more hedging. Conversely, correct reasoning was characterized by declarative, confident assertions, suggesting that high certainty (low tentativeness) is a reliable linguistic proxy for mathematical accuracy. This "Verbalized Uncertainty" [20] serves as a potent diagnostic signal, allowing the classifier to detect when the model is "guessing" based on weak correlations rather than identifying a clear signal.

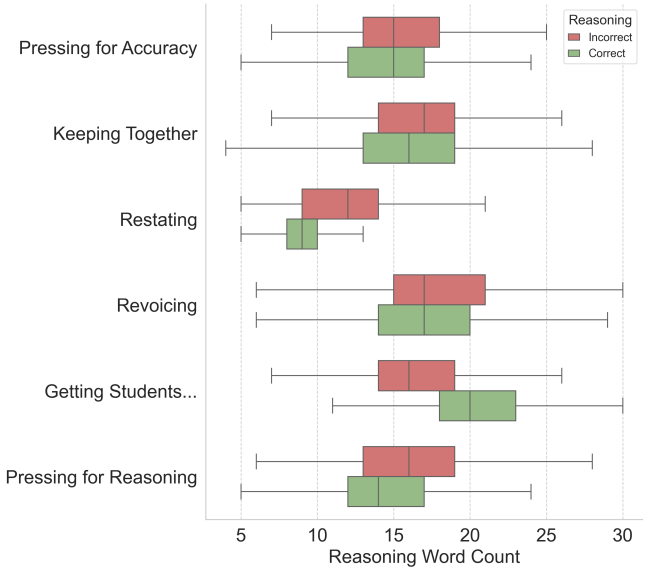


Figure 3: Reasoning Length Distribution. Distribution of reasoning lengths for Correct (Green) vs. Incorrect (Red) predictions. All differences are statistically significant ($p < 0.05$). Reasoning for correct responses is generally more concise than for incorrect responses, with the notable exception of *Getting Students to Relate*, where correct identification of complex student-to-student interactions requires significantly longer explanations.

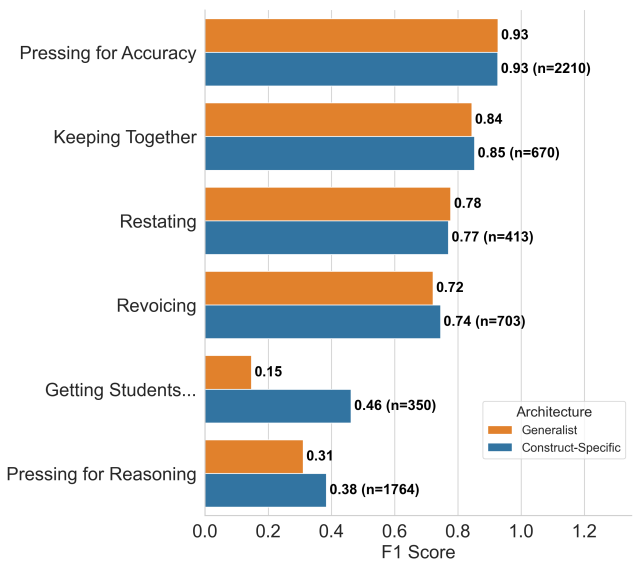


Figure 4: Performance Comparison: Generalist vs. Construct-Specific Models. While generalist models perform comparably on constructs like *Pressing for Accuracy* ($F1 \approx 0.93$), construct-specific models provide critical gains for complex moves. Notably, for *Getting Students to Relate* ($n = 350$), the specialist architecture triples the F1 score (0.46 vs. 0.15), recovering signal that is otherwise drowned out by global patterns.

5.3.3 *Causal Chaining and Logical Validity*

Valid mathematical derivations are fundamentally characterized by explicit causal structures. Likewise, we found that correct responses exhibited a significantly higher ($p = 0.015$) density of Causation markers (e.g., "therefore," "because," "implies," "yields") than misclassified responses. Correct reasoning contained a mean density of 0.18 causal markers per 100 words, compared to only *0.12* for misclassified responses. This finding confirms that accuracy is strongly correlated with the use of directional logical connectors: correct models explicitly link premises to conclusions, whereas incorrect models often present disconnected assertions without the requisite connective tissue.

5.3.4 *Conceptual Differentiation and Structural Complexity*

We analyzed the density of Differentiation markers (e.g., "but," "however," "except") to determine if valid reasoning involved more robust error-checking or case handling. The analysis yielded a null result ($p = 0.212$), with no significant difference in differentiation density between correct ($\mu = 1.63$) and incorrect ($\mu = 1.74$) reasoning. This indicates that incorrect reasoning is often just as syntactically complex as valid reasoning. Reasoning of misclassified responses frequently employs contrastive language not to distinguish valid mathematical constraints, but rather to express confusion or hedge between incorrect options, resulting in a linguistically sophisticated but logically flawed explanation.

5.3.5 *Reasoning Length as a Signal*

Besides the LIWC analysis, we also evaluated whether the reasoning length analysis differed based on correct or incorrect responses. Figure 3 reveals distinct verbosity profiles for valid vs. invalid predictions: For the majority of constructs (e.g., *Pressing for Reasoning*, *Keeping Everyone Together*), valid reasoning (Green boxplots) is **significantly shorter** than incorrect ones (Red boxplots). This suggests that valid reasoning is precise, evidence-based, and directed. In contrast, misclassified reasoning tends to be verbose and circuitous, often characterized by the model "waffling" or attempting to rationalize a weak connection with filler text. The "Relating" Exception: The sole statistically significant exception is *Relating to Others*, where correct reasoning is longer ($Median \approx 22$ words) than misclassifications. This makes intuitive sense: correctly identifying a "Relate" move requires describing a complex interaction between the teacher, Student A, and Student B. A misclassification in this construct often fails to capture this tripartite structure, resulting in shorter, more vague reasoning.

5.4 Construct-Specific Detection (RQ4)

Finally, we investigated whether we could optimize performance by training Construct-Specific Verifiers—dedicated Random Forest models trained exclusively on partitions of the dataset corresponding to a single TalkMove. As illustrated in Figure 4, these targeted models demonstrated distinct performance profiles based on construct complexity. We observed high verification reliability for **Pressing for Accuracy** ($F1 = 0.86$) and **Restating** ($F1 = 0.78$). This indicates that for moves defined by specific keywords (e.g., "exact words", "vocabulary"), the classifier is effective at distinguishing valid reasoning from misclassifications. In contrast, inferential

constructs proved more challenging. **Pressing for Reasoning** achieved an F1 of 0.48. Similarly, **Getting Students to Relate to Another's Ideas** ($n = 1,918$) achieved an F1 of 0.46. While low in absolute terms, this represents a significant signal recovery compared to the generalist baseline ($F1 = 0.15$), confirming that training on a dedicated subset captures distinct tripartite references (Teacher-Student-Student) that are otherwise diluted in the global dataset.

5.4.1 Linguistic Markers of Correctness: A Construct-by-Construct Analysis

To understand *what* the specialist classifiers are learning beyond global heuristics, we analyzed feature importance scores to extract the "vocabulary of correctness" for each TalkMove. Table 2 details the distinguishing signatures and provides in-context examples of correct vs. incorrect reasoning.

This analysis reveals that reasoning for each pedagogical move has a *linguistic fingerprint*:

Explicit Moves (e.g., Restating): Reliability is driven by correct reasoning containing synonyms for exactness (*verbatim, repeats, word-for-word*). Misclassifications typically involve words related to modification (*clarify, rephrase*), indicating the model is confusing a repetition with a re-explanation.

Inferential Moves (e.g., Pressing for Reasoning): The signal relies on causal language (*because, justify, why*). Misclassifications often use vague cognitive terms (*thinking, thoughts*), failing to distinguish between a general question and a specific request for evidence.

Social Moves (e.g., Relating, Keeping Together): Correctness is marked by collective nouns (*class, peer, classmate, we*). The absence of these group-orienting terms is a strong predictor of misclassifications, where the model falsely attributes a social move to a generic teacher-student interaction.

6. DISCUSSION

This study shows that model-generated rationales (i.e., elicited explanations produced alongside automated discourse labels) contain systematic linguistic signals that predict whether an LLM's annotation is correct. Across various LLMs, prediction models, constructs, and verification architectures, we find that these rationales support reliable detection of misclassifications (F1 $\approx$ 0.83), demonstrating that error and uncertainty are frequently externalized when models are prompted to explain their decisions. This finding aligns with prior work showing that reasoning and verbalized confidence signals encode information about model uncertainty and correctness [17, 35, 44]. Rather than treating explanations as auxiliary artifacts, our results position them as a practical verification layer that can be leveraged to improve the reliability of large-scale automated coding in EDM, where LLM-based classification is increasingly used for complex, high-inference constructs [18, 26].

These findings contribute to ongoing debates about the role and reliability of LLM explanations. Prior work has cautioned that explanations may function as post-hoc rationalizations that are not causally responsible for model predictions [38, 15, 2]. Our results support a more application-oriented interpretation: even when elicited rationales are not guaranteed to be faithful to internal model mechanisms, their linguistic structure remains predictively diagnostic of correctness. In high-inference educational coding tasks where reliability has been shown to vary widely across prompts and constructs [24, 40], valid annotations are more often supported by rationales grounded in observable textual evidence and explicit causal logic, whereas incorrect annotations more frequently rely on speculative or self-referential language. This enables downstream verification without requiring access to model internals or computationally expensive multi-sample strategies such as self-consistency [41] or SelfCheck-style approaches [23].

The linguistic analysis reveals a consistent pattern across models and constructs. Correct annotations are associated with concise, declarative rationales that explicitly connect utterance features to constructs using causal connectors and evidence-grounded justification. In contrast, misclassifications are marked by increased epistemic hedging, performative metacognitive language, and greater verbosity. This pattern is consistent with prior analyses showing that LLM uncertainty and error are often expressed through longer, more hedged reasoning traces [44] and with broader accounts of confabulation and faithfulness failures in generated explanations [15, 2, 48]. LIWC-based and related linguistic profiling work further suggests that AI-generated text exhibits systematic shifts in hedging, insight, and causation markers when semantic grounding is weak [28, 48]. Importantly, incorrect rationales in our data are not linguistically simplistic; they are often syntactically complex and rhetorically fluent, reinforcing that surface sophistication alone is not a reliable proxy for validity.

We also observe substantial variation in detectability across LLM architectures. Misclassifications generated by GPT-5 and o3 were comparatively easy to identify, while errors from Claude 4.5 Sonnet were more difficult to distinguish from correct annotations. This result is consistent with prior findings that LLM annotation reliability varies significantly across models and prompt formulations [18, 24]. It also suggests that as models improve in rhetorical fluency, incorrect rationales increasingly approximate the discourse patterns of correct ones, creating a moving-target problem for explanation-based verification. Explanation-based detection remains effective, but its discriminative margin may narrow as stylistic quality improves. This reinforces the need to combine rationale-based verification with complementary approaches such as calibration signals and construct-specific prompting strategies [17, 35].

Our results further highlight a trade-off between model generality and construct specificity in LLM-based coding. Prior work shows that LLM performance drops for ambiguous or high-inference constructs and that targeted prompt and example design can substantially improve results [36, 29, 18]. Consistent with this pattern, we find that specialist verifiers yield substantial gains for certain discourse constructs, suggesting a practical pipeline in which general-purpose LLM annotation is paired with construct-targeted verification layers. Linguistic profiles of correct rationales—such as explicit relational or causal structure—provide actionable features

Table 2: Linguistic Markers of Correctness (TF-IDF Analysis). A breakdown of the lexical markers that distinguish correct reasoning from misclassifications across six TalkMoves. (+) denotes positive predictors, (-) denotes negative predictors.

Construct	Key Markers	Correct Example (+)	Incorrect Example (-)
Pressing for Reasoning	(+) *justify, because, explain why* (-) *share, thinking, thoughts*	"The teacher asks 'How do you know', prompting the student to explain their **justification**."	"The teacher prompts students to write... eliciting explanation of their **thinking**."
Restating	(+) *verbatim, repeats, exactly* (-) *clarify, rephrase, confirm*	"The teacher repeats the student's response ('No') **verbatim**."	"The teacher repeats a key part... for **clarification**, changing the words."
Revoicing	(+) *paraphrases, repeats, wording* (-) *question, check, understanding*	"The teacher **paraphrases** the student's idea... changing **wording** to clarify."	"The teacher reframes... by suggesting 'dis' might mean 'wrong,' checking **understanding**."
Pressing for Accuracy	(+) *precise, specific, term* (-) *check, confirm, correctness*	"The teacher asks the student to use the **precise** mathematical **term**."	"The teacher **checks** whether a specific way of writing is valid..."
Keeping Everyone Together	(+) *class, shared, orienting* (-) *agreement, explain, question*	"Prompts a student to **read aloud** to the **class**, orienting listeners."	"The teacher asks a **question** to ensure all students are following."
Getting Students to Relate	(+) *classmate, peer, react* (-) *connect (vague), link, interaction*	"The teacher explicitly asks students to **react** to [Student]'s idea."	"The teacher asks students to **connect** their answer to the previous problem."

for designing these construct-specific verifiers and prompts.

Finally, these results have direct implications for educational data mining pipelines that rely on automated discourse annotation. LLM-based coding is increasingly used to analyze tutoring dialogue, classroom discourse, and other forms of unstructured educational text [1, 47, 37, 43]. However, prior studies document persistent false positives and construct-level reliability gaps [40, 24]. Because downstream methods such as discourse modeling and instructional quality measurement are sensitive to systematic labeling errors, explanation-based verification offers a scalable quality-control layer. By using rationale-derived signals to triage outputs, high-confidence annotations can be accepted automatically while flagged cases are routed for human review or construct-specific reanalysis. This selective acceptance strategy supports scalable annotation while preserving inferential validity, advancing more trustworthy deployment of LLMs for educational discourse analysis.

6.1 Limitations

Our study focused on a specific math instructional context. While the concept of detection via reasoning is likely generalizable, the specific vocabulary (e.g., "numerator", "fraction") may be domain-specific. Future work should test the transferability of these Correctness markers to other domains, such as English Language Arts or Science. All rationales analyzed in this study were elicited through a specific prompting strategy that explicitly asked models to justify their labels. Different prompt formulations, levels of instruction, or constraints on explanation length may influence how uncertainty and error are linguistically expressed. As a result, the strength and form of the verification signal may vary under alternative prompting regimes. Our models operate on textual features of model-generated rationales using traditional NLP representations. While this design choice improves interpretability and computational efficiency, it does not capture deeper semantic or discourse-level structure that may further distinguish valid from invalid explanations, particularly for high-inference constructs. More expressive representations may improve detection performance but also introduce additional complexity and cost.

7. CONCLUSION

As LLMs become increasingly integrated into educational data mining workflows, the reliability of automated annotation remains a central challenge. This work shows that when models are prompted to justify their labels, the resulting model-generated rationales contain systematic linguistic signals that can be used to detect misclassifications at scale. Across multiple state-of-the-art LLMs, instructional constructs, and verification architectures, we demonstrate that these rationales support accurate prediction of annotation correctness, offering a practical mechanism for quality control in automated discourse analysis.

Beyond performance gains, our findings provide insight into how and when automated annotations fail. Correct annotations are supported by concise, causally grounded explanations that reference observable features of the discourse, while incorrect annotations are more often accompanied by hedging, performative metacognitive language, and increased verbosity. These patterns suggest that elicited explanations externalize model uncertainty in a way that is both systematic and exploitable, even when explanations are not guaranteed to reflect internal decision processes.

From an applied perspective, this work reframes explanations from passive artifacts into actionable verification signals. Rather than treating LLM outputs as binary predictions, ed-

ucational researchers can leverage model-generated rationales to selectively accept high-confidence annotations and flag uncertain cases for further review. This approach enables scalable annotation while mitigating the propagation of false positives into downstream analyses such as sequential modeling, instructional quality measurement, and network-based representations of classroom discourse.

More broadly, this study contributes to ongoing discussions about the role of explanations in machine learning systems. While explanations may not always be faithful representations of internal reasoning, our results show that they can nonetheless be predictively useful for assessing correctness in real-world, high-inference tasks. By operationalizing explanation-based verification in an educational setting, this work offers a practical path toward building more trustworthy, transparent, and robust LLM-powered analytics for learning and instruction.

8. REFERENCES

[1] H. Acosta, S. Lee, H. Bae, C. Feng, J. Rowe, K. Glazewski, C. Hmelo-Silver, B. Mott, and J. C. Lester. Recognizing multi-party epistemic dialogue acts during collaborative game-based learning using large language models. *International Journal of Artificial Intelligence in Education*, 35(2):677–701, 2025.

[2] C. Agarwal, S. H. Tanneru, and H. Lakkaraju. Faithfulness vs. plausibility: On the (un) reliability of explanations from large language models. *arXiv preprint arXiv:2402.04614*, 2024.

[3] O. Biran and K. McKeown. Human-centric justification of machine learning predictions. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[4] Z. Cai, B. Eagan, N. Dowell, J. Pennebaker, D. Shaffer, and A. Graesser. Epistemic network analysis and topic modeling for chat data from collaborative learning environment. In *Proceedings of the 10th international conference on educational data mining*, Jan. 2017.

[5] Z. Cai, A. Siebert-Evenstone, B. Eagan, D. W. Shaffer, X. Hu, and A. C. Graesser. ncoder+: a semantic tool for improving recall of ncoder coding. In *International Conference on Quantitative Ethnography*, pages 41–54. Springer, Oct. 2019.

[6] H. Chen, W. Du, and Y. Ji. Explaining predictive uncertainty by looking back at model explanations. *arXiv preprint arXiv:2201.03742*, 2022.

[7] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[8] D. Demszky and J. Liu. M-powering teachers: Natural language processing powered feedback improves 1:1 instruction and student outcomes. In *Proceedings of the 10th ACM Conference on Learning @ Scale*, pages 59–69, New York, NY, USA, 2023. ACM.

[9] D. Demszky, J. Liu, Z. Mancenido, J. Cohen, H. C. Hill, D. Jurafsky, and T. B. Hashimoto. Measuring conversational uptake: A case study on student-teacher interactions. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 1638–1653, 2021.

[10] S. D'Mello and A. Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, 2012.

[11] S. D'Mello, A. Olney, and N. Person. Mining collaborative patterns in tutorial dialogues. *Journal of Educational Data Mining*, 2(1):1–37, 2010.

[12] F. Gilardi, M. Alizadeh, and M. Kubli. Chatgpt outperforms crowd-workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 2023.

[13] X. Z. . K. C. Gyeonggeon Lee, Minji Yun. Artificial intelligence in science education research: Current states and challenges. *Journal of Science Education and Technology*, 2025.

[14] A. Jacovi and Y. Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, July 2020.

[15] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

[16] Z. Jiang, J. Araki, H. Ding, and G. Neubig. How can we know when language models know? on the calibration of language models. In B. Roark and A. Nenkova, editors, *Transactions of the Association for Computational Linguistics*, volume 9, pages 962–977, Cambridge, MA, 2021. MIT Press.

[17] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[18] C. Kim, F. Ke, A. Barrett, and N. Zhang. From code to insight: How LLMs help and hinder qualitative research. In *Proceedings of the 26th International Conference on Artificial Intelligence in Education*, pages 28–35, Palermo, Italy, 2025. Springer.

[19] S. Lin, J. Hilton, and O. Evans. Teaching models to express their uncertainty in words. In *Transactions on Machine Learning Research*, 2022.

[20] S. Lin, J. Hilton, and O. Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

[21] X. Liu, A. F. Zambrano, R. S. Baker, A. Barany, J. Ocumpaugh, J. Zhang, M. Pankiewicz, N. Nasiar, and Z. Wei. Qualitative coding with gpt-4: Where it works better. *Journal of Learning Analytics*, 12(1):169–185, Mar. 2025.

[22] A. Madsen, S. Chandar, and S. Reddy. Are self-explanations from large language models faithful? In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 295–337, Bangkok, Thailand, Aug. 2024.

[23] P. Manakul, A. Liusie, and M. J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural*

Language Processing, pages 9004–9017. Association for Computational Linguistics, 2023.

[24] J. McClure, D. Smyslova, A. Hall, and S. Jiang. Deductive coding's role in ai vs. human performance: A comparative analysis of reliability. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 809–813, Atlanta, GA, 2024. International Educational Data Mining Society.

[25] S. M. Mohammad, P. Sobhani, and S. Kiritchenko. Stance and sentiment in tweets. *ACM Trans. Internet Technol.*, 17(3), June 2017.

[26] S. P. Neshaei, R. L. Davis, P. Mejia-Domenzain, T. Nazaretsky, and T. Käser. Bridging the data gap: Using LLMs to augment datasets for text classification. In C. Mills, G. Alexandron, D. Taibi, G. Lo Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 119–132, Palermo, Italy, July 2025. International Educational Data Mining Society.

[27] Z. A. Pardos, R. S. Baker, M. O. San Pedro, S. M. Gowda, and S. M. Gowda. Affective states and state tests: Investigating how affect throughout the school year predicts end of year learning outcomes. In *Proceedings of the third international conference on learning analytics and knowledge*, pages 117–124, 2013.

[28] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn. *The development and psychometric properties of LIWC2015*. University of Texas at Austin, 2015.

[29] M. J. Petrilli. Next-gen classroom observations, powered by ai. *Education Next*, 24(4):47–55, 2024.

[30] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in Neural Information Processing Systems*, 28, 2015.

[31] E. Siedahmed, J. Ocumpaugh, J. Ferris, D. Kodwani, D. Worden, and N. Heffernan. Nonstandard english and the automated scoring of open-ended math problems. In *Proceedings of the 18th International Conference on Educational Data Mining*, Palermo, Italy, 2025. International Educational Data Mining Society.

[32] J. Su, J. Healey, P. Nakov, and C. Cardie. Between underthinking and overthinking: An empirical study of reasoning length and correctness in llms. *arXiv preprint arXiv:2505.00127*, 2025.

[33] A. Suresh, J. Jacobs, C. Harty, M. Perkoff, J. H. Martin, and T. Sumner. The talkmoves dataset: K–12 mathematics lesson transcripts annotated for teacher and student discursive moves. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022)*, pages 4654–4662, Marseille, France, 2022. European Language Resources Association.

[34] L. Tao, Y.-F. Yeh, B. Kai, M. Dong, T. Huang, T. A. Lamb, J. Yu, P. H. Torr, and C. Xu. Can large language models express uncertainty like humans? *arXiv preprint arXiv:2509.24202*, 2025.

[35] K. Tian, E. Mitchell, A. Zhou, A. Sharma, R. Rafailov, H. Yao, C. Finn, and C. D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*, 2023.

[36] N. Tran, D. Litman, and A. Godley. Using large language models to analyze students' collaborative argumentation in classroom discussions. In *Proceedings of the Artificial Intelligence in Measurement and Education Conference (AIME-Con): Full Papers*, pages 111–125, 2025.

[37] N. Tran, B. Pierce, D. Litman, R. Correnti, and L. C. Matsumura. Analyzing large language models for classroom discussion assessment. *arXiv preprint arXiv:2406.08680*, 2024.

[38] M. Turpin, J. Michael, E. Perez, and S. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 74952–74965. Curran Associates, Inc., 2023.

[39] K. Vanacore, A. Barany, S. Mehta, J. Ocumpaugh, R. S. Baker, N. Al Nahin, V. Kuvar, C. Mills, and O. Henkel. Downshifting: The nuanced antecedents of seeking easier content. In *Proceedings of the 19th International Conference of the Learning Sciences-ICLS 2025, pp. 637-645*, pages 637–645. International Society of the Learning Sciences, 2025.

[40] K. Vanacore and R. F. Kizilcec. How well do large language models recognize instructional moves? establishing baselines for foundation models in educational discourse. *arXiv preprint arXiv:2512.19903*, 2025.

[41] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

[42] J. Wei, X. Wang, D. Schuurmans, M. Bosma, b. ichter, F. Xia, E. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022.

[43] J. Whitehill and J. LoCasale-Crouch. Automated evaluation of classroom instructional support with llms and bows: Connecting global predictions to specific feedback. *arXiv preprint arXiv:2310.01132*, 2023.

[44] Z. Xia, J. Xu, Y. Zhang, and H. Liu. A survey of uncertainty estimation methods on large language models. *https://arxiv.org/abs/2503.00172*, 2025.

[45] A. F. Zambrano, X. Liu, A. Barany, R. S. Baker, J. Kim, and N. Nasiar. From ncoder to chatgpt: From automated coding to refining human coding. In G. Arastoopour Irgens and S. Knight, editors, *Advances in Quantitative Ethnography*, page 470–485, Cham, 2023. Springer Nature Switzerland.

[46] A. F. Zambrano, Z. Wei, J. Zhang, R. S. Baker, J. Ocumpaugh, A. Barany, X. Liu, Y. Zhou, L. Paquette, and J. Ginger. Data plus theory equals codebook: Leveraging llms for human-ai codebook development. *Journal of Educational Data Mining*, 18(1):25–65, 2026.

[47] R. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using large language models to detect self-regulated

learning in think-aloud protocols. In *Proceedings of the 17th international conference on educational data mining*, pages 157–168, 2024.

[48] S. Zhang, T. Yu, and Y. Feng. Truthx: Alleviating hallucinations by editing large language models in truthful space. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8908–8949, 2024.

[49] C. Ziems, W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang. Can large language models transform computational social science? *Computational Linguistics*, 50(1):237–291, 2024.

Leveraging Privileged Multimodal Data to Model Cognitive States in an Interactive Learning Platform

Deniz Sonmez Unal
University of Pittsburgh
Pittsburgh, PA 15260
des204@pitt.edu

Theresa Mowad
Lehigh University
Bethlehem, PA 18015
tgm320@lehigh.edu

Catherine M. Arrington
Lehigh University
Bethlehem, PA 18015
cma205@lehigh.edu

Erin Solovey
Worcester Polytechnic Institute
Worcester, MA 01609
esolovey@wpi.edu

Erin Walker
University of Pittsburgh
Pittsburgh, PA 15260
eawalker@pitt.edu

ABSTRACT

Student models in intelligent tutoring systems (ITS) often rely on large-scale behavioral data (e.g., correctness, response time), which provides only indirect evidence of cognition. While multimodal data such as think-alouds or screen recordings can offer richer insight into students' cognitive states, such data is rarely available at scale, limiting the practicality of multimodal models. To address this, we leverage Learning Using Privileged Information (LUPI) framework for cognitive state modeling in ITS. Our approach incorporates high-quality think-aloud data and written action descriptions from a subset of students at training time to inform model decision boundaries, while using only behavioral data at inference. This setup mimics real-world deployment where rich modalities are unavailable for most learners. We evaluate this method in modeling rule learning, an established framework for conceptual development during problem solving, and find improved predictive accuracy, particularly for nuanced cognitive states not well captured by behavioral logs alone.

Keywords

Learning Using Privileged Information, Rule Learning, Multimodal Learning Analytics

1. INTRODUCTION

Interactive learning platforms such as intelligent tutoring systems (ITS) aim to provide personalized learning support by modeling students' knowledge, behavior, and cognitive states [38]. These models are often built using large amounts of data collected during student interactions with the system. Among these, behavioral log data such as correctness, response time, or hint usage are the most widely used sources of information, largely because they are easy to collect and available across many systems. However, while behavioral data are abundant and convenient, they are also discrete and indirect, often failing to capture the more subtle or continuous aspects of learners' cognitive processes [19].

In recent years, the field has increasingly turned to multimodal approaches to capture richer representations of student learning [3, 10, 13]. These approaches integrate behavioral logs with other data sources such as eye tracking, facial expressions, neural signals, and verbal protocols to gain deeper insight into students' thought processes and cognitive states. Among these, think-aloud protocols are particularly valuable for identifying real-time cognitive activity due to their interpretability [21]. Think-aloud protocols are widely used across HCI and cognitive science as a method for capturing users' internal reasoning in real time [14, 28]. They involve asking individuals to verbalize their thoughts while performing a task in order to gain access to moment-to-moment reactions and cognitive processing that are otherwise difficult to infer. In educational contexts, think-aloud protocols serve a similar purpose: they reveal how learners engage with problems, what they understand, and where they struggle. This form of verbal data has been shown to provide insights into learners' cognitive processes, and specifically in uncovering processes like self-regulated learning strategies [17, 43] and metacognitive monitoring and control [25, 31] that remain invisible in log data. Complementing these verbalizations, another heavily used modality is external observations which captures behaviors that learners may not articulate explicitly. This may include researcher coding of video data, screen recordings, or in situ observations of classroom behaviors. Prior work has shown such observations can reveal cognitive and affective states, and instances of disengagement [1, 27].

Even though multimodal methods are extremely helpful in providing a broader picture of students' cognition, a key limitation of many multimodal methods is that they rely on non-deployable signals or data that are easy to collect in controlled lab settings, but infeasible for real-world environments. As a result, models trained on rich multimodal data often suffer from limited generalizability and are difficult to apply in practice [26]. To address this challenge, we propose framing cognitive state modeling within the Learning

Deniz Sonmez Unal, Theresa Mowad, Catherine Arrington, Erin Solovey, and Erin Walker. Leveraging Privileged Multimodal Data to Model Cognitive States in an Interactive Learning Platform. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 155–166. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039992

Using Privileged Information (LUPI) paradigm [39]. LUPI is a machine learning technique in which models are trained using additional information, called privileged information, that is available only at training time but not at test time. This privileged information can include richer or more explanatory features that help the model learn a better decision boundary during training. At inference time, the model operates solely on the standard input features. It ensures generalization while benefiting from the additional insight gained during training. In this work, we hypothesize that LUPI enables the use of rich but harder-to-collect information such as verbal think-alouds or action descriptions during model training, while maintaining deployability by relying only on behavioral data at inference time. In doing so, it offers a powerful approach to bridge the gap between what learners experience and what can be captured externally.

In this work, we model phases of rule learning [7] as our target cognitive states. The rule learning framework offers a formalization for cognitive states, such as moving from unfamiliarity to partial understanding to fluent application, and reflects core aspects of skill acquisition [33]. Our key contributions are as follows: (1) We introduce a novel application of LUPI to model rule learning states in ITS contexts, using multimodal data that combines behavioral logs with verbal protocol data. (2) We demonstrate that incorporating verbal protocol data during training significantly improves model performance, particularly for cognitive states that are challenging to infer from behavior alone.

2. RELATED WORK

2.1 Multimodal Student Modeling

Beyond behavioral log data, researchers have increasingly turned to multimodal approaches to capture a richer picture of learning processes. These approaches integrate multiple data streams such as eye-tracking, speech, hand-writing to improve student modeling and provide insight into cognitive and affective states that are difficult to detect from logged behavioral information alone [8].

Textual data has been widely explored within multimodal student modeling, since it provides direct evidence of learners' reasoning and cognitive states in real time [2]. Within this broader category, think-aloud protocols represent a particularly valuable form of textual data. Recent advances in educational data mining and AI in education have increasingly incorporated think-aloud protocols as rich sources of cognitive and metacognitive data for machine learning applications. Zhang et al. demonstrated the potential of large language models to automatically detect self-regulated learning (SRL) behaviors from transcribed think-alouds, enabling predictive modeling of learning strategies without the need for manual annotation [43]. Similarly, Li et al. leveraged topic modeling and supervised classification on think-aloud transcripts to predict students' reasoning activities during computer-assisted learning, showing that TAP-derived features can robustly model cognitive patterns [21]. Tabarsi et al. triangulated verbal and behavioral data, using coded think-aloud segments alongside video and interaction logs to train data-driven detectors for identifying programming difficulties among novice learners [35].

Another complementary modality is human observation, where researchers code student behaviors to capture aspects of learning not visible in interaction logs. Early work in ITS relied heavily on expert observers to identify cognitive and affective states. For example, Baker et al. [1] used classroom observations of Cognitive Tutor users to code when students were gaming the system, and these observations provided the training data for generalizable automated detectors. Similarly, the Baker-Rodrigo Observation Method Protocol (BROMP) [27] has been used to code affective states such as boredom, frustration, and confusion. Other studies have relied on expert coding of more complex behaviors from screen replays or video recordings [34]. D'Mello et al. combined human annotation of affect during learning sessions with system logs to train multimodal models of affect detection [12]. Paquette et al. [29] examined expert coding of disengagement in online learning environments. They showed that coders rely on contextual cues rather than isolated actions when making judgments about student behavior. Recent studies have explored human-coded observational data as features in predictive models. Tian et al. demonstrated how textual artifacts such as rubric-based classroom observations and field notes can be encoded and used in supervised learning to predict instructional quality and engagement [36]. Wu utilized graph convolutional networks (GCNs) on structured classroom observation datasets to evaluate teaching effectiveness at scale [42]. In another study, Petković et al. combined manual coding of teacher nonverbal behaviors with computer vision and machine learning techniques to assess instructional immediacy and its impact on student engagement [30]. In our work, we use written notes from screen recordings as a form of human observation. This allows us to capture detailed traces of students' actions during problem solving and relate them to underlying cognitive states.

2.2 Learning Using Privileged Information

The Learning Using Privileged Information (LUPI) paradigm was first introduced by Vapnik and Vashist [39]. The key idea is that during training, the model has access to additional *privileged* information that is not available at test time but can help the learner form a more accurate decision boundary. This setup mirrors real-world scenarios in which richer explanatory features (e.g., expert feedback, annotations, or multimodal data) are accessible during the development of a model but are impractical or impossible to collect consistently during deployment. Vapnik describes this idea as taking inspiration from human learning: a teacher may provide additional explanations, hints, or demonstrations when introducing a challenging concept, enabling the student to build a deeper understanding. However, once the student sits for an exam, these supports are no longer available, and the student must rely only on what they have internalized.

Researchers have used LUPI in a number of different HCI and educational settings. Vrigkas et al. [40] used LUPI for a human activity recognition task. They introduced a model that combined standard video features with additional data such as audio, pose data, or semantic annotations as privileged information. Their results showed that including this privileged information improved recognition accuracy across several activity datasets. Another line of work applies LUPI to affective computing, focusing on the

challenge of transferring models trained in the lab to real-world settings. Makantasis et al. [24] introduced the use of privileged information to bridge this gap. In their framework, high-quality multimodal signals available in controlled lab settings (e.g., audio, facial features, and physiological measures such as ECG and EDA) were treated as privileged information, while raw video frames were considered the only modality available "in the wild". They showed that performance could approach that of full multimodal models, even in noisy, uncontrolled conditions. Wang et al. [41] also used LUPI in emotion recognition, where EEG signals and video features were treated as complementary modalities: each served as privileged information for improving the other task. This work illustrated how LUPI can exploit correlations across modalities to enhance models when one modality is unavailable at test time. In the education domain, Zhao et al. [44] explored LUPI for graduate admissions prediction. They proposed S3VM+, which combined semi-supervised learning with LUPI to handle biases in committee decisions across years and to exploit large amounts of unlabeled application data. By treating yearly committee differences as privileged information, their approach outperformed standard baselines in identifying top-performing students, demonstrating LUPI's potential in educational data mining.

Our work builds on this line of research by applying the LUPI paradigm to model cognitive states in intelligent tutoring systems. Similar to the activity recognition and affective computing studies, our approach addresses the gap between rich but impractical signals and deployable features. Like Zhao et al., we operate in an educational context, but instead of admissions, we focus on modeling fine-grained cognitive states during problem solving.

2.3 Rule Learning

Inductive reasoning plays a central role in learning. It involves observing instances of a phenomenon, identifying patterns and relationships among them, and forming general rules that can be applied to new situations. The sequence of steps in this process is often described as the phases of rule learning. In cognitive science, these phases have traditionally been examined through variations of the rule attainment task [4, 7]. Such tasks are designed to capture three key phases of rule learning: rule search, rule discovery, and rule following. Typically, participants respond to a series of stimuli, indicating whether they believe the stimuli follow a rule or appear random. For example, in one version, learners see sequences of numbers that conform to a simple mathematical pattern (e.g., +2 leading to 32, 34, 36...) [20]. In another, they view numbered circles where the position of a blue circle shifts according to an underlying rule [5]. Across these tasks, specific behavioral markers make it possible to identify rule learning phases independently of the exact stimuli. This offers a standardized framework for analyzing learning processes across contexts.

Building on this tradition, our study adopts an extended set of five rule learning phases as the formalization of cognitive states (Table 1). These phases reflect how the constructs originally studied in abstract tasks translate into real-world educational problem solving:

Rule Search: Segments where learners express uncertainty or speculate without confidence about how to proceed.

Rule Discovery: Segments where learners suddenly recognize a useful principle or pattern that helps solve the problem.

Rule Following: Segments where learners apply a correct rule confidently and without hesitation.

Rule Violation: Segments where learners attempt to apply a previously correct rule but make a slip or error.

Follow Wrong: Segments where learners confidently apply a rule they believe is correct, but it reflects a misconception.

While most prior work has focused on abstract laboratory tasks, these phases have also been shown to manifest in real-world educational problem solving [37, 33]. In this study, we adopt the phases of rule learning as our formalization of cognitive states. This framing allows us to capture how learners progress from uncertainty to understanding and application. Higher-level abstractions, such as knowledge components or skills, are useful for structuring instruction but often too coarse to capture the nuanced, moment-to-moment shifts that occur during problem solving. In contrast, rule learning phases parallel these abstractions while making visible more subtle states that are otherwise difficult to define or detect. At the same time, the phases are sufficiently well-defined in cognitive science to provide a stable target for modeling.

3. LEARNING TASK DESIGN

Our study was conducted using the ASSISTments platform, an online learning environment that that supports the creation of custom problems, interactive scaffolds, and adaptive feedback. [18]. We authored 9 probability problems in the platform. Each problem in our task included multiple probability rules, which corresponded to the underlying skills or concepts required for successful problem solving. The rules covered in our problem set were: 1) identifying the number of favorable outcomes, 2) identifying the number of possible outcomes, 3) calculating probability of an event, 4) finding the probability of only one event when two events can occur together, 5) finding the probability of either of the two events happening but not both, 6) finding the conditional probability of an event, and 7) finding the probability of dependent events. Students were given access to on-demand scaffolding that allows them to break down problems into subproblems as needed (see Figure 2). Each scaffolded subproblem was associated with a single probability rule only. In addition to requesting scaffolds, students could also ask for a hint or reveal the answer if they were completely stuck.

To capture students' cognitive states throughout the learning process, we introduced a knowledge indication panel that prompts learners to self-report their current rule learning phase. Students respond to this panel whenever a new problem or scaffold appears on screen, before they can proceed with solving (see Figure 1). Their choices reflect phases such as "I know how to solve this problem" or "I don't know how to solve this problem" to initially label students as being in a rule search or rule following state.

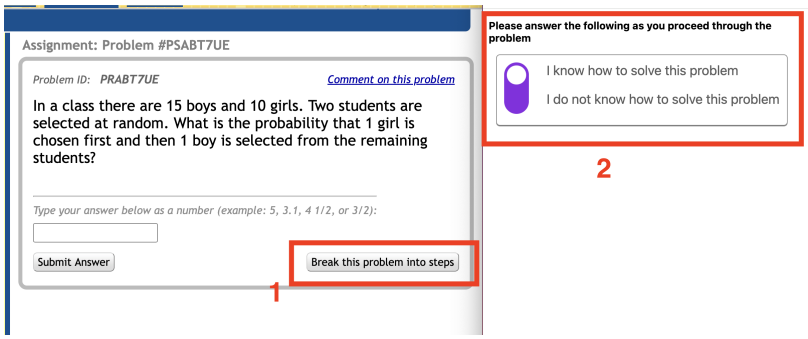


Figure 1: Screenshot of a sample problem. 1: Participants ask to divide the problem into smaller steps by clicking on this button. 2: The knowledge indication panel. Participants need to choose one of the two states on the toggle to answer the current problem on the screen.

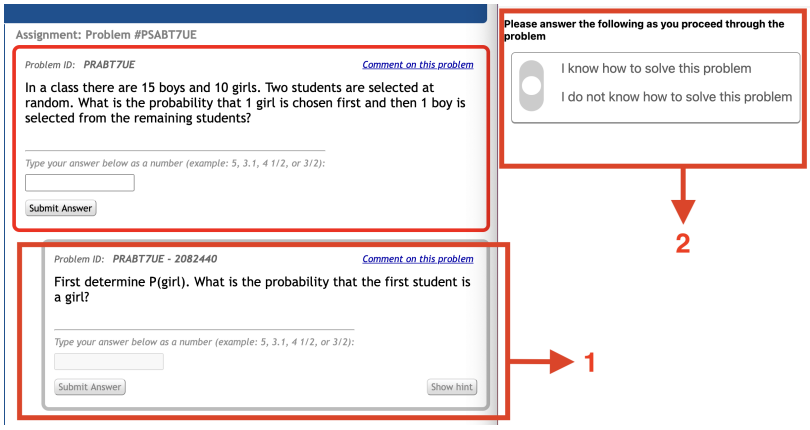


Figure 2: A sample problem showing how scaffolding is provided. 1: Once the participant clicks on the "Break this problem into steps" button shown in Figure 1, the first substep of the problem unfolds. 2: Participants continue to respond to the toggle to indicate their knowledge on the current problem for the substeps of the problem as well. For each new problem shown on the screen (either a main problem or a substep), the toggle is reset and the current problem is disabled until the participant switches the toggle.

In cases where a student's problem-solving response contradicts their indicated knowledge state (e.g., incorrect answer despite claiming to know how to solve the problem), the system triggers additional prompts to refine our understanding of their learning status. These follow-up questions help disambiguate whether the student was having a misconception (follow wrong state), was just making a guess (rule search state), made a simple mistake like a typo or miscalculation (rule violation state), or encountering a different kind of difficulty (Figure 3).

4. DATASET

4.1 Experimental Procedures

42 university students completed a combination of problem solving tasks and a rule learning task. All participants gave their informed consent prior to their inclusion and received course credit. They agreed to their audio and visual information being recorded and the collected data was de-identified prior to analysis. All study procedures were reviewed and approved by the university's Institutional Review Board. Participants began the experiment by solving a set of probability problems using ASSISTments. Initially, they were introduced to a practice set of problems to familiarize themselves with the platform. After completing the practice problems, participants proceeded to the pretest, which included 7 probability questions designed to be completed within 15 minutes. Upon completion of the pre-test, participants engaged in the rule learning task. Participants were instructed on how they could access the on-demand scaffolding by clicking on "Break This Problem Into Steps" button as well as how to use the right hand panel to indicate whether they knew how to solve the problem. After the practice, participants proceeded to solve 9 probability problems that were designed to be completed within 20 minutes using ASSISTments. They were allowed to use pen, paper, and a calculator. They were instructed to think aloud while solving the problems. The final phase involved a post-test consisting of 7 problems similar to those in the pre-test.

We collected a multimodal dataset during a structured problem solving task in the ASSISTments online tutoring system. The dataset includes behavioral interaction logs, verbal think-aloud data, and screen recordings of participants' sessions.

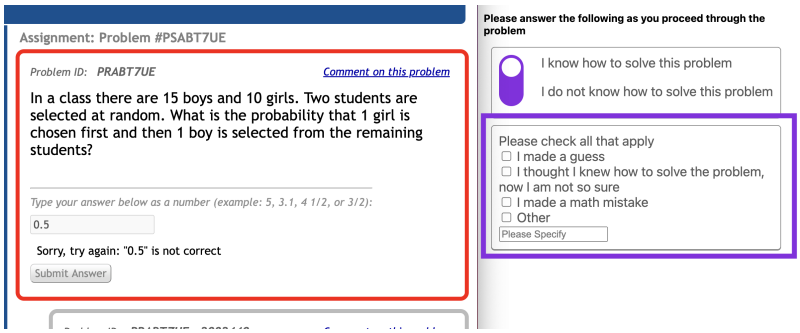


Figure 3: If the participant claims to know how to solve the current problem but provides an incorrect response or asks for scaffolding, another panel with additional options for the knowledge indication (highlighted in purple) unfolds to clarify the reason for the incorrect answer or the help request.

4.2 Behavioral Log Data

As participants interacted with the problems, we recorded detailed behavioral logs. These included the current action the participant was performing on the ASSISTments platform (submitting an answer, requesting a hint, requesting to break the problem into steps, or revealing the solution), correctness of the participants' answers to the problems if the action was an answer attempt, the items participants click on the knowledge indication panel, and timestamps of each action. The resulting dataset contains 2942 rows, with each row representing an action taken by a specific participant during the session.

4.3 Verbal Think-Aloud Data

During the task, students were asked to think aloud, verbalizing their reasoning as they worked through each problem. These verbalizations were recorded and subsequently transcribed to produce sentence-level textual data aligned with the students' actions. The transcriptions capture cognitive and metacognitive insights, such as moments of uncertainty, strategic decisions, and explanations of problem-solving steps.

4.4 Observation Notes from Screen Recordings

In addition to verbal data, we collected screen recordings of each session. These recordings were reviewed by trained coders to generate written descriptions of student actions, particularly for cases where system logs lacked sufficient context. These descriptions include observations such as hesitation before submitting an answer, switching between problem elements, or interpreting problem prompts. When combined with the think-aloud transcripts, these descriptions help contextualize the behavioral trace with higher granularity.

4.5 Labeling Cognitive States

To label cognitive states, we used a hybrid annotation approach based on both self-report data and manual coding. As mentioned in Section 3, Students were asked to complete a knowledge indication panel prior to solving each problem or scaffolded subproblem. This panel prompted them to indicate their perceived level of knowledge about the underlying rule, providing a continuous stream of self-reported rule learning states.

In cases where students provided verbal think-alouds aligned with specific actions, we used these transcriptions, along with the screen-recorded behaviors, to refine the state labels. Annotation was guided by a structured coding scheme for rule learning phases adapted from Unal et al. [33] (see Table 1), allowing coders to disambiguate ambiguous or inconsistent self-reports based on verbal evidence.

On top of the five rule learning states described above, we also observed a few instances of gaming the system in the screen recordings. Following Baker et al. [1], we flagged behaviors such as rapidly clicking through hints, repeatedly guessing until correct, or otherwise exploiting system feedback. These cases were coded but excluded from analysis, as gaming reflects disengagement rather than the cognitive processes underlying rule learning.

To ensure the reliability of our coding scheme, two coders independently annotated 20% of the dataset. Interrater agreement was assessed using Cohen's kappa, yielding a value of 0.84. Frequency of each label is presented in Table 2.

5. METHOD

We implemented a LUPI framework using knowledge distillation which has become one of the standard ways for training models with information that is only available at learning time [22]. In our setup, different "teacher" models are trained with both behavioral features from interaction logs and privileged features from either student think-alouds or observation notes from screen recordings or a combination of both privileged data sources. The teacher models' outputs are then used to guide a "student" model that only has access to behavioral features (see Section 5.2for more details). This design allows the student models to benefit from the richer representations learned by their teachers, while remaining deployable in real-world settings where privileged features are not available.

5.1 Feature Representation

In this modeling scheme, we have 2 sets of features. The first set is regular features that are extracted from student interaction logs within the ASSISTments platform. These features reflect observable student behavior and are available for real-time inference. Our second feature set is called the privileged feature set. Privileged features consisted of

Table 1: Coding Scheme Labels and Descriptions.

Label	Description	Example
Rule Search	Segments where learners express uncertainty or lack of knowledge about how to proceed. They may be speculating, experimenting, or trying different approaches without confidence.	"I'm not sure if this is right... maybe I should divide these numbers first and see what happens."
Rule Discovery	Segments where the learner suddenly recognizes a useful pattern or principle that helps solve the problem.	"Wait, I see now—we need to subtract the overlapping part from the total."
Rule Following	Segments where the learner applies a correct rule with confidence, explaining their reasoning clearly and arriving at an answer without hesitation.	"Since the events are independent, I just multiply 3/10 by 2/5 to get 6/50."
Rule Violation	Segments where the learner attempts to apply a previously correct rule but makes a mistake, often due to a slip or calculation error.	"Oops, I added those fractions and got 7/12... oh, I should have found a common denominator first."
Follow Wrong	A case where the learner confidently applies a rule they believe to be correct, but it is actually a misconception. They consistently follow the wrong logic.	"To find the total probability, we always add the numbers—so 0.3 plus 0.6 gives 0.9." (when multiplication is required).

Table 2: Frequency of cognitive states in the dataset.

Follow Wrong	Rule Discovery	Rule Following	Rule Search	Rule Violation
259	446	1248	898	91

verbal thinkaloud data, in which students articulated their reasoning while solving problems, and post-hoc observation notes transcribed from screen recordings. See Table 3 for the full list of features and their descriptions.

Verbal data streams were encoded using the all-MiniLM-L6-v2, a Sentence Transformer model that maps sentences to a 384 dimensional dense vector space to produce dense vector embeddings [32]. For instances where no think-aloud or observation note was available, we inserted a special "non textual" tag to mark the absence of verbal data.

5.2 Models & Training

The training pipeline involves two types of models: a teacher model trained with both behavioral and privileged features, and a student model trained solely on behavioral features using soft labels generated by the teacher. The idea is that the teacher, having access to richer information that privileged features provide, produces a probability distribution over the labels rather than a single hard label. The student model is then trained on this distribution instead of hard one-hot targets. Because the teacher's output reflects how it weighs similarities between classes, for example, assigning probability mass to both Rule Search and Follow Wrong states when the evidence is ambiguous, the student model gains a better sense of which states are easily confused. This provides a smoother and more informative training signal, allowing the student model to learn about subtle distinctions that are especially difficult to detect from behavioral features alone. As a result, the student can generalize more effectively to uncertain or hard-to-classify cases.

Let $x \in \mathbb{R}^d$ represent the behavioral feature vector and $x^* \in \mathbb{R}^{d^*}$ the privileged feature vector derived from participants' verbal think-alouds and/or observation notes from screen recordings. The teacher model f_T, receives the concatenated input $x_{full} = [x, x^*] \in \mathbb{R}^{d+d^*}$ and produces logits $z_T = f_T(x_{full}) \in \mathbb{R}^k$ where k is the number of cognitive states. The teacher is trained to minimize the standard cross-entropy loss:

$$\mathcal{L}\text{CE} = -\sum i = 1^K y_i \log\left(\text{softmax}(\mathbf{z}_T)_i\right) \tag{1}$$

where $y \in \{0,1\}^K$ is the one-hot encoded ground truth label.

For distillation, we applied temperature scaling to soften the teacher's output distribution. The softened probabilities are defined as:

$$p_T^{(i)} = \frac{\exp(z_T^{(i)}/T)}{\sum_{j=1}^K \exp(z_T^{(j)}/T)} \tag{2}$$

where $T > 1$ is the temperature parameter. Higher temperatures produce smoother distributions, providing the student with more informative gradients.

The student model f_S is trained on behavioral features x and outputs logits $\mathbf{z}_S = f_S(x)$. Its probability distribution with temperature scaling is given by:

$$p_S^{(i)} = \frac{\exp(z_S^{(i)}/T)}{\sum_{j=1}^K \exp(z_S^{(j)}/T)} \tag{3}$$

The student is optimized using a hybrid loss that combines cross-entropy with ground-truth labels and the KL diver-

Table 3: Regular and privileged features and their explanations.

Space	Feature	Description
x	Action	The type of the action on the current problem. Possible action types are answer attempts, hint or scaffolding requests, self reflection actions on the knowledge indication prompt, and next-button clicks.
	FirstAction	Whether the type of the first action on the current problem was an answer attempt or hint request.
	Rule	The rules associated with the current problem will be represented with a 7-dimensional vector (for the 7 distinct rules in our problem set). Each problem is tagged with 1 to 3 rules.
	TimeOnProblem	Total time spent on the current problem measured in milliseconds.
	FirstRT	First response time to the current problem measured in millisecond.
	IsMainProblem	Whether the current problem is a main problem or a sub-problem.
	AttemptCount	Number of attempts on the current problem.
	NumRule	A 7-dimensional vector representing the cumulative count of exposures to each rule up to the current problem.
	RollingMeanTime	The rolling mean of the time spent on previous NumRule problem steps for each rule that is associated with the current problem.
	SuccessRate	Proportion of correctly solved steps in the previous NumRule steps for each rule that is associated with the current problem.
${x_1}^*$	Verbal thinkalouds	Sentences in the transcriptions of student think-alouds were represented with dense sentence embeddings.
${x_2}^*$	Observation notes	Verbal descriptions of students' actions as seen on the screen recordings were represented using the same methods as we did for ${x_1}^*$.

gence with the teacher's soft labels:

$$\mathcal{L} = \alpha \mathcal{L}_{\text{CE}}(y, \text{softmax}(\mathbf{z}S)) + (1-\alpha)T^2 \mathcal{L}\text{KD}(p_T, p_S) \quad (4)$$

where

$$\mathcal{L}\text{KD}(p_T, p_S) = \sum i = 1^K p_T^{(i)} \log\left(\frac{p_T^{(i)}}{p_S^{(i)}}\right). \quad (5)$$

Here, $\alpha \in [0,1]$ controls the balance between matching the hard ground-truth labels and distilling knowledge from the teacher's softened predictions. The T^2 factor accounts for the scaling effect of the temperature on gradients.

Both teacher and student models used the same feedforward neural network architecture with two hidden layers. Hidden layers had 128 units in teacher model and 64 units in the student model. Each hidden layer used ReLU activation functions and batch normalization. The output layer used a softmax activation to produce probability distributions over the cognitive states. Models were trained using the Adam optimizer with a learning rate of 0.001 , batch size 32, and trained for 50 epochs. To select these settings, we performed a grid search over key hyperparameters, including learning rates $\{1e-2, 1e-3, 1e-4\}$, hidden layer sizes $\{64, 128, 256\}$, batch sizes $\{16, 32, 64\}$, temperature values $\{1, 2, 4, 8\}$, and hybrid loss weights $\alpha \in \{0.25, 0.5, 0.75\}$. The final configuration was chosen based on average F1 scores across folds on the validation set.

Because observation notes were produced by researchers, they were consistently high-quality and did not require further stratification. In contrast, think-alouds varied in quality so we categorized participants' think-aloud data into three groups to account for this variability. Participants who consistently articulated their reasoning and problem-solving strategies were classified as high-quality think-aloud participants ($n = 18$). Those who spoke but primarily read the problem text aloud or stated answers without reflecting on their reasoning were labeled as low-quality think-aloud participants ($n = 17$). Finally, participants who remained silent throughout were assigned to the silent group ($n = 7$). For model training, we stratified participants across these three groups when creating training and validation splits to ensure balanced representation. We used 5-fold cross validation at the participant level. We randomly oversampled participants who provided high quality think-alouds in the training folds to make sure that teacher models were exposed to richer privileged features and to strengthen the supervision signal for the student (LUPI) models. No data from the same participant appeared in both training and validation sets within a fold.

For evaluation, we computed per-class F1 scores on the validation set and. The baseline model was a behavioral-only network trained using cross-entropy loss with hard labels. Performance of the LUPI-trained student model was compared against this baseline and the privileged teacher to assess gains attributable to LUPI.

6. RESULTS

To evaluate the impact of incorporating privileged information during training, we compare performance of seven models using average F1 scores over folds. All models were trained and evaluated on the same participant-level cross-validation splits to ensure comparability. The teacher models differed in the privileged features (behavioral features were available to all models) they were given during training:

Teacher-Full had access to both think-alouds and observation notes, *Teacher-TA* had only think-alouds, and *Teacher-Obs* had only observation data. Each teacher produced soft labels (class probability distributions), which were then used to train the corresponding student (LUPI) models: *LUPI-Full*, *LUPI-TA*, and *LUPI-Obs*, that relied exclusively on behavioral features. In this setup, the student models did not differ in the features they used, but in the type of labels they were trained using: they learned from soft label distributions distilled from their respective teachers. Finally, we trained a baseline behavioral-only model using hard labels, without access to any privileged features or soft targets from a teacher.

Because evaluation was conducted using 5-fold participant-level cross validation on a relatively small dataset, we focus on the consistency and magnitude of observed performance trends rather than formal statistical significance testing. We first examined the overall performance of each model (Figure 4). The teacher models outperformed both the baseline and the LUPI models with *Teacher-Obs* achieving the highest average F1 score. This indicates that observation notes from screen recordings provided the strongest supervision signal. In contrast, *Teacher-TA* lagged behind the other teacher models which suggests think-alouds were a noisier source of privileged information.

The performance gains of the LUPI models over the baseline were modestly positive even though they did not match the teacher models' performance. This is natural as the LUPI models are smaller and rely only on behavioral features at inference. Importantly, the LUPI models achieved these gains while maintaining standard errors comparable to the baseline model, suggesting that the distillation process improved performance without increasing variability across folds. These results suggest that the distillation process provides a robust representation that generalizes well to unseen data, even in the absence of privileged information.

Having established the overall performance results, we next turn to a class-wise analysis of F1 scores to better understand where different types of privileged information contributed most to model improvements. As shown in Figure 5, performance varied across cognitive states. All models achieved very high F1 scores for Rule Following (above 0.94), suggesting that once participants are consistently applying the correct rule, behavioral information alone is largely sufficient for accurate classification. In Rule Discovery, performance differences were comparatively modest, with the baseline achieving an F1 score of approximately 0.81, LUPI models ranging between 0.79-0.81, and teacher models reaching 0.83-0.84. Unlike the more difficult states, the gains from privileged information in Rule Discovery were relatively small, suggesting that behavioral features already captured much of the information needed for classification. A similar pattern was observed in Rule Search, where the baseline already performed strongly (approximately 0.86 F1). Most LUPI models remained close to baseline performance, while Teacher-Obs achieved the strongest results (approximately 0.89 F1). This suggests that privileged information still provided additional benefit in distinguishing exploratory reasoning patterns, although the improvements were less pronounced than those observed in more subtle cognitive states such as Follow Wrong and Rule Violation.

In contrast, privileged information had the strongest impact on the more subtle states. For example, in Follow Wrong, the baseline achieved an F1 of 0.78, while LUPI models improved slightly (0.79-0.80) and teacher models, particularly *Teacher-Obs* substantially higher performance (0.85). This reflects the fact that observation notes capture students' underlying strategies and misconceptions more directly than verbalizations or behavioral logs. Similarly, Rule Violation proved to be the most challenging state overall. The baseline was lowest at 0.59 but LUPI models improved to (0.68) and teacher models pushed F1 substantially higher with *Teacher-Full* at 0.82 and *Teacher-Obs* at 0.85. Interestingly, while *Teacher-TA* did not achieve comparable scores, the corresponding *LUPI-TA* model outperformed it. This can be attributed to the hybrid loss used in training, which allowed the LUPI model to balance between its teacher model's soft predictions and the direct behavioral signal. In cases where think-alouds were noisy or not informative, the LUPI model effectively "listened" more to the behavioral data, resulting in stronger generalization than its teacher.

Our results also highlight important differences between the types of privileged information used by the teacher models. Across most cognitive states, Teacher-Obs achieved the strongest or near-strongest performance, while Teacher-Full also consistently outperformed the baseline and LUPI models. This suggests that observation-based privileged information provided a particularly strong supervision signal for detecting students' cognitive states. At the same time, the competitive performance of Teacher-Full indicates that combining modalities can still provide complementary information, even if the additional think-aloud features did not always translate into higher F1 scores. When the modalities were examined separately, Teacher-Obs generally outperformed Teacher-TA. One explanation to why Teacher-Obs mostly outperformed Teacher-TA can be that the observation data was a stronger signal on its own. Even though think-alouds provided more raw data, they were often noisier, since many verbalizations are generic and occur across multiple states. For example, students may express surprise after giving an incorrect answer both when they are in a Follow Wrong phase (reflecting a deeper misconception or flawed mental model) and when they are in a Rule Violation phase (reflecting a simple slip despite otherwise correct knowledge). In contrast, observation data such as watching how students solve the problem on the calculator (on their computer screen) or sequence their inputs provides evidence of the strategies and mental models they are using. Because of this, observation features may have offered more discriminative information for distinguishing subtle differences between cognitive states.

These differences also carried over to the student models. While the gains over the baseline were more modest than those observed for the teacher models, the LUPI variants demonstrated that privileged information could still improve performance even when only behavioral features were available at inference. The effect was especially noticeable in the more challenging cognitive states, where the quality of the privileged supervision signal appeared to matter most. For example, in Follow Wrong, the Teacher-TA model per-

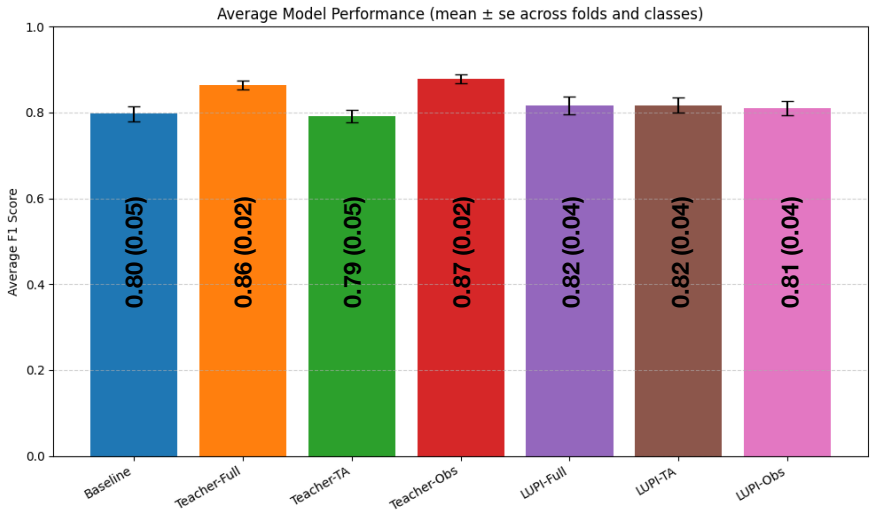


Figure 4: Comparison of Average F1 Scores and standard errors of the Models.

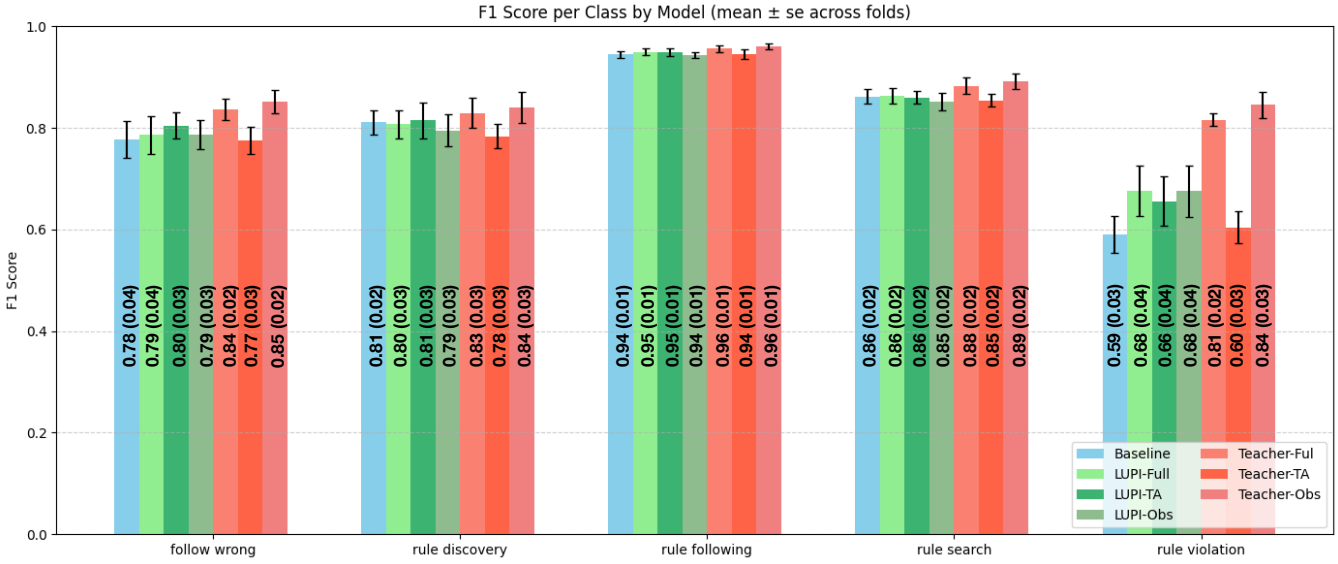


Figure 5: Mean class-wise F1 Scores and Standard Errors of the Models.

formed comparatively poorly, suggesting that think-alouds alone were often too noisy to reliably distinguish deeper misconceptions from neighboring states. However, the corresponding LUPI-TA model achieved stronger performance, likely because the hybrid distillation objective allowed it to balance between the teacher's soft predictions and the ground-truth hard labels derived from behavioral data. In this sense, the student model was able to retain the most informative aspects of the privileged signal while reducing the impact of noisy verbalizations. A similar pattern appeared in Rule Violation, the most difficult state overall. Here, observation-based privileged information proved especially valuable, as both Teacher-Obs and LUPI-Obs substantially outperformed their think-aloud counterparts. This suggests that observation data provided more discriminative cues about students' underlying strategies and slips in reasoning, while think-alouds alone were often insufficiently specific. The strong performance of LUPI-Full further indicates that combining privileged modalities can still be beneficial when the student model learns to integrate comple-

mentary supervision signals through distillation.

7. DISCUSSION

In this work, our goal was to investigate whether incorporating privileged information through a LUPI framework can improve the modeling of cognitive states in a realistic learning environment. We used the phases of rule learning as our target cognitive states, because they offer an appropriate granularity for capturing nuanced, moment-to-moment shifts in cognition that are not easily represented in higher-level abstractions such as mastery / non-mastery in traditional knowledge tracing [6] or broader stages of skill acquisition in cognitive psychology such as cognitive and associative stages [15]. We compared the performance of teacher models with access to privileged information (think-alouds and observation data), student models trained on behavioral data only, optimized using soft labels distilled from those teachers, and a baseline model trained on the same behavioral data, optimized using hard labels. Our findings showed three main patterns. First, teacher models outperformed behavioral-only baselines, demonstrating that privileged information provides valuable insight into the learning process. Second, student models trained with teacher soft labels (LUPI) consistently outperformed the baseline despite relying only on behavioral data at test time. Third, these gains were most pronounced for subtle states such as Follow Wrong and Rule Violation, which are difficult to distinguish from behavioral features alone. The results also revealed differences between modalities. Observation data often provided a stronger standalone signal than think-alouds, but the best results were achieved by combining both. The models demostrated strong generalization as all evaluations were conducted on unseen participants. The consistency of performance across folds highlights the robustness of our approach and the transfer of privileged information benefits to new learners is both reliable and effective.

Beyond predictive performance, the utility of modeling rule learning phases lies in the potential to inform feedback. By detecting nuanced states such as Follow Wrong or Rule Violation in real time, systems could provide targeted interventions that directly address misconceptions or slips without needing additional context about the specific problem, rather than treating all errors as equivalent. Similarly, identifying transitions into Rule Discovery opens opportunities for reflective prompts that help consolidate new understanding. This phase-level precision creates possibilities for adaptive support that is both more timely and more tailored than what is achievable with traditional correctness-based models.

Despite these promising results, our work has several limitations. First, the sample size was modest, which is common in this line of research [11]. While participant-level cross-validation mitigated the risk of overfitting, larger studies will be needed to validate the robustness of our findings. Second, while observation data proved valuable, it was generated through researcher coding of screen recordings, which may limit scalability. Future work should explore automated extraction of similar data. Third, think-aloud data were highly variable in quality, consistent with prior findings that verbal protocols can range from deeply reflective to surface-level or uninformative. Although we stratified participants by think-aloud quality in our cross-validation splits, the variability highlights the need for methods to reduce noise.

Our research can lead to several future directions. Firstly, while this work focused on modeling rule learning phases, the proposed learning framework is not limited to this context. In principle, any type of privileged information that provides deeper access to students' cognitive or affective states could be incorporated at training time. For example, sensor-based modalities such as facial expressions, posture, or physiological signals have long been shown to capture affective dynamics relevant for learning [16], but their reliance on specialized equipment limits classroom deployment. Sensor-free affect detectors developed using behavioral proxies such as timing patterns, hint use, or keystroke dynamics already aim to approximate such states [9]. A promising direction for future work is to train LUPI models on multimodal or sensor-based privileged signals, while using sensor-free behavioral detectors as the deployable counterpart. This would allow researchers to take advantage of the richness of lab-collected affective data while maintaining scalability in real-world systems. A natural extension is to neuroscience, where privileged signals such provide direct access to neural mechanisms that underlie rule learning and other higher-order cognitive processes. Prior research has shown that distinct phases of rule learning are associated with signature patterns of brain activation, including shifts between prefrontal and parietal regions as learners move from rule search to rule following [20]. While such data are difficult to collect outside of controlled laboratory environments, they offer a uniquely fine-grained and objective window into cognition. A LUPI framework could allow us to leverage these neural signatures as privileged information during training, improving the ability of deployable models to infer subtle cognitive states in real time from behavioral logs alone.

Furthermore, an additional line of inquiry is to explore participant level training, where models are individualized based on richer data collected from each learner. Prior work in multimodal learning analytics and neuroscience has shown that such approaches can improve classification accuracy when sufficient data per participant is available [23]. While our study was not designed at that level of granularity, this type of approach could be especially valuable in settings with larger datasets per learner.

Finally, another promising line of research lies in improving the quality of verbal data used as privileged information. Thinkalouds are valuable for revealing reasoning processes, but as observed in our study and others [45], their quality varies widely across learners. Large language models (LLMs) have recently shown promise in evaluating, coding, and even scaffolding verbal explanations in educational contexts [43]. Leveraging LLM based evaluation could help filter noisy or surface level thinkalouds to enhance the reliability of verbal data as privileged information, and even provide real time feedback to learners as they articulate their reasoning. These directions point to a broader role for LUPI in learner modeling, serving as a bridge between modalities that are rich but impractical at scale and the behavioral data streams that are readily available in classrooms and tutoring systems.

8. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under awards #1835307 and #1912474.

9. REFERENCES

[1] R. S. Baker, A. T. Corbett, K. R. Koedinger, and A. Z. Wagner. Off-task behavior in the cognitive tutor classroom: When students" game the system". In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 383–390, 2004.

[2] P. Blikstein. Multimodal learning analytics. In *Proceedings of the third international conference on learning analytics and knowledge*, pages 102–106, 2013.

[3] P. Blikstein and M. Worsley. Multimodal learning analytics and education data mining: Using computational technologies to measure complex learning tasks. *Journal of learning analytics*, 3(2):220–238, 2016.

[4] P. W. Burgess and T. Shallice. Bizarre responses, rule detection and frontal lobe lesions. *Cortex*, 32(2):241–259, 1996.

[5] B. Cao, W. Li, F. Li, and H. Li. Dissociable roles of medial and lateral pfc in rule learning. *Brain and behavior*, 6(11):e00551, 2016.

[6] A. T. Corbett and J. R. Anderson. Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[7] C. Crescentini, S. Seyed-Allaei, N. De Pisapia, J. Jovicich, D. Amati, and T. Shallice. Mechanisms of rule acquisition and rule following in inductive reasoning. *Journal of Neuroscience*, 31(21):7763–7774, 2011.

[8] M. Cukurova, M. Giannakos, and R. Martinez-Maldonado. The promise and challenges of multimodal learning analytics. *British Journal of Educational Technology*, 51(5):1441–1449, 2020.

[9] R. S. d Baker, S. M. Gowda, M. Wixon, J. Kalka, A. Z. Wagner, A. Salvi, V. Aleven, G. W. Kusbit, J. Ocumpaugh, and L. Rossi. Towards sensor-free affect detection in cognitive tutor algebra. *International Educational Data Mining Society*, 2012.

[10] D. Di Mitri, J. Schneider, M. Specht, and H. Drachsler. From signals to knowledge: A conceptual model for multimodal learning analytics. *Journal of Computer Assisted Learning*, 34(4):338–349, 2018.

[11] X. Du, J. Yang, J.-L. Hung, and B. Shelton. Educational Data Mining: A Systematic Review of Research and Emerging Trends. *Information Discovery and Delivery*, 48(4):225–236, 2020.

[12] S. K. D'mello, S. D. Craig, A. Witherspoon, B. McDaniel, and A. Graesser. Automatic detection of learner's affect from conversational cues. *User modeling and user-adapted interaction*, 18(1):45–80, 2008.

[13] A. Emerson, W. Min, J. Rowe, R. Azevedo, and J. Lester. Multimodal predictive student modeling with multi-task transfer learning. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 333–344, 2023.

[14] K. A. Ericsson and H. A. Simon. Verbal reports as data. *Psychological review*, 87(3):215, 1980.

[15] P. M. Fitts. Perceptual-Motor Skill Learning. In *Categories of Human Learning*, pages 243–285. Academic Press, 1964.

[16] M. Giannakos, M. Cukurova, and S. Papavlasopoulou. Sensor-based analytics in education: Lessons learned from research in multimodal learning analytics. In *The multimodal learning analytics handbook*, pages 329–358. Springer, 2022.

[17] J. A. Greene, V. M. Deekens, D. Z. Copeland, and S. Yu. Capturing and modeling self-regulated learning using think-aloud protocols. In *Handbook of self-regulation of learning and performance*, pages 323–337. Routledge, 2017.

[18] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[19] K. Holstein, Z. Yu, J. Sewall, O. Popescu, B. M. McLaren, and V. Aleven. Opening up an intelligent tutoring system development environment for extensible student modeling. In *International conference on artificial intelligence in education*, pages 169–183. Springer, 2018.

[20] F. Li, B. Cao, H. Gao, L. Kuang, and H. Li. Different brain potentials evoked at distinct phases of rule learning. *Psychophysiology*, 49(9):1266–1276, 2012.

[21] S. Li, X. Huang, T. Wang, J. Zheng, and S. P. Lajoie. Using text mining and machine learning to predict reasoning activities from think-aloud transcripts in computer assisted learning. *Journal of Computing in Higher Education*, 37(1):477–496, 2025.

[22] J. Liu, D. Capurro, A. Nguyen, and K. Verspoor. Improving text-based early prediction by distillation from privileged time-series text. *arXiv preprint arXiv:2301.10887*, 2023.

[23] R. Liu, E. Walker, L. Friedman, C. M. Arrington, and E. T. Solovey. fnirs-based classification of mind-wandering with personalized window selection for multimodal learning interfaces. *Journal on multimodal user interfaces*, 15(3):257–272, 2021.

[24] K. Makantasis, K. Pinitas, A. Liapis, and G. N. Yannakakis. From the lab to the wild: Affect modeling via privileged information. *IEEE Transactions on Affective Computing*, 15(2):380–392, 2023.

[25] M. Montague and B. Applegate. Middle school students' mathematical problem solving: An analysis of think-aloud protocols. *Learning Disability Quarterly*, 16(1):19–32, 1993.

[26] J. Ning, Z. Ma, J. Yao, Q. Wang, and B. Zhang. Personalized learning supported by learning analytics: a systematic review of functions, pathways, and educational outcomes. *Interactive Learning Environments*, pages 1–23, 2025.

[27] J. Ocumpaugh et al. Baker rodrigo ocumpaugh monitoring protocol (bromp) 2.0 technical and training manual. *New York, NY and Manila, Philippines: Teachers College, Columbia University and Ateneo Laboratory for the Learning Sciences*, 60, 2015.

[28] E. L. Olmsted-Hawala, E. D. Murphy, S. Hawala, and K. T. Ashenfelter. Think-aloud protocols: a

comparison of three think-aloud protocols for use in testing data-dissemination web sites for usability. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 2381–2390, 2010.

[29] L. Paquette, A. M. de Carvalho, and R. S. Baker. Towards understanding expert coding of student disengagement in online learning. In *CogSci*, 2014.

[30] U. Petković, J. Frenkel, O. Hellwich, and R. Lazarides. Modeling teacher nonverbal immediacy: Refinement and analysis. *Adaptive Behavior*, page 10597123251374920, 2025.

[31] A. Ramachandran, C.-M. Huang, E. Gartland, and B. Scassellati. Thinking aloud with a tutoring robot to enhance learning. In *Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction*, pages 59–68, 2018.

[32] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[33] D. Sonmez Unal, E. Solovey, C. M. Arrington, and E. Walker. Modeling the phases of rule learning during problem solving with an interactive learning environment. *User Modeling and User-Adapted Interaction*, 35(1):7, 2025.

[34] Ö. Sümer, P. Goldberg, S. D'Mello, P. Gerjets, U. Trautwein, and E. Kasneci. Multimodal engagement analysis from facial videos in the classroom. *IEEE Transactions on Affective Computing*, 14(2):1012–1027, 2021.

[35] B. Tabarsi, H. Reichert, R. Qualls, T. Price, and T. Barnes. Exploring novices' struggle and progress during programming through data-driven detectors and think-aloud protocols. In *2023 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 179–183. IEEE, 2023.

[36] Z. V. Tian, M. Sun, A. Liu, S. Sarkar, and J. Liu. Instructional improvement: leveraging computer-assisted textual analysis to generate insights from educational artifacts. In *Research Handbook on Classroom Observation*, pages 190–207. Edward Elgar Publishing, 2025.

[37] D. S. Unal, C. M. Arrington, E. Solovey, and E. Walker. Using thinkalouds to understand rule learning and cognitive control mechanisms within an intelligent tutoring system. In *International Conference on Artificial Intelligence in Education*, pages 500–511. Springer, 2020.

[38] K. VanLehn. The behavior of tutoring systems. *International journal of artificial intelligence in education*, 16(3):227–265, 2006.

[39] V. Vapnik and A. Vashist. A new learning paradigm: Learning using privileged information. *Neural networks*, 22(5-6):544–557, 2009.

[40] M. Vrigkas, E. Kazakos, C. Nikou, and I. A. Kakadiaris. Human activity recognition using robust adaptive privileged probabilistic learning. *Pattern Analysis and Applications*, 24(3):915–932, 2021.

[41] S. Wang, Y. Zhu, L. Yue, and Q. Ji. Emotion recognition with the help of privileged information. *IEEE Transactions on Autonomous Mental Development*, 7(3):189–200, 2015.

[42] S. Wu. Application of artificial intelligence graph convolutional network in classroom grade evaluation. *Scientific Reports*, 15(1):32044, 2025.

[43] J. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using large language models to detect self-regulated learning in think-aloud protocols. In *Proceedings of the 17th international conference on educational data mining*, pages 157–168, 2024.

[44] Y. Zhao, B. Lackaye, J. G. Dy, and C. E. Brodley. A quantitative machine learning approach to master students admission for professional institutions. *International Educational Data Mining Society*, 2020.

[45] L. Zheng, A. He, C. Qi, H. Zhang, and X. Gu. Cognitive echo: Enhancing think-aloud protocols with llm-based simulated students. *British Journal of Educational Technology*, 2025.

MTBU-Bench: a Multimodal Benchmark for Evaluating Video-LLMs in Teacher Behavior Understanding

Zhiyi Duan
Department of Computer Science
Inner Mongolia University
Hohhot 010021, China
duanzy@imu.edu.cn

Jiangshan Guan
Department of Computer Science
Inner Mongolia University
Hohhot 010021, China
32409140@mail.imu.edu.cn

Lei Gao
Department of Computer Science
Inner Mongolia University
Hohhot 010021, China
32409159@mail.imu.edu.cn

Qianli Xing*
College of Computer Science and Technology
Jilin University
Changchun, China
qianlixing@jlu.edu.cn

ABSTRACT

Video-LLMs hold significant potential for analyzing classroom dynamics to inform pedagogy and drive educational innovation. There is an urgent need to assess the capability of Video-LLMs in educational applications to identify their potential and limitations. Existing educational evaluation datasets often suffer from three major shortcomings: reliance on small-scale, single-label human annotations; a focus on unimodal analysis; and the absence of granular annotations required for multi-step reasoning. To address these gaps, we first introduce an efficient Human-LLM collaborative pipeline for high-quality multimodal annotation. We then propose **MTBU-Bench**, a comprehensive multimodal benchmark designed to assess teacher behavior understanding. It comprises over 300 classroom scenarios across 15 K-12 and higher education disciplines, organized into four evaluation tasks: Action Recognition, Temporal Boundary Extraction, Spatial Localization, and Behavioral Description. Extensive evaluations of twelve mainstream Video-LLMs reveal critical shortcomings in fine-grained temporal understanding, specifically in the precise localization of action boundaries and the discrimination of similar behaviors. Our benchmark and findings provide a foundation for the targeted development and application of Video-LLMs in educational research. The code and dataset are publicly available at our project homepage: `https://github.com/271213/MTBU-Bench`.

Keywords

Video-LLMs, Multimodal benchmark, Human-LLM collaborative annotation, Teacher Behavior Understanding

*Corresponding author.

Zhiyi Duan, Jiangshan Guan, Lei Gao, and Qianli Xing. MTBU-Bench: a Multimodal Benchmark for Evaluating Video-LLMs in Teacher Behavior Understanding. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 167–179. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
`https://doi.org/10.5281/zenodo.21039705`

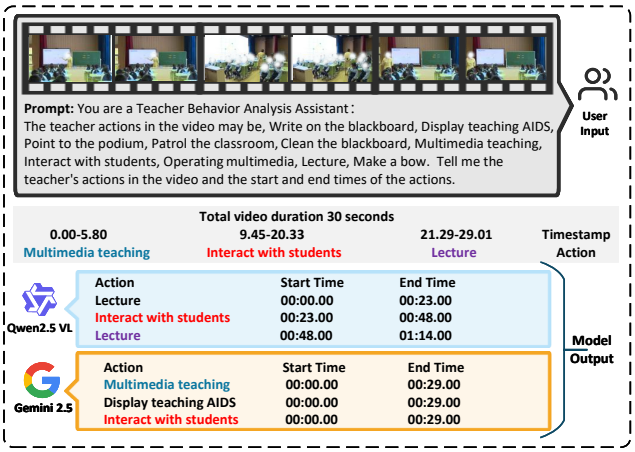


Figure 1: An example of the current Video-LLMs limitation in temporal boundary extraction from videos.

1. INTRODUCTION

In recent years, Video large language models (Video-LLMs), which integrate visual encoders with large-scale pre-trained language models, have demonstrated remarkable performance in video understanding tasks [17, 19, 16]. These models are capable of jointly processing spatiotemporal visual features and semantic reasoning tasks, unlocking new possibilities for comprehending complex scenes [44, 14]. Among the domains empowered by such advancements, education stands out as a critical application area where video understanding facilitates teaching assessment and instructional strategy optimization [45]. In particular, through analyzing instructional actions, spatial positions, and interaction targets within authentic classroom environments, teacher behavior understanding not only provides an objective reflection of teaching effectiveness but also fosters pedagogical self-reflection and offers interpretable insights for professional development [32, 1]. Consequently, evaluating the reasoning capabilities of Video-LLMs in educational settings has become an essential research challenge.

Mainstream Video-LLMs typically rely on static image-based representations, extracting keyframes from input videos to conduct semantic modeling and action recognition [9, 40]. However, such keyframe-driven paradigms pose inherent limitations when applied to fine-grained analysis of teacher behavior in authentic classroom settings [37]. Specifically, the omission of continuous temporal information compromises the model's ability to accurately delineate action boundaries, often leading to temporal ambiguity as shown in Figure 1. Moreover, for semantically similar behaviors such as "write on the blackboard" versus "clean the blackboard" existing models still struggle with disambiguation, resulting in frequent semantic confusion. These challenges significantly hinder the effectiveness of current Video-LLMs in educational behavior understanding tasks.

Although prior studies have explored video understanding tasks within educational contexts [29, 35], existing benchmarks remain inadequate for systematically evaluating the reasoning capabilities of Video-LLMs. This inadequacy stems from three primary limitations: 1) Current datasets predominantly rely on manually annotated, single teacher action labels, which are costly to produce and prone to annotator bias. They are typically confined to small-scale instructional videos [37, 23]; 2) Existing research is largely limited to unimodal video analysis [11, 12], failing to leverage the complementary strengths of multimodal data, thereby constraining the comprehension of rich classroom interaction dynamics; 3) Present benchmarks tend to focus on coarse, single-label descriptions of teacher action [6], lacking fine-grained annotations of behavior, interaction targets, and contextual features, which are essential for aligning with the multi-step reasoning processes of Video-LLMs.

To address the aforementioned challenges, we propose **MTBU-Bench**, the first multimodal evaluation benchmark tailored for assessing Video-LLMs on teacher behavior understanding tasks. **MTBU-Bench** spans 15 academic disciplines (Math, Physics, Biology, Chemistry, Science, Geography, History, Chinese, English, Music; Higher education: Medicine, Political - History, Engineering, Liberal Arts, Natural Sciences.) across K-12 and higher education, comprising over 500 complete, authentic classroom videos from more than 300 distinct teaching scenarios. As illustrated in Figure 2, each instance includes multimodal representations of teacher behavior along with five annotated dimensions: behavior category, spatial location, interaction target, video description, and Question–answer (QA) pairs. To ensure both annotation efficiency and multimodal consistency, we introduce a multi-stage Human–LLM collaborative annotation pipeline. This workflow integrates expert annotations, senior expert validation, and LLM-assisted behavioral description generation. Based on this, we conduct a comprehensive evaluation of ten mainstream Video-LLMs (e.g., GPT-4V, Gemini 2.5 Pro), focusing on four core tasks: teacher action recognition, teacher behavior temporal boundary extraction, teacher behavior spatial localization, and teacher behavior description. Experimental results reveal notable limitations in current Video-LLMs, particularly in recognizing similar behaviors, aligning interaction targets, and modeling environmental context.

2. RELATED WORKS

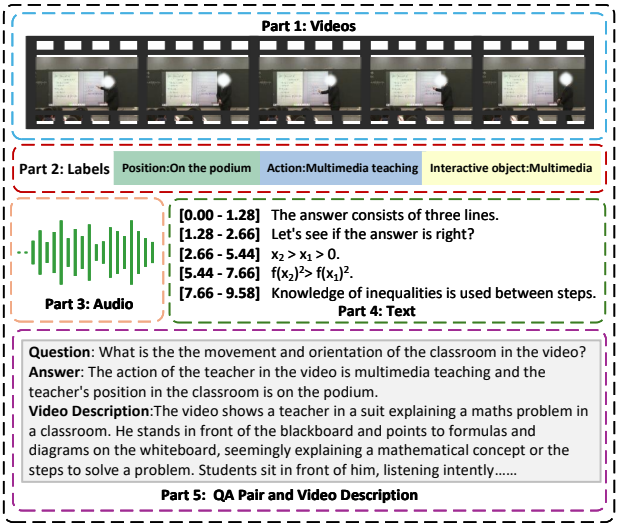


Figure 2: Data composition of MTBU-Bench.

2.1 Video Understanding Dataset

Video understanding aims to automatically generate natural language descriptions of key events, human actions, and scene contexts within videos, and has been widely applied in educational analysis, video retrieval, and supplementary instruction [27, 7, 38]. With the rapid development of multimodal understanding and generative models, video captioning has evolved from single-sentence summaries to structured, hierarchical, and fine-grained representations. The MiraData dataset, employing the GPT-4V model to generate detailed descriptions from six distinct perspectives [24]. The YouCookII dataset focuses on instructional videos in cooking scenarios, providing temporal boundaries and English descriptions for each procedural step [49]. The ViTT dataset further extends to multi-domain instructional videos, offering short tags averaging three words per segment [21]. To pursue higher-quality and more detailed long-form descriptions, the Koala-36M dataset provides 36 million filtered video clips with detailed, structured captions averaging 200 words [42]. Recent research has also pushed into grounding these descriptions, with ViCaS linking captions to pixel-level segmentation masks [3], and HD-EPIC further grounding fine-grained actions and objects within 3D digital twins from an egocentric perspective [34].

2.2 Datasets for Teacher Analysis

Compared to general video datasets, educational video data place greater emphasis on multimodal alignment, authentic instructional context, and structured annotations. The EduNet dataset, encompassing 20 categories of teacher and student behaviors across 7,851 video clips [37]. TBU is a multi-task teacher behavior dataset that includes three kinds of labels across 200 classroom videos [6]. TCDA focuses on teacher dress code assessment containing 11,879 annotated images with 25 attribute categories teacher dress [12]. TAQR is the benchmark dataset dedicated to teacher action quality ranking, covering four core instructional behaviors 1,200 video clips [11]. However, all of these datasets are limited to manually annotated unimodal data and lack multi-

Table 1: Comparison of MTBU-Bench with existing teacher behavior datasets.

Dataset	Modalities	Scale	Subject	Ed. Stage	Open Source	Tasks
TAR [23]	Visual	13,288 clips	N/A	Vocational	✗	Action Recognition
TAQR [11]	Visual	1,200 clips	9	High School	✗	Action Quality Ranking
EduNet [37]	Visual, Audio	7,851 clips	N/A	K-12	✓	Action Recognition
TBU [6]	Visual, Text	8,390 clips, 188 untrimmed	N/A	K-12, College	✗	Action Recognition, Location, Human-base Behavior Description
MM-TBA [20]	Visual, Audio, Text	4,839 videos	2	Middle School, High School	✓	Action Detection, Teacher Lecture Evaluation, Instructional Design
MTBU(Ours)	**Visual, Audio, Text**	**8,000 clips, 500 untrimmed**	**15**	**K-12, Higher Education**	✓	**Action Recognition, Temporal Boundary Extraction, Spatial Localization, Behavior Description & QA**

dimensional fine-grained video descriptions, making them insufficient for directly evaluating the reasoning capabilities of Video-LLMs. A detailed comparison of MTBU-Bench with these existing teacher behavior datasets is summarized in Table 1.

2.3 Evaluation of Video-LLMs

With the development of Video-LLMs, a series of evaluation benchmarks has also emerged [47, 48], each with a distinct focus. Video-MME [15] aims for comprehensive evaluation, covering diverse video types, durations from 11 seconds to 1 hour. ALLVB [39] concentrates on ultra-long video understanding, using videos averaging nearly two hours to assess long-sequence modeling capabilities [17]. Other benchmarks focus on more fine-grained abilities[28], such as MotionBench [18], which specifically evaluates the understanding of fine-grained motion. However, these general-purpose benchmarks, despite their advancements in comprehensiveness, duration, and granularity, do not address the specific needs of educational scenarios, such as the fine-grained understanding of teacher behavior[2, 30]. **MTBU-Bench** is the first comprehensive multimodal benchmark designed to fill the crucial gap of evaluating Video-LLMs understanding of teacher multimodal behavior.

3. MTBU-BENCH

In this section, we introduce MTBU-Bench, the first comprehensive multimodal benchmark tailored for assessing Video-LLMs on teacher behavior understanding tasks. Spanning 15 academic disciplines across K-12 and higher education, the dataset comprises over 500 complete, authentic classroom videos from more than 300 distinct teaching scenarios. To ensure annotation quality and multimodal consistency, we construct the benchmark through a Human-LLM Collaborative Annotation pipeline that effectively integrates expert knowledge with automated generation capabilities. We organize the dataset into four core task subsets: Multimodal Teacher Action Recognition, Teacher Behavior Temporal Boundary Extraction, Teacher Behavior Spatial Localization, and Teacher Behavior Description, designed to systematically evaluate the fine-grained perception and reasoning capabilities of large models in complex educational environments.

3.1 Data Construction Workflow

As shown in Figure 3, we propose a Human-LLM Collaborative Annotation pipeline to construct **MTBU-Bench**, including video collection, Annotation System Construction, data segmentation and annotation, quality assurance, multimodal extraction and alignment, behavior description generation, question-answer pair generation, description and QA Pair verification.

Video Collection. We collect 500 complete classroom teaching videos spanning multiple educational levels from K-12 to higher education. It covers 15 academic subjects and includes both public lectures and competition-based lessons. The recorded classroom scenarios exhibit diverse camera perspectives, varied instructional formats, and heterogeneous spatial layouts, totaling more than 300 distinct teaching environments. These videos faithfully capture pedagogical characteristics, teacher behaviors, and classroom interaction patterns across different educational stages, offering strong representativeness and practical applicability for multimodal educational research.

Annotation System Construction. To support multimodal reasoning and enable comprehensive modeling of teacher behaviors, we design a structured annotation system composed of four key dimensions. First, we define a set of ten core instructional actions based on their high frequency and semantic distinctiveness in real-world classrooms. Second, we annotate the spatial position of the teacher during action execution to provide grounding for spatiotemporal localization. Third, we annotate the primary interaction object to reflect communicative targets and teaching intent. Finally, we generate structured natural language descriptions that integrate visual, auditory, and contextual cues, offering rich semantic supervision for downstream Video-LLMs. This four-dimensional annotation schema ensures interpretability, consistency, and task alignment across the benchmark.

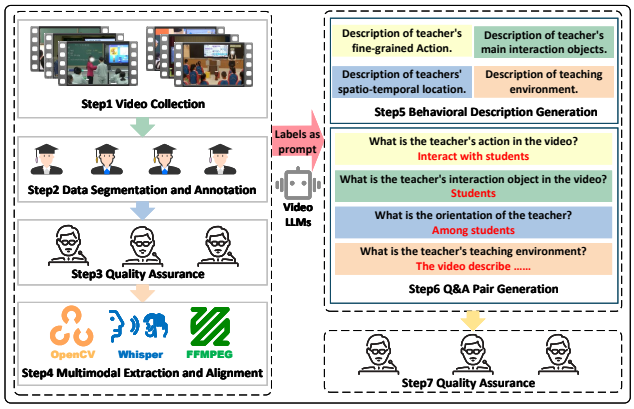


Figure 3: Construction pipeline of MTBU-Bench.

Data Segmentation and Annotation. To construct a multimodal dataset tailored for teacher behavior understanding, we adopt behavior segments as the fundamental annotation unit. Raw instructional videos are temporally segmented and annotated to ensure that each segment exhibits content completeness, clear temporal boundaries, and semantic consistency with respect to the behavior it represents. The segmentation is conducted using videoManager, with each segment corresponding to the full execution of a distinct teaching action. The start time is defined by the onset of an observable behavioral cue, while the end time is marked by the action's completion or transition into the next behavior. The granularity of segmentation follows the principle of units of semantic behavior, balancing multimodal observability with the pedagogical coherence of instructional behaviors [25].

To ensure the high quality, accuracy, and pedagogical domain fidelity of the MTBU-Bench dataset, the annotation process was driven by a highly qualified, cross-disciplinary team. This team consisted of one primary researcher, three experienced frontline teachers (each with over 10 years of classroom teaching experience), and four graduate students (two specializing in Education and two in Computer Science).

The annotation process followed a rigorous collaborative workflow. Initially, the primary researcher and the frontline teachers collaboratively finalized the behavior labels to ensure they accurately reflected authentic instructional dynamics. Subsequently, they conducted comprehensive training for the four graduate students. The annotators were provided with detailed annotation guidelines, which included strict label definitions, concrete annotation examples, and standardized operating procedures for the annotation tools.

During the execution phase, the annotation tasks were divided into four subsets. Rather than being separated by educational stages or subject areas, each subset encompassed content from all disciplines and educational levels. All four graduate students were jointly responsible for the data segmentation and annotation tasks across all four subsets. Each behavior instance was meticulously annotated with four key attributes: teacher action, spatial position during the action, interaction target, and start and end timestamps.

Quality Assurance. To guarantee the overall reliability of the dataset, the three frontline teachers independently conducted a Quality Assurance phase, rigorously reviewing and validating the temporal boundaries and semantic labels of the annotated clips. Furthermore, to evaluate inter-annotator agreement, a 10% data overlap was maintained across the subsets. Any discrepancies were resolved through group discussion and majority voting. The final annotation consistency was quantitatively assessed using Fleiss' Kappa, yielding a score of 0.949, which indicates a "high agreement" level [13]. This review process focuses primarily on verifying and harmonizing the temporal boundaries of annotated behaviors, and enhancing the overall consistency and scientific rigor of the dataset.

Multimodal Extraction and Alignment. We employ a suite of open-source tools including OpenCV[1], Whisper[2], and FFmpeg[3] to automate the extraction and alignment of visual, audio, and textual modalities from classroom videos. Through a timestamp-based synchronization mechanism, teacher actions in the visual stream, spoken content in the audio stream, and semantic information in the text stream are precisely aligned to form temporally consistent multimodal behavior segments. This stage ensures that each annotated teacher behavior instance maintains high consistency and semantic completeness across visual, auditory, and linguistic modalities.

Behavior Description Generation. To enhance model understanding in classroom teacher behavior recognition tasks, we construct a four-dimensional description for each annotated video segment. Each dimension is designed to complement behavior semantics from a distinct perspective, thereby improving the model's capability to recognize fine-grained instructional behaviors in complex classroom environments. Building upon high-quality annotations, we leverage the Video-LLMs to generate four types of structured natural language descriptions for each behavior clip, as detailed below:

- **Fine-grained description of teacher actions**. This dimension provides fine-grained descriptions of core instructional actions to distinguish semantically different but visually similar behaviors. For example, "clean the blackboard" and "write on the blackboard" share similar visual cues standing in front of the blackboard and using hand gestures yet their pedagogical intents diverge, often leading to misinterpretation by Video-LLMs. By documenting micro-actions (e.g., hand-raising, body-turning, or using teaching aids) and subtle temporal patterns, this dimension enhances the model's capacity to disambiguate near-identical actions and capture critical shifts in instructional orientation.
- **Spatial location of behavior execution**. This dimension captures the teacher's spatial position, which serves as a critical indicator of instructional intent. For instance, positions such as the classroom corners or the

[1]https://github.com/opencv/opencv
[2]https://github.com/openai/whisper
[3]https://github.com/FFmpeg/FFmpeg

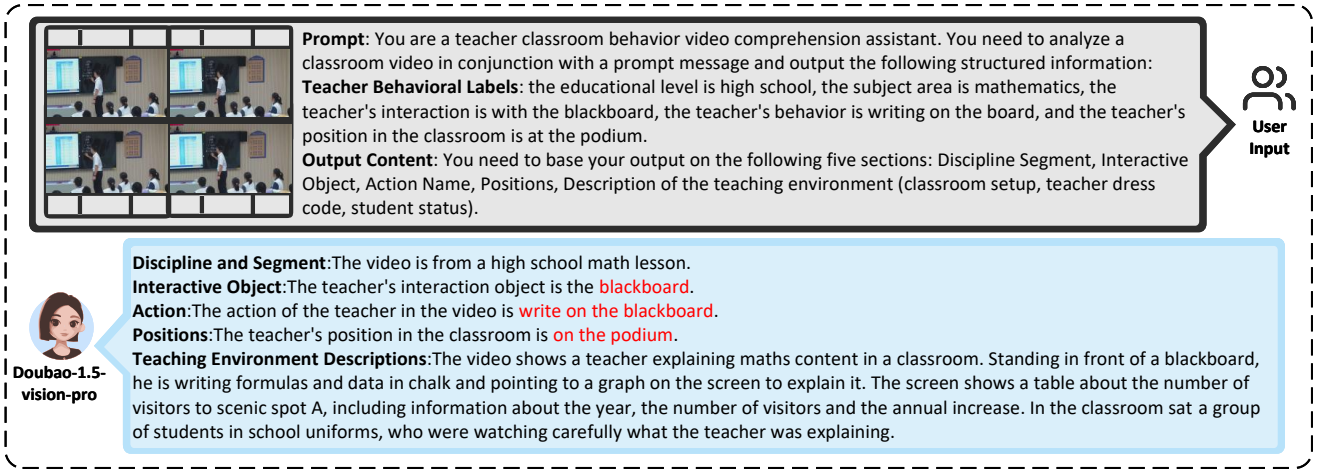


Figure 4: An example of Video-LLMs for teacher behavior description.

blackboard center correlate with specific actions like "patrol the classroom" or "lecture". Explicitly encoding spatial context enhances the model's contextual grounding and classification accuracy by leveraging the strong spatial dependencies of instructional behaviors. Ultimately, spatial location functions as a discriminative feature that significantly improves action recognition performance.

- **Primary interaction target during behavior execution**. This dimension explicitly identifies interaction targets such as students, blackboards, or multimedia devices to clarify the object associated with each instructional behavior. Distinguishing these targets is essential for interpreting pedagogical intent and serves as a primary criterion for differentiating between lecture driven actions and interactive driven actions. For instance, engaging with students signifies interactive behavior, while manipulating multimedia denotes a control action. Ultimately, these annotations support downstream tasks like pedagogical strategy analysis and classroom structure modeling.

- **Instructional environment context**. This dimension captures environmental cues such as classroom layout, teacher attire, and student status to provide essential context for interpreting instructional actions. For example, laboratory settings with lab benches often imply demonstrative actions, while traditional attire combined with a projection screen aligns with "lecture" or "multimedia teaching" scenarios. These cues are pivotal for filtering non-instructional behaviors and enhancing model precision and robustness. Ultimately, these features ensure semantic coherence in behavioral analysis and reinforce the reliability of multimodal alignment.

These structured descriptions provide rich semantic grounding for subsequent behavior understanding and multimodal evaluation tasks. Behavior description generation process is shown in Figure 4.

Question–Answer Pair Generation. Building upon the structured behavior descriptions, we further employed Video-LLMs to generate QA pairs for each classroom behavior segment. Each QA pair targets one of four key dimensions: the teacher's instructional action, spatial position, primary interaction object, and the surrounding instructional environment. The integration of QA pairs not only enriches the semantic depth of the dataset enabling models to acquire more fine-grained insights into teacher behaviors but also strengthens cross-modal alignment ability. Moreover, this QA-based design lays a high-quality foundation for downstream applications such as instructional question answering systems and fine-tuning large models for classroom behavior analysis and evaluation in educational settings [46, 36, 43].

Description and QA Pair Verification. Domain experts review the model-generated descriptions and question–answer pairs with respect to four key dimensions: instructional actions, spatial positioning, primary interaction targets, and the surrounding classroom environment.

3.2 Dataset Subset Composition

As shown in Figure 5, **MTBU-Bench** comprises four distinct subsets, each tailored to a specific evaluation objective: multimodal teacher action recognition, teacher behavior temporal boundary extraction, teacher behavior spatial localization, and teacher behavior description. These subsets are designed with a clear structural organization, comprehensive annotations, and a strong task-driven focus, making the dataset well-suited for advancing research in educational video understanding across multiple dimensions.

3.2.1 Multimodal Teacher Action Recognition Subset

Teacher Action Labels. The Multimodal Teacher Action Recognition Subset (MTAR Subset) is designed to provide standardized data for instructional action recognition, focusing on ten of the most frequently observed teacher actions in classroom settings: Write on the blackboard: The teacher uses chalk or a marker to write text, formulas, or draw dia-

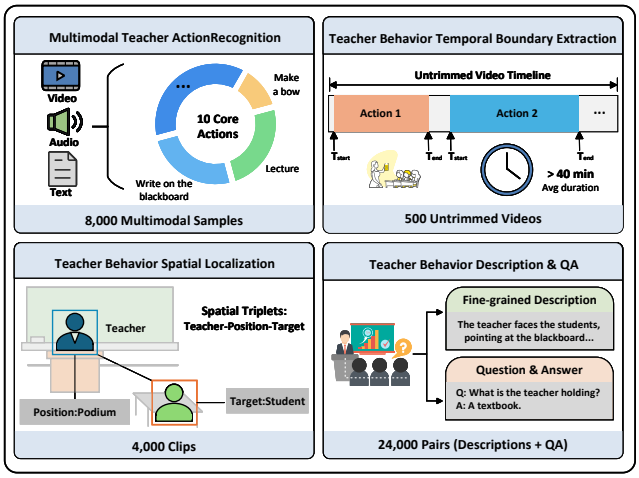


Figure 5: Statistics and task illustrations of the four subsets in MTBU-Bench.

grams on the physical blackboard or whiteboard. Clean the blackboard: The teacher uses an eraser or cloth to remove content from the physical blackboard or whiteboard. Multimedia teaching: The teacher utilizes digital platforms (e.g., presenting slides, playing videos) as the primary medium to deliver instructional content. Operating multimedia: The physical action of the teacher manipulating technological devices (e.g., typing on keyboard, clicking mouse, adjusting projector) to control the flow of digital content. Display teaching aids: The teacher physically holds up or demonstrates physical objects, models, or printed materials to the students to facilitate understanding. Interact with students: The teacher engages in direct, two-way communication with students, including asking questions, answering queries, or guiding group discussions. Lecture: The teacher delivers verbal explanations of academic concepts to the entire class, typically standing still without intense physical interactions with props or the board. Patrol the classroom: The teacher walks dynamically among the classroom to monitor progress, observe independent work, or provide localized guidance. Point to the podium: The teacher uses hand gestures or a pointer to direct students' visual attention to specific content on the blackboard or projection screen. Make a bow: A formal, ritualistic action where the teacher bows to the students, typically occurring at the very beginning or end of a lesson. [22, 26, 5].The data distribution is shown in Table 2.

This subset contains a total of 8,000 multimodal samples, covering the ten core instructional action. Each sample includes a high-quality video clip, synchronized audio, and its transcribed text, ensuring precise alignment across the three modalities. This subset is suitable for training and evaluating multimodal action recognition models and supports research directions such as cross-modal retrieval, audio-visual joint modeling, and multimodal feature fusion. It serves as foundational training data for multimodal models in educational scenarios.

3.2.2 *Teacher Behavior Temporal Boundary Extraction Subset*

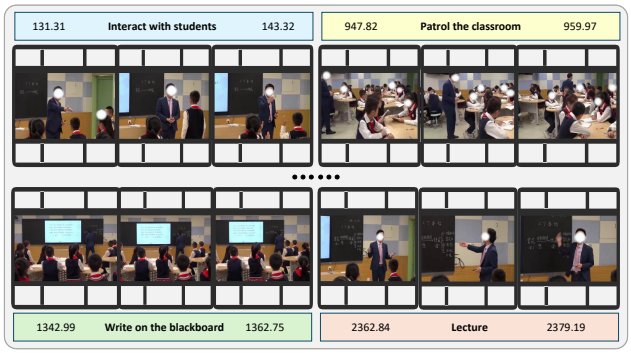


Figure 6: Examples from the Teacher Behavior Boundary Extraction Subset.

To address the bottlenecks of current Video-LLMs in long video understanding and temporal localization, we construct the Teacher Behavior Temporal Boundary Extraction Subset. Unlike most existing benchmarks that rely on short trimmed clips, this subset comprises 400 complete classroom teaching videos, with an average duration exceeding 40 minutes per session.

These untrimmed videos fully preserve the continuity and complexity of authentic instructional activities. We provide dense, frame-level annotations, precisely marking the start and end timestamps of every teacher action throughout the entire lesson. This setup presents a significant challenge for models: they are required not only to recognize specific action categories but also to precisely delineate action boundaries within a long-horizon context, thereby testing their capability to handle temporal redundancy and transition ambiguity in real-world continuous video streams. Examples from the Teacher Spatiotemporal Behavior Localization Subset are provided in Figure 6.

3.2.3 *Teacher Behavior Spatial Localization Subset*

To evaluate the fine-grained spatial reasoning capabilities of Video-LLMs within complex classroom environments, we construct the Teacher Behavior Spatial Localization Subset (TBSL Subset). To ensure the reliability of spatial reasoning evaluation, this subset consists of 4,000 high-quality short video clips carefully selected from the 8,000 clips in the MTAR subset. We strictly filtered for instances where the teacher's spatial orientation and positional information are highly discernible, and the continuous action duration exceeds 10 seconds.

Teacher Spatial Position and Interaction Target Labels. Each sample is annotated with a structured behavior triplet, comprising the teacher action, their spatial position relative to the classroom layout, and the corresponding primary interaction target. Spatial positions include being On the podium: where the teacher is located in the defined instructional zone at the front of the classroom, typically near the blackboard and main desk; Among students: where the teacher is located within the student seating area, moving through the aisles or standing next to desks; and Behind the classroom: where the teacher is positioned at the very back of the room, typically to observe the entire class's sta-



Question: What is the object of the teacher's interaction in the video?
Answer: The teacher's interaction object is the teaching AIDS.

Video Description: The video is from a high school math lesson. The action of the teacher in the video is **display teaching aids.**
The teacher's position in the classroom is **on the podium**.
The teacher's interaction object is the **teaching AIDS**.
The video shows a teacher teaching in a classroom, standing in front of a blackboard, holding a bottle of drink and explaining the lesson content to students. On the screen are the words "Renjiao A version Selective Compulsory I Chapter 3 Equations of Conic Curves" and "3.1.1 Ellipse and its Standard Equation (Lesson 1)", with the name of the teacher next to it. The students were wearing uniform green jackets and sitting at their desks, listening attentively to the lectures.



Question: The orientation information for the classroom in the video is?
Answer: the teacher's position in the classroom is from among students to behind the classroom.

Video Description: The video is from a high school math lesson.
The action of the teacher in the video is **patrol the classroom.**
The teacher's position in the classroom is **from among students to behind the classroom**.
The teacher's eyes are interacting with the **students**.
The video shows a scene inside a classroom where students wearing uniform white school uniforms sit at their desks and study carefully. A teacher walks back and forth in front of the classroom, seemingly inspecting and guiding the students. The entire classroom is simply decorated and well-lit, creating an atmosphere of focused learning.

Figure 7: Example data of Teacher Behavior Description Subset.

Table 2: Subset Data Distribution.

MTAR Subset	
Teacher actions	Number
Clean the blackboard	500
Operating multimedia	500
Display teaching aids	834
Lecture	790
Interact with students	954
Make a bow	682
Multimedia teaching	867
Patrol the classroom	796
Point to the podium	873
Write on the blackboard	1,204

TBSL Subset	
Spatial positions	Number
On the podium	2,256
Among students	1,531
Behind the classroom	213
Interaction target	Number
Blackboard	1,073
Multimedia	463
Teaching aids	798
Students	1,145
Multimedia PPT	521

tus or during student-led presentations. Furthermore, primary interaction targets include the "Blackboard" for traditional physical writing surfaces; "Multimedia PPT" for digital slides or educational software interfaces displayed on a screen; "Multimedia" for hardware devices such as computers, projectors, or interactive smartboards; "Teaching aids" for physical objects, biological specimens, geometric models, or chemical apparatus; and "Students" for individual students or the class as a whole. The data distribution is shown in Table 2. These triplets provide fine-grained and semantically rich supervision for modeling spatial reasoning in instructional scenarios. The teacher's interaction target offers implicit cues regarding their spatial orientation; this correlation between interaction semantics and spatial positioning is critical for Video-LLMs to build accurate representations of instructional context.

3.2.4 Teacher Behavior Description Subset

The Teacher Behavior Description Subset is dedicated to supporting video-to-text generation and question answering tasks within educational contexts. For each behavior segment, the subset provides a detailed natural language description that encapsulates semantic elements such as the type of action, spatial location, and interaction target. To further enhance the model's reasoning capabilities in complex instructional scenarios, each description is paired with a corresponding video-based QA pair, enabling both generative and comprehension-based evaluations of multimodal language models, example data is displayed in Figure 7.

In total, the subset contains 24,000 natural language descriptions aligned with corresponding video segments, along with 24,000 QA pairs grounded in the semantic content of those segments. These 24,000 samples are directly derived from the 8,000 MTAR clips by generating three distinct descriptions and corresponding QA pairs from different semantic perspectives for each clip. Each description averages 26 words in length and captures rich semantic dimensions, including teacher actions, classroom environment, and interactive elements. This subset provides a standardized benchmark resource for fine-tuning and evaluating multimodal large models in educational scenarios.

4. EXPERIMENTS

4.1 Experimental Setup

To evaluate the fundamental capabilities of mainstream Video-LLMs on the teacher behavior understanding task, we conducted a systematic zero-shot evaluation of ten representative models, including GPT-4V, Gemini 2.5 Pro, Video-LLaVA, LLaMA 4-17B, Qwen2.5-Omni, Spark X1 (education-oriented), Grok3, Qwen-QVQ, Claude 3.5 Sonnet, and Kimi-latest. All experiments were conducted under a zero-shot inference setting without any task-specific fine-tuning or training. The evaluation was performed on the four core subtasks of **MTBU-Bench** Teacher Action Recognition, Teacher Behavior Temporal Boundary Extraction, Teacher Behavior Spatial Localization Subset , and Behavior Description, using a unified standard to ensure comparability and reproducibility of the results.

In terms of test scale, we constructed a comprehensive zero-shot evaluation dataset tailored to the specific characteristics of each task. For Teacher Action Recognition, Spatial Localization, and Behavior Description, we utilized a dataset comprising 4,000 samples, with 400 samples drawn from each of the 10 core teacher action categories. For Teacher Behavior Temporal Boundary Extraction, considering the current limitations of Video-LLMs in long-context understanding, we curated a specialized test set of action-intensive segments selected from the complete annotated classroom videos. This set consists of 100 clips with a duration of 4 minutes and 100 clips with a duration of 8 minutes.

Table 3: Model performance on teacher action recognition performance. Bold values denote the best performance; Italics values indicate the second-best.

Model	Accuracy	Precision	Recall	F1 Score
Open-source LLM				
Qwen2.5-Omni	*0.5575*	0.6425	**0.5675**	**0.6030**
Qwen2.5-Omni-Single	0.5150	0.6071	0.5225	*0.5616*
Qwen-QVQ	0.4925	0.6339	0.4925	0.4667
LLaMA4-17B	0.4675	*0.6561*	0.4675	0.4577
Video-LLaVA	0.0800	0.0146	0.0800	0.0245
Closed-source LLM				
Gemini 2.5 Pro-Omni	**0.5650**	**0.7363**	*0.5621*	0.5606
Gemini 2.5 Pro	0.5325	0.5511	0.5325	0.5212
Claude 3.5 Sonnet	0.4800	0.6274	0.4800	0.4611
GPT-4V	0.4725	0.5737	0.4725	0.4602
Grok3	0.4100	0.6318	0.4100	0.3848
Kimi-latest	0.3775	0.6191	0.3775	0.3367
Education-oriented LLM				
Spark X1	0.5125	0.5813	0.5075	0.4748

All models were evaluated on the same sampled data to ensure consistency and reliability of the experimental results.

4.2 Teacher Action Recognition

For this task, we specifically introduced two Omni-LLMs, Qwen2.5-Omni and Gemini 2.5 Pro-Omni, to evaluate their performance with full multimodal inputs (video, audio, and text). To isolate and assess the specific contribution of the auditory modality to classroom behavior understanding, we also established a baseline vision-language setting for all ten models. This comparative setup allows us to validate the benefits of supplementary audio information.

We evaluate model performance on both overall accuracy and individual instructional actions, using Accuracy, Precision, Recall, and F1 Score as the primary metrics. As summarized in Table 3, the multimodal Gemini 2.5 Pro-Omni (0.5650 Acc), Qwen2.5-Omni (0.5575 Acc) significantly outperforms its unimodal variant, Gemini 2.5 Pro (0.5325 Acc), Qwen2.5-Omni-Single (0.5150 Acc). This demonstrates that inputs encompassing video, audio, and text enhance a model's ability to capture diverse cues from complex instructional scenes. Furthermore, Gemini 2.5 Pro-Omni achieves the best overall accuracy (0.5650) and a standout precision (0.7363). Closely following are Qwen2.5-Omni and Spark X1. In contrast, LLaMA4-17B displays a high precision (0.6561) but a comparatively low recall (0.4675), suggesting a more conservative prediction strategy.

For individual action categories. The results reveal that actions with distinct and concrete visual features (such as "Write on the blackboard" and "Display teaching aids") tend to be recognized more reliably across models. Qwen2.5-Omni and Spark X1 are particularly strong on such categories, achieving up to 100% accuracy. In contrast, abstract or weakly visual actions such as "Point to the podium" and "Make a bow" remain challenging across the board, often exhibiting recognition accuracy below 10%, as presented in Figure 8. These findings underscore the necessity of cross-modal fusion to resolve visual ambiguity and to improve semantic discrimination in fine-grained instructional behavior understanding.

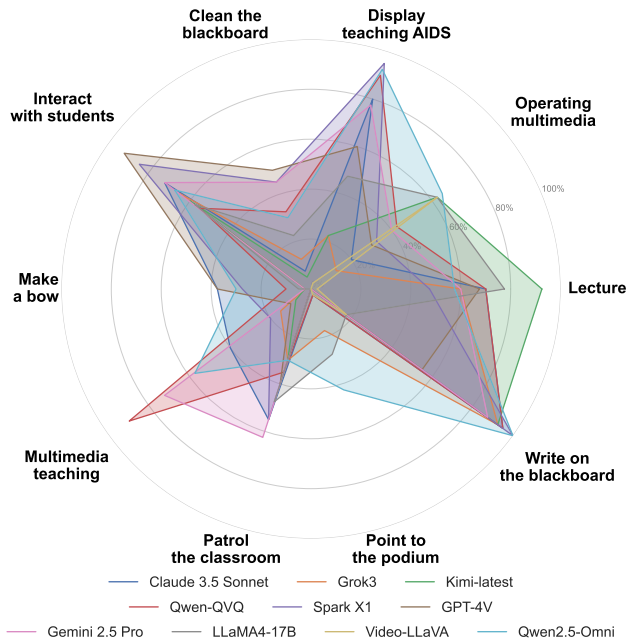


Figure 8: Per-category model recognition accuracy.

Compared to unimodal video inputs, multimodal integration demonstrates superior performance in teacher action recognition tasks. Specifically, the audio modality contributes linguistic content and ambient classroom cues, helping to disambiguate semantically similar actions that are visually confounded. For instance, "Write on the blackboard" and "Clean the blackboard", which appear visually alike but can be distinguished through corresponding audio signals. When writing on the blackboard, teachers often verbalize the content being written, whereas during clean the blackboard, they typically remain silent. The textual modality further enhances the model's understanding of instructional goals and behavioral intent, addressing the limitations of visual data in modeling contextual dependencies.

4.3 Teacher Behavior Temporal Boundary Extraction

To evaluate the temporal localization capabilities of Video-LLMs in continuous video streams, we conducted specific experiments on the Teacher Behavior Temporal Boundary Extraction Subset (TBTB Subset). Although the TBTB Subset provides complete classroom annotations with an average duration exceeding 40 minutes, current Video-LLMs still suffer from significant computational bottlenecks and reasoning deficiencies when processing ultra-long temporal contexts (Long-context), making it difficult to effectively comprehend the complexity of a full lesson. Therefore, we curated a test set by extracting "action-intensive" segments from the TBTB Subset, consisting of 100 clips with a duration of 4 minutes and 100 clips with a duration of 8 minutes. This design aims to maximally test the models' temporal

Table 4: Performance comparison of Teacher Behavior Temporal Boundary Extraction on 4-minute and 8-minute clips. The metrics include tIoU and mAP at different IoU thresholds (0.3 and 0.7).

Model	4min			8min		
	tIoU	mAP@0.3	mAP@0.7	tIoU	mAP@0.3	mAP@0.7
Gemini 2.5 Pro	36.48%	48.89%	6.32%	23.12%	22.43%	0.00%
Qwen2.5-Omni	34.20%	39.67%	4.21%	20.32%	17.64%	0.00%
GPT-4V	29.32%	21.47%	0.00%	16.38%	15.52%	0.00%
Kimi-latest	32.57%	41.09%	3.33%	18.24%	19.73%	0.00%

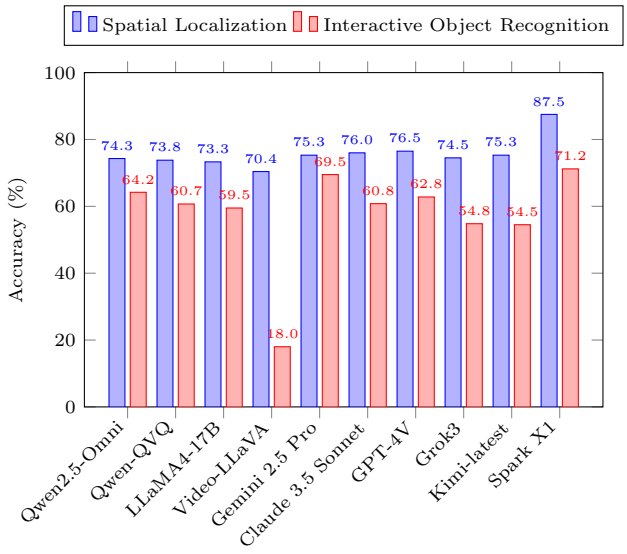


Figure 9: Comparison of Video-LLMs on Spatial Localization and Interactive Object Recognition.

sensitivity within a range accessible to their current capabilities. For evaluation metrics, we employ temporal Intersection over Union (tIoU) to measure the overlap between predicted segments and ground truth annotations. Additionally, we use mAP@0.3 and mAP@0.7 to assess model performance under two distinct strictness levels: "coarse localization" and "fine-grained localization," respectively. The experimental data is presented in Table 4.

The experimental results expose a "Precision Bottleneck" in temporal grounding among current models. While models demonstrate a degree of usability in coarse localization (mAP@0.3) on 4-minute videos (with Gemini 2.5 Pro reaching up to 48.89%), indicating their ability to perceive the approximate intervals of action occurrences; however, performance drops precipitously once precise boundary regression (IoU=0.7) is required. Even the best-performing model achieves an mAP@0.7 of only 6.32% in the shorter 4-minute setting, while some models score 0. This reveals that existing Video-LLMs function more as "semantic event detectors" rather than localizers with high-resolution temporal regression capabilities, failing to meet the demand for precise timestamps in educational analysis.

4.4 Teacher Behavior Spatial Localization

As video duration increases, models exhibit Catastrophic Performance Decay. When the test video duration extends from 4 minutes to 8 minutes, the average coarse localization accuracy (mAP@0.3) across all models drops by approximately 50%, highlighting the difficulty models face in maintaining attention over longer contexts. Most critically, in the 8-minute setting, the mAP@0.7 scores for all tested models fall to zero (0.00%). This "zero-shot failure" phenomenon indicates that as the temporal context expands, the redundancy and accumulation of visual information completely overwhelm the models' fine-grained perception mechanisms, resulting in a total loss of the ability to precisely delineate action boundaries in long continuous streams.

The data reveals a severe Divergence between Semantic Retrieval and Temporal Alignment in Video-LLMs. Although tIoU metrics maintain a range of 16%-37%, suggesting that models can successfully comprehend and retrieve approximate content segments containing the target actions (i.e., "finding the content"), the extremely low mAP@0.7 proves that this semantic understanding does not translate into precise geometric or temporal alignment (i.e., "failing to localize accurately"). This suggests that current architecture designs may over-rely on semantic correlation calculations similar to "image-text matching," while lacking dedicated mechanisms to explicitly model temporal boundary mutations and state transitions.

This dimension measures the model's ability to accurately identify the spatial location and the interaction target of the teacher while performing a specific instructional action.

As shown in Figure 9, all models achieved high performance in the spatial localization task, with accuracy exceeding 70%. This suggests a generalizable capability in recognizing stable spatial configurations commonly found in classrooms. For instance, the podium area is often characterized by frontal blackboards and multimedia screens, while interactions among students are typically associated with dense seating arrangements and more dynamic visual contexts.

In terms of interactive object recognition, model performance varied more significantly. Notably, Spark X1 achieved the highest accuracy in this dimension (71.2%), substantially outperforming other models. This can be attributed to its education-oriented design, which enhances its capacity to attend to pedagogically relevant visual cues and instructional targets. Models such as Qwen2.5-Omni and Gemini 2.5 Pro also demonstrated balanced performance across both spatial and interaction-level reasoning, reflecting strong cross-modal alignment.

Table 5: Text similarity evaluation of different models. Bold values denote the best; Italics values indicate the second-best.

Model	BLEU	METEOR	ROUGE	CIDEr
Open-source LLM				
Qwen2.5-Omni	0.2303	0.4773	0.4783	**0.5097**
Video-LLaVA	0.1939	0.4264	0.4330	0.3546
LLaMA4-17B	0.2171	0.4634	0.4639	0.4501
Qwen-QVQ	0.2231	0.4717	0.4706	0.4878
Closed-source LLM				
Claude 3.5 Sonnet	0.2273	0.4753	0.4750	0.4874
Grok3	0.2193	0.4661	0.4653	0.4133
Kimi-latest	0.2141	0.4612	0.4593	0.3932
GPT-4V	0.2276	0.4701	0.4735	0.4555
Gemini 2.5 Pro	**0.2327**	**0.4795**	**0.4810**	0.4973
Education-oriented LLM				
Spark X1	0.2251	0.4740	0.4737	0.5077

The two tasks are inherently interrelated: when teachers are positioned on the podium, their interaction targets are predominantly the blackboard or multimedia devices; When interacting with students, their spatial position typically shifts among students seating area. This spatial-interaction coupling provides mutual semantic cues that can reinforce model prediction accuracy. Together, accurate recognition of spatial positions and interaction targets offers deeper insights into instructional intent and classroom orchestration strategies.

4.5 Teacher Behavior Description

We further evaluate models' ability of video understanding by the teacher behavior description task, using two complementary approaches: traditional textual similarity metrics and semantic scoring via LLM-as-a-judge. In traditional metric-based evaluation, the evaluation results are shown in Table 5 in terms of ROUGE [31], CIDEr [41], BLEU [33], METEOR [4]. Qwen2.5-Omni achieves the highest CIDEr score (0.5097), reflecting its superior ability to reproduce fine-grained instructional details that align with expert references. Gemini 2.5 Pro leads in BLEU (0.2327), METEOR (0.4795), and ROUGE (0.4810), demonstrating a strong balance between surface-level lexical accuracy and semantic fluency. GPT-4V and Claude 3.5 Sonnet exhibit relatively balanced but slightly lower scores across all metrics, while education-oriented model Spark X1 maintains competitive performance, particularly in CIDEr (0.5077), which highlights its alignment with the educational domain.

To further enhance the objectivity and robustness of the evaluation, we introduce the LLM-as-a-Judge [8, 10] mechanism, employing DeepSeek-R1 to compute semantic alignment score for the model-generated teacher behavior descriptions. To ensure the reliability of this automated scoring mechanism, we further introduce GPT-5 to conduct an independent re-check on all teacher behavior description samples and calculate the Pearson correlation coefficient (γ) between its scores and the original scores from DeepSeek-R1. The results show an overall correlation of $\gamma = 0.87$, indicating a high consistency between the two scoring methods.

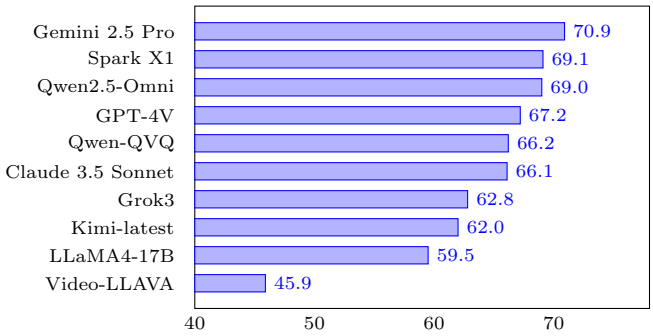


Figure 10: LLM-as-a-judge Evaluation Results for Teacher Behavior Description.

For the samples where $\gamma < 0.8$ (totaling 212 instances), we further invited three experts in the field of education to perform a manual re-evaluation to verify the reasonableness of the scores. This multi-layer evaluation mechanism ensures that both lexical similarity and deep semantic consistency are taken into account, thereby significantly increasing the reliability of our benchmarking results.

Specifically, we design a rigorous 100-point weighted deduction scheme to quantify semantic fidelity, as detailed in Table 6. The scoring framework comprises six dimensions: Core Action Accuracy (30%) serves as the primary metric, imposing the most severe penalties for action misclassification to ensure fundamental validity. Spatiotemporal Localization (15%) and Interactive Object Recognition (15%) jointly evaluate the precision of spatial grounding and target alignment. Content Consistency (20%) strictly penalizes logical contradictions and hallucinations, while Environmental Detail Fidelity (10%) and Linguistic Specification (10%) assess the richness of contextual cues and terminological correctness. This hierarchical weighting strategy ensures that the evaluation prioritizes action-centric semantic precision while maintaining the overall structural coherence of the generated descriptions.

Evaluation scores are presented in Figure 10. Scoring results reveal that Gemini 2.5 Pro(70.9), Spark X1(69.1) and Qwen2.5-Omni(69.0) achieve relatively high semantic alignment, indicating their strong capacity to model key instructional components such as action intent, spatial positioning, and interaction targets. GPT-4V(67.2), Qwen-QVQ(66.2) and Claude 3.5 Sonnet (66.1) followed closely, suggesting a high degree of linguistic coherence, though it exhibited occasional misinterpretations of behavioral details. As an automated semantic evaluation framework, LLM-as-a-judge provides interpretable and discriminative metrics that go beyond surface-level textual overlap, offering deeper insights into whether large models truly comprehend instructional behavior. Scoring examples from DeepSeek-R1 are provided in Figure 11.

5. CONCLUSION

In this work, we addressed the critical need to evaluate Video-LLMs in educational contexts, particularly for teacher behavior understanding. We proposed a multi-stage Human-LLM collaborative annotation pipeline that ensured anno-

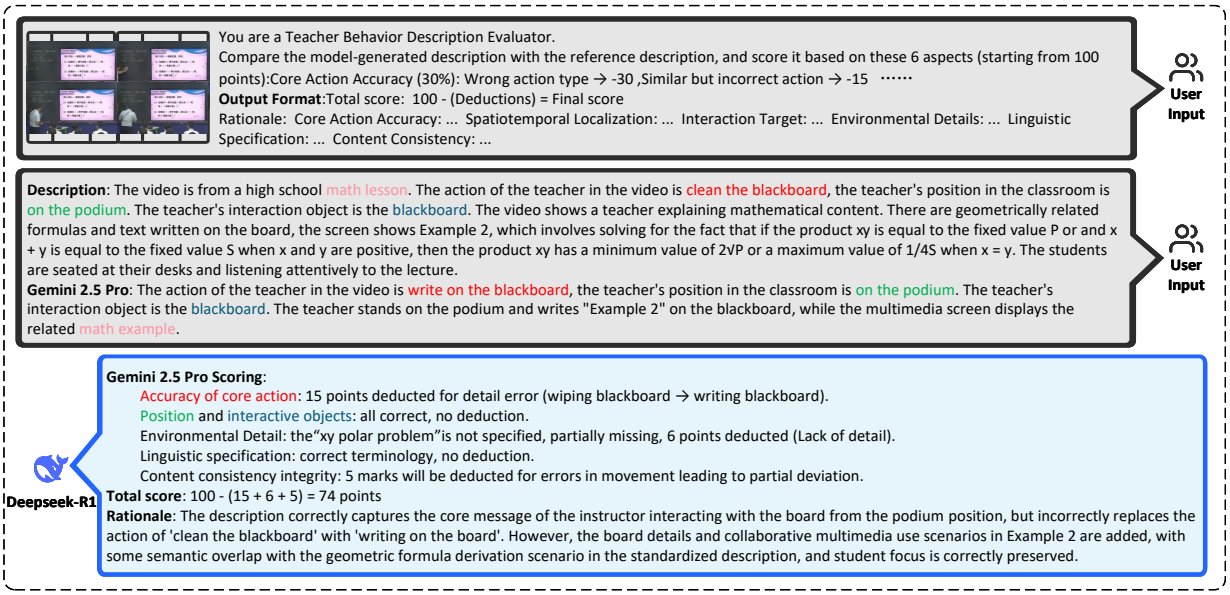


Figure 11: A scoring example of using DeepSeek-R1 to evaluate for teacher behavior description.

Table 6: Semantic evaluation criteria for LLM-as-a-judge ($\alpha + \beta + \gamma + \delta + \epsilon + \zeta = 100\%$).

Evaluation Dimension & Weight	Deduction Rules	Example
Core Action Accuracy ($\boldsymbol{\alpha}\%$)	Directional error: $-\alpha$ Detail-level error: $-\alpha/2$	Describing "clean the blackboard" as "lecture" Describing "lecture" as "multimedia teaching"
Spatiotemporal Localization ($\boldsymbol{\beta}\%$)	Spatial location error: $-\beta$	"on the podium" misjudged as "among students"
Interactive Object Recognition ($\boldsymbol{\gamma}\%$)	Interaction target error: $-\gamma$	"blackboard" misjudged as "multimedia"
Environmental Detail Fidelity ($\boldsymbol{\delta}\%$)	Completely missing details: $-\delta$ Partially missing details: $-\delta/2$	Missing all environmental cues Teacher attire "suit" misjudged as "sportswear"
Linguistic Specification ($\boldsymbol{\epsilon}\%$)	Terminology error: $-\epsilon/2$ to $-\epsilon$ Logical inconsistency: $-2/\epsilon$ to $-\epsilon$	Describing "podium" as "desk" Describe "clean" and "write" at the same time.
Content Consistency Integrity ($\boldsymbol{\zeta}\%$)	Completely off-topic: $-\zeta$	Misjudging a math lesson as historical lesson

tation efficiency and multimodal consistency. Through this pipeline, we introduced **MTBU-Bench**, the first multimodal benchmark tailored for this task, spanning 15 K-12 and higher education disciplines across 300+ classroom scenarios. Through large-scale evaluations of ten mainstream Video-LLMs across action recognition, spatiotemporal localization and behavior description, we uncovered key insights. Firstly, audio and text modalities were critical for disambiguating visually similar behaviors by incorporating verbal cues and contextual intent. Secondly, abstract actions posed significant challenges, highlighting limitations in interpreting weakly visualized behaviors. Thirdly, all models struggled with fine-grained distinctions in interaction targets and temporal boundary extraction in long-form videos. Overall, **MTBU-Bench** provided a rigorous framework for advancing Video-LLMs in education, while our findings highlighted urgent needs for improved temporal modeling, cross-modal alignment, and handling of abstract instructional behaviors. Furthermore, we recognize that our current classroom videos are predominantly sourced from mainland China, which introduces a limitation in cross-cultural generalization. Since instructional norms and spatial layouts vary significantly across cultures, scaling this benchmark to encompass global diverse educational settings remains an important direction for future research. We hope it will inspire further developments in instructional video understanding, MLLMs alignment, and behavior-aware educational applications.

Ethical Statement

The construction of MTBU-Bench was conducted in strict adherence to ethical guidelines for human subjects research and was formally approved by our Institutional Review Board (Approval No. IMU20260402-0017). To safeguard the privacy of participating educators and K-12 students, all Personally Identifiable Information (PII) was systematically removed, and robust facial blurring was applied to recognizable individuals. The dataset is released exclusively for non-commercial, academic research under a strict Data Use Agreement to prevent misuse.

6. ACKNOWLEDGMENTS

This work was funded by the National Natural Science Foundation of China (Nos. 62567005 and 62406127), and Natural Science Foundation of Inner Mongolia Autonomous Region of China (No. 2025MS06004).

7. REFERENCES

[1] K. Ahmad et al. Data-driven artificial intelligence in education: A comprehensive review. *IEEE Trans. Learn. Technol.*, 17:12–31, 2023.

[2] M. A. AlAfnan et al. ChatGPT as an educational tool: Opportunities, challenges, and recommendations for communication, business writing, and composition courses. *J. Artif. Intell. Technol.*, 3(2):60–68, 2023.

[3] A. Athar et al. Vicas: A dataset for combining holistic and pixel-level video understanding using captions with grounded segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19023–19035, June 2025.

[4] S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. ACL Workshop Intrinsic Extrinsic Eval. Measures Mach. Transl. Summarization*, pages 65–72, 2005.

[5] X. B. Bruce, Y. Liu, and K. C. C. Chan. Multimodal fusion via teacher-student network for indoor action recognition. In *Proc. AAAI Conf. Artif. Intell.*, volume 35, pages 3199–3207, 2021.

[6] T. Cai, Y. Xiong, C. He, C. Wu, and S. Zhou. Tbu: A large-scale multi-mask video dataset for teacher behavior understanding. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024.

[7] N. Chaturvedi. LLMs and NLP for generalized learning in AI-enhanced educational videos and powering curated videos with generative intelligence. In *Proc. 1st Workshop NLP Sci. (NLP4Science)*, pages 148–154, 2024.

[8] D. Chen et al. MLLM-as-a-judge: Assessing multimodal LLM-as-a-judge with vision-language benchmark. In *Forty-first International Conference on Machine Learning*, pages 6562–6595, 2024.

[9] DeepMind. Gemini 2.5 Pro, 2025. Accessed: 2025-07-30.

[10] M. Desmond et al. EvalAssist: LLM-as-a-judge simplified. In *Proc. AAAI Conf. Artif. Intell.*, volume 39, pages 29637–29639, 2025.

[11] M. Fang, X. Du, Q. Liu, Y. Zhou, Q. Liang, and S. Liu. Which is the better teacher action? a new ranking model and dataset. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7695–7699. IEEE, 2024.

[12] M. Fang et al. A teacher classroom dress assessment method based on a new assessment dataset. In *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, pages 7251–7259, 2024.

[13] J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychol. Bull.*, 76(5):378–382, 1971.

[14] C. Fu et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis, 2024. arXiv preprint.

[15] C. Fu et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24108–24118, June 2025.

[16] D. Guo et al. Deepseek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. arXiv preprint.

[17] Y. Guo et al. Vtg-llm: Integrating timestamp knowledge into video LLMs for enhanced video temporal grounding. In *Proc. AAAI Conf. Artif. Intell.*, volume 39, pages 3302–3310, 2025.

[18] W. Hong, Y. Cheng, Z. Yang, W. Wang, L. Wang, X. Gu, S. Huang, Y. Dong, and J. Tang. Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8450–8460, June 2025.

[19] B. Huang, X. Wang, H. Chen, Z. Song, and W. Zhu. Vtimellm: Empower llm to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14271–14280, 2024.

[20] C. Huang, J. Zhu, Y. Ji, W. Shi, M. Yang, H. Guo, J. Ling, P. De Meo, Z. Li, and Z. Chen. A multi-modal dataset for teacher behavior analysis in offline classrooms. *Scientific Data*, 12(1):1115, 2025.

[21] G. Huang et al. Multimodal pretraining for dense video captioning, 2020. arXiv preprint.

[22] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3192–3199, 2013.

[23] J. Jia, J. Song, Q. Hu, S. Tang, and S. Xu. Tar: A dataset of teacher-teaching action recognition. In *2023 8th International Conference on Image, Vision and Computing (ICIVC)*, pages 676–681. IEEE, 2023.

[24] X. Ju et al. Miradata: A large-scale video dataset with long durations and structured captions. *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 37:48955–48970, 2024.

[25] R. Khirodkar et al. Harmony4D: A video dataset for in-the-wild close human interactions. *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 37:107270–107285, 2024.

[26] Y. Kong and Y. Fu. Human action recognition and prediction: A survey. *Int. J. Comput. Vis.*, 130(5):1366–1401, 2022.

[27] K. Li, Y. Wang, Y. He, Y. Li, Y. Wang, Y. Liu, Z. Wang, J. Xu, G. Chen, P. Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024.

[28] Z.-Z. Li et al. From system 1 to system 2: A survey of reasoning large language models, 2025. arXiv preprint.

[29] Z. Liao. CH-CC: A chinese multimodal classroom atmosphere analysis dataset based on teachers' behavior and voice, 2023.

[30] B. Lin et al. Video-LLaVA: Learning united visual representation by alignment before projection, 2023.

arXiv preprint.

[31] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[32] Q. Liu et al. Exploration of the characteristics of teachers' multimodal behaviours in problem-oriented teaching activities with different response levels. *Brit. J. Educ. Technol.*, 55(1):181–207, 2024.

[33] K. Papineni et al. BLEU: A method for automatic evaluation of machine translation. In *Proc. Annu. Meet. Assoc. Comput. Linguistics*, pages 311–318, 2002.

[34] T. Perrett et al. Hd-epic: A highly-detailed egocentric video dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23901–23913, June 2025.

[35] A. Sabuncuoglu and T. M. Sezgin. Multimodal group activity dataset for classroom engagement level prediction, 2023. arXiv preprint.

[36] M. R. Salehi et al. ActionAtlas: A videoqa benchmark for domain-specialized action recognition. *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 37:137372–137402, 2024.

[37] V. Sharma, M. Gupta, A. Kumar, and D. Mishra. Edunet: A new video dataset for understanding human activity in the classroom environment. *Sensors*, 21(17):1424–8220, 2021.

[38] Y. Shi, R. Liang, and Y. Xu. EducationQ: Evaluating LLMs' teaching capabilities through multi-agent dialogue framework, 2025. arXiv preprint.

[39] X. Tan, Y. Luo, Y. Ye, F. Liu, and Z. Cai. Allvb: All-in-one long video understanding benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7211–7219, 2025.

[40] Y. Tang, J. Bi, S. Xu, L. Song, S. Liang, T. Wang, D. Zhang, J. An, J. Lin, R. Zhu, A. Vosoughi, C. Huang, Z. Zhang, P. Liu, M. Feng, F. Zheng, J. Zhang, P. Luo, J. Luo, and C. Xu. Video understanding with large language models: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 36(2):1355–1376, 2026.

[41] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[42] Q. Wang et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8428–8437, June 2025.

[43] J. Xiao, N. Huang, H. Qin, D. Li, Y. Li, F. Zhu, Z. Tao, J. Yu, L. Lin, T.-S. Chua, et al. Videoqa in the era of llms: An empirical study. *International Journal of Computer Vision*, 133(7):3970–3993, 2025.

[44] J. Xu, Z. Guo, J. He, H. Hu, T. He, S. Bai, K. Chen, J. Wang, Y. Fan, K. Dang, B. Zhang, X. Wang, Y. Chu, and J. Lin. Qwen2.5-omni technical report, 2025.

[45] L. Yan et al. Practical and ethical challenges of large language models in education: A systematic scoping review. *Brit. J. Educ. Technol.*, 55(1):90–112, 2024.

[46] Z. Yu, D. Xu, J. Yu, T. Yu, Z. Zhao, Y. Zhuang, and D. Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9127–9134, 2019.

[47] Z. Zhang, Z. Jia, H. Wu, C. Li, Z. Chen, Y. Zhou, W. Sun, X. Liu, X. Min, W. Lin, and G. Zhai. Q-bench-video: Benchmark the video quality understanding of lmms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3229–3239, June 2025.

[48] B. Zhou, H. Yang, D. Chen, J. Ye, T. Bai, J. Yu, S. Zhang, D. Lin, C. He, and W. Li. Urbench: A comprehensive benchmark for evaluating large multimodal models in multi-view urban scenarios. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 10707–10715, 2025.

[49] L. Zhou, C. Xu, and J. Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, pages 7590–7598, 2018.

Measuring Math or Mouse Skills? Behavioral Dimensions and Multiple Pathways to Test-Taking Efficiency in Computer-Based Testing

Xinman Liu*
Stanford University
xinman@stanford.edu

Eunjung Myoung*
Stanford University
ejmyoung@stanford.edu

ABSTRACT

This study investigates behavioral signatures in computer-based testing and their relationship to students' test-taking efficiency, contributing a granular behavioral taxonomy that moves beyond rigid archetypes. Using process data from the 2017 NAEP Grade 8 Mathematics assessment (N=1,642), we developed an unsupervised-to-supervised learning framework utilizing Principal Component Analysis and Gaussian Mixture Modeling to identify five core behavioral dimensions (i.e., Navigation Intensity, Problem-Solving Tool Use, UI Interface Interaction, Deliberate Pausing, and Content Interaction Over UI), followed by supervised methods including tree-based model to predict efficiency patterns. Through engineering 43 features, including a Relative Action Intensity Score to standardize interaction density across heterogeneous items, our analysis categorized students into 13 distinct behavioral clusters with efficiency rates ranging from 13% (Impulsive Rushers) to 73% (Strategic Navigators). Our XGBoost model achieved an AUC of 0.729 (Adjusted AUC = 0.458), outperforming all previous approaches from the 2019 NAEP Data Mining Competition. Importantly, we reveal that UI complexity significantly impacts efficiency, with inefficient students disproportionately interacting with interface elements rather than content. Our granular behavioral taxonomy demonstrates that while inefficiency concentrates in specific patterns (particularly impulsive rushing), efficiency emerges through multiple strategic pathways. These results provide evidence-based guidelines for computer-based tests (CBT) design, recommending reduced UI complexity to minimize extraneous cognitive load and accommodate diverse student strategies.

Keywords

computer-based testing, test-taking behavioral patterns, test-taking efficiency, educational data mining, process data, gaussian mixture modeling

*Equal contribution.

Xinman Liu, and Eunjung Myoung. Measuring Math or Mouse Skills? Behavioral Dimensions and Multiple Pathways to Test-Taking Efficiency in Computer-Based Testing. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 180–192. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039982

1. INTRODUCTION

Recent technological advancements in the test-taking landscape have shifted traditional paper-based exams toward computer-based tests (CBT). This transition introduces new interaction demands, as students must navigate complex digital interfaces by clicking, dragging, typing, and scrolling while solving assessment items [9]. Such changes highlight the importance of understanding how students engage with CBT interfaces. For instance, for students with limited digital literacy, this can impose additional cognitive load, making it difficult to efficiently complete tests and potentially affecting performance [13]. More critically, this raises concerns about test validity—whether assessments truly measure cognitive abilities in the intended domain or inadvertently assess digital interface proficiency [10, 22, 23]. On the other hand, the transition to CBT enables collection of detailed behavioral data, including clicks, drags, navigational paths, and precise timestamps, which are the data previously unavailable in paper-based tests [14].

This presents opportunities to examine test-taking behaviors comprehensively and investigate how interaction patterns relate to efficiency. Using process data from the 2017 NAEP Grade 8 Mathematics assessment, which contains rich behavioral information from over 1,600 students, this study develops new approaches for understanding and predicting test-taking efficiency in digital environments [20]. Two key research questions guide this study: **(1) What distinct behavioral patterns emerge in students' interaction with computer-based tests?** and **(2) Which behavioral patterns distinguish efficient from inefficient test-taking approaches?**

Our study makes several key contributions: First, we develop a five-dimensional behavioral framework to capture distinct engagement patterns, leading to the identification of 13 behavioral clusters that reveal multiple pathways to efficiency while showing inefficiency concentrates in specific problematic patterns. Second, using these behavioral dimensions as features in supervised machine learning models, we achieve improved predictive performance on test-taking efficiency that substantially outperforms previous approaches including winners of the 2019 NAEP Data Mining Competition [19, 6, 12, 25]. Finally, we provide evidence-based CBT interface design recommendations, particularly regarding UI complexity reduction to minimize extraneous cognitive load. [1]

[1]GitHub Code:
https://github.com/yo-lxmmm/math-or-mouse-skills

2. RELATED WORK

The shift toward computer-based testing has enabled researchers to apply educational data mining techniques to understand student behaviors and predict performance outcomes [5]. Process data analysis has emerged as a key approach for examining the sequence and timing of student actions during digital assessments [14].

In particular, the 2019 NAEP Data Mining Competition generated significant research interest in predicting test-taking efficiency, resulting in multiple methodological approaches published in a special issue of the Journal of Educational Data Mining [3]. Bosch (2021) [6] employed AutoML feature engineering using TSFRESH and Featuretools, generating over 4,000 automatically extracted features and achieving an Adjusted AUC of 0.331. However, the limited interpretability of this approach makes it difficult to derive actionable insights for practitioners. Zehner et al. (2021) [25] adopted a top-down psychometric approach, using latent speed and ability models to categorize students into four distinct test-taking types, achieving moderate performance with Adjusted AUC of 0.270. Patel et al. (2021) [19] utilized educational process analysis and fuzzy mining algorithms, focusing on temporal sequences and behavioral prototypes, and achieving Adjusted AUC of 0.326. In particular, they specifically discuss how observed "flipping behaviors" could inform interface design improvements. Levin (2021) [12] won the competition with Adjusted AUC = 0.315 by combining process mining with expert feature engineering, using approximately 330 features reduced to fewer than 30 through iterative selection to build a predictive model.

While these approaches established critical benchmarks, they often relied on binary classifications or rigid process sequences that overlook the multi-dimensional nature of student-interface interactions. Our study advances this literature by proposing a novel five-dimensional behavioral framework spanning across interaction intensity, tool use, and UI engagement, to identify 13 distinct student clusters. By accounting for both interpretability and performance, our model achieves an Adjusted AUC of 0.458, significantly outperforming the original competition winners while offering new insights into how interaction complexities impact testing efficiency.

3. METHODS

3.1 Data Explanation

For this research, we utilized the process data from the 2017 National Assessment of Educational Progress (NAEP) Grade 8 Mathematics assessment. NAEP is the largest and only representative ongoing assessment that measures U.S. students' knowledge and learning experiences in various subjects (e.g., mathematics, reading, science, writing, civics, economics, etc.) across the nation, states, and 27 urban districts, dating back to 1969 [4]. NAEP officially transitioned from paper-based to digitally-based assessment (DBA) for mathematics and reading in 2017 [20]. This transition to DBA has allowed NAEP to collect and archive "process data." Process data includes new types of detailed information beyond merely analyzing students' right or wrong. It captures students' interactions with the assessment platform and tasks, time-stamped records of student-initiated actions (e.g., clicking drawing tools) after viewing an item, logged processes students follow to arrive at answers, the steps or strategies students use for problem-solving, and the resources or tools utilized during assessments. The process data offers valuable opportunities for researchers to explore student test-taking behaviors and determine whether students use their time efficiently during tests [5].

The complete process data from the 2017 NAEP Grade 8 mathematics assessment, as well as Grade 4 mathematics, are available through restricted access by request. However, for this research, we utilized a limited dataset of Grade 8 process data provided by the Educational Testing Service for the NAEP Data Mining 2019 Competition. The competition organizers publicly provided NAEP process data for approximately 2,400 students who took the NAEP test in early 2017, and participants were asked to use this process data to build a behavioral detection model predicting whether students would spend their time efficiently on the test. While the original dataset includes survey questionnaires capturing students' demographic characteristics, their performance, and learning opportunities both in and outside of school, and their broader educational experiences, the competition dataset contains limited information, focusing exclusively on students' test-taking processes and their efficiency (see Appendix A for the full variable list that we were able to access). Nevertheless, acknowledging these constraints, we aimed to maximize the potential of the available data to explore and understand student behaviors in digital assessments.

For the analysis, we utilized four key variables from the complete dataset: (1) StudentID, (2) Observable, (3) EventTime, and (4) Efficiently Completed Block B. The dataset consisted of a total of 1,642 students. From the outset, the competition organizers designated 1,232 students as the training set and 410 students as the test set, and we maintained this original division in our analysis. However, there were no inherent differences in data collection methods or conditions between these two groups; all 1,642 students completed two 30-minute test sections referred to as Block A and Block B, under identical testing conditions. In terms of efficiency, the training set includes 744 efficient, 488 inefficient test-takers and the test set has 248 efficient and 162 inefficient test-takers.

Among the selected variables, the Observable variable was particularly important for our research interest, as it contained information about the specific actions taken by students during the test. Of the initial 42 actions recorded, we included only 38 actions in our analysis. Four actions related solely to procedural interactions necessary for beginning or ending the test (i.e., "Yes," "No," "OK," and "Leave Section") were excluded. The EventTime variable allowed us to examine the duration of individual student actions and the total amount of time students spent on test items.

Regarding the efficiency variable (i.e., Efficiently Completed Block B), the NAEP Mathematics test consisted of two 30-minute blocks (Block A and Block B), each with a fixed set of questions. The objective of the competition was to predict students' efficiency in Block B using their process data from Block A. That is, all predictor features used in our analyses were derived from students' interactions in Block A, and the efficiency outcome was derived from Block B.

The outcome variable was a binary indicator (i.e., True or False) showing whether students spent their time efficiently or inefficiently in Block B. The competition organizers operationalized "efficiency" as: (1) completing all problems in Block B, and (2) allocating a reasonable amount of time to each problem [20]. To determine what constituted a "reasonable amount of time," they conceptually defined it as the minimum possible time needed to solve each problem, and operationalized it by ranking the total time each student spent on each problem in Block B and establishing the 5th percentile as the lower-bound cutoff, applied per problem, such that a student was labeled as "inefficient" if their time on any problem fell below the 5th percentile of all students' time on that same problem, and "efficient" only if they completed all problems and no problem fell below this threshold. Importantly, this operationalization is purely relative and time-based, without consideration of item correctness or validated psychometric standards. The competition organizers themselves acknowledged that defining this threshold is difficult and has limitations [20]. Despite this, the definition of efficiency is built into the competition dataset, and we adopt it as such throughout our study.

3.2 Feature Engineering

Following NAEP's guidance that researchers must develop meaningful variables from process data before deriving insights [16], we conducted feature engineering as described below. As part of our preprocessing and feature engineering approach, we first applied an exploratory Principal Component Analysis (PCA) [11] for feature engineering to the 38 student actions captured by the "Observable" variable. The main purpose of this initial PCA was dimensionality reduction—to uncover underlying dimensions reflecting distinct student test-taking behaviors, which then provided a data-driven foundation for subsequent feature engineering. Building on the categories identified through our initial PCA of student actions, we engineered several quantitative features to capture students' test-taking behavior. The resulting PCA dimensions provided an empirically driven basis for the subsequent feature-engineering phase. Specific variables derived from these PCA-based behavioral dimensions are detailed further in the Results section. We noted that this PCA represents the initial step in the project's analytic workflow, guiding the process of feature engineering. A second, separate PCA as the method for unsupervised learning was conducted later as part of the analytic methods and is described in detail in the subsequent section.

3.2.1 PCA for Feature Engineering

The decision to retain how many principal components was guided by a visual inspection of a scree plot of eigenvalues, where we observed a notable decrease and leveling off of the explained variance after the fourth component, suggesting that additional components beyond the fourth provided diminishing explanatory value. We therefore decided to maintain four dimensions for interpretability, ensuring that meaningful and coherent dimensions representing students' test interactions. Based on the PCA results depicted in Figure 1, we labeled each principal component and categorized the 38 actions for each dimension. We labeled the first dimension as *Problem-Solving Support Actions* which is associated with the actions that directly support students' problem-solving process, such as activating scratchwork modes, drawing, and using calculators. The second dimension is labeled as *Open Response Actions*, and it involves tools directly used for inputting answers, including opening and closing the equation editor. The third dimension was named *Navigation Actions*, which includes the action to navigate between test items, such as clicking Next, Back button, or item-entry actions. The fourth dimension is named *UI Interaction Actions*, and it pertains to interface modifications that are indirectly related to problem-solving, such as zooming, scrolling, and managing timers.

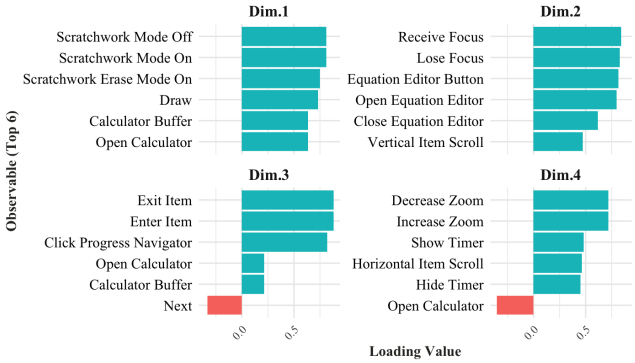


Figure 1: Top Six Contributing Observables in Four Principal Components

Additionally, we found seven actions (e.g., click choice, drop choice, clear answer) that did not align with any specific principal component, as these actions represented essential answering behaviors required across all item types, regardless of students' interaction patterns. These actions were considered together with the four dimensions for feature engineering with the label of *Essential Answer Actions*. Detailed descriptions of 38 actions grouped under each principal component are presented in Table 1. These new categories of the actions were utilized further to create new variables with feature engineering.

3.2.2 Feature Engineering

In addition to the interaction pattern categories identified through our initial PCA of student actions, we engineered several quantitative features to capture different dimensions of student test-taking behavior. As presented in Table 2, we engineered 43 features across five categories of variables: *Interaction Patterns, Time Management, Answer Timing, Hiatus Time, and Navigation Patterns.*

Interaction Patterns. To quantify the density of student interactions across the different action categories as defined above, we developed the Relative Action Intensity Score (RAIS) metric, which standardizes student action counts relative to the average student behavior, and enables meaningful comparisons across students and items by controlling for question-specific interaction items. Specifically, the overall RAIS was calculated by first determining the ratio of a student's action count on each item to the average action count for that item across all students, then averaging these ratios across all items attempted by the student. Values greater than 1.0 indicating above-average interaction intensity and values less than 1.0 indicating below-average interaction intensity.

Table 1: Categorization of 38 Observable Actions into Five Action Categories Based on PCA Results

PC	Name of Category	Actions	# of Actions
Dim 1	Problem-Solving Support Actions	Open Calculator, Close Calculator, Move Calculator, Calculator Buffer, Scratchwork Mode On, Scratchwork Mode Off, Scratchwork Draw Mode On, Scratchwork Erase Mode On, Scratchwork Highlight Mode On, Highlight, Draw, Erase, Clear Scratchwork	13
Dim 2	Open Response Actions	Receive Focus, Lose Focus, Open Equation Editor, Close Equation Editor, Equation Editor Button	5
Dim 3	Navigation Actions	Next, Back, Enter Item, Exit Item, Click Progress Navigator	5
Dim 4	UI Interaction Actions	Change Theme, Hide Timer, Show Timer, Increase Zoom, Decrease Zoom, Text-to-Speech, Vertical Item Scroll, Horizontal Item Scroll	8
Extra	Essential Answer Actions	Click Choice, Math Keypress, Drop Choice, First Text Change, Last Text Change, Clear Answer, Eliminate Choice	7

Table 2: Engineered Features by Category

Categories	Features	# of Features
Interaction Patterns	overall RAIS, problem-solving support RAIS, open response RAIS, navigation RAIS, UI interaction RAIS, essential answer RAIS	6
Time Management	average time per question, time per question variability, essential answer time, essential answer average interval, problem-solving time, problem-solving average interval, open response time, open response average interval, navigation time, navigation average interval, UI interaction time, UI interaction average interval, total time, essential answer proportion, problem-solving proportion, open response proportion, navigation proportion, UI interaction proportion, problem-solving to answer ratio, navigation to answer ratio, open response to answer ratio, UI interaction to answer ratio	22
Answer Timing	time to first answer variability, median time to first answer	2
Hiatus Time	very rapid hiatus percentage (<0.16s), rapid hiatus percentage (0.16-0.5s), normal hiatus percentage (0.5-2.6s), deliberate hiatus percentage (2.6-6.8s), extended hiatus percentage (>6.8s), median hiatus	6
Navigation Patterns	total question switches, forward switches, backward switches, forward movement percentage, unique questions attempted, average revisits per question, maximum revisits per question	7

We then developed dimension-specific RAIS metrics by applying the same computation above to each of the action categories identified from the initial PCA: problem-solving support RAIS measures intensity of optional tool usage (e.g., calculator, scratchwork, drawing tools); open response RAIS measures intensity of interactions with text input and equation tools; navigation RAIS measures forward and backward movements between questions and within the test; UI interaction RAIS measures interface adjustments such as zooming, theme changes, and timer visibility; and essential answer RAIS measures intensity of actions directly required to submit answers. The rationale behind engineering these dimension-specific RAIS metrics is to provide more nuanced behavioral patterns by distinguishing between students who may have the same overall activity levels but distribute their actions differently across interaction types.

Time Management. We engineered 22 temporal features to capture time management patterns. For each student, we calculated average time per question using the difference between the earliest and latest timestamp on each question, as well as time per question variability to measure consistency in time allocation. We then quantified the time spent on each of the five action categories (i.e., problem-solving time, open response time, navigation time, UI interaction time, and essential answer time) by summing the time hiatuses between consecutive actions within each category, with an arbitrarily set upper threshold of 30 seconds to avoid overestimation from extended pauses between different question attempts. For each action category, we also calculated average interval measures (i.e., problem-solving average interval, open response average interval, navigation average interval, UI interaction average interval, and essential answer average interval) to characterize the dispersion of each action type (i.e., how "spaced out" the intervals are per action type). To be more specific, the lower values of these variables suggest

more dense engagement in a category and higher values indicate more dispersed behavior, that students do not perform the certain action regularly.

For comparison across students, we derived proportions of total time spent on each category (problem-solving proportion, open response proportion, navigation proportion, UI interaction proportion, essential answer proportion) and efficiency ratios (problem-solving to answer ratio, navigation to answer ratio, open response to answer ratio, UI interaction to answer ratio) to assess how students distributed their time across different types of interactions.

Answer Timing. We captured students' answering approaches with two features: median time to first answer (i.e., seconds between entering a question and first answer attempt) and time to first answer variability (standard deviation of how quickly a student responds upon seeing a question, measured across all questions). These metrics serve as indicators of comprehension speed, decision-making approaches, and impulsivity in responding. They were calculated separately for different question types (e.g., multiple-choice, fill-in-the-blank, and matching) to account for the varying response mechanisms.

Hiatus Time. We analyzed time gaps between consecutive actions to quantify students' interaction rhythms (i.e., how fast they perform clicks). Based on the overall hiatus distribution across all students, we established five categories and calculated the percentage of actions in each: very rapid hiatus percentage (<0.16s), rapid hiatus percentage (0.16-0.5s), normal hiatus percentage (0.5-2.6s), deliberate hiatus percentage (2.6-6.8s), and extended hiatus percentage (>6.8s). We also calculated median hiatus as an overall indicator of click pace.

Navigation Patterns. We engineered 7 features to capture navigation behaviors: total question switches, forward switches (to later questions), backward switches (to earlier questions), forward movement percentage (proportion of forward-directed moves to capture a predilection toward linear answering strategies), unique questions attempted, average revisits per question, and maximum revisits per question (where a revisit was defined as returning to a previously seen question after having navigated away from it). These metrics collectively capture students' sequencing strategies and tendencies toward linear progression versus non-linear back-and-forth exploration of the test content.

Upon engineering all features, we standardized most of them using z-score normalization to ensure comparability across different scales. However, we preserved the original values of RAIS features (already normalized relative to average student behavior), any proportion/percentage features (bounded between 0-1), and ratio features (inherently comparable across students) to maintain their interpretability. To prevent data leakage, test data features were standardized using means and standard deviations derived exclusively from the training dataset.

3.3 Analytic Method

The analytic approach of this study was conducted in two distinct phases: unsupervised learning and supervised learning. This methodological structure allowed for an initial exploration of behavioral patterns without predefined labels, subsequently informing predictive modeling of efficiency.

3.3.1 Unsupervised Learning

We implemented unsupervised learning techniques to systematically identify and describe students' test-taking behaviors. Following the creation of the engineered features, we conducted a second PCA to further distill these features into coherent behavioral patterns that reflect underlying student test-taking strategies. Based on the results of this PCA, we applied Gaussian Mixture Modeling (GMM; [21]) to cluster students into groups characterized by similar test-taking behaviors as captured by the PCA. We leverage GMM for this task as it offers a probabilistic approach to clustering. Specifically, unlike deterministic methods such as k-means that assign each student to a fixed cluster membership, GMM acknowledges students' varying combination and extent of behavioral patterns as existing along a continuum with fuzzy boundaries, offering a probabilistic interpretation [21].

3.3.2 Supervised Learning

The supervised learning phase of this research consisted of systematically applying predictive modeling techniques based on variables derived from the previous unsupervised learning stage. Using the training data (n = 1232), we trained predictive models based on the PCA-derived behavioral dimensions, feature-engineered variables, and student clusters from our unsupervised analyses. Specifically, we employed tree-based methods—Decision Tree, Random Forest [7], and XGboost—to predict students' test-taking efficiency using all engineered features.

For Decision Tree modeling, we implemented 10-fold cross-validation to identify the optimal complexity parameter, exploring values from 0.0001 to 0.1. This tuning process balanced model complexity and predictive power while preventing overfitting. The final tree was visualized to examine decision pathways that distinguished efficient from inefficient test-takers. We then implemented the Random Forest model using ranger for computational efficiency, applying a similar 10-fold cross-validation but with hyperparameter optimization including mtry values of 2, 4, 6, 8, 10, and 12 (variables considered at each split), min.node.size values of 1, 2, and 3 (minimum observations in terminal nodes), and gini splitrule criterion. This approach generates an ensemble of decorrelated trees to reduce variance while maintaining predictive power. Lastly, for XGBoost, we employed a multiparameter tuning strategy with 5-fold cross-validation, optimizing boosting rounds (50, 100), max depth (3, 6), learning rate (0.1, 0.3), and subsampling rates for both observations and features (0.7-1.0). For all three models, we evaluated and compared accuracy, sensitivity, specificity, and Area Under the Curve (AUC) on the held-out test data (n = 410), then extracted the top 15 most important features identified by the model with the highest AUC. We also report the Adjusted AUC (which is equivalent to the Gini coefficient), calculated as $2 \times (\text{AUC} - 0.5)$, to directly compare with prior competition results [20]. Note that unlike conventional AUC, which has a chance level of 0.5, the Adjusted AUC is rescaled such that a value of 0 indicates chance-level performance and 1 indicates perfect discrimination. The

supervised-learning analytic framework provided not only predictive power but also interpretative clarity regarding how distinct behavioral patterns contributed to efficient or inefficient test-taking behaviors.

4. RESULTS

4.1 Unsupervised Learning

4.1.1 PCA for Behavioral Dimensions

As the first unsupervised learning, we conducted the PCA with 43 feature-engineered variables. The decision to retain how many principal components was again guided by a visual inspection of a scree plot of eigenvalues, where we observed a notable decrease and leveling off of explained variance after the fifth component. Therefore, we decided to maintain five dimensions which collectively explained 62% of the variance in user interactions. Figure 2 displays the top ten contributing variables for each retained principal component. Based on the factor loadings we labeled each dimension (see Table 3).

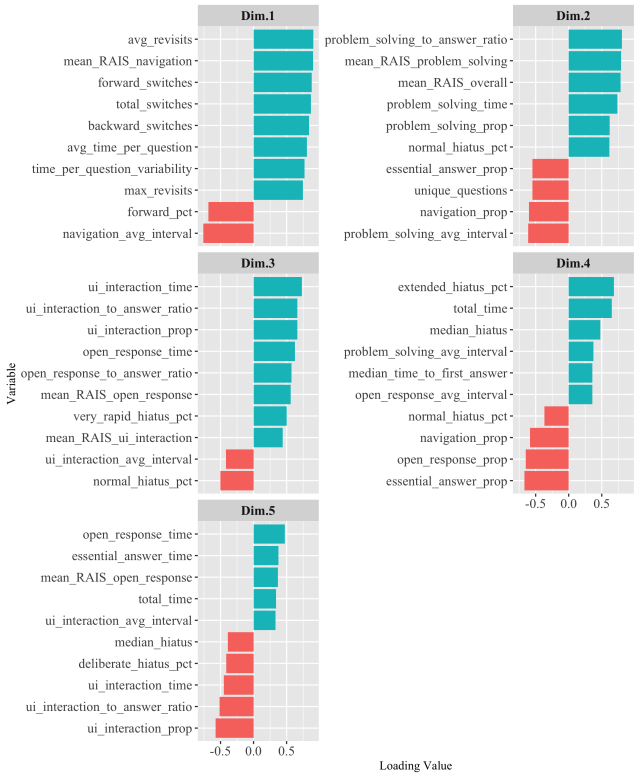


Figure 2: Top Ten Contributing Variables in Five Principal Components

Dimension 1: Navigation Intensity. This dimension is primarily characterized by variables such as average revisits, forward switches, total switches, and backward switches, all of which are strongly and positively associated with item-switching behavior. In contrast, navigation average interval and forward percentage negatively correlated with this dimension Given this, we interpret Dimension 1 as a measure of navigation intensity. A higher score suggests that students frequently revisit and switch between different sections within short time gaps, engaging in more active and fragmented navigation. Conversely, a lower score indicates students navigate less frequently and with longer intervals between transitions, suggesting a more deliberate, linear, or less exploratory approach to problem-solving.

Dimension 2: Problem-Solving Tool Use. Dimension 2 is marked by high positive loadings on variables related to problem-solving behaviors, such as problem solving to answer ratio, problem solving time, and mean RAIS problem solving. In contrast, it exhibits negative loadings for variables associated with other types of engagement, including essential answer prop and navigation prop. This suggests that students with higher scores in this dimension primarily engage in problem-solving tool use, focusing their efforts on problem-solving rather than other behaviors such as submitting essential answers or navigating between sections.

Dimension 3: UI Interface Interaction. This dimension is predominantly associated with UI interaction-related variables, such as UI interaction time, UI interaction to answer ratio, and UI interaction prop, all of which exhibit strong positive loadings. Therefore, we interpret Dimension 3 as measuring the extent to which students engage with the user interface. A higher score in this dimension indicates students who frequently interact with the UI elements. It aligns with the negative value of the UI interaction average interval. The higher factor loading of this variable (i.e., the negative value of this dimension) indicates students less frequently interact with UI and with longer intervals between transitions, which suggests a more deliberate and less exploratory approach to UI interaction.

Dimension 4: Deliberate Pausing. Dimension 4 is characterized by positive loadings on variables related to time spent before taking action, including total time, extended hiatus percentage, and median time to first answer. Meanwhile, it exhibits negative loadings on variables reflecting active engagement, such as essential answer proportion and open response proportion. This pattern suggests that students with higher scores in this dimension engage in deliberate pausing and contemplation, as indicated by longer total time spent, more frequent and extended hiatuses, and slower initial responses. Conversely, lower scores in this dimension indicate a higher relative proportion of active engagement behaviors, such as providing open-ended responses, submitting essential answers, and frequently navigating through the interface. These students tend to engage in actions more frequently rather than deliberately pausing for extended periods.

Dimension 5: Content Interaction Over UI. This dimension is positively associated with variables related to content engagement, such as open response time and essential answer time, while it is negatively associated with UI interaction variables. Based on this pattern, we interpret Dimension 5 as a measure of content-focused interaction versus UI-focused interaction. Students with higher scores in this dimension spend more time engaging with content, such as responding to open-ended questions and providing essential answers. In contrast, students with lower scores demonstrate a greater focus on UI interactions rather than content engagement.

Table 3: The Characteristics of Each Dimension and The Label of Them

Principal Component	Positive Scores (+)	Negative Scores (-)	Name of Dimension
1: Navigation Intensity vs. Efficiency	Frequent question-switching, multiple revisits, non-linear approach	Linear progression, fewer revisits, efficient navigation	Navigation Intensity
2: Problem-Solving Tool Utilization vs. Direct Answering	Heavy use of calculators and tools, high problem-solving time	Focus on direct answering with minimal tool use	Problem-Solving Tool Use
3: Interface Interaction vs. Content Focus	Significant time spent adjusting UI and test environment	Minimal interface adjustment, greater focus on content	UI Interface Interaction
4: Deliberate Pausing vs. Continuous Engagement	Extended pauses, deliberate reflection before answers	Continuous engagement, fewer pauses, faster responses	Deliberate Pausing
5: Content Interaction vs. Interface Adjustment	More time on responses and engaging with question content	More time adjusting interface relative to answering	Content Interaction over UI

4.1.2 Gaussian Mixture Modeling

Following the dimension reduction through the second PCA, we employed GMM to identify distinct clusters of students based on their test-taking behaviors as captured by the principal components. To determine the optimal number of clusters, we evaluated models with 2 to 16 clusters using the Bayesian Information Criterion (BIC). As shown in Figure 3, the VVE model (i.e., variable volume, variable shape, equal orientation) achieved the highest BIC values, peaking at 13 clusters before beginning to plateau[2]. In the context of clustering students by behaviors, the VVE model thus suggests that students' test-taking strategies may have varying prevalence (variable volume) and the degrees of dispersion across dimensions (variable shape); nonetheless, while they demonstrate different behavioral tendencies, the underlying relationships between these behavioral dimensions remain consistent across groups (equal orientation). Based on this analysis and considerations of interpretability, we selected the 13-cluster VVE as the optimal GMM model for capturing the diversity of behavioral patterns in our data.

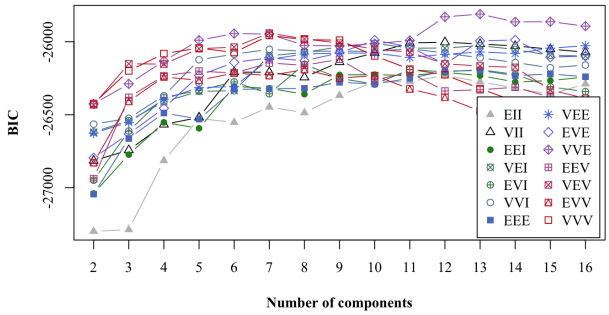


Figure 3: Gaussian Mixture Model Selection

The heatmap in Figure 4 visualizes the 13 identified student clusters, each with distinct behavioral patterns across the five principal components. Note that the dimension values are normalized across the student population, with a value of 0 representing the average behavior, while values further from 0 (either positive or negative) indicate more extreme deviations from the typical behavior patterns. The magnitude of a value indicates how strongly the particular behavioral pattern characterizes that cluster.

Table 4 presents the number of students per cluster and the percentage of efficient students within the cluster, along with more detailed interpretations of their key characteristics and notable behavioral dimension scores. Based on these hallmarks, we coined the names for each cluster. Note that the taxonomic naming convention of each cluster profile only captures the most distinct characteristics of the cluster for interpretability purposes, and should not be understood as defining the cluster or categorizing the students themselves. Furthermore, while cluster labels are grounded in observed behavioral patterns, they remain interpretive in nature.

The GMM clustering results on identified principal components reflect several implications. Firstly, there are multiple pathways to efficiency, but inefficiency concentrates in a few specific behavioral patterns. Specifically, while the percentage of efficient students vary somewhat gradually from Cluster 13 (73%) to Cluster 2 (41%), there is a dramatic drop to Cluster 3 (13%), where the overwhelming majority (87%) of students exhibit inefficient test-taking behavior. This *Impulsive Rushers* group (Cluster 3) demonstrates extremely low deliberative pausing (Dim 4: -4.87), minimal problem-solving tool utilization (Dim.2: -2.20), and disproportionate UI interface interaction over actual content engagement (Dim.5: -1.77). This combination of behaviors portrays the cluster as students who fiddle around with UI

[2]Note that typically when using BIC for model selection, the lower BIC values indicate better models; however in the Mclust implementation in R, the reported BIC is calculated with the negative sign from the standard BIC formula is removed, therefore higher values indicate better models.

Table 4: Description of Cluster Profiles (Ordered by % of Efficient Students per Cluster)

Cluster #	n	% of Efficient Students	Student Cluster Profile
13	98	73%	**Strategic Navigators:** Exhibit extremely high navigation intensity (Dim.1: 4.96) with limited tool utilization (Dim.2: -2.56). They take deliberate pauses when needed (Dim.4: 0.55). Their strategic navigation between questions appears highly effective.
11	120	68%	**Deliberate Pausers:** Characterized by taking thoughtful pauses (Dim.4: 1.63) while minimizing navigation switching (Dim.1: -1.22), tool usage (Dim.2: -1.66), and UI interaction (Dim.3: -0.89), reflecting a focused approach.
5	151	66%	**Interaction Minimalists:** Show consistently low scores across most dimensions, particularly avoiding excessive navigation (Dim.1: -2.65) and UI interactions (Dim.3: -1.22). They take a streamlined and economical approach with continuous progression and minimal usage of supplementary tools.
1	134	64%	**Balanced Reflectors:** Demonstrate positive but moderate scores across most dimensions with notable deliberate pausing (Dim.4: 1.24) and moderate navigation patterns (Dim.1: +0.91).
8	26	62%	**Reflective Pausers:** Show notably reduced navigation activity (Dim.1: -3.55) but very high scores on deliberate pausing (Dim.4: 4.19). They spend more time considering their responses but generally do not switch back and forth between questions.
12	106	61%	**Navigation-Tool Integrators:** Exhibit high navigation intensity (Dim.1: 3.80) with above-average tool usage (Dim.2: 1.99) while minimizing interface distractions (Dim.3: -1.44). They strategically leverage both navigation and problem-solving tools.
6	201	60%	**Interface-Oriented Moderates:** Exhibit moderate UI interaction engagement (Dim.3: 1.35) while showing below-average navigation and problem-solving tool usage. Their somewhat balanced approach avoids extremes in any dimension while moderately interacting with the UI.
7	109	59%	**Tool Maximizers:** Distinguished by very high problem-solving tool usage (Dim.2: +4.79) with moderate interface interaction (Dim.3: +0.63), reflecting that they mostly rely heavily on tools such as the scratchpad and the calculator.
4	41	59%	**Interface-Active Switchers:** Demonstrates extremely high navigation intensity (Dim.1: +4.24) and interface interaction (Dim.3: +3.42), signifying frequent transitions between test items with extensive usage of UI interface elements.
9	53	55%	**Tool-Interface Dwellers:** Show extremely high interface interaction (Dim.3: 4.26) combined with high tool utilization (Dim.2: 2.34) but very low content interaction (Dim.5: -3.46). They focus heavily on UI elements and problem-solving tools rather than direct content engagement.
10	118	53%	**Interface Avoiders:** Characterized by limited usage of available assessment infrastructure with minimal UI interface engagement (Dim.3: -2.16).
2	44	41%	**Conservative Users:** Show consistently negative scores across all dimensions, particularly low in navigation (Dim.1: -2.43) and tool utilization (Dim.2: -3.07), suggesting minimal interactions.
3	31	13%	**Impulsive Rushers:** Exhibit extremely low deliberate pausing (Dim.4: -4.87) and tool utilization (Dim.2: -2.20). They also show substantial imbalance toward interface adjustment over content interaction (Dim.5: -1.77), spending more time manipulating UI elements than engaging with questions. Overall they show a combination of hurried progression with minimal pausing and poor content focus.

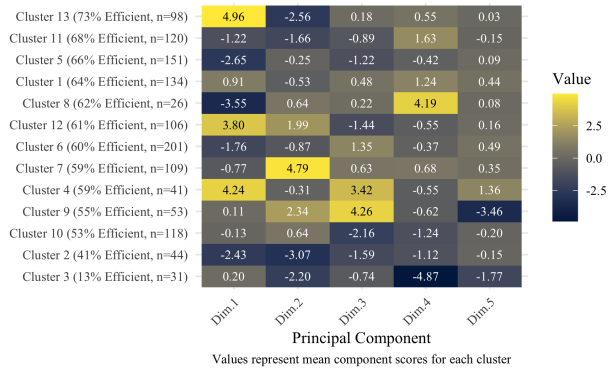


Figure 4: Cluster Profile Across Principal Components (ordered by % of efficient students per cluster)

interface changes without engaging in the actual assessment content, rushing through questions while barely leveraging support tools like the scratchpad and calculator.

Secondly, deliberative pausing (Dim.4) emerges as an important factor for achieving efficiency. Including *Strategic Navigator* (73% efficient students), *Deliberate Pausers* (68% efficient students), *Balanced Reflectors* (64% efficient students), and *Reflective Pausers* (62% efficient students)—Clusters exhibiting above-average deliberate pausing behaviors—all achieved efficiency rates exceeding 62%. Conversely, as aforementioned, *Impulsive Rushers* (which has the lowest percentage of efficient students) showed the most extreme negative value on this dimension (Dim.4: -4.87). This result aligns with Wise's [24] findings that "rapid-guessing behavior"—which test-takers spend very short time responding to multiple-choice items—indicates disengaged test-taking regardless of whether the assessment is low- or high-stake.

Thirdly, when interpreting clusters, the behavioral dimensions would be better to be considered and compared holistically rather than as isolated, deterministic factors. For example, *Strategic Navigators* (Cluster 13) achieved the highest efficiency (73% efficient students) through a distinctive combination of extremely high navigation intensity (Dim.1: 4.96) and minimal problem-solving tool utilization (Dim.2: -2.56), reflecting that they strategically move between questions and do not rely on computational aids such as the calculator or scratchwork features. This suggests that their efficiency may stem not only from their behavioral strategy but potentially also from stronger content knowledge or computational proficiency [1], though such interpretation requires further validation through external achievement data or cognitive measures beyond what current process data alone can provide.

4.2 Supervised Learning

4.2.1 Tree-Based Models

The three tree-based algorithms, all trained on the full set of 43 engineered features, yielded varying performance. As shown in Table 5, XGBoost outperforms both Decision Tree and Random Forest models across accuracy (0.688), sensitivity (0.859), and AUC (0.729). Random Forest achieved marginally better performance than Decision Tree in terms of accuracy (0.659 vs. 0.656) and AUC (0.698 vs. 0.682), though Decision Tree exhibited slightly higher sensitivity (0.815) than Random Forest (0.806). We note that all three of our tree-based models have outperformed the best AUC scores of the winners of the NAEP Data Mining Competition 2019 [19]. Specifically, our best model XGBoost with Adjusted AUC of 0.458 represents a 45.4% improvement over the original competition winner Levin [12] (2021) with Adjusted AUC of 0.315. While these gains are meaningful, an Adjusted AUC of 0.458 reflects only moderate predictive strength, indicating that behavioral process data alone provides useful but limited signal for predicting test-taking efficiency.

Table 5: Model Performance Comparison of Tree-Based Methods

Metric	Decision Tree	Random Forest	XGBoost
Accuracy	0.656	0.659	0.688
Sensitivity	0.815	0.806	0.859
Specificity	0.414	0.432	0.426
AUC	0.682	0.698	0.729
Adjusted AUC	0.364	0.396	0.458

A notable pattern across all three models was the substantial gap between sensitivity and specificity metrics. While sensitivity values ranged from 0.806 to 0.859, specificity values were considerably lower, ranging from 0.414 to 0.432. This indicates that the models were much more successful at identifying efficient test-takers than detecting inefficient ones, which can be partly attributed to the class imbalance in the dataset.

As shown in Figure 5, the feature importance analysis from the best-performing XGBoost model (with highest AUC) demonstrates that median time to first answer is the most influential feature, highlighting that the deliberative thinking process between viewing a question and beginning to answer serves as a critical efficiency marker. Other highly ranked features included navigation time (time spent moving between questions), and total time (overall test duration). Overall, we can see that among the top 15 features, most are related to navigation, hiatus, and essential answer actions, indicating how students move through the test, their rhythm and pacing of interactions, and their engagement with the actual test content contribute most significantly to predicting their efficiency. These findings align with our unsupervised learning results, particularly the identification of dimensions like "Navigation Intensity" and "Deliberate Pausing" in our PCA.

Our decision tree modeling (see Appendix B), which visualizes specific decision paths and identifies key threshold values that distinguish efficient from inefficient test-taking behaviors, complements the results from our XGBoost feature important analysis. Unlike feature importance rankings, which only indicate variables' predictive power, this decision tree reveals specific conditional relationships between variables and their directional impact on efficiency. At the root node, total time serves as the primary splitting criterion, with both lower and higher values potentially leading

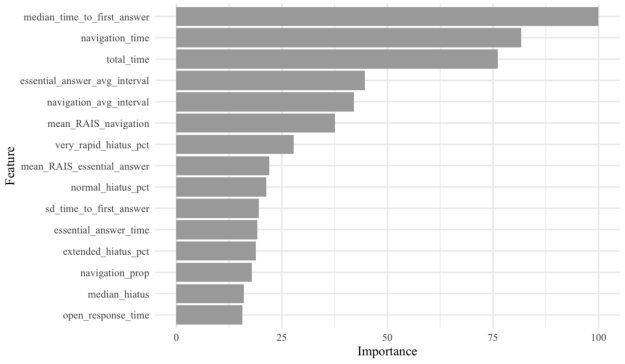


Figure 5: Top 15 Features by Importance in XGBoost

to efficiency depending on subsequent behaviors. This immediately challenges the notion that faster test completion consistently yields greater efficiency.

5. DISCUSSION

In this study, we aimed to understand students' distinctive interaction behaviors with the testing interface and their impact on test-taking efficiency. To achieve this, we applied unsupervised learning (i.e., PCA and GMM) and supervised learning (i.e., tree-based models) to NAEP 8th-grade Mathematics process data. Our findings highlight key behavioral dimensions that affect test taking efficiency and provide insights into optimizing CBT.

PCA revealed five major behavioral dimensions: Navigation Intensity, Problem-Solving Tool Use, UI Interaction, Deliberate Pausing, and Content Interaction Over UI. These dimensions captured distinct engagement patterns among students. GMM clustering further demonstrated the diversity of test-taking strategies, identifying 13 student clusters with varying efficiency levels. Notably, Navigation Intensity and Deliberate Pausing emerged as strong indicators of efficiency, aligning with previous research on test-taking strategies [19]. Our analysis found that disengagement and impulsive rushing—characterized by minimal pausing, high UI interaction, and low content engagement—significantly reduce efficiency. Conversely, high-efficiency clusters demonstrated a balance of navigation, deliberate pausing, and content-focused engagement rather than excessive reliance on UI interactions or problem-solving tools. The tree-based models further corroborated these findings, highlighting *median time to first answer* as the strongest predictor of efficiency. This affirms the significance of deliberative engagement; as such, supporting behaviors like structured pauses and reduced rapid navigation could improve test-taking performance.

While our supervised tree-based models suggest that features related to UI interaction and tool usage do not have important predictive powers (i.e., we may not able to label the students' efficiency based on the UI interaction), our unsupervised learning discloses that these features distinguish clusters with higher percentages of efficient students from those without. The 60 percentage point efficiency gap between our highest and lowest performing clusters (Strategic Navigators at 73% vs. Impulsive Rushers at 13%) suggests that interface design choices can substantially impact students' test-taking behavior and which can eventually hinder the interpretation of the test score. From this, we can interpret that reducing the extraneous cognitive load from UI complexity is still a crucial consideration for optimizing interface designs [8, 18, 17].

These findings have broader implications beyond traditional NAEP mathematics assessments. To enhance engagement and efficiency while taking the test, interfaces should minimize unnecessary UI elements that may distract students from content engagement [18], preventing "mouse skills" from masking their actual "math skills." Additionally, adaptive testing interfaces that recognize and respond to inefficient navigation patterns could offer real-time support to students struggling with disengagement or impulsive rushing. Assessment designers should also consider how structured pauses and deliberate engagement can be facilitated to help enhance students' test-taking efficiency [2]. While our analysis is limited to the context of computer-based testing on NAEP, future validation across different computer-based tests across different subject areas would strengthen the practical applicability of our taxonomy.

5.1 Limitations and Future Work

This study has several limitations that should be considered when interpreting the findings. First, one major limitation is the lack of demographic data. As noted in the data description, the dataset does not include information on socio-economic status (SES), gender, or prior digital exposure, which may affect students' interaction behaviors and efficiency in CBT. However, there is an opportunity to obtain these variables by contacting the NAEP through a formal request. Future research could explore these factors further if access to such data is granted. For instance, the role of SES in digital literacy and assessment performance warrants further investigation. The lack of digital literacy skills can exacerbate existing educational inequalities, disproportionately affecting students from lower socio-economic backgrounds who may have limited access to technology and digital learning resources.

Second, as another crucial limitation, the dataset does not include item-level accuracy information (i.e., information about whether students answered correctly or incorrectly on each item), and the efficiency metric itself is inherited from the competition design [20]. This raises concerns about construct validity [15]. For instance, high-performing students who solve problems quickly and correctly may be systematically labeled as inefficient, while students who spend time on problems without productive engagement may be labeled as efficient. Future research should integrate additional data sources, such as item-level accuracy and cognitive measures, to both develop efficiency measures that jointly consider time and correctness, and to further validate the behavioral cluster interpretations identified in this study.

Third, in terms of analytic method, this study solely focused on individual student behaviors, without considering item types, which may significantly impact interaction patterns. For example, actions such as "Lose Focus" or "Receive Focus" occur exclusively in "Fill in the Blank" item format, yet our analysis did not account for such item-specific behaviors.

Future studies with access to a larger and more diverse set of item types should investigate how students' behaviors vary by question format.

We hope, addressing these limitations will help advance research on computer-based testing and student interaction behaviors. Expanding datasets to include demographics, diverse item types, and performance-based efficiency metrics will allow for a more comprehensive analysis. Additionally, refining predictive models by incorporating more behavioral and cognitive factors can improve our understanding of effective computer-based test-taking strategies and inform adaptive assessment designs that better support student learning.

6. REFERENCES

[1] S. Araneda, D. Lee, J. Lewis, S. G. Sireci, J. A. Moon, B. Lehman, B. Arslan, and M. Keehner. Exploring relationships among test takers' behaviors and performance using response process data. *Education Sciences*, 12(2):104, 2022.

[2] R. Bachhel and R. G. Thaman. Effective use of pause procedure to enhance student engagement and learning. *Journal of Clinical and Diagnostic Research*, 8(8):XM01, 2014.

[3] R. S. Baker, N. T. Heffernan, T. Patikorn, C. M. Forsyth, and I. R. Katz. Editorial acknowledgments and introduction to the special issue for the NAEP data mining competition. *Journal of Educational Data Mining*, 13(2):i–i, 2021.

[4] A. E. Beaton, R. L. Linn, and G. W. Bohrnstedt. Alternative approaches to setting performance standards for the national assessment of educational progress (NAEP). Technical report, American Psychological Association, 2012.

[5] Y. Bergner and A. A. Von Davier. Process data in NAEP: Past, present, and future. *Journal of Educational and Behavioral Statistics*, 44(6):706–732, 2019.

[6] N. Bosch. AutoML feature engineering for student modeling yields high accuracy, but limited interpretability. *Journal of Educational Data Mining*, 13(2):55–79, 2021.

[7] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[8] A. Darejeh, N. Marcusa, G. Mohammadi, and J. Sweller. A critical analysis of cognitive load measurement methods for evaluating the usability of different types of interfaces: guidelines and framework for Human-Computer Interaction. *arXiv preprint arXiv:2402.11820*, 2024.

[9] A. El-Hashash. Role of digital formative assessment in improving the assessment and monitoring of students' learning and their significance during the COVID-19 pandemic. *Open Journal of Educational Research*, 2(1):9–12, 2022.

[10] T. M. Haladyna and S. M. Downing. Construct-irrelevant variance in high-stakes testing. *Educational Measurement: Issues and Practice*, 23(1):17–27, 2004.

[11] I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A*, 374(2065), 2016.

[12] N. A. Levin. Process mining combined with expert feature engineering to predict efficient use of time on high-stakes assessments. *Journal of Educational Data Mining*, 13(2):1–15, 2021.

[13] F. Li, L. Cheng, X. Wang, L. Shen, Y. Ma, and A. Y. M. A. Islam. The causal relationship between digital literacy and students' academic achievement: a meta-analysis. *Humanities and Social Sciences Communications*, 12(1):108, 2025.

[14] M. A. Lindner and S. Greiff. Process data in computer-based assessment: Challenges and opportunities in opening the black box. *European Journal of Psychological Assessment*, 39(4):241–251, 2023.

[15] S. Messick. Validity of psychological assessment: Validation of inferences from persons' responses and performances as scientific inquiry into score meaning. *American Psychologist*, 50(9):741, 1995.

[16] National Center for Education Statistics. NAEP process data. National Assessment of Educational Progress, U.S. Department of Education, n.d. Accessed: May 5, 2026.

[17] S. Oviatt. Human-centered design meets cognitive load theory: designing interfaces that help people think. In *Proceedings of the 14th ACM international conference on Multimedia*, pages 871–880, 2006.

[18] S. Oviatt, A. Arthur, and J. Cohen. Quiet interfaces that help students think. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pages 191–200, 2006.

[19] N. Patel, A. Sharma, T. Shah, and D. Lomas. Modeling NAEP test-taking behavior using educational process analysis. *Journal of Educational Data Mining*, 13(2):16–54, 2021.

[20] T. Patikorn and N. Heffernan. NAEP data mining competition 2019, 2019.

[21] L. Scrucca, M. Fop, B. T. Murphy, and A. E. Raftery. mclust 5: Clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):289, 2016.

[22] A. Skulmowski. Guidelines for choosing cognitive load measures in perceptually rich environments. *Mind, Brain, and Education*, 17(1):20–28, 2023.

[23] J.-F. Wang, T.-H. Wang, and C.-H. Huang. Investigating students' answering behaviors in a computer-based mathematics algebra test: A cognitive-load perspective. *Behavioral Sciences*, 12(8):293, 2022.

[24] S. L. Wise. Rapid-guessing behavior: Its identification, interpretation, and implications. *Educational Measurement: Issues and Practice*, 36(4):52–61, 2017.

[25] F. Zehner, B. Eichmann, T. Deribo, S. Harrison, D. Bengs, N. Andersen, and C. Hahnel. Applying psychometric modeling to aid feature engineering in predictive log-data analytics: The NAEP EDM competition. *Journal of Educational Data Mining*, 13(2):80–107, 2021.

APPENDIX
A. DATASET VARIABLES

Table A: Variables in Publicly Accessible Raw Datasets

Variable Name	Description	Values/Categories	Sample Size
Student ID	Unique identifier for each student	Numeric ID	N = 1232
Block	Test block identifier	Block A, Block B	
Accession Number	A unique identification of a problem/item	Item IDs (e.g., "VH356862")	N = 24
Item Type	The type of the item	MCSS (Multiple-choice single-selection question), GridMS (Grid-style multiple-selection question), MatchMS (Match multiple-selection question), ZonesMS (Multiple selection of pre-defined zones), FillInBlank (Single fill in the blank question), MultipleFillInBlank (Multiple fill in the blank question), CompositeCR (Composite constructed response question (e.g., open-ended)), BQMCSS (Background/Survey MCSS question)	N = 8
Observable	Action taken by student	Enter Item, Next, Exit Item, Click Choice, Open Calculator, Move Calculator, Calculator Buffer, DropChoice, Eliminate Choice, Vertical Item Scroll, Receive Focus, Math Keypress, Lose Focus, Close Calculator, Click Progress Navigator, First Text Change, Yes, Leave Section, Scratchwork Mode On, Draw, Scratchwork Mode Off, Scratchwork Erase Mode On, Erase, Scratchwork Draw Mode On, Scratchwork Highlight Mode On, Highlight, Clear Answer, Change Theme, TextToSpeech, Open Equation Editor, Equation Editor Button, Clear Scratchwork, OK, Close Equation Editor, Back, Increase Zoom, Decrease Zoom, Hide Timer, Show Timer, Last Text Change, No, Horizontal Item Scroll	N = 42
Extended Info	Additional metadata about the action	Various metadata strings. Metadata of the student action in a JSON format e.g. how much did the student scroll, what key she pressed on the calculator, what digit she typed as a response, etc.	Average 356 actions per student in the training set and 117 actions in the test set
Event Time	The timestamp of when the action was taken	Datetime format (e.g., "2017-02-10 14:44:54.167")	
Efficiency	Outcome variable indicating efficient completion of Block B*	TRUE/FALSE**	

* All predictor variables of both training and test sets of the data are from Block A interactions, while the outcome variable (i.e., EfficientlyCompletedBlockB) is derived from Block B performance.
** TRUE means "efficient" and FALSE means "inefficient". The efficiency is defined based on the usage of the time as 1) being able to complete all problems in Block B, and 2) being able to allocate a reasonable amount of time to solve each problem. Detailed information can be found here.

B. DECISION TREE

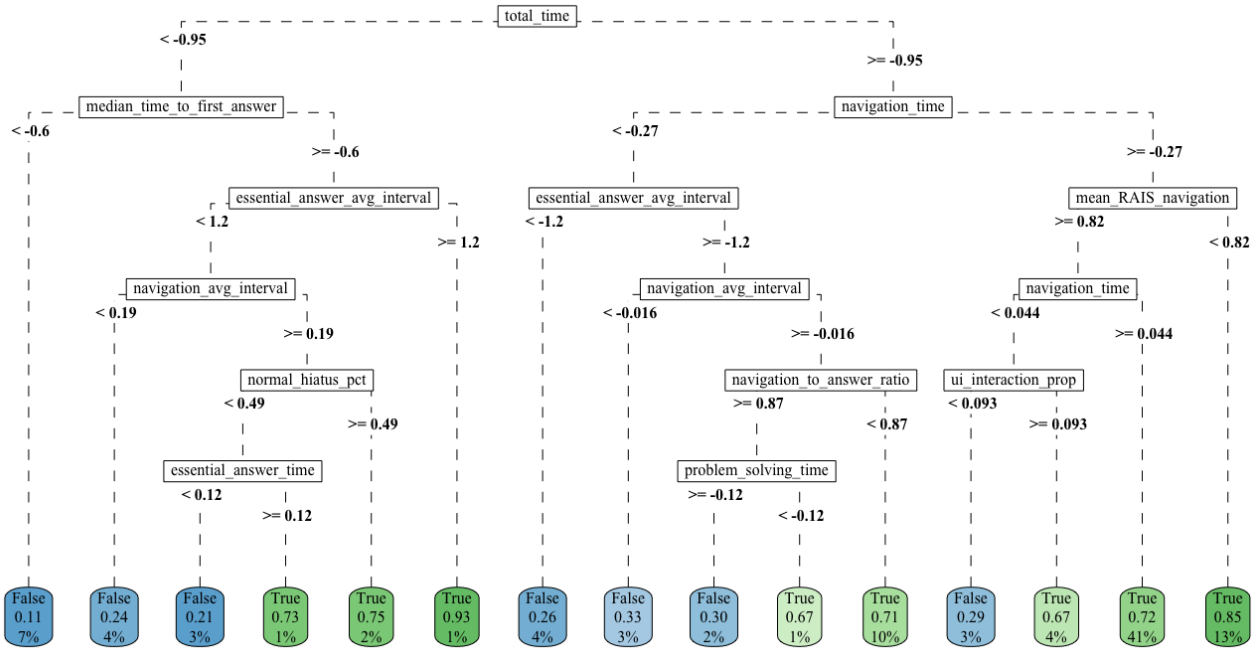


Figure 6: Decision Tree for Efficiency (n = 1232)

Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory

Amit Paikrao
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India
amitpaikrao@iitb.ac.in

Vishwas Badhe
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

Indrayani Nishane
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

Jyoti Shaha
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

Ramkumar Rajendran
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

Ritayan Mitra
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

ABSTRACT

Modern electrical grid systems demand engineers capable of adaptive reasoning, systematic justification, and principled decision-making under uncertainty. However, while many graduates excel technically, they often struggle with these expert-level competencies, revealing gaps in our understanding of expertise development. To address this, we need to identify the granular cognitive and epistemic shifts that occur along the expertise continuum. Drawing on Epistemic Frame Theory, this study examines how epistemic frames shape the developmental trajectory of expertise across second-year, third-year, and final-year undergraduates, compared against field professionals (experts). In the study, participants completed an open-ended electrical grid design task in a computer-based environment. We applied Min-Max normalization to verbal protocols to neutralize verbosity bias and reveal latent structural shifts. Our findings highlight a two fundamental developmental trajectories. An ascending trajectory captures the adoption of a professional lens, wherein students transition from academic idealism to professional pragmatism and from reactive debugging to proactive modeling. Conversely, a descending trajectory reflects the shedding of novice habits, indicating a gradual disengagement from surface-level and trial-and-error strategies. Notably, experts' strategic silence emerged as a marker of proficiency. We observed that experts' silence is a signal of proficiency. The high-frequency behaviors in novices, such as visual monitoring and explicit naming of components, fade away as knowledge becomes tacit. These findings suggest that expertise is defined as much by what is shed (novice habits) as what is acquired (professional lens). These insights suggest that engineering curricula should incorporate earlier authentic practice to bridge the theoretical trap fostering the transition from explicit theoretical rules to implicit professional norms.

Keywords

Epistemic Frame Theory, Trajectory of Expertise, Continuum of Expertise, Min-Max Normalization, Engineering Education

1. INTRODUCTION

A purpose of formal education is to enable students to build domain-specific knowledge, ways of thinking, and professional practices that are essential for successful learning and to develop expertise in it [24, 28, 19]. The most important part is that expertise is not a general body of knowledge; it is grounded in numerous context-specific experiences. Expert knowledge is typically deeply tied to particular domains and develops through sustained participation in authentic practice [18]. The expertise acquired in a particular domain shapes professional expertise in that domain. [15]. Expertise is developed through progressive stages, with each stage characterized by qualitative transformations in how individuals perceive situations and make decisions [13, 20, 4, 27] [18].

The modeling of expertise through progressive stages is extensively discussed in foundational research, most notably in *the Dreyfus model of skill acquisition* [13], *the Benner mode* [4], *the Two expertise model* proposed by Hatano and Inagaki [27, 7]. *The skill acquisition model* proposed by Dreyfus and Dreyfus [13] is stage-based model, it conceptualize expertise as a progression through qualitatively distinct stages of novice, advanced beginner, competent, proficient and expert. This model emphasize changes in rule-following, intuition, and situational perception as individuals accumulate skills. Such models have been influential across professional domains. Benner [3] adapted Dreyfus' five-stage model of

Amit Paikrao, Vishwas Badhe, Indrayani Nishane, Jyoti Shaha, Ramkumar Rajendran, and Ritayan Mitra. Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 193–204. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039873

skill acquisition to nursing domain. Hatano and Inagaki [27], and later Carbonell in 2019 [7], differentiate between *routine experts* and *adaptive experts*. The routine experts are ones who perform efficiently in familiar situations and Adaptive Experts are who can transfer knowledge and creatively solve novel problems. Adaptive expertise builds on routine expertise, is shaped by task variety, and reflects a dynamic rather than static view of knowledge. Alongside the stage-based models of expertise, there are *cognitive theories* for problem solving strategies [9, 2, 34], *Mental models and situation awareness* that explain experts' superior anticipation of system behavior [22, 16] and *knowledge-based model* [15, 16, 17]. These existing models are predominantly cognitive, focusing on technical skills, social participation, and situational experiences.

Recent studies apply distinct theoretical lenses to understand the trajectory of professional development. Lyon [36] utilizes the Dreyfus model to structure the linear stages of teaching development in dental education. Dehn et al.[11] analyze the guided Self-Determination model, revealing the complex, multidimensional factors that drive skill acquisition beyond simple proficiency in nursing education. Furthermore, Kua et al.[32] examine Hatano's concept of adaptive expertise to highlight the necessity of cognitive flexibility in educational settings; they are specifically examining the epistemology. Although these models of expertise are useful for characterizing differences between novices and experts, they offer limited guidance for designing targeted interventions as they lack empirical evidence about when and how key transformations in expertise actually occur. Moreover, beyond cognitive dimensions such as skills and knowledge, existing models provide little clarity about which additional dimensions (e.g., identity, values, or professional judgment) contribute to more fine-grained developmental progressions.

This theoretical limitation is further compounded by a measurement challenge. As individuals develop expertise, what they choose to verbalize changes systematically. Empirical expert–novice studies show that experts often articulate fewer foundational concepts than novices because much of their domain knowledge becomes automated and tacit [1, 9, 34]. Consequently, traditional frequency-based analysis methods risk misinterpreting reduced verbalization as disengagement or conceptual weakness rather than recognizing it as a potential marker of proficiency. Addressing this issue requires both a multidimensional theoretical framework and analytical methods capable of this verbosity confound.

In this paper, we propose a multi-model expertise development, operationalized through Epistemic Frame Theory (EFT) [44, 42]. EFT conceptualizes a network of interconnected components, where individuals' engagement and practice within a community shape the development and organization of epistemic frames (Shaffer, 2006a). Epistemic frame theory suggests that every profession has unique collections of *Skills* (practical actions professionals perform), *Knowledge* (conceptual understanding professionals possess), *Values* (what are the believes of professionals and what they consider important), *Identity* (how professionals see themselves and their role), and *Epistemology* (how professionals justify decisions and determine valid reasoning) that construct an epistemic frame.

We move beyond simple usage counts to uncover the changing meaning of these epistemic frames. By applying Min-Max normalization to verbal protocols across four cohorts (second year, third year, final year, and a field professional), we neutralize verbosity bias to reveal two complementary developmental trajectories:

- Ascending Trajectories: The adoption of the professional lens
- Descending Trajectories: The shedding of novice habits

This developmental map provides a benchmark for designing stage-appropriate educational interventions. It identifies not only which expert-like behaviors emerge and should be reinforced through scaffolding, but also which persistent novice habits require explicit pedagogical attention to accelerate their decline. By revealing both what students must develop and what they must learn to abandon, this work enables educators to provide targeted support at the precise developmental moments when students need it most.

Understanding what students must develop alongside what they must shed enables educators to provide precisely timed support at critical developmental transitions. To empirically establish these ascending and descending trajectories, this study addresses two primary research questions:

1. What distinct developmental patterns emerge across the novice-to-expert continuum when epistemic codes are classified by their trajectory slopes?
2. How does the qualitative application of epistemic frames transform from second-year undergraduate students to practicing field professionals?

2. RELATED WORK

To understand how expertise develops, we must first understand what distinguishes experts from novices. Extensive research across multiple domains has documented systematic cognitive differences between expert and novice problem-solvers. Experts possess highly interconnected knowledge structures that support efficient chunking, reduce cognitive load. They can enable flexible, principle-based problem solving through forward reasoning and multiple solution pathways, along with extended reflection and evaluation [8, 9, 34, 50, 10, 39]. In contrast, novices rely on fragmented and superficial representations, they attend primarily to explicit problem features and known-unknowns rather than underlying principles, which limits transfer and strategy diversity [8, 9, 49, 40]. Novices predominantly employ means-end or backward inference strategies, resulting in trial-and-error approaches, minimal evaluation, and susceptibility to irrelevant information [34, 41, 26, 39, 38].

These expert-novice differences extend beyond problem solving. In categorization and representation tasks, experts emphasize deep structural relations while novices focus on surface or aesthetic features [25, 48, 31, 12]. This reflects the

role of tacit, experience-based knowledge in expert performance—knowledge that novices have not yet developed.

However, this cognitive divide is not a static binary state. Research across domains demonstrates that the sharp distinctions between expert and novice represent endpoints of a developmental continuum rather than discrete categories. Expertise emerges through gradual, qualitative transformation. The Dreyfus model captures this progression well, describing a move from rigid, rule-based reasoning (Novice) to context-sensitive, situational understanding (Expert) [13]. Similar developmental stages characteristics have been noted in medicine, where expertise emerges through iterative refinement of pattern recognition [4]. Yet recognizing that expertise develops along a continuum raises a fundamental question: what, precisely, is developing?

Expertise is more than cognitive transformation; rather, it is fundamentally multidimensional. Epistemic Frame Theory proposes that professional expertise emerges through the integration of five interconnected dimensions: Skills, Knowledge, Values, Identity, and Epistemology (SKIVE) [42, 45]. These dimensions do not develop in isolation, but they integrate into a cohesive professional lens over time. For an engineer, a design decision is never purely technical. It represents an interplay between technical knowledge, identity as a problem-solver, and values regarding system reliability and societal impact [42]. This EFT-grounded perspective fundamentally reshapes how we must study expertise development: rather than tracking a single cognitive trajectory, we must examine how all five SKIVE dimensions evolve simultaneously—and potentially at different rates.

Understanding this multidimensional development has important practical implications. This multidimensional view helps explain why engineering graduates struggle in professional practice despite strong technical performance in coursework [46, 51]. The transition from student to professional requires transformation not only in knowledge and skills, but also in values, identity, and epistemology. Without all dimensions developing in coordination, individuals may possess technical competence yet lack the integrated professional judgment required for effective practice. Understanding how this multidimensional development unfolds raises critical questions.

How do these dimensions develop across the undergraduate journey? When do specific transformations occur? Do all dimensions develop at the same rate, or do some lag behind? Most critically, does expertise follow only an upward trajectory of acquisition, or does it also require the active shedding of novice behaviors? Answering these questions requires methodological approaches that can capture the activation of knowledge, values, and identity as students engage with authentic professional tasks.

Think-aloud protocols (TAPs) offer one such approach. TAPs have long served as the gold standard for examining cognitive processes in expert-novice research [**?**]. In engineering education, these protocols provide a rich verbal trace of how students at different levels navigate design tasks, identify hidden constraints, and justify their actions through professional norms. While traditional observation only records what a student do, think-aloud protocols reveal the why exposing the activation of specific epistemic frame elements during problem-solving. When enhanced by eye-tracking data (ET-RTA) [37], these protocols provide even deeper insight into the cognitive processes underlying professional reasoning.

However, possessing a methodological tool does not guarantee a complete developmental picture. Existing research has documented endpoints of the expertise continuum such as novice versus expert characteristics, but provides limited insight into the developmental stages between them. We lack detailed understanding of the specific transformations that occur at each year of undergraduate training. We do not know which dimensions develop early and which lag behind. Crucially, we do not know whether expertise development is purely acquisitional or whether it also involves the strategic abandonment of novice strategies. Without this granular developmental map showing both what students must gain and what they must shed, educators cannot design stage-appropriate interventions or provide targeted scaffolding at the precise moments students need support.

This study addresses this gap by revealing the bidirectional nature of expertise development across the undergraduate-to-professional continuum. Through a cross-sectional design comparing second-year, third-year, and final-year students with practicing professionals on an authentic engineering task, we created a detailed benchmark of developmental characteristics at each stage. This benchmark reveals not only what grows but also what must be shed—providing the complete picture needed to design effective, stage-appropriate educational interventions.

3. METHODOLOGY

3.1 Study Design and Participants

This study employs a cross-sectional research design to capture the developmental trajectory of expertise from the undergraduate to professional practice. The study was conducted with 15 electrical engineering undergraduate students categorized by their academic level: Second-Year (SE, n=6), Third-Year (TE, n=5), Final-Year (BE, n=4), and an industrial expert (Exp, n=1). The industrial expert has been working in the power system grid industry for more than 15 years. This distribution enables a granular comparison of epistemic frame shifts across key curricular milestones. All procedures were approved by the Institutional Review Board (IRB), and informed consent was obtained from all participants for the use of their behavioral and verbal data.

3.2 The "Power to the People" game

Present research utilized "Power to the People," (developed by (https://www.rhombicogames.com), a high-fidelity power system role-playing game. This game requires participants to manage complex grid operations like optimization, load balance, mitigating faults, etc., as load demand of city expand dynamically. This environment provides a rich context for process mining, as it forces participants to move beyond theoretical knowledge to professional application.

3.3 Data Collection Method

Data collection was structured into three distinct phases designed to ensure data validity and minimize the novelty effect of the game interface:

Orientation and Familiarization (30-45 mins): To ensure that subsequent data captured cognitive processes rather than interface-learning hurdles, participants underwent a standardized demo and checklist-driven practice. They practiced installing substations and managing generators in a neutral simulation environment ("Siem Reap, Cambodia").

Gameplay and Gaze Capture (30-40 mins): Participants transitioned to the experimental task centered on the "Agra, India" city layout. During this phase, a Tobii Pro Nano (60 Hz) eye tracker was used to record eye-gaze data. This eye-tracking layer serves as a digital trace of the participant's visual attention, providing a non-intrusive log of their interaction with the grid.

Eye-Gaze Enhanced Retrospective Think-Aloud (ET-RTA): Post-gameplay, we conducted structured interviews (average duration 70-80 minutes) using the ET-RTA protocol. Participants viewed their gameplay video overlaid with their gaze plots, which served as a cued-recall stimulus. This method allows the extraction of rich process level verbal protocols grounded in actual behavioral traces, reducing the risk of post-hoc rationalization [37].

3.4 Data Transformation and Coding

To bridge the gap between qualitative learning sciences and quantitative data mining, our approach treats unstructured verbal data as a source for computational modeling. This study employs a data processing pipeline that transforms coded verbalizations into normalized feature vectors. By calculating the slope of these vectors across academic cohorts, we establish a mathematical framework for mining the trajectories of expertise development. This methodology aligns with the EDM community's focus on modeling complex learner behavior through reproducible, quantitative metrics.

The recorded ET-RTA interviews were transcribed and analyzed using a hybrid coding approach. We employed the EFT framework deductively to categorize broad epistemic codes, while inductively deriving sub-codes like $S.Monitoring$, and $E.Real.Life$ etc. from the verbal data. The detailed codebook is available **here**. The reliability of coding was assessed using Cohen's kappa (κ) as a measure of inter-rater reliability. In the first round, two independent educational technology researchers coded 30% of the interview responses. This was followed by a second round of discussion to resolve disagreements and establish consensus. All codes demonstrated a high level of agreement between coders ($\kappa = 0.80$). This value indicates substantial inter-rater agreement [33] suggesting that the coding procedures were reliable across constructs.

The resulting dataset was normalized by calculating the relative frequency of codes per cohort (see section 4), enabling a quantitative analysis of the expertise trajectory, shifting the focus from qualitative description to data-driven pattern recognition.

4. ANALYSIS

To investigate the developmental trajectory of expertise, this study employs a cross-sectional quantitative design. By comparing four cohorts: second-year, third-year, and final-year undergraduate students, alongside practicing professional, we construct a developmental benchmarks representing distinct stages of professional growth. This approach allows us to identify patterns of expertise development across the undergraduate-to-professional continuum.

The challenge in analyzing the cross-sectional datasets is that code frequencies often exhibit extreme variance. Verbal protocols often generate massive amounts of unstructured data that are difficult to compare across different levels of expertise. A significant hurdle we encountered in this process was the natural variation in participant verbosity; for instance, a talkative novice might produce a higher raw frequency of codes than a concise expert, potentially skewing the developmental signal. In raw frequency analysis, the visual prominence of a code is strictly determined by its cumulative count [47]. This creates a visual flattening effect for lower-frequency codes. For example, if a skill code $.Troubleshooting$ appears 50 times and an epistemology code $E.Design$ appears 5 times, the graphs will be dominated by the skill code, and the visuals will be very flat for the epistemology code. In a standard line chart, the $S.Troubleshooting$ line will oscillate high on the Y-axis, whereas the $E.Design$ line, even if it triples from 2 to 6 mentions (a 200% growth), will appear as a nearly flat line at the bottom of the chart. One might visually conclude that nothing happened with the epistemology code, simply because its geometric slope is dwarfed by the magnitude of the skill code.

4.1 Two-Step Normalization Procedure

To enable fair comparison across cohorts despite both unequal sample sizes and scale differences, we employed a two-step normalization procedure.

Step 1: Group Averaging: To neutralize the influence of cohort size (SE=6, TE=5, BE=4, Exp=1), we first calculated group-averaged frequencies by $\bar{X}_{SE} = \sum X_{SE}/6$, $\bar{X}_{TE} = \sum X_{TE}/5$, $\bar{X}_{BE} = \sum X_{BE}/4$, and $\bar{X}_{Exp} = \sum X_{Exp}/1$. This yields a single representative value per code per stage.

Step 2: Min-Max Normalization: Even after group averaging, codes exhibit extreme variance in absolute magnitude (e.g., high-frequency $S.Troubleshooting$ vs. low-frequency $E.Design$.To address this, we applied standardization techniques such as min-max normalization to systematically transform these narratives into a quantifiable dataset. Min-max normalization as shown in equation 1 [21] standardizes the dataset, ensuring that all features contribute proportionately to the final result. By placing all data points on a common scale, we could exclude the impact of outliers on their broader conclusions [5, 47].

Let $C_{i,t}$ denote the averaged raw frequencies of code i at developmental stage t, where $t \in \{\text{SE}, \text{TE}, \text{BE}, \text{Exp}\}$. The normalized weight $W_{i,t}$ is calculated as:

$$W_{i,t} = \frac{C_{i,t} - \min(C_i)}{\max(C_i) - \min(C_i)} \quad (1)$$

where $\min(C_i)$ and $\max(C_i)$ represent the minimum and maximum frequencies of code i observed across the four temporal stages.

This transformation maps each code's emphasis to the unit interval $[0, 1]$. A value of 1 indicates the stage of maximum relative emphasis, while 0 indicates the minimum. By scaling the raw frequencies into a standardized range $[0, 1]$, we were able to conduct a fair comparison of all the code elements across the second-year (SE), third-year (TE), final-year (BE), and expert groups, regardless of their total word count. This mathematical adjustment was central to our study, as it enabled us to visualize and quantify the movement of expertise. Specifically, it allowed us to model the expertise continuum as a series of developmental trajectories. The normalized values of all the codes helped us to plot the trajectories of all the codes as shown in Figure 1.

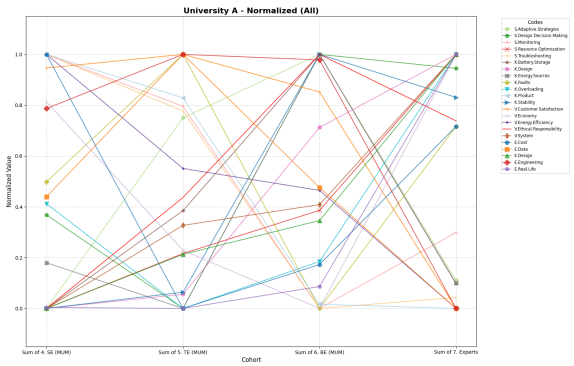


Figure 1: Developmental Trajectories of epistemic codes using min-max normalization

4.2 Slope

Figure 1 shows the trajectories of all the codes across cohorts, but to quantify the directional magnitude of each code, we derived a slope for each trajectory. This slope represents the mean rate of shift from the novice state (SE) to the expert state (Exp). Given that the timeline consists of four distinct stages (SE, TE, BE, Exp), the slope is defined as the total normalized displacement averaged over the temporal span:

$$\lambda_i = \frac{W_{i,\text{Exp}} - W_{i,\text{SE}}}{3} \quad (2)$$

The resulting λ_i values range from -0.33 (a complete shift from maximum usage at SE to minimum usage at Expert) to $+0.33$ (the inverse). The overall range of the λ_i values is therefore 0.66.

4.3 Statistical Thresholding and Classification

To categorize the diverse trajectories into meaningful epistemic patterns, we established a slope-based threshold τ to distinguish substantive developmental trends from transitional volatility.

An analysis of the distribution of slopes across the combined dataset for all $N = 44$ across 22 unique codes revealed a mean of $\mu = 0.02$ and a standard deviation of $sd = 0.24$. To ensure robustness in identifying distinct developmental trajectories, we selected a threshold of $\tau = \pm 0.15$, derived from the data distribution and focused on quartile alignment (see Figure 2). Specifically, this threshold approximates the absolute boundaries of the interquartile range (IQR = 0.42, spanning Q1 ≈ -0.17 to Q3 $\approx +0.25$): the negative threshold (-0.15) is close to Q1 (capturing the lower quartile's tail for negative shifts), while the positive threshold ($+0.15$) falls between the median (Q2 ≈ 0.02) and Q3 (highlighting the upper quartile for positive shifts). This approach effectively filters out central noise within the IQR, ensuring the classification prioritizes the most pronounced shifts in the outer quartiles while accounting for the distribution's slight positive skew.

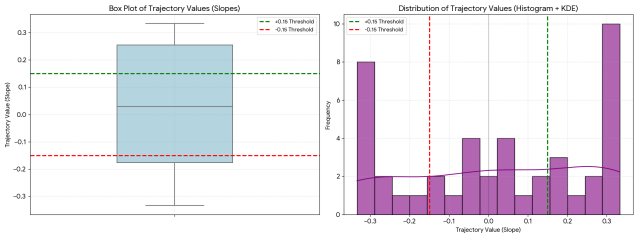


Figure 2: Statistical distribution of trajectory slopes (λ) for all codes. (Left) Box plot displaying the Interquartile Range (IQR) with Q1 ≈ -0.17 and Q3 ≈ 0.24. The dashed lines at ± 0.15 align closely with these quartiles, justifying their use as discriminatory thresholds. (Right) Histogram. The red (-0.15) and green ($+0.15$) thresholds clearly separate the central "transitional" bell curve from the significant negative and positive tails (Major Decline and Major Growth).

Using $\tau = 0.15$, we categorized the epistemic trajectories into four distinct types as shown in Table 1.

Table 1: Threshold Value Range for Categorizing Trajectory Types

Trajectory Type	Mathematical Definition
Type 1: Major Growth	$\lambda_i > +0.15$
Type 2: Minor Growth	$0 < \lambda_i \leq +0.15$
Type 3: Minor Decline	$-0.15 \leq \lambda_i < 0$
Type 4: Major Decline	$\lambda_i < -0.15$

4.4 Integration of Qualitative Protocol

To interpret the trajectories derived from the slope-based classification, we employed a qualitative approach. While the quantitative metrics (slope λ) identify which behaviors change, they do not explain how the nature of that behavior transforms. Therefore, we aligned the normalized code weights (W_i) with their corresponding verbal protocol segments. For each significant trajectory, we conducted a the-

matic analysis of the excerpts to extract the semantic and cognitive shifts driving the statistical trend, ensuring that every quantitative claim is grounded in participant evidence.

5. RESULTS

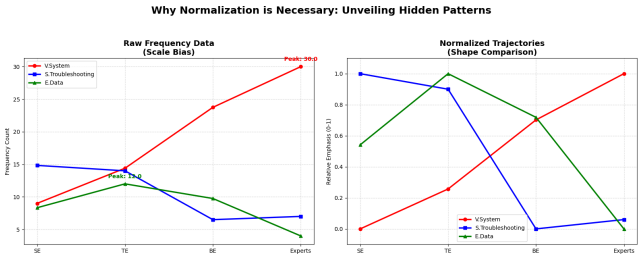


Figure 3: Comparative plots of raw frequency (left) and normalized trajectories (right) for selected codes: V.System, S.Troubleshooting, and E.Data.

To illustrate how normalization helps visualize epistemic trajectories, we compare the raw and normalized distributions of three representative codes: *V.System* (Values: System Stability), *S.Troubleshooting* (Skills: Troubleshooting), and *E.Data* (Epistemology: Data-driven Justification) (Figure 3). The raw frequency graph (left) exhibits a significant scale bias, where high-volume codes like *V.System* (RED) peaking at ≈ 30, exerts a magnitude bias that suppresses lower-frequency codes like *E.Data* (GREEN). *E.Data* is less frequent peaking at ≈ 12 and appears flat across the four cohorts. If we have to make the inference from this, the development across the 4 cohorts will be stagnant when viewed on a shared raw axis with *V.System*.

In contrast, the min-max normalized graph (right) performs a crucial analytical equalization. By scaling each trajectory independently to a standardized $[0, 1]$ range, we eliminate verbosity-induced noise and reveal the relative intensity of each epistemic element within its own developmental lifecycle. This transformation allows the trajectories' distinct shapes to emerge as the primary visual feature. We can now observe, for instance, that while *S.Troubleshooting* (BLUE) might have a high raw volume in the sophomore year, its relative importance to the expert's frame is negligible compared to the sharp uptrend of *E.Data*. Without this normalization, our analysis would be trapped in a frequency bias, leading to erroneous conclusions. Such as certain professional traits are absent simply because they are spoken with more precision and fewer words. Normalization thus serves as a lens that brings the structure of the epistemic frame into focus, allowing us to quantify the movement of expertise across the continuum with mathematical parity.

5.1 RQ1

What distinct developmental patterns emerge across the novice-to-expert continuum when epistemic codes are classified by their trajectory slopes?

Following the application of min-max normalization, we performed a slope-based classification to categorize the developmental trends across 22 codes. Utilizing the thresholds defined in Table 1, four major trajectories of the movement of epistemic frames across the four cohorts emerged. These trajectories are linear (Type 1: Major Growth; Type 4: Major Decline) and non-linear (Type 2: Minor Growth; Type 3: Minor Decline) (Figure 4.)

The sequence of normalized weights W_i corresponding to each trajectory is presented in Table 2.

Table 2: Sequence of the SE, TE, BE, Exp weights in four trajectories

Trajectory Type	Sequence of normalized weights	Shape of the Trajectory
Type 1: Major-Growth	$W_{\text{SE}} < W_{\text{TE}} < W_{\text{BE}} < W_{\text{Expert}}$	Uptrend
Type 2: Minor-Growth	$W_{\text{SE}} < W_{\text{TE}} \leq W_{\text{BE}} > W_{\text{Expert}}$	Hump
Type 3: Minor-Decline	$W_{\text{SE}} > W_{\text{TE}} \geq W_{\text{BE}} < W_{\text{Expert}}$	Dip
Type 4: Major-Decline	$W_{\text{SE}} > W_{\text{TE}} > W_{\text{BE}} > W_{\text{Expert}}$	Downtrend

In this paper, we focus our analysis exclusively on the two most significant graphs: *Type 1 (Major Growth)* and *Type 4 (Major Decline)*. Type 1 (Major Growth: Uptrend): These trajectories represent "expert-like" competencies that show a consistent increase in relative intensity as students move toward professional practice. Type 4 (major Decline: Downtrend): These trajectories represent "novice-heavy" habits that are systematically abandoned or automated as expertise develops.

We have intentionally excluded Type 2 and Type 3 from the current discussion. These categories exhibit high intra-group volatility, characterized by non-linear fluctuations such as "humps" (peaks at TE or BE) and "dips" (Dips at TE or BE). In the context of trajectory modeling, these volatile patterns often contain a high degree of statistical noise or represent complex, intermediate cognitive restructuring that requires a more granular longitudinal analysis. Therefore, the detailed examination of these fluctuating patterns is beyond the scope of this paper.

5.2 RQ2

How does the qualitative nature of epistemic frames evolve from Second-Year students to practicing professionals?

This section talks about the two primary developmental trajectories: *Type 1 (Major Growth)* and *Type 4 (Major Decline)* revealed through the application of min-max normalization and slope-based classification. For each trajectory, we present the quantitative findings followed by a qualitative analysis of representative excerpts. This mixed-methods structure allows us to demonstrate not only the statistical magnitude (through W_i) of the change but also the qualitative evolution of the learner's reasoning strategies from the Second Year (SE) to the Expert level.

5.2.1 Type 1: Major Growth: Uptrend

Figure 5 shows the type 1 trajectory with the slope $\lambda \geq +0.15$. This Trajectory represents the acquisition of high-level professional frames. The key codes that are under are:

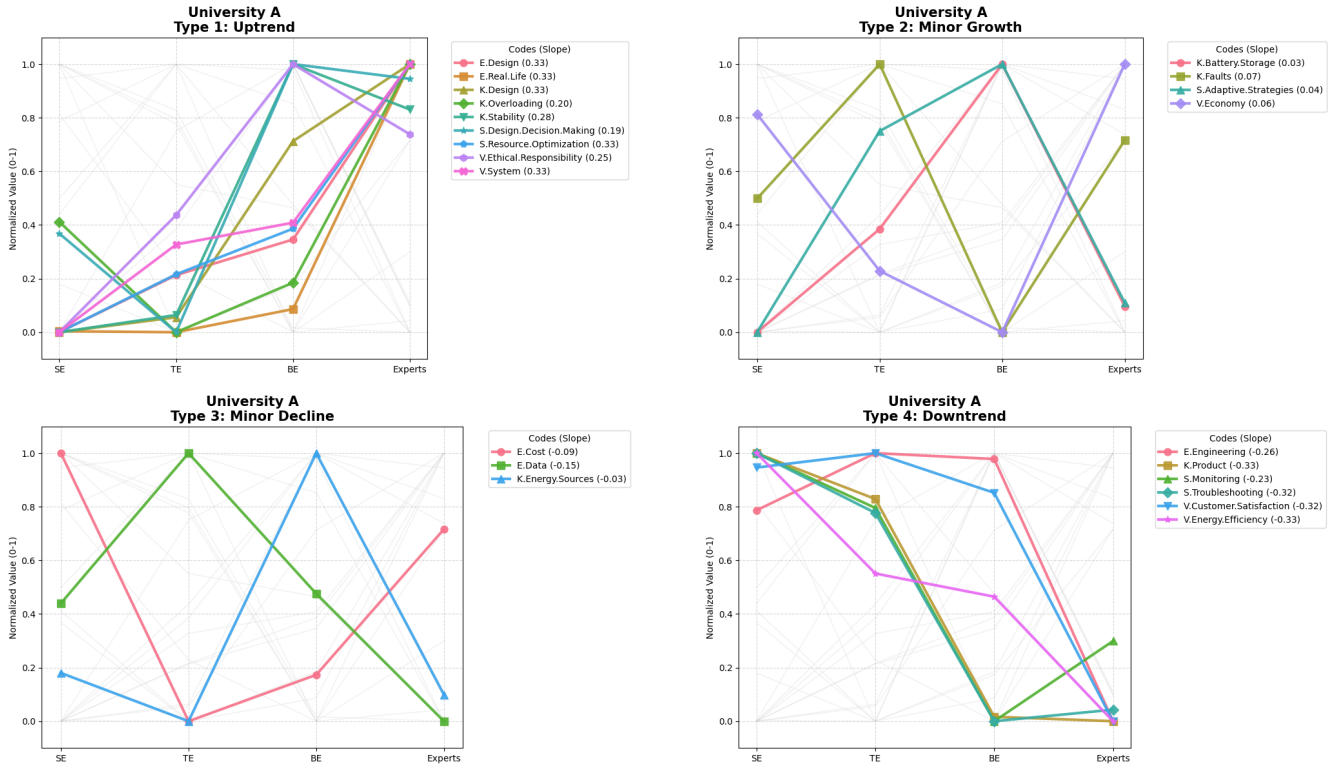


Figure 4: Classification of trajectories into four types with respect to slope threshold (± 0.15).

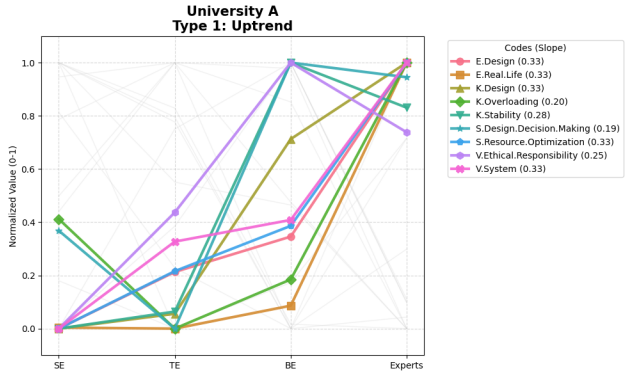


Figure 5: The Ascending Trajectory of epistemic frames with $\lambda \geq +0.15$.

- E.Design: 0.33
- E.Real.life: 0.33
- K.Design: 0.33
- K.Overloading: 0.20
- K.Stability: 0.28
- S.Design.Decision.Making: 0.19
- S.Resource.Optimization: 0.33
- V.Ethical.Responsibility: 0.25
- V.System: 0.33

These codes exhibit a consistent increase in normalized weight from the Second Year (SE) to the Expert (EXP). The sequence of the weights W_i for this Type 1 trajectory can be seen Table 2 in row 1, signaling the successful adoption of the learner into the community of practice.

E.Real.Life ($\lambda = 0.33$): It is the most significant developmental signal observed in Uptrend. The trajectory reveals a late-blooming pattern where the code is virtually flat in SE (0.0) and TE (0.12) cohorts, reflecting an academic isolation where real-world constraints are invisible to the student. A noticeable shift occurs in the BE cohort (0.58), but the Expert cohort reaches saturation (1.0). Qualitative analysis shows that Experts exclusively use field heuristics. Experts replace theoretical citations with professional intuition. As one expert noted, *"In a real grid, you utilize diversity factors... you don't design for the peak of every single house simultaneously."*. This expert behavior shows the effective replacement of the classroom lens of the novice with the field lens of the practitioner.

S.Design.Decision.Making ($\lambda = 0.19$): This epistemological shift is mirrored in the evolution of design logic, revealing a four-stage maturation moving from surface heuristics to deep structural analysis. Novices (SE) initially relied on cost and quality heuristics, operating on the naive assumption that price equals quality. As one student noted, *"I thought the more expensive the thing, the better. So I bought it... I thought it would be better if the cost was higher."* This

indicates a decision-making process driven by budget variables rather than technical fit. By the Third Year (TE), this evolves into Attribute Matching, where students align static component features with spatial needs. For instance, a TE student stated, *"...because this city is small. So I have used a small local substation."* While logical, this approach remains static and spatial. A higher-order complexity emerges in the Final Year (BE), characterized by Multi-Variable Analysis. These students balance conflicting variables like terrain, weather data, and future load before execution. An excerpt illustrates this data-driven approach: *"I looked at precipitation graphs... we have a lot of precipitation... I looked at the terrain where I can see that there are some places where we have trees."* Finally, Experts transcend the game's explicit variables to incorporate Real-World constraints that are invisible in the game interface. They describe the grid's topology before placing a single component, noting, *"First, I am identifying the load centers... the ring main unit will provide redundancy here before I even connect the generation."* The expert protocol reveals a distinct pre-design phase absent in novices, in which contingencies for future load growth are calculated before implementation.

V.System ($\lambda = 0.33$):The data further highlights a transformation in how participants define value, specifically capturing the prioritization of grid stability. In the SE cohort, this value appeared frequently but was framed negatively in terms of the avoidance of game penalties, with students noting, "I removed the line because the message came." By the Expert phase, the normalized frequency remained high, but the definition of the system changed from a set of game rules to an internalized standard of care. They impose an intrinsic benchmark of system health that dictates grid robustness, regardless of whether the external agent explicitly penalizes them or not. An expert illustrates this internalized standard, stating, "I try to keep the system healthy... If there is a breakdown, how to restore it immediately, how to stay healthy, we work from this perspective..." This reflects a shift in role identity from a builder following instructions to a guardian of continuity.

V.Ethical.Responsibility ($\lambda = 0.25$): Concurrently, this trajectory tracks the expansion of the moral boundary. Novices (0.20) framed ethics transactionally, viewing the utility to customer relationship as a commercial contract for profit. One SE student explicitly stated, *"ofcourse, Customer satisfaction is my priority... if I promise that I will give this to customers, I will give it first... that will give me profit..."* This perspective shifts toward environmental stewardship in the Final Year (BE). These students are willing to sacrifice economic efficiency to protect the aesthetic and environmental integrity of the region. A final-year student noted, *"Also coal... of course it's cost is less, but over the longer period of time it will get extinct. And also there will be a lot of pollution and if this polluted air gets to Taj Mahal of course there will be consequences to its beauty..."* Finally, values shift to a non-negotiable civic duty for Experts (1.0), who feel it is the fundamental purpose of their profession. The expert views the grid not as a commercial asset, but as a public utility essential for life. Stripping away all complex variables, one expert revealed the singular mission of the grid: *"We just want the system to deliver electricity to every consumer. That's all we want. We work from this perspective."* This indicates that universal accessibility is the foundational axiom of their mental model, creating an obligation to serve everyone regardless of difficulty.

S.Resource.Optimization ($\lambda = 0.33$): Finally, this trajectory tracks the sophistication of efficiency strategies. SE novices employed naive spatial symmetry, such as "splitting the load 50/50" based purely on visual geometry. One SE student explained, *"I will give an even distribution. If there are 100 residents, 50 will be distributed by one line and 50 by another line. So that the load on the line will be less. I have split it like that."*. TE students advanced to component-level optimization (e.g., matching a small substation to a small city), reflecting a textbook understanding of capacity. However, the trajectory peaks as Experts integrate complex variables—cost, line loss, and future demand—into a single decision matrix. Experts demonstrate a well-thought-out optimized system, with reasoning well grounded in real-world practices. Continuously reflecting on field experience, an expert noted, *"The installed capacity must be higher than the requirement. So, I was thinking of the maximum install capacity power plant...I learned from that practical experience that segregation of equal loading is very much insufficient."*

5.2.2 Type 4: Major Decline: Downtrend

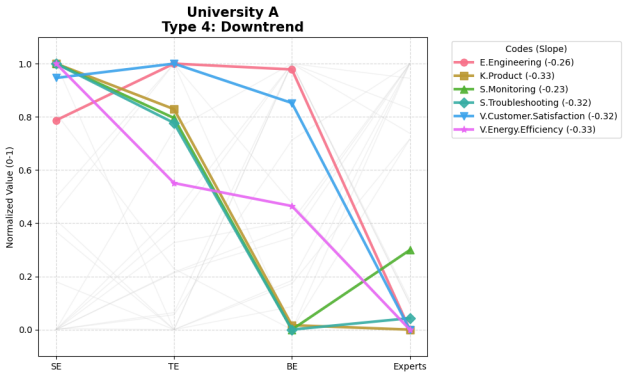


Figure 6: The Descending Trajectories of Cognitive Automation ($\lambda \leq -0.15$)

Figure 6 shows the type 4 trajectory with the slope $\lambda \leq -0.15$. This trajectory represents the fading away of the novice habits. The sequence of the weights W_i for this Type 4 trajectory can be seen Table 2 in row 4. This decrease in weights W_i are critical as they identify behaviors that are prominent in novices(SE) but recede as expertise develops (Exp). This trend does not indicate skill loss, but rather the *cognitive automation* of foundational processes.

The Type 4 trajectories ($\lambda \leq -0.15$) exhibited a negative slope. The key codes fall under:

- S.Troubleshooting (-0.32)
- S.Monitoring (-0.23)
- V.Customer.Satisfaction (-0.32)
- K.Product (-0.33)
- V.Energy.Efficiency (-0.33)

- E.Engineering (-0.26)

S.Monitoring ($\lambda = -0.23$):The most dramatic decline was observed in this trajectory, which captures the reliance on visual verification. SE participants recorded the highest weight (1.0), primarily associated with surface monitoring (constantly scanning the interface for color-coded warnings). An SE student noted, *"I am checking the red line... it turned red again."* This reflects a reactive stance where the student awaits visual feedback from the game mechanics to validate the grid's state. In contrast, Experts (< 0.1), possessing a proactive mental model. Experts analyse the load distribution internally before placement. Consequently, the need for explicit verbal monitoring vanishes; they trust their internal simulation, whereas the novice requires constant external validation.

S.Troubleshooting ($\lambda = -0.32$): Similarly, this trajectory tracks the shift from a trial-and-error loop of patching crashes to a proactive design strategy. Novices (SE) explicitly described a reactive dependency on game notifications, stating, "I got the message twice. I removed them as the message came... Because the last time was short circuit. I wish there was a help button... I thought I was doing it wrong and that's why he is giving me notice." When Third-Year (TE) students encounter errors, they understand the underlying reasons but still perform actions strictly by the rules, indicating a transitional learning phase. A qualitative leap occurs in the Final Year (BE), where students stop reactive error fixing and start simulating alternative strategies to prevent errors entirely. As one BE student explained, "I would have added wind turbine if there was some height. this is like, you know, just alternative way..." This marks the transition to a proactive design strategy that resolves contingencies upfront, eliminating the need for reactive fixes.

The interesting finding for this code is, as expertise develops, this code vanishes. In opposition to the students, the Expert starts designing the grid for reliability and, as they do so, accounts for contingencies. The contingency-oriented behavior can be seen through *"..whatever the fault occurs further, lets say this point, is prone to the overloading, we can have one fault, but as I have this one substation, it is backup for me. So that I can distribute power properly or evenly to particular area..."*. This expert's behavior is typically because they employ a well-thought-out plan in advance.

K.Product ($\lambda = -0.33$):A similar trend characterizes a shift from declarative labeling to tacit integration. The data shows a monotonic decline from SE (1.0) to Expert (0.0). SEs record the highest weight because they must explicitly verbalize their understanding of surface features, often appearing overwhelmed by game mechanics. An SE student noted, *"...I chose a wind turbine, but I didn't get the output... I thought maybe it's a problem with wind or connection maybe..."* This excerpt highlights a focus on the object itself rather than the underlying reason for its failure. In the intermediate phases, the nature of knowledge becomes procedural. While TEs and BEs still frequently reference product names, they begin to connect these labels to broader concepts, attaching conditions and rules to facts. They select objects based on specific conditions rather than just availability. The Expert, conversely, invokes the tacit professional world—referencing concepts like Site Survey and Load Center—that precedes the product. The quantitative drop signifies that the game interface has become transparent; they think through the component to the solution. For the professional, the wind turbine is not a standalone object to be bought, but a conditional solution that emerges from a rigorous assessment. As one expert explained, *"So, in real life, there are many more constraints... we do site survey, we do load center survey... If it is in hilly areas, with good wind speed, we go for wind power turbines... If it is near to the coal mines... we go for coal..."* The expert zooms out beyond the game graph to reference the site survey protocol.

E.Engineering ($\lambda = -0.21$): An interesting deviation from the linear downtrend was found in this trajectory, which followed an inverted-U shape. It peaked in the Third Year (TE) (1.0), reflecting a cohort immersed in heavy theory ("Power Systems I"). TEs attempted to force complex academic constraints, such as per-unit systems or reactance values, onto the game to validate their knowledge. As one TE student noted, *"I was thinking about the 'per-unit' system... to see if the game follows the standard reactance values."* Strikingly, the Expert cohort dropped to zero. Experts did not cite academic theories, voltage or current Laws, or even Electricity Rules by competent authorities, because these norms are internalized and hence tacit. This finding suggests a theoretical trap in the TE year that must be transcended to achieve professional fluency.

V.Energy.Efficiency ($\lambda = -0.33$) along with *V.Customer.Satisfaction* ($\lambda = -0.22$): Lastly, value-based codes showed marked declines. Novices verbalized these as distinct, affective goals, often worrying about "sad people" or "happy faces" on the interface. For experts, these are not separate values to be maximized but embedded constraints within the primary goal of *V.System*; they do not speak of efficiency separately because a stable grid is inherently efficient. The disappearance of these explicit codes marks the transition from explicit student values to implicit professional norms.

6. DISCUSSION

This study aimed to map the granular developmental trajectory of electrical engineering expertise. By quantifying the shift in epistemic frames from the Second Year (SE) to the Expert level, our results provide the empirical evidence for the traditional accumulation model of expertise. The data reveals that expertise development is not merely a linear addition of technical skills, but a complex, trajectories: the simultaneous Adoption of a Professional Lens (Ascending Trajectories) and the Shedding of Novice Habits (Descending Trajectories). This section discusses the mechanisms driving these patterns.

6.1 Adoption of a Professional Lens: Game Rules to Professional lens

The ascending trajectories of *E.Real.Life* and *V.System* illustrate a fundamental shift in the learner's Locus of Control. Novices (SE) operate within an academic frame, where value is defined externally by game mechanics or grading rubrics. This aligns with Engstrom's [14] distinction between school-going activity, where the object is the grade, and work activity, where the object is the societal value.

Their high frequency of *V.System* indicates a defensive mechanism, they act to avoid error messages rather than to ensure system health.

In contrast, the Expert frame is driven by an Internalized Standards. The synchronization of *V.System* and *V.Ethical.Responsibility* in the expert cohort suggests that for a professional, technical decisions are inseparable from their social consequences. This confirms Shaffer's [44] core tenet that professional expertise is not just "knowing how" (Skills) or "knowing that" (Knowledge), but "knowing with" adopting the values and identity of the community of practice. This trajectory reflects Lave and Wenger's [35] theory of Legitimate Peripheral Participation: novices remain on the periphery of the engineering community, engaging academic problems without context, while experts demonstrate full participation through internalized professional values and identity. The delayed emergence of *V.Ethical.Responsibility* codes suggests that undergraduate curricula provide limited opportunity for authentic engagement with the community of practice where these dimensions are enacted [52]. The delayed emergence of these codes points to a critical gap in the curriculum. While students acquire technical competence early, they suffer from contextual isolation, lacking the situated learning environment described by Lave and Wenger [35]. They possess the syntax of engineering (the math) without the semantics (the professional meaning). The sharp rise in the Expert cohort confirms that true fluency requires the adoption of this professional lens. This is a process that currently appears to happen largely after the graduation.

6.2 Shedding Novice Habits: From Reactive Monitoring to Proactive Design

The steep decline in *S.Monitoring* and *K.Product* highlights the mechanism of how novices display a high interaction with game interface, constantly verbalizing labels and verifying visual feedback from the system. They constantly verbalize component labels and verify visual feedback, outsourcing working memory to external representations—a phenomenon known in cognitive science as distributed cognition [29]. This external scaffolding is necessary and adaptive at early stages; without robust internal models, novices require the interface to confirm their understanding at each step.

However, the persistence of these behaviors in third-year (TE) and final-year (BE) students signals problematic dependency. This reflects what Braver [6] terms reactive control: students respond to feedback rather than anticipating consequences beforehand. They rely on the interface to tell them whether a design works, treating error messages as cues to act upon rather than using principled reasoning to avoid errors in the first place. While *S.Design* was visible in their protocols, it lacked correlation with *K.Design* and *E.Design* novices were mimicking engineering actions without the supporting epistemic structure to predict outcomes.

In contrast, experts demonstrate proactive control [6, 30], they simulate outcomes internally before acting. The expert's relative silence during problem-solving is not an absence of cognition but evidence of internalized mental models. The epistemic frame has become so compressed that intermediate reasoning steps no longer require conscious verbalization or external verification. Experts have effectively shed the scaffolding behaviors that once supported their learning, achieving what Dreyfus and Dreyfus [13] describe as the fluid, intuitive performance of expertise—no longer relying on context-free rules or external feedback but on internalized professional judgment.

6.3 The "Theoretical Frame": The Intermediate Effect

An interesting finding was the trajectory of *E.Engineering*, which peaked in the Third Year (TE). This aligns with the intermediate effects observed in medical education by Schmidt and Rikers [43], where intermediate learners process information more laboriously than both novices and experts due to the activation of elaborate but encapsulated causal networks. The TE cohort, immersed in heavy theory courses, attempted to force rigid academic abstractions e.g., ($Per - UnitSystems$) onto a dynamic problem where they were ill-suited. This represents rigidity of the academic, as there is more emphasis on theoretical precision (*E.Engineering*) that temporarily overrides practical intuition (*E.Real.Life*). If not corrected, this leads to graduates who are technically rigorous but practically paralyzed. The drop to near-zero in the expert cohort confirms Glaser's [23] observation that expertise relies on flexible, context-sensitive heuristics rather than the rigid application of weak methods.

6.4 The Expert as Benchmark

A key methodological consideration in this study concerns the asymmetric sample composition, particularly the inclusion of a single expert participant. While this represents a limitation in terms of statistical generalizability, the design employs the expert not as a representative mean but as "a gold standard benchmark". This methodological approach aligned with Epistemic Frame Theory's use of professional exemplars to define the target epistemic state toward which learners progress [44]. This benchmark serves to identify the "ceiling" of the professional lens, establishing the developmental endpoint against which student trajectories can be mapped.

The resulting trajectories reveal systematic patterns: ascending codes (e.g., *E.Real.Life*, *V.Ethical*) show consistent growth from SE through Expert, while descending codes (e.g., *S.Monitoring*, *K.Product*) demonstrate systematic decline. These patterns emerge not from statistical aggregation across multiple experts, but from the structural positioning of student cohorts relative to the professional benchmark. The consistency of these trajectories across multiple SKIVE dimensions—with some ascending, some descending, and some exhibiting non-linear patterns—suggests that the observed trends reflect genuine developmental phenomena rather than artifacts of individual expert variation.

7. CONCLUSION AND IMPLICATIONS

This study highlights the holistic nature of the expertise gap: students do not primarily lack knowledge; rather, they operate within a fractured epistemic frame in which theory, design, and ethics are treated as separable academic tasks. In contrast, expert practice reflects a comprehensive epistemic frame, in which technical reasoning, ethical judgment, and contextual awareness are inseparable and enacted simultane-

ously through an internalized professional lens. By applying min-max normalization to verbal protocols, we effectively neutralized the verbosity bias that often obscures qualitative comparisons. This methodological innovation allowed us to visualize expertise as two simultaneous trajectories: the *Adoption of Professional lens* (Uptrends) and the *Shedding of Novice Habits* (Downtrends). Our analysis confirms that the silence of an expert is as significant as the speech of a novice. The monotonic decline of codes like *K.Product* and *S.Monitoring* provides empirical evidence that expertise involves the internalization of rules. Conversely, the ascent of *E.Real.Life* and *V.System* maps the transition from a reactive, rule-based game lens to a proactive, ethical field lens.

The revealed trajectories provide actionable insights for curriculum design, specifically for bridging the gap between academic theory and professional practice. The late emergence of *E.Real.Life* suggests that current curricula may isolate students in academic idealism for too long. To accelerate novices' development toward expert-like competencies, programs should integrate real-world constraints. The trajectory of *V.Ethical.Responsibility* implies that ethics training must evolve. Rather than treating ethics as a separate compliance module, it should be embedded in technical design tasks.

8. LIMITATIONS AND FUTURE WORK

We acknowledge that the single-expert (n=1) limits claims about variability in expert performance across engineering sub-domains or institutional contexts. Future work should expand the expert cohort to examine whether these trajectory slopes remain stable across diverse professional populations. Such expansion would enable investigation of whether the professional lens exhibits domain-specific variations or represents a more universal engineering epistemic frame. Nevertheless, the current benchmark approach provides sufficient granularity to identify which expert-like competencies emerge at which undergraduate stages and which novice behaviors require explicit intervention—actionable insights that inform curriculum design even in the absence of expert population statistics.

Additionally, the cross-sectional design infers developmental trajectories rather than tracking them directly. Future research should employ longitudinal designs to validate whether individual learners typically traverse these inferred paths. Furthermore, to empirically triangulate the phenomenon of "expert silence" (specifically in *S.Monitoring*), future work will integrate Multimodal Learning Analytics (MMLA).

9. ACKNOWLEDGMENTS

This study was funded by SBI Foundation Hub for Data & Analytics for India (grant number DO/2023-SBIF002-007).

10. REFERENCES

[1] K. Anders Ericsson. Deliberate practice and acquisition of expert performance: a general overview. *Academic emergency medicine*, 15(11):988–994, 2008.

[2] K. C. Anderson and G. Leinhardt. Maps as representations: Expert novice comparison of projection understanding. 20(3):283–321.

[3] P. Benner. Using the dreyfus model of skill acquisition to describe and interpret skill acquisition and clinical judgment in nursing practice and education. *Bulletin of science, technology & society*, 24(3):188–199, 2004.

[4] P. Benner. From novice to mastery-ii, the dreyfus and dreyfus model of skill acquisition in nursing practice. *Educating Nurses*, 2022.

[5] S. Bhanja and A. Das. Impact of data normalization on deep neural network for time series forecasting. *ArXiv*, abs/1812.05519, 2018.

[6] T. S. Braver. The variable nature of cognitive control: a dual mechanisms framework. *Trends in cognitive sciences*, 16(2):106–113, 2012.

[7] K. B. Carbonell and J. J. van Merrienboer. Adaptive expertise. *The Oxford handbook of expertise*, page 262, 2019.

[8] W. G. Chase and H. A. Simon. Perception in chess. 4(1):55–81.

[9] M. T. H. Chi, P. J. Feltovich, and R. Glaser. Categorization and representation of physics problems by experts and novices*. 5(2):121–152.

[10] J. Day. What is an expert? *Radiography*, 8(2):63–70, 2002.

[11] P. Dehn, S. Munch Simonsen, and M. L. Olesen. Multidimensional factors determine skill acquisition development in guided self-determination: A qualitative study. *Scandinavian Journal of Caring Sciences*, 37(2):549–560, 2023.

[12] M. Deininger, S. R. Daly, K. H. Sienko, and J. C. Lee. Novice designers' use of prototypes in engineering design. 51:25–65.

[13] S. E. Dreyfus. The five-stage model of adult skill acquisition.

[14] Y. Engeström. *Learning by expanding: An activity-theoretical approach to developmental research*. Orienta-Konsultit, Helsinki, 1987.

[15] M. Eraut. The acquisition and use of educational theory by beginning teachers. *Action and reflection in teacher education*, 69:88, 1994.

[16] M. Eraut. Non-formal learning and tacit knowledge in professional work. *British journal of educational psychology*, 70(1):113–136, 2000.

[17] M. Eraut. Transfer of knowledge between education and workplace settings. In *Workplace learning in context*, pages 217–237. Routledge, 2004.

[18] K. A. Ericsson, R. T. Krampe, and C. Tesch-Römer. The role of deliberate practice in the acquisition of expert performance. *Psychological review*, 100(3):363, 1993.

[19] M. Fenton-O'Creevy and S. Hutchinson. Building the foundations of professional expertise: creating a dialectic between work and formal learning. *Learning and Teaching in Higher Education*, (4-1):69–92, 2010.

[20] Y. Fetisova. Applying the expert-novice paradigm in tennis coaching: Improving coaches' knowledge, diagnostic skills and understanding of the tennis serve technique.

[21] G. C. Fleming, M. Klopfer, A. Katz, and D. Knight. What engineering employers want: An analysis of technical and professional skills in engineering job advertisements. *Journal of Engineering Education*,

113(2):251–279, 2024.

[22] D. R. Gentner and A. Stevens. Lawrence erlbaum associates. *Hillsdale, New Jersey*, pages 7–14, 1983.

[23] R. Glaser. Education and thinking: The role of knowledge. *American Psychologist*, 39(2):93–104, 1984.

[24] S. Goldman and A. Petrosino. Cognition and technology group at vanderbilt.(1999). design principles for instruction in content domains: Lessons from research on expertise and learning. *Handbook of applied cognition*, pages 595–627, 1999.

[25] P. T. Hardiman, R. Dufresne, and J. P. Mestre. The relation between problem categorization and problem solving among experts and novices. 17(5):627–638.

[26] J. A. Harsh, M. Campillo, C. Murray, C. Myers, J. Nguyen, and A. V. Maltese. "seeing" data like an expert: An eye-tracking study using graphical data representations. 18(3):ar32.

[27] G. Hatano and K. Inagaki. Two courses of expertise. , 6:27–36, 1984.

[28] G. Hatano and Y. Oura. Commentary: Reconceptualizing school learning using insight from expertise research. *Educational researcher*, 32(8):26–29, 2003.

[29] J. Hollan, E. Hutchins, and D. Kirsh. Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(2):174–196, 2000.

[30] L. L. JACOBY. The role of cognitive control. *Dual-process Theories in Social Psychology*, page 383, 1999.

[31] B. D. Jee, D. Gentner, D. H. Uttal, B. Sageman, K. Forbus, C. A. Manduca, C. J. Ormand, T. F. Shipley, and B. Tikoff. Drawing on experience: How domain knowledge is reflected in sketches of scientific structures and processes. 44(6):859–883.

[32] J. Kua, W.-S. Lim, W. Teo, and R. A. Edwards. A scoping review of adaptive expertise in education. *Medical Teacher*, 43(3):347–355, 2021.

[33] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[34] J. Larkin, J. McDermott, D. P. Simon, and H. A. Simon. Expert and novice performance in solving physics problems. 208(4450):1335–1342.

[35] J. Lave and E. Wenger. *Situated learning: Legitimate peripheral participation*. Cambridge University Press, Cambridge, UK, 1991.

[36] L. J. Lyon. Development of teaching expertise viewed through the dreyfus model of skill acquisition. *Journal of the Scholarship of Teaching and Learning*, pages 88–105, 2015.

[37] A. Paikrao, D. Nath, R. Rajendran, and R. Mitra. Enhancing retrospective think-aloud protocols with eye-gaze data. In *International Conference on Computers in Education*, 2025.

[38] A. Paikrao, J. Shaha, R. Rajendran, and R. Mitra. Incremental expertise: The continuum of expertise in electrical engineering grid design through the lens of epistemic frames. *International Association for Development of the Information Society*, 2025.

[39] A. M. PAIKRAO and R. MITRA. Expert-novice differences in electrical circuit analysis based on the order of attention. 2023.

[40] A. M. Persky and J. D. Robinson. Moving from novice to expertise and its implications for instruction. 81(9):6065.

[41] D. Rosengrant, C. Thomson, T. Mzoughi, M. Sabella, C. Henderson, and C. Singh. Comparing experts and novices in solving electrical circuit problems with the help of eye-tracking. pages 249–252.

[42] A. A. Rupp, M. Gushta, R. J. Mislevy, and D. W. Shaffer. Evidence-centered design of epistemic games: Measurement principles for complex learning environments.

[43] H. G. Schmidt and R. M. Rikers. How expertise develops in medicine: knowledge encapsulation and illness script formation. *Medical Education*, 41(12):1133–1139, 2007.

[44] D. W. Shaffer. Epistemic frames for epistemic games. 46(3):223–234. Number: 3.

[45] D. W. Shaffer, D. Hatfield, G. N. Svarovsky, P. Nash, A. Nulty, E. Bagley, K. Frank, A. A. Rupp, and R. Mislevy. Epistemic network analysis: A prototype for 21st-century assessment of learning. 1(2):33–53.

[46] S. R. Singer, N. R. Nielsen, and H. A. Schweingruber. *Discipline-based education research: understanding and improving learning in undergraduate science and engineering*. The National academies press.

[47] D. Singh and B. Singh. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, 97:105524, 2020.

[48] M. U. Smith. Expertise and the organization of knowledge: Unexpected differences among genetic counselors, faculty, and students on problem categorization tasks. 29(2):179–205.

[49] J. Sweller. Cognitive load during problem solving: Effects on learning. 12(2):257–285.

[50] J. Sweller. Cognitive load theory. In *Psychology of Learning and Motivation*, volume 55, pages 37–76. Elsevier.

[51] J. Trevelyan. Transitioning to engineering practice. *European Journal of Engineering Education*, 44(6):821–837, 2019.

[52] E. Wenger et al. Communities of practice: Learning as a social system. *Systems thinker*, 9(5):2–3, 1998.

No Single Metric Tells the Whole Story: Interpreting Multiple Pattern Mining Metrics

Andres Felipe Zambrano[1], Jaclyn Ocumpaugh[2], Ryan S. Baker[3], Zhanlan Wei[1], Xiner Liu[1], Hyeongjo Kim[4], Qianhui Liu[4], Jeffrey Ginger[4], Luc Paquette[4], Amanda Barany[1]
[1]University of Pennsylvania
[2]University of Houston
[3]Adelaide University
[4]University of Illinois at Urbana-Champaign
{afzambrano97, jlocumpaugh, ryanshaunbaker, luc.paquette, amanda.barany}@gmail.com, {zhanlanw, xiner}@upenn.edu, {hk61, ql29, ginger}@illinois.edu

ABSTRACT
Pattern mining techniques are widely used in Educational Data Mining to uncover meaningful patterns in learner behaviors, discourse, and learning processes. However, a range of metrics is used to quantify pattern strength, often producing different conclusions. In this study, we systematically compare popular metrics from Association Rule Mining, Sequential Pattern Mining, Epistemic Network Analysis (ENA), and Ordered Network Analysis (ONA) by examining (a) similarities in the patterns highlighted by each method and metric, (b) their associations with validated external measures of learning and motivation, and (c) expert ratings of theoretical alignment and perceived interestingness. Our results reveal two clusters of metrics: one emphasizing frequent, common, and theoretically expected patterns (*Support, Confidence, Cosine, Jaccard*, and *ENA/ONA connection weights*), and another emphasizing less frequent patterns driven more by statistical dependence than by prevalence (e.g., *Added Value, Conviction*, the *Phi coefficient*, and *Lift*). This second group is sometimes rated as more interesting but is less theoretically aligned. Our findings suggest that combining multiple metrics and methods can provide complementary perspectives, either reinforcing conclusions when results converge or signaling the need for deeper investigation when they diverge. Doing so can help us to better understand complex learning processes and their relationships with educational outcomes.

Keywords
Association Rule Mining, Sequential Pattern Mining, Epistemic Network Analysis, Ordered Network Analysis, Interestingness Metrics.

1. INTRODUCTION
A strength of the Educational Data Mining (EDM) community is the diverse range of modeling approaches and algorithms it employs, often to study the same phenomena. One area where this is the case is pattern mining techniques such as Association Rule Mining (ARM; [1]), which have been widely applied to uncover interpretable patterns in student behaviors, problem-solving actions, and learning processes [11, 20, 22, 24].

ARM techniques aim to identify implication rules of the form X→Y that capture relationships among events, items, or discrete variable values based on their co-occurrence in data [1]. To achieve this, ARM employs a range of methods to mine potentially meaningful patterns [21], along with metrics to assess which patterns merit further analysis or consideration [19, 33]. Each of these metrics emphasizes different aspects of these patterns, such as their frequency (*Support*) or conditional reliability (*Confidence*). Others highlight associations that deviate from expectations under statistical independence (*Lift and Cosine*) and capture the extent to which the occurrence of one event increases the likelihood of another beyond its base rate (*Added Value*). The choice of metric influences which patterns are identified and how they are interpreted, particularly in applied domains such as EDM, where interpretability and relevance to learning processes are central.

More recently, Epistemic Network Analysis (ENA; [27]) has emerged as an alternative approach for discovering important relationships between discrete variable values. Initially developed to study thematic complexity in discourse data, ENA models learning processes by counting patterns of co-occurrence and applying cosine normalization (distinct from the *Cosine* similarity metric used in ARM) to estimate the strength of connections between codes, while accounting for differences in discourse length or interaction volume across the students or groups being analyzed [7, 27]. A key strength of ENA lies in its network representations, which support visual interpretation of complex relationships among codes and facilitate exploratory analyses of learning processes. However, the quantification of ENA patterns is typically achieved by comparing its *Connection Weights*, which are conceptually similar to ARM's *Support*. Unlike other ARM metrics such as *Cosine*, *Lift*, or *Added Value*, *ENA Connection Weights* do not explicitly normalize for the base rates of individual codes. As a result, ENA may prioritize patterns involving more prevalent codes, while potentially obscuring stronger and more informative associations among less frequent codes.

ARM and ENA each have many variants, including widely used variants that extend their base approach to explicitly account for the temporal order of events or codes. For ARM, the most popular extension is Sequential Pattern Mining (SPM; [2]) and for ENA, it is Ordered Network Analysis (ONA; [34]). While these variants enable the analysis of sequential patterns rather than only unordered co-occurrences, they largely inherit the conceptual strengths and

Andres Felipe Zambrano, Jaclyn Ocumpaugh, Ryan S. Baker, Zhanlan Wei, Xiner Liu, Hyeongjo Kim, Qianhui Liu, Jeffrey Ginger, Luc Paquette, and Amanda Barany. No Single Metric Tells the Whole Story: Interpreting Multiple Pattern Mining Metrics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 205–216. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039720

limitations of their underlying methods, including how associations are quantified, normalized, and ultimately interpreted.

The existence of multiple analytic approaches to analyzing the relationship between discrete variable values raises an important question for educational researchers: What method do I use? Each of these algorithms and metrics within it seeks to quantify the strength/importance of connections in data, but it is not clear which is most likely to yield meaningful and useful conclusions about learning processes.

Prior work in EDM has attempted to address this question by comparing different ARM metrics against patterns deemed relevant by human experts, in order to assess which metrics reveal patterns that are most meaningful in educational contexts. For example, Merceron and Yacef [19] examined four ARM metrics in a small-scale case study of student assessment data, arguing in favor of metrics that are intuitive for educators and robust to limited data. Their results highlight base-rate-adjusted metrics (i.e., *Cosine* and *Lift*) as especially suitable for educational applications.

Building on this work, Bazaldua et al. [5] asked human experts to evaluate the importance of a range of 21 ARM metrics. Specifically, they asked the human raters which patterns were *theoretically aligned* and *interesting* (to the human raters). They found that *Phi coefficient*, *Conviction*, and *Jaccard* were among the strongest predictors of human judgments. However, *Jaccard*, *Phi*, *Conviction*, and 16 other metrics were actually ***negatively*** associated with human ratings of interestingness. This finding demonstrates the complexity of identifying patterns that researchers genuinely find interesting, rather than patterns that merely reflect expected relationships, yielding accurate but ultimately trivial conclusions.

These studies investigated expert judgments about the importance and interestingness of the identified patterns but did not evaluate how useful the patterns were in terms of research findings. An alternative and complementary approach to those used in prior studies is therefore to assess discovered patterns in terms of whether they are related to external measures of key educational constructs, such as situational interest, self-efficacy, learning gains, and other motivational or cognitive variables. We can then ask human experts to evaluate not only the patterns themselves but also the identified associations. Accordingly, this study extends prior work by addressing the following research questions: **(RQ1)** which pattern mining methods, and strategies for aggregating connections or sequences, are most effective at identifying patterns among coded events that are associated with external measures of learning and motivation, and **(RQ2)** which of the metrics produces associations to external measures of learning and motivation that are meaningful to domain experts? This approach enables an explicit comparison of pattern mining methods not only in terms of what domain experts find interesting, but also what they find important, namely, their consequential validity.

2. RELATED WORK

2.1 Association Rule Mining

Association Rule Mining (ARM) is a data mining technique used to identify associations among items, events, or codes in large datasets [1]. ARM seeks to discover rules expressed as implication statements of the form X→Y, where the antecedent (X) consists of one or more conditions whose occurrence is associated with an increased likelihood of the consequent (Y). For example, a rule identified through ARM might indicate that when a student visits a point of interest and uses a scientific tool to measure a physical variable (antecedent), the student is likely to make a scientific observation (consequent). Importantly, the antecedent-consequent relationship in ARM does not imply temporal order. In this example, the scientific observation may have occurred before the tool use or the visit to the point of interest. Rather than modeling a sequence of events, ARM captures conditional relationships among sets of events or constructs.

Sequential Pattern Mining (SPM) extends ARM by explicitly incorporating temporal order into the discovery of patterns [2]. Rather than identifying unordered co-occurrences, SPM focuses on discovering frequent sequences of events in which antecedent events must temporally precede consequent events within a specified temporal window, transaction, or conversation. This makes SPM particularly suitable for modeling processes that unfold over time, such as learning trajectories or sequences of student actions. For example, SPM could be used to evaluate whether a visit to a point of interest and scientific tool use (antecedent) temporally precedes the student's scientific observation (consequent).

In the EDM literature, both ARM and SPM have been used in a range of projects. ARM has been used to identify patterns of student actions within learning management systems, such as combinations of content access and assessment activities, which were then examined to better understand differences between high- and low-performing students [24]. Similarly, Hwang et al. [11] applied ARM to show that students who struggled with foundational concepts were more likely to fail subsequent, dependent items. SPM has been used to capture the temporal structure of learning and collaboration. For example, Perera et al. [23] analyzed trace data from student teams working in a professional software development environment and identified distinct temporal patterns of tool use and interaction associated with effective coordination, leadership, and task management. Martínez et al. [17] applied SPM to interaction traces from student collaborations using a shared interactive tabletop. Their analysis revealed systematic differences associated with group achievement: successful groups engaged in parallel, discussion-rich interactions and actively externalized their thinking, whereas lower-achieving groups tended to interact sequentially with the same resources.

Kinnebrew et al. [12] extended SPM by proposing Differential Sequence Mining, an exploratory pattern mining approach designed to identify activity sequences that occur differentially across groups or phases of learning. Applied to interaction data from an intelligent tutoring system, this method revealed that high-performing students more frequently engaged in monitoring behaviors, such as assessing their progress and reflecting on recent actions. Overall, this body of work demonstrates the value of ARM and SPM in revealing interpretable temporal patterns in learning data that support a deeper understanding of diverse learning processes.

2.1.1 Interestingness Metrics

In ARM and SPM, patterns can be discovered using several methods [21]. The choice of method depends on the type of patterns researchers aim to highlight (e.g., ordered sequences of events or co-occurrences within a specific time window) [21], as well as the computational resources required for different data sizes and the size of the candidate itemset space (i.e., the number of possible combinations of events or codes that can be formed from the available set of items) [16].

Across these methods, the discovery process typically involves identifying frequent itemsets—sets of events or codes that co-occur within the units of analysis (e.g., student study sessions, problem

attempts, or conversations)—based on a minimum *Support* threshold. *Support* measures how frequently a rule or pattern occurs in the dataset and is defined as the proportion of transactions or instances in which both the antecedent (X) and the consequent (Y) appear together (P(XY)). As such, *Support* captures prevalent or common patterns, favoring rules that represent behaviors or events that occur frequently.

After identifying itemsets with high *Support* (i.e., frequent co-occurrence), implication rules of the form X→Y are generated and evaluated using a variety of metrics that quantify different aspects of the relationship between the antecedent (X) and the consequent (Y). The most commonly used metric is *Confidence*, which estimates the conditional probability of observing the consequent given the antecedent, P(Y|X). As such, it emphasizes conditional reliability, or how consistently the occurrence of X is associated with the occurrence of Y. *Confidence*-based rules capture asymmetric relationships and are particularly useful for identifying cases in which one behavior reliably accompanies another, even when the reverse association does not hold.

Though widely used, *Support* and *Confidence* also have well-documented limitations [19]. For example, while high *Support* characterizes dominant or typical behaviors, it may overlook less frequent but potentially meaningful patterns. Similarly, *Confidence* does not account for the overall frequency of the consequent and may therefore overestimate the strength of associations involving high-frequency events or codes.

To address these limitations, a range of so-called *interestingness metrics*, explicitly normalize association rules by the marginal frequencies (base rates) of the antecedent and/or consequent [33]. Metrics such as *Cosine* capture the similarity between the occurrence vectors of X and Y by normalizing their joint frequency by the geometric mean of their individual frequencies, thereby emphasizing the strength of association while reducing the influence of highly frequent events. Similarly, *Lift* compares the observed co-occurrence of X and Y with what would be expected if they were statistically independent, highlighting associations that are stronger than chance. *Added Value* also measures the extent to which the occurrence of X increases the likelihood of Y beyond its baseline probability. More broadly, Tan et al. [33] catalogued more than twenty such metrics, all designed to surface non-trivial or surprising patterns that may be less frequent but more informative about underlying learning processes.

Among these metrics, Merceron and Yacef [19] recommend *Lift*, *Added Value*, and *Cosine* as especially suitable metrics for educational data, after first filtering out rules with low *Support* and *Confidence*. They argue that these metrics are intuitive for non-data-mining experts (e.g., teachers or school administrators) while remaining informative for identifying non-trivial associations. At the same time, they emphasize that no single metric is sufficient for identifying pedagogically meaningful rules. Different metrics surface different types of patterns—such as frequent behaviors, reliable conditional relationships, or associations that stand out relative to overall frequency—and their usefulness depends both on the research goal and on the interpretability of the resulting rules for educators and researchers. Accordingly, they emphasize the need to compare rules across multiple metrics to obtain a more nuanced view of learner behavior and to support more robust and meaningful conclusions.

Motivated by this need to identify which of the many available *interestingness metrics* are most informative for educational stakeholders, Bazaldua et al. [5] conducted a systematic comparison of a wide range of metrics using interaction data focused on affective states and learner behaviors in an educational platform, together with ratings provided by three domain experts familiar with the learning context. In their study, a large set of association rules was first generated from student interaction logs and filtered using minimum *Support* and *Confidence* thresholds. The experts then rated the interestingness of each rule using a Likert scale, providing a human benchmark against which rankings of rules induced by different *interestingness metrics* could be compared. The authors evaluated not only *Cosine*, *Lift*, and *Added Value*, but also several additional metrics—including the *Phi coefficient*, *Conviction*, and *Jaccard*—drawn from the set of metrics compiled by Tan et al. [33]. Their results showed that multiple metrics were predictive of expert judgments, with metrics such as the *Phi coefficient*, *Conviction*, and *Jaccard* exhibiting stronger alignment with human ratings. However, only *Lift* showed a positive association with expert judgments across the 21 evaluated metrics, while most showed significant negative associations. This finding highlights that what human analysts consider interesting may differ substantially from what *interestingness metrics* prioritize.

2.2 Epistemic Network Analysis

Epistemic Network Analysis (ENA; [27]) is a technique used in the quantitative ethnography (QE) community to model the structure of relationships among a set of theoretically meaningful constructs or codes. ENA has been applied across a wide range of contexts, including analyses of learning processes in digital learning environments [28], task performance [25], gaze patterns [3], team communication [31], social media interactions [9], and video game player behavior [39].

ENA operationalizes relationships among constructs by quantifying their co-occurrence within segments of coded data, typically organized into conversations or horizons (analogous to transactions in ARM and SPM) and aggregated across learners, groups, or tasks. Connections between constructs are accumulated using a moving window that links codes appearing in the same or recent prior lines of discourse, thereby capturing local temporal proximity without imposing strict sequential constraints [7, 27]. These co-occurrence counts are aggregated for each unit of analysis (e.g., learners or groups), then normalized using cosine or spherical normalization (distinct from the *Cosine interestingness metric* used in ARM) to account for differences in the amount of coded data across units. Finally, dimensionality reduction techniques are applied to project the resulting normalized vectors into a two-dimensional space, facilitating statistical and visual comparison across units and groups.

In ENA visualizations, each unit of analysis is represented both as a weighted network graph and as a point in a two-dimensional space corresponding to the network's centroid (or mean). In these network representations, nodes correspond to events or constructs used to code the data, and edges represent the relative strength of co-occurrence between pairs of constructs. Spatial proximity between constructs should not be interpreted as a direct measure of association strength. In ENA, node positions are derived from the overall structure of variance in the network space, so constructs that appear closer together may reflect similar contributions to the underlying dimensions rather than consistently stronger co-occurrence. On the other hand, the relative positions of constructs and unit centroids are interpretable, as they reflect which constructs and connections most characterize a given unit's network. For example, if a student *i* is positioned closer to construct A than to construct C, it indicates that the interaction patterns associated with

student *i* more strongly involve construct A. In this way, differences in the positions of units of analysis can also be interpreted as differences in their underlying connection patterns, enabling researchers to compare individual networks, group means, and network differences within a shared analytical space (acknowledging the limitations inherent to any dimensionality reduction).

ENA's primary strength lies in its ability to represent and compare complex patterns of pairwise associations among multiple constructs in rich, temporally indexed data, while maintaining a strong emphasis on interpretability through visual representations. However, ENA also embodies specific analytic assumptions that shape the types of patterns it can reveal. Standard ENA models capture only pairwise (dyadic) associations and treat connections as symmetric, focusing on co-occurrence rather than conditional dependence. As a result, ENA connection weights tend to be highly correlated with the *Support* metric commonly used in frequency-based pattern mining approaches [38]. Consequently, *ENA connection weights* are subject to similar limitations as *Support*, which does not take into account the base rates of individual codes, and does not capture the nuances emphasized by conditional metrics (e.g., *Confidence*) or base-rate-adjusted metrics such as *Cosine*, *Lift*, *Added Value*, and related *interestingness metrics*.

Chronological ordering has been incorporated into the ENA framework through Ordered Network Analysis (ONA; [34]). Like SPM, this method constrains connections based on temporal precedence and thus distinguishes temporally between X→Y and Y→X. However, unlike *Confidence* and many *interestingness metrics*, ONA does not model these relationships in terms of conditional probability or statistical independence. Additionally, like ENA, ONA relies on co-occurrences as the primary mechanism for highlighting important patterns, and is thus also susceptible to identifying patterns that are driven primarily by high base rates rather than by genuine implication-based relationships between codes.

2.3 Contributions of this Study

Although ARM and SPM offer a wide range of *interestingness metrics* and support the identification of patterns involving three or more codes or events, it remains unclear whether all of these metrics and higher-order patterns (i.e., patterns involving three or more codes or events) meaningfully enhance interpretation in educational research. Methods such as ENA and ONA were developed to provide qualitative researchers with a quantitative framework for identifying meaningful patterns that can then be interpreted through domain expertise to "close the interpretative loop" and support robust, data-driven conclusions about educational phenomena [27]. Merceron and Yacef [19] argue that pattern mining should rely on metrics beyond *Support*, rather than primarily on co-occurrence-based measures such as those used in ENA. However, Bazaldua et al. [5] show that these alternative metrics do not necessarily align more closely with the patterns human researchers find interesting. This discrepancy reflects the challenge of defining what makes a pattern "interesting," a term generally used without reflection in early mathematical work on *interestingness metrics*. Instead of relying on the term alone, we evaluate metrics based on whether the patterns they produce connect to validated external measures of important motivational and cognitive constructs. We then build on Bazaldua et al. [5] and ask domain experts to assess whether these associations (not only the patterns themselves) are meaningful and interesting. By doing so, we can better assess which metrics are most informative for educational research and for supporting the interpretative process.

3. METHODS

3.1 Educational Context

The dataset for this study was collected from eight 5-day summer camps conducted in 2024 and 2025 as part of the *What if Hypothetical Implementations in Minecraft* (WHIMC) project [13], and has been previously published in [40]. WHIMC uses *Minecraft: Java Edition* to engage learners in exploring hypothetical astronomy scenarios through "What if" questions (e.g., "What if Earth had no moon?" or "What if Earth orbited a colder sun?"). During the first three days of each camp, students, guided by pedagogical agents (NPCs) and human facilitators, evaluated the habitability of a range of hypothetical worlds and real exoplanets modeled in Minecraft. During the final two days, students explored a Mars map generated from real terrain data and designed shelters capable of sustaining human life.

During each camp, students completed twelve motivational and learning assessments (see Table 1). On day 1, they completed Boeder et al.'s [6] interest development scale and Gadbury's [10] adaptation of LoPresto and Murrell's [15] astronomy knowledge test. On day 3, students completed Gadbury's [10] astronomy and Minecraft interest survey. On day 4, they completed Linnenbrink-Garcia et al.'s [14] situational interest scale and Britner and Pajares' [8] self-efficacy scale, and they also repeated the day 1 instruments. Three surveys (situational interest, astronomy interest, and *Minecraft* interest) were not administered during the first camp; these instruments were added beginning with the second camp.

Table 1. Interest and knowledge assessments.

Instrument	Day	Camps	N
Boeder et al.'s Interest Development	1	8	104
Gadbury et al.'s Knowledge Assessment	1	4	52
NSS Astronomy Knowledge Assessment 1	1	4	46
Camp-specific Knowledge Assessment	2	3	31
Gadbury et al.'s Astronomy Interest	3	7	82
Gadbury et al.'s Minecraft Interest	3	7	82
Boeder et al.'s Interest Development	4	7	88
Linnenbrink-Garcia et al.'s Situational Interest	4	7	88
Britner & Pajares' Self-efficacy	4	8	101
Gadbury et al.'s Knowledge Assessment	4	4	47
Camp-specific Knowledge Assessment	4	4	44
NSS Astronomy Knowledge Assessment 2	4	4	43

The knowledge assessment was revised between the 2024 and 2025 camps to better align with core astronomy concepts. A second instrument, based on the National Science Standards (NSS; [26]), was introduced. This instrument assessed concepts addressed during camp activities (e.g., eclipses, tides, seasons, rotation speed, and orbital periods) on day 1, as well as students' ability to transfer those concepts to constructs not covered during the camp on day 5. A separate knowledge assessment was also administered at the beginning of days 2 and 4 to evaluate concepts targeted in the exploration worlds; it was not administered on day 1, which was primarily devoted to orientation and familiarization with facilitators and the game environment rather than learning-focused activities. Normalized learning gains were calculated for each knowledge assessment as the ratio of a student's observed improvement (or decline) to the maximum possible change [18].

A total of 118 students from urban and rural areas across four states participated in the summer camps. The sample included 72 male, 40 female, and 3 non-binary students, as well as 3 students who

preferred not to disclose their gender. Participants self-identified as 44 White, 49 African American, 7 Hispanic/Latinx, 2 Native American, 10 Other, and 6 who preferred not to disclose their race/ethnicity. Participation was voluntary, and written assent and parental consent were obtained for all students. 13 students were excluded from the analyses because they did not complete any surveys or lacked recorded activity interaction logs on multiple days. No significant demographic differences were observed between the excluded and included participants.

3.2 Coding of Gameplay Log Data

Our current study uses previously coded data [40] from the first three days of the summer camp, during which students explored different "What if" worlds to identify physical variables, observe environmental changes, and assess potential human habitability. These days were selected because the worlds exhibit similar, scientifically-grounded behaviors, allowing a single codebook to be applied consistently across all three. The structured nature of these activities also supports greater consistency across camps held in different locations.

Students were free to explore the worlds at their own pace (moving quickly or slowly, independently or collaboratively) while observing areas of interest, measuring physical variables, and sharing observations with other players through written messages within the virtual environment. The interaction logs recorded each student's X, Y, and Z coordinates every three seconds, along with the in-game commands used to carry out these actions. The codes (i.e., behavioral constructs) reported in [40] were derived from these log data and corresponded directly to seven observable student actions during the exploration period (see Table 2).

Table 2. Codebook proposed by [40].

Code	Definitions
Non-stopping/ Racing	The student has stopped for less than 6 seconds during the last minute. A stop corresponds to moving less than three blocks during a period of three seconds.
Social Movement	The student was less than 20 blocks from another player during the entire last minute.
Individual Movement	The student was more than 35 blocks away from any other player during the entire last minute.
Point of Interest	The student is inside a point of interest for 10 seconds or more. Every 10 seconds within the Point of Interest trigger this code again.
Talk to NPCs	The student is within 4 blocks or less of an NPC for 10 seconds or more. Every 10 seconds close to the NPC trigger this code again.
Science Tool	The student uses a scientific tool to measure a physical variable.
Scientific Observation	In-game observations in which students describe the virtual world, ask a science-related question, or attempt to comprehend scientific concepts (e.g., "Can humans survive without the moon?").

Whenever a student action met the definition of a code, a new utterance was created and coded under the corresponding category, with each utterance assigned to a single category. When actions could co-occur (e.g., using a tool while visiting a point of interest), two separate utterances were generated.

The threshold for the Non-Stopping code was established by examining the distributions of students' movement speeds and stops, taking into account the three-second sampling interval of the location data. We defined the Non-Stopping code by the number of stops instead of rapid movement indicators because of the nature of the WHIMC learning environment, where students are expected to conduct activities that require brief pauses in movement (i.e., measure physical variables or make observations).

The threshold for the Individual Movement code was based on the maximum in-game field of view (35 blocks). For the Social Movement code, we used a distance of approximately half that value (20 blocks) to capture instances in which students were in close proximity to others, rather than merely observing them from a distance.

Points of Interest refer to specific in-game locations that students are expected to visit. A duration threshold of 10 seconds was selected for this code because 95% of visits to these areas lasted at least that long. Similarly, interactions with NPCs occur through text boxes that automatically appear when students are within 4 blocks of an NPC; this distance was therefore used as the threshold for defining the Talk to NPC code. The 10-second duration threshold for this code follows the same rationale used for the Points of Interest code.

Finally, the Scientific Observations code captures in-game written observations in which students describe aspects of the virtual world, pose science-related questions, or attempt to reason about scientific concepts. The dataset was manually coded by two human coders [40], who initially achieved strong interrater reliability (Cohen's $\kappa > 0.75$) for each of the three subtypes of scientific observations (Description, Questioning, and Reasoning). For the purposes of [40], these subtypes were subsequently aggregated into a single code for analysis, a methodological decision we follow here.

3.3 Statistical Analysis Compared

In this study, we compare a range of metrics to determine how the choice between ARM and ENA-based methods might affect our results. We do so by applying metrics derived from these methods to the coded behavioral data and then comparing those results to the external knowledge and motivational variables. Specifically, for each identified rule or connection, we computed Spearman's rank correlation coefficients (ρ) between the corresponding ENA connection weight or ARM metric and each external education-related measure. This analysis aimed to identify which pattern mining metrics most consistently highlighted patterns associated with constructs assessed using validated knowledge and motivation instruments.

Based on previous research [5, 19], we selected the following ARM and SPM metrics for analysis: *Support*, *Confidence*, *Cosine*, *Lift*, *Added Value*, *Jaccard*, *Conviction*, and the *Phi coefficient*. These metrics were analyzed alongside the *Connection Weights* generated by ENA and ONA.

For ENA and ONA, connections are computed using a moving-window approach with a stanza size of four instances, which is the most commonly used window size and aggregation method in the QE literature [30]. To ensure methodological consistency and facilitate comparability across approaches, the same window size is used to define transactions for ARM and SPM, following prior work that compared these methods [38].

3.3.1 Understanding each Pattern Mining Metric

To examine the similarity of results from different pattern mining methods and metrics, we first assess the associations among all metrics across both ARM- and ENA-based approaches over the full

set of identified patterns. To this end, we employ a mixed-effects linear model comparing metric values across patterns and students. This approach was chosen over simple correlation analyses to account for the non-independence introduced by multiple patterns identified for the same student. In addition, all metrics were rank-transformed prior to analysis to enable the assessment of non-linear associations between them.

Six metrics examined in this study are inherently symmetric. Specifically, *ARM Support*, *Jaccard*, *Cosine*, *Lift*, *Phi*, and *ENA Connection Weights* assign one rule to both X→Y and Y→X. In contrast, three asymmetric metrics (i.e., *Confidence*, *Added Value*, and *Conviction*) can yield different values depending on the rule direction. All SPM and ONA metrics are also inherently asymmetric because they depend on temporal order. To facilitate comparisons between symmetric and asymmetric metrics, we split the symmetric rules into two rules (X→Y and Y→X) with identical numerical values.

To further understand the types of rules emphasized by each pattern mining method and metric, we conducted a complementary analysis. For each method, we identified the ten highest-ranked rules and computed five descriptive statistics for each rule. These statistics included the base rate (*Support*) of the antecedent, consequent, and their co-occurrence; the conditional probability of the consequent given the antecedent (equivalent to *Confidence*); and the ratio of the co-occurrence base rate to the individual base rates (equivalent to *Lift*). Because rules involving a single item naturally exhibit higher overall support and are not represented in ENA, this analysis was restricted to rules involving two distinct items or codes (three or more naturally exhibit lower support and are also not represented in ENA).

3.3.2 Connections with External Measures

For each metric and method, we counted the number of statistically significant correlations between pattern-level metrics (e.g., *Support*, *Confidence*, or *ENA Connection Weights*) and external knowledge or motivational measures. Importantly, the purpose of this analysis was not to interpret individual significant correlations but to compare how frequently different metrics and methods identified associations with external constructs. Accordingly, no correction for multiple comparisons was applied, because the count of significant associations served as a comparative indicator rather than as evidence supporting specific inferential claims.

ARM, SPM, ENA, and ONA differ in the number and types of patterns they can identify (e.g., ENA is limited to dyadic relations), which may inflate the number of significant associations with external measures for methods that generate larger pattern sets, such as ARM and SPM. However, after applying an initial filtering step that removed patterns with *Support* or *Confidence* below 0.05, a standard procedure recommended by Merceron and Yacef [19], all patterns involving rules with more than two codes were eliminated, indicating that such higher-order patterns were not common in this dataset. This result suggests that the observed differences in the number of significant associations across methods are not driven by ARM-based approaches identifying more complex patterns but rather reflect substantive differences in how each metric and method quantifies dyadic patterns.

The only adjustment made to the count of significant associations concerned the symmetric nature of certain metrics. As noted earlier, *ARM Support*, *Jaccard*, *Cosine*, *Lift*, *Phi*, and *ENA Connection Weights* assign identical values to the rules X→Y and Y→X. Consequently, these methods would be expected to yield approximately half as many significant associations as methods that distinguish between X→Y and Y→X (though not exactly half, due to significant correlations involving single-code patterns). For this reason, we treated X→Y and Y→X as separate rules when computing correlations, even for symmetric metrics where their numerical values were identical.

Additionally, patterns involving single events (i.e., marginal probabilities of individual codes) were included. For this type of pattern, ONA provides a comparable representation through self-transitions (X→X). In contrast, ENA and some ARM metrics, such as *Conviction* or *Phi*, do not offer directly analogous metrics for single-code rules, which may slightly reduce the number of significant associations identified by these methods. However, because this limitation is specific to those methods and metrics, and because single-event associations have substantial interpretive value for understanding direct relationships between individual codes and outcomes, these rules were retained.

3.4 Human Ratings of Pattern Importance

A large proportion of our analysis is based on metrics that do not involve human judgement, but instead describe other aspects of the patterns' frequency, reliability, and non-randomness. These metrics, like the other measures in this study, are important for quantifiably describing patterns, but do not necessarily indicate whether those associations are meaningful to expert researchers. In both the EDM and QE communities, the goal is not merely to identify patterns, but to determine which observed patterns support meaningful conclusions [27]. In other words, we are interested in which research methods (or metrics) are most likely to lead to consequential validity [37].

Building on this perspective, we examine which of the statistically significant correlations highlighted by each technique are theoretically meaningful and relevant to researchers. As in previous research that has sought to distinguish ARM metrics of interestingness from human ratings of interestingness [5], we used a five-point Likert scale to evaluate both the *theoretical alignment* and the *perceived interestingness* of each identified correlation. Five educational researchers (all coauthors) independently rated the correlations while blinded to the pattern mining technique and metric that produced each result. All raters evaluated the same set of 203 statistically significant correlations, defined as those identified by at least one pattern mining method or metric. After completing the ratings, raters were also asked to describe their rationale for judging correlations as interesting.

The five ratings for each dimension were aggregated using a 40% trimmed mean, in which the highest and lowest ratings were discarded, and the remaining three were averaged. Trimmed means are a standard robust aggregation procedure that reduces the influence of outliers [36]. Finally, correlations falling in the top 10% for both interestingness and theoretical alignment were examined across pattern mining methods and metrics to determine which techniques identified these correlations and whether the methods agreed on the direction of the associations.

4. RESULTS

4.1 Method-Metric Similarity

We first consider how the observed associations among the different ARM- and ENA-based metrics compare to one another. Overall, the results show a general pattern of positive association across most metrics (Figure 1). However, important differences emerge between patterns that highlight simple frequency and those that account for base rate differences in the components of a pattern.

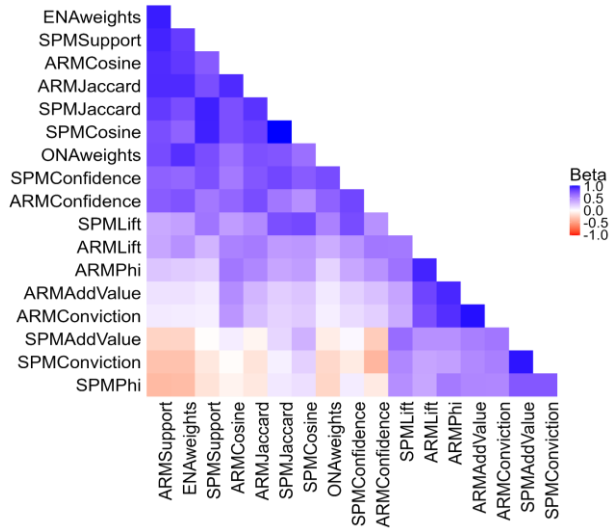


Figure 1. Associations between multiple pattern mining methods and metrics.

In particular, *Support*, *Confidence*, *Cosine*, and *Jaccard* (for both ARM and SPM), as well as ENA and ONA connection weights, are strongly associated with one another, indicating a tendency to highlight similar patterns. In contrast, *Added Value*, *Conviction*, and *Phi* exhibit negative associations with this group of metrics when computed in the context of SPM, and exhibit near-zero associations when computed for ARM. *Lift* occupies an intermediate position, showing moderate associations with *Phi* as well as with *Cosine*, but not reaching the same level of strong association observed within either of the two main clusters of measures. These results show that *Added Value*, *Conviction*, and *Phi* emphasize patterns that differ systematically from those highlighted by the remaining measures.

We further demonstrate the distinctions in these metrics by examining the details of the baseline probabilities associated with each metric. Table 3 reports the average base rates of the antecedent (P(X)), the consequent (P(Y)), their co-occurrence (P(XY)), the conditional probability of the consequent given the antecedent (P(Y|X)), and the ratio (*Lift*) of the joint probability (P(XY)) to the product of the individual probabilities (P(X)P(Y)) for the top 10 rules identified by each method and metric (i.e., the highest-scoring rules per metric). Multiple threshold values were tested (e.g., top 5 rules to top 20), yielding consistent results. Across the combined top-10 lists from the 18 methods and metrics considered, 31 unique patterns were identified, with most appearing among the top-ranked rules for at least 12 methods. Antecedent and consequent base rates ranged from 0.22 to 0.44, co-occurrence probabilities ranged from 0.08 to 0.16, and *Lift* values ranged from 0.96 to 1.28, indicating that different methods emphasize patterns with distinct statistical characteristics.

As expected based on the definitions of these metrics (see [33]), methods that primarily emphasize high-frequency constructs, such as *Support*, *ENA*, and *ONA Connection Weights*, tend to highlight rules in which both the antecedent and the consequent had relatively high base rates. In contrast, metrics such as *Lift*, *Phi*, *Added Value*, and *Conviction* did not emphasize rules composed of high base rate antecedents and consequents. Instead, these metrics tended to highlight rules in which the antecedent and consequent co-occurred in ways that were not primarily driven by their individual frequencies.

Table 3. Base rates and co-occurrence statistics for top-10 rules identified by each method, sorted on joint probability.

Method	Metric	P(X)	P(Y)	P(XY)	P(Y|X)	P(XY) / (P(X)P(Y))
ARM	Support	0.390	0.390	0.157	0.411	1.034
ENA	Weights	0.387	0.387	0.157	0.400	1.047
ONA	Weights	0.407	0.385	0.155	0.381	0.988
SPM	Support	0.410	0.360	0.153	0.373	1.032
ARM	Cosine	0.360	0.360	0.152	0.423	1.167
ARM	Jaccard	0.360	0.360	0.152	0.423	1.167
SPM	Jaccard	0.387	0.354	0.149	0.397	1.085
SPM	Cosine	0.391	0.348	0.149	0.385	1.094
SPM	Confidence	0.343	0.371	0.144	0.421	1.134
ARM	Confidence	0.302	0.439	0.135	0.440	1.016
SPM	Lift	0.337	0.325	0.132	0.396	1.202
ARM	Add. Value	0.314	0.314	0.126	0.393	1.277
ARM	Conviction	0.314	0.314	0.126	0.393	1.277
ARM	Lift	0.314	0.314	0.126	0.393	1.277
ARM	Phi	0.314	0.314	0.126	0.393	1.277
SPM	Phi	0.272	0.305	0.092	0.335	1.102
SPM	Conviction	0.367	0.221	0.081	0.221	1.003
SPM	Add. Value	0.376	0.223	0.081	0.220	0.964

Consistent with its definition, *Confidence* emphasizes rules in which the consequent (Y), but not necessarily the antecedent (X), had a high base rate, reflecting its focus on maximizing the probability of the consequent given the antecedent. Notably, *Cosine*, although conceptually related to *Lift*, appeared to reflect a trade-off among these tendencies, highlighting patterns that fell between these extremes in terms of base rates and conditional relationships relative to the other metrics and methods considered.

4.2 Correlations with External Instruments

One measure of consequential validity is the degree to which the patterns identified in these metrics are associated with meaningful outcome measures. As such, Table 4 summarizes the total number of statistically significant correlations identified between each pattern mining metric and the external knowledge and motivational instruments. Correlations involving knowledge assessments administered at the beginning of the camp were grouped under the Pre-test category. Similarly, correlations involving knowledge assessments collected at the end of the camp were grouped under Post-test, and correlations related to learning gains, computed from multiple knowledge assessments, were grouped under the Learning category.

Across all motivational and knowledge measures, *Support* emerged as the pattern mining metric that identified the largest number of significant associations with the external instruments. Specifically, ARM *Support* and SPM *Support* yielded 143 and 142 significant correlations, respectively. These were followed by *Confidence*, which produced 140 significant correlations for ARM and 121 for SPM. Similarly, *ONA* and *ENA Connection Weights* showed comparatively high numbers of significant associations (127 and 120, respectively), indicating that metrics that do not adjust for code base rates were consistently related to external measures of knowledge and motivation. Notably, across all instruments, no metric outside this set of six produced the highest number of significant correlations, underscoring the consistency of these trends across external measures.

Table 4. Significant correlations identified with each pattern mining metric across all the knowledge and motivational measures.

Method	Metric	Pre Test	Post Test	Learning	Initial Sit. Int.	Final Sit. Int	Self-Efficacy	Astronomy Int.	Minecraft Int.	Total
ARM	Support	20	**44**	37	13	11	**6**	9	**3**	**143**
SPM	Support	20	43	38	13	**12**	4	**10**	2	142
ARM	Confidence	17	**44**	**40**	**16**	10	5	7	1	140
ONA	Weights	17	38	36	13	10	2	9	2	127
SPM	Confidence	15	38	31	14	10	5	5	**3**	121
ENA	Weights	**22**	40	32	8	8	2	6	2	120
SPM	Conviction	14	36	32	14	9	4	6	2	117
SPM	Added Value	12	36	28	11	9	4	5	**3**	108
ARM	Jaccard	18	31	29	7	10	3	5	**3**	106
ARM	Cosine	17	26	32	4	9	3	8	**3**	102
SPM	Jaccard	18	32	30	6	8	2	4	2	102
SPM	Cosine	12	24	29	5	5	2	4	**3**	84
SPM	Phi	9	35	30	1	1	1	1	0	78
ARM	Added Value	11	22	18	1	2	2	3	0	59
ARM	Lift	6	26	18	0	0	4	0	0	54
ARM	Conviction	9	20	19	0	0	1	0	0	49
SPM	Lift	6	18	14	0	3	5	1	1	48
ARM	Phi	10	14	14	0	0	0	0	0	38

In contrast, metrics designed to adjust for base rates or to emphasize non-trivial associations were less consistently associated with the external instruments. Specifically, the *Phi coefficient* (38 significant correlations for ARM and 78 for SPM), *Lift* (54 for ARM and 48 for SPM), and the ARM versions of *Conviction* and *Added Value* (49 and 59 significant correlations, respectively) exhibited substantially fewer associations with knowledge and motivational measures, a pattern that was consistent across all instruments considered. These findings suggest that patterns involving highly frequent codes and their co-occurrences may be more strongly associated with differences in student interest, self-efficacy, prior knowledge, or learning gains, even when such patterns are partly driven by high base rates rather than strong conditional dependence. In other words, adjusting for base rates does not, in this case, appear to produce patterns that are more closely aligned with external educational measures.

It is important to note that symmetric metrics—such as *Support* (for ARM) and *ENA Connection Weights*—do not distinguish between X→Y and Y→X, unlike asymmetric metrics, including SPM-based metrics, *ONA Connection Weights*, and ARM metrics such as *Confidence*, *Added Value*, and *Conviction*. As a result, symmetric metrics inherently represent only half of the distinct directional patterns that can be identified by asymmetric rules. As noted earlier, symmetric rules were therefore counted twice when tallying significant correlations to ensure comparability with asymmetric metrics.

Notably, after applying this adjustment, ARM *Support* and *ENA Connection Weights* yielded totals of significant associations that were nearly identical (and even greater for the case of *Support*) to those obtained using asymmetric metrics such as SPM-based metrics, *Confidence*, and *ONA Connection Weights*. This pattern suggests that incorporating temporal ordering (as in SPM and ONA), or conditional directionality (as captured by *Confidence*), does not necessarily provide substantial additional information beyond what is already captured by co-occurrence alone, at least with respect to associations with external measures. In the context of this data set, the primary relationships between codes and educational constructs appear to be driven more by co-occurrence than by the direction or temporal ordering of events.

4.3 Human Ratings Results

Finally, we examine which patterns were rated as important by expert human judges. Table 5 presents the number of significant correlations identified by each method among the top 10% of correlations rated as most *theoretically aligned* and *interesting* by human raters. Due to tied ratings, this resulted in 38 correlations rated as most *theoretically aligned* (average score ≥ 4.67) and 22 correlations rated as most *interesting* (average score ≥ 4.00). The table also shows how often each metric identified a significant correlation in the direction that raters judged to be *theoretically aligned* or *interesting*. Notably, ratings of *theoretical alignment* differed markedly from ratings of *perceived interestingness* (Spearman's $\rho = -0.781$, $p < .001$) and there was no overlap between the top 10% of correlations identified for the two ratings.

Table 5. Significant associations identified by each metric among highly rated theoretically aligned and interesting patterns. Symmetric rules were counted twice to ensure comparability with asymmetric metrics.

		Theoretical Alignment		*Interestingness*	
Method	**Metric**	**Significant**	**Direction**	**Significant**	**Direction**
ARM	Support	28	28	13	6
ONA	Weights	26	26	10	5
ARM	Confidence	25	25	12	8
SPM	Confidence	25	25	13	9
SPM	Support	25	25	13	5
ARM	Cosine	23	23	8	6
ARM	Jaccard	22	22	9	5
SPM	Cosine	22	22	11	6
SPM	Jaccard	21	21	9	5
ENA	Weights	18	18	8	2
ARM	Lift	9	5	13	10
ARM	Add. Value	8	4	11	10
SPM	Add. Value	18	4	13	9
SPM	Conviction	18	4	15	10
SPM	Lift	6	4	9	7
ARM	Conviction	3	1	5	4
ARM	Phi	6	1	7	7
SPM	Phi	11	1	10	8

Support identified the largest number of correlations with high *theoretical alignment* (N = 28 for ARM and N = 25 for SPM), followed by *ONA Connection Weights* (N = 26), *Confidence* (N = 25 for both ARM and SPM), *Cosine* (N = 23 for ARM and N = 22 for SPM), *Jaccard* (N = 22 for ARM and N = 21 for SPM), and *ENA connection weights* (N = 18). Notably, all correlations highlighted by these methods exhibited a direction that was consistent with both theoretical expectations and raters' judgments. These results indicate that this cluster of metrics, which also showed stronger mutual associations and similar pattern emphasis (see Section 4.1), tends to identify patterns that align closely with theory and human expectations.

In contrast, the second cluster of metrics (*Added Value*, *Conviction*, the *Phi coefficient*, and, to a lesser degree, *Lift*) tend not to highlight rules that were *theoretically aligned*. For example, although *SPM Added Value* and *Conviction* each identified 18 significant correlations that raters judged as *theoretically aligned*, only four of those correlations exhibited the expected direction.

This mismatch between the theoretically expected direction and the actual direction of the associations surfaced by this second cluster of pattern mining metrics helps explain why those associations ranked higher when they were evaluated primarily in terms of *perceived interestingness* rather than *theoretical alignment*. *Added Value* (N = 10 for ARM and N = 9 for SPM), *Conviction* (N = 10 for SPM and N = 4 for ARM), and *Lift* (N = 10 for ARM and N = 7 for SPM) were the metrics that identified the largest number of correlations whose direction aligned with what human raters found interesting, even when those directions did not match theoretical expectations.

All raters agreed that, beyond their specific interest in particular knowledge or motivational measures, one important reason for labeling a pattern as interesting was the surprising direction of the observed association, a finding that helps explain the negative correlation between *theoretical alignment* and *perceived interestingness* in human ratings. For example, one researcher noted: "Things I want to understand (e.g., individual movement followed by racing indicates positive [situational] interest) [were] given a 4 or 5 because I don't know what is happening." As such, metrics such as *Added Value* and *Conviction* appear particularly interesting because they tend to highlight patterns that deviate from researchers' expectations and prompt further inquiry.

However, returning to the specific rule mentioned by the researcher (i.e., Individual Movement → Racing positively correlated with situational interest), this association was positive and statistically significant only when evaluated using *Added Value* and *Conviction* ($\rho = 0.33, p = 0.002$ for both). In contrast, all other metrics in the first cluster (*Support*, *Confidence*, *Jaccard*, *Cosine*, and *ENA* and *ONA Connection Weights*) consistently indicated a negative association. This discrepancy raises an important question: Are the rules highlighted by certain metrics genuinely interesting and meaningful? Or are some pattern mining methods or metrics instead emphasizing misleading associations that are an artifact of their mathematical formulation?

Another result labeled as interesting by the reviewers involved the same behavioral pattern (Individual Movement → Racing) and its positive association with post-test scores. Notably, although this positive association reached statistical significance for only two metrics (*Lift*: $\rho = 0.35, p = 0.03$ and *Added Value*: $\rho = 0.28, p = 0.05$), all pattern mining methods and metrics agreed on the direction of the association. This finding suggests, consistent with the associations among metrics reported in Section 4.1, that pattern mining metrics are not always misaligned, but may also differ in the strength or significance with which they detect the same underlying relationship. In other words, it is perhaps not that any specific metric is "wrong," but rather that greater agreement across metrics increases confidence that an identified correlation or pattern reflects a valid relationship.

This complementarity between metrics helps explain why, although *Added Value*, *Conviction*, and *Lift* often identify correlations that are rated as more interesting, they do not consistently point in the direction that researchers find meaningful. At the same time, nearly all metrics were able to identify at least some correlations that researchers rated as interesting, and all metrics also surfaced correlations whose direction conflicted with researchers' interestingness judgments. These findings indicate that no single metric reliably captures all aspects of what researchers find interesting, and that different metrics emphasize distinct, and sometimes competing, patterns.

5. DISCUSSION AND CONCLUSION

5.1 Overview

Previous work has compared pattern mining metrics with human ratings of perceived interestingness [5]. This study extends that line of research by first examining how metrics from ARM, SPM, ENA, and ONA relate to one another. It then evaluates the consequential validity of each metric by analyzing the patterns they identify with respect to both their associations with external measures of motivation and knowledge and researchers' ratings of *theoretical alignment* and *perceived interestingness*.

Although the metrics analyzed in this study generally highlighted overlapping sets of patterns, our results revealed two broad clusters. The first cluster consists of metrics that tend to emphasize relatively common and theoretically expected patterns, including *Support*, *Confidence*, *Cosine*, *Jaccard*, and *ENA* and *ONA Connection Weights*. In terms of consequential validity, these metrics accounted for the largest number of significant associations with knowledge and motivational outcomes, with *Support*, *Confidence*, and *ENA* and *ONA Connection Weights* proving particularly fruitful. In contrast, the second cluster includes metrics (*Added Value*, *Conviction*, *Phi*, and, to a lesser extent, *Lift*) that place greater emphasis on associations not primarily driven by overall frequency. These metrics tend to reveal patterns that are less theoretically expected and less prevalent overall, but that are, for the same reason, sometimes rated as more interesting by researchers.

5.2 Interpretation

ARM- and ENA-based methods have been widely used to identify the most common event or code sequences in a dataset and their associations with variables of interest in educational research. However, consistent with earlier findings [5, 19], our study shows that different metrics do not always agree and often highlight different patterns. Prior work therefore recommends that researchers proceed with caution, particularly when using metrics that distinguish between patterns driven by high base rates and those that control for base rates to reveal associations beyond statistical independence (e.g., *Cosine*, *Lift*, and *Added Value*; [19]). When metrics disagree, these studies emphasize the need for researchers and instructors to rely on domain knowledge and theoretical judgment when interpreting results.

This emphasis on interpretive value echoes calls in the literature that stress the importance of being able to "close the interpretive loop"—that is, to explain why a particular pattern is associated with

a specific group of students or a given outcome—in order to understand the substantive meaning and real-world implications of observed patterns [27, 29].

Our findings suggest that many patterns judged to be theoretically aligned are driven largely by base rates, raising the question of whether existing theory is truly comprehensive or instead reflects primarily the most frequent patterns, potentially overlooking less common but still meaningful ones. When a pattern appears surprising to researchers, it may simply be incidental. However, if theory has been developed largely from statistical analyses that privilege high base rates, then patterns highlighted by complementary metrics may be particularly valuable for extending and refining theoretical frameworks. From this perspective, moments when different metrics yield associations pointing in opposing directions are especially important. Such discrepancies prompt deeper investigation into what each pattern mining method and metric emphasizes, how the underlying codes, labels, or events are defined, and how researchers, drawing on their theoretical knowledge and intuition [4], can determine whether a result is substantively meaningful or instead an artifact of the mathematical assumptions underlying a given method.

5.3 Limitations and Future Work

An important consideration when interpreting these findings is the granularity of the underlying data. The analyses in this study rely on a coded representation of events derived from a predefined codebook, reflecting a relatively coarse-grained abstraction of learner activity. While this abstraction supports interpretability and alignment with theoretical constructs, it may also mask finer-grained behavioral distinctions that could influence the patterns identified. In datasets with more fine-grained representations (e.g., raw clickstream or low-level interaction logs), the semantic interpretability of resulting patterns depends on how low-level events are defined and contextualized. At the same time, the number of possible itemsets typically increases with a finer granularity, expanding the pattern search space. Consequently, differences in data granularity may influence both the behavior of pattern mining metrics and the types of patterns identified as meaningful or interesting.

As a result, the findings reported here are most directly applicable to contexts in which data have been aggregated into theoretically meaningful categories. The extent to which these results generalize to datasets with different levels of granularity remains an open question. It is plausible that metrics emphasizing frequency (e.g., *Support*, *Confidence*) may identify fewer patterns or prioritize a different set of itemsets and rules in finer-grained datasets, where co-occurrences tend to be sparser and less likely to meet minimum frequency thresholds. In contrast, metrics that control for base rates (e.g., *Lift*, *Added Value*) may still highlight associations that occur infrequently but are stronger relative to their expected occurrence, as they account for the baseline frequency of individual events when evaluating associations. Future work should therefore examine how data representation choices, including levels of granularity and the type of coding scheme, interact with pattern mining methods to shape both statistical results and their interpretability.

Overall, as the field seeks to better understand how empirical methods shape theoretical interpretations (and vice versa), methodological differences represent an important area of inquiry. Our results highlight meaningful differences among metrics commonly used to analyze data patterns in the EDM community and provide empirical evidence regarding which methods may better support consequential validity. Given our findings about which metrics align to human judgments of *theoretical alignment*, future work should further examine the extent to which existing theoretical frameworks might benefit from expansion informed by evidence from metrics that are less likely to produce findings that align with current theory.

Additionally, recent work has proposed the use of *multiverse analysis* [35], in which each plausible analytic decision used to define or refine the problem space gives rise to a separate analysis, or "universe." Although this approach was originally developed to better understand predictive modeling, it could be adapted to incorporate the multiple metrics examined in this study (e.g., ARM and SPM versus ENA and ONA). Such analyses can strengthen confidence in findings that are robust across analytic choices or help clarify why different methods lead to divergent conclusions, thereby informing which approach is most appropriate for a given research question and theoretical framework. Of course, it is rarely feasible to exhaustively explore all possible analytic paths, as illustrated by the fact that our current study does not encompass all available pattern mining approaches. Nevertheless, our results help identify a focused subset of measures (e.g., a small number of carefully chosen metrics from each identified cluster) that could be selected to support such multiverse analyses.

In conclusion, this work compares multiple pattern mining methods and metrics to illustrate how methodological choices can influence findings related to consequential validity, including associations with external learning and motivational constructs as well as expert judgments of theoretical alignment and interestingness. Our results demonstrate that no single metric is sufficient to capture the full complexity of learning phenomena (cf. [32], on explainable AI). Rather than viewing disagreements across metrics as weaknesses to be resolved, we argue that such divergences represent productive tensions that can stimulate deeper inquiry and support theoretical refinement.

6. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (Grant No. DRL2301173). Andres Felipe Zambrano thanks the Ministerio de Ciencia, Tecnología e Innovación and the Fulbright-Colombia commission for supporting his doctoral studies through the Fulbright-MinCiencias 2022 scholarship.

7. REFERENCES

[1] Agrawal, R., Imieliński, T. and Swami, A. 1993. Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 1993), 207–216.

[2] Agrawal, R. and Srikant, R. 1995. Mining sequential patterns. *Proceedings of the Eleventh International Conference on Data Engineering* (1995), 3–14.

[3] Andrist, S., Collier, W., Gleicher, M., Mutlu, B. and Shaffer, D. 2015. Look together: analyzing gaze coordination with epistemic network analysis. *Frontiers in Psychology*. Volume 6-2015, (2015).

[4] Arastoopour Irgens, G. and Eagan, B. 2023. The Foundations and Fundamentals of Quantitative Ethnography. *Advances in Quantitative Ethnography* (Cham, 2023), 3–16.

[5] Bazaldua, D.A.L., Baker, R. and Pedro, M.O.S. 2014. Comparing Expert and Metric-Based Assessments of

Association Rule Interestingness. *Educational Data Mining* (2014).

[6] Boeder, J.D., Postlewaite, E.L., Renninger, K.A. and Hidi, S.E. 2021. Construction and validation of the Interest Development Scale. *Motivation Science*. 7, 1 (2021), 68–82. https://doi.org/10.1037/mot0000204.

[7] Bowman, D., Swiecki, Z., Cai, Z., Wang, Y., Eagan, B., Linderoth, J. and Shaffer, D.W. 2021. The Mathematical Foundations of Epistemic Network Analysis. *Advances in Quantitative Ethnography* (Cham, 2021), 91–105.

[8] Britner, S.L. and Pajares, F. 2006. Sources of science self-efficacy beliefs of middle school students. *Journal of Research in Science Teaching*. 43, 5 (2006), 485–499. https://doi.org/10.1002/tea.20131.

[9] Dubovi, I. and Tabak, I. 2021. Interactions between emotional and cognitive engagement with science on YouTube. *Public Understanding of Science*. 30, 6 (2021), 759–776. https://doi.org/10.1177/0963662521990848.

[10] Gadbury, M. 2024. Addressing Misconceptions in Science through Minecraft. *Proceedings of the 18th International Conference of the Learning Sciences-ICLS 2024, pp. 1370-1373* (2024).

[11] Hwang, G.-J. and Hsiao, C.-L. 2003. A computer-assisted approach to diagnosing student learning problems in science courses. *Journal of Information Science and Engineering*. 19, 2 (2003), 229–248.

[12] Kinnebrew, J.S., Loretz, K.M. and Biswas, G. 2013. A contextualized, differential sequence mining method to derive students' learning behavior patterns. *Journal of Educational Data Mining*. 5(1), (2013), 190–219.

[13] Lane, H.C., Gadbury, M., Ginger, J., Yi, S., Comins, N., Henhapl, J. and Rivera-Rogers, A. 2022. Triggering STEM interest with Minecraft in a hybrid summer camp. *Technology, Mind, and Behavior*. 3, 4 (2022). https://doi.org/10.1037/tmb0000077.

[14] Linnenbrink-Garcia, L., Durik, A.M., Conley, A.M.M., Barron, K.E., Tauer, J.M., Karabenick, S.A. and Harackiewicz, J.M. 2010. Measuring situational interest in academic domains. *Educational and psychological measurement*. 70, 4 (2010), 647–671. https://doi.org/10.1177/0013164409355699.

[15] LoPresto, M.C. and Murrell, S.R. 2011. An astronomical misconceptions survey. *Journal of College Science Teaching*. 40, 5 (2011), 14.

[16] Mabroukeh, N.R. and Ezeife, C.I. 2010. A taxonomy of sequential pattern mining algorithms. *ACM Comput. Surv.* 43, 1 (Dec. 2010). https://doi.org/10.1145/1824795.1824798.

[17] Martinez, R., Yacef, K., Kay, J., Al-Qaraghuli, A. and Kharrufa, A. 2011. Analysing frequent sequential patterns of collaborative learning activity around an interactive tabletop. *EDM 2011 - Proceedings of the 4th International Conference on Educational Data Mining* (Germany, 2011), 111–120.

[18] Marx, J.D. and Cummings, K. 2007. Normalized change. *American Journal of Physics*. 75, 1 (Jan. 2007), 87–91. https://doi.org/10.1119/1.2372468.

[19] Merceron, A. and Yacef, K. 2008. Interestingness Measures for Association Rules in Educational Data. *Educational Data Mining* (2008), 57–66.

[20] Merceron, A. and Yacef, K. 2004. Mining Student Data Captured from a Web-Based Tutoring Tool: Initial Exploration and Results. *Journal of Interactive Learning Research*. 15, 4 (Oct. 2004), 319–346.

[21] Mooney, C.H. and Roddick, J.F. 2013. Sequential pattern mining – approaches and algorithms. *ACM Comput. Surv.* 45, 2 (Mar. 2013). https://doi.org/10.1145/2431211.2431218.

[22] Pechenizkiy, M., Calders, T., Vasilyeva, E. and De Bra, P. 2008. Mining the student assessment data: Lessons drawn from a small scale case study. *Educational Data Mining 2008* (2008).

[23] Perera, D., Kay, J., Koprinska, I., Yacef, K. and Zaïane, O.R. 2009. Clustering and Sequential Pattern Mining of Online Collaborative Learning Data. *IEEE Transactions on Knowledge and Data Engineering*. 21, 6 (2009), 759–772. https://doi.org/10.1109/TKDE.2008.138.

[24] Romero, C., Ventura, S. and García, E. 2008. Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*. 51, 1 (Aug. 2008), 368–384. https://doi.org/10.1016/j.compedu.2007.05.016.

[25] Ruis, A.R., Rosser, A.A., Quandt-Walle, C., Nathwani, J.N., Shaffer, D.W. and Pugh, C.M. 2018. The hands and head of a surgeon: Modeling operative competency with multimodal epistemic network analysis. *The American Journal of Surgery*. 216, 5 (2018), 835–840. https://doi.org/10.1016/j.amjsurg.2017.11.027.

[26] Sadler, P.M., Coyle, H., Miller, J.L., Cook-Smith, N., Dussault, M. and Gould, R.R. 2010. The astronomy and space science concept inventory: development and validation of assessment instruments aligned with the k–12 national science standards. *Astronomy Education Review*. 8, 1 (2010), 010111.

[27] Shaffer, D.W., Collier, W. and Ruis, A.R. 2016. A Tutorial on Epistemic Network Analysis: Analyzing the Structure of Connections in Cognitive, Social, and Interaction Data. *Journal of Learning Analytics*. 3, 3 (Dec. 2016), 9–45. https://doi.org/10.18608/jla.2016.33.3.

[28] Shaffer, D.W., Hatfield, D., Svarovsky, G.N., Nash, P., Nulty, A., Bagley, E., Frank, K., Rupp, A.A. and Mislevy, R. 2009. Epistemic Network Analysis: A Prototype for 21st-Century Assessment of Learning. *International Journal of Learning and Media*. 1, 2 (May 2009), 33–53. https://doi.org/10.1162/ijlm.2009.0013.

[29] Shaffer, D.W. and Ruis, A.R. 2024. Theories All the Way Across: The Role of Theory in Learning Analytics and the Case for Unified Methods. *Theory Informing and Arising from Learning Analytics*. K. Bartimote, S.K. Howard, and D. Gašević, eds. Springer Nature Switzerland. 187–201.

[30] Siebert-Evenstone, A.L., Arastoopour Irgens, G., Collier, W., Swiecki, Z., Ruis, A.R. and Williamson Shaffer, D. 2017. In Search of Conversational Grain Size: Modeling Semantic Structure using Moving Stanza Windows. *Journal of Learning Analytics*. 4, 3 (Dec. 2017), 123–139. https://doi.org/10.18608/jla.2017.43.7.

[31] Sullivan, S., Warner-Hillard, C., Eagan, B., Thompson, R.J., Ruis, A.R., Haines, K., Pugh, C.M., Shaffer, D.W. and Jung, H.S. 2018. Using epistemic network analysis to identify targets for educational interventions in trauma team communication. *Surgery*. 163, 4 (2018), 938–943. https://doi.org/10.1016/j.surg.2017.11.009.

[32] Swamy, V., Radmehr, B., Krco, N., Marras, M. and Käser, T. 2022. Evaluating the Explainers: Black-Box Explainable Machine Learning for Student Success Prediction in MOOCs. *Proceedings of the 15th International Conference on Educational Data Mining* (Durham, United Kingdom, July 2022), 98–109.

[33] Tan, P.-N., Kumar, V. and Srivastava, J. 2002. Selecting the right interestingness measure for association patterns. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2002), 32–41.

[34] Tan, Y., Ruis, A.R., Marquart, C., Cai, Z., Knowles, M.A. and Shaffer, D.W. 2023. Ordered Network Analysis. *Advances in Quantitative Ethnography* (Cham, 2023), 101–116.

[35] Tang, Y., Harvey, E., Yao, C., Yu, R., Kizilcec, R. and Brooks, C. 2025. Understanding Predictive Models of Student Success with a Multiverse Analysis. *Proceedings of the 18th International Conference on Educational Data Mining* (Palermo, Italy, July 2025), 518–525.

[36] Tukey, J.W. 1992. The Future of Data Analysis. *Breakthroughs in Statistics: Methodology and Distribution*. S. Kotz and N.L. Johnson, eds. Springer New York. 408–452.

[37] Winne, P.H. 2020. Construct and consequential validity for learning analytics based on trace data. *Computers in Human Behavior*. 112, (Nov. 2020), 106457. https://doi.org/10.1016/j.chb.2020.106457.

[38] Zambrano, A.F., Baker, R.S., Mehta, S. and Barany, A. 2024. Epistemic Association Rule Networks: Incorporating Association Rule Mining into the Quantitative Ethnography Toolbox. *Advances in Quantitative Ethnography* (Cham, 2024), 3–17.

[39] Zambrano, A.F., Barany, A., Ocumpaugh, J., Nasiar, N., Vandenberg, J., Goslen, A., Esiason, J., Rowe, J. and Hutt, S. 2025. Unlocking Gameplay Insights with Epistemic (Ordered) Network Analysis: Understanding the Potential of Video Games to Foster Authentic Scientific Practices in STEM Education. *Journal of Science Education and Technology*. 31, (Mar. 2025), 1164–1177. https://doi.org/10.1007/s10956-025-10213-4.

[40] Zambrano, A.F., Wei, Z., Ocumpaugh, J., Barany, A., Baker, R.S., Liu, X., Zhou, Y., Paquette, L. and Ginger, J. 2026. Exploring Player Archetypes in a Minecraft-Based Learning Environment. *International Journal of Serious Games*. (2026), 19–40.

Pattern-based Knowledge Component Extraction from Student Code Using Representation Learning

Muntasir Hoq
North Carolina State University
mhoq@ncsu.edu

Griffin Pitts
North Carolina State University
wgpitts@ncsu.edu

Tirth Bhatt
North Carolina State University
tjbhatt@ncsu.edu

Aum Pandya
North Carolina State University
apandya4@ncsu.edu

Andrew Lan
University of Massachusetts Amherst
andrewlan@cs.umass.edu

Peter Brusilovsky
University of Pittsburgh
peterb@pitt.edu

Bita Akram
North Carolina State University
bakram@ncsu.edu

ABSTRACT

Personalized instruction aims to provide learners with support that adapts to their individual knowledge. Discovering and tracing Knowledge Components (KCs) is an important step in building accurate models of student learning. However, KC discovery in computer science education is challenging due to the open-ended nature of programming, wide variability in student solutions, and intertwined use of programming structures in code. We address these challenges with a *pattern-based KC* discovery method that uses a data-driven approach to define KCs as recurring structural patterns in student code that reveal persistent patterns of struggle and mastery in students' solutions. We then evaluate the discovered KCs using expert evaluation and statistical student modeling to demonstrate their effectiveness in capturing student learning and struggles. We propose a framework for modeling students' learning by deriving pattern-based KCs from student code through a three-stage process. First, an attention-based code representation model identifies Abstract Syntax Tree subtrees most relevant to code correctness. Second, a Variational Autoencoder abstracts these subtrees into a smooth latent space, capturing structural similarity across student submissions. Third, the resulting representations are clustered into pattern-based KCs. To assess the effectiveness of pattern-based KCs for modeling students' learning, we adapt the Deep Knowledge Tracing model to incorporate these KCs, demonstrating significant improvements in predictive performance over baseline KT methods. Additionally, the learning curve analysis showed alignment between the derived KCs and learning theory.

Muntasir Hoq, Griffin Pitts, Tirth Bhatt, Aum Pandya, Andrew Lan, Peter Brusilovsky, and Bita Akram. Pattern-based Knowledge Component Extraction from Student Code Using Representation Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 217–228. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040127

Keywords

Knowledge component, Student modeling, CS education, Code analysis, Knowledge tracing

1. INTRODUCTION

Student modeling provides the basis for which adaptive educational systems can track knowledge acquisition, adapt instructional strategies, and provide personalized, targeted support [37]. A central part of student modeling is discovering Knowledge Components (KCs) that students practice during problem-solving. Here, KCs refer to discrete units of acquired knowledge or skill inferred from performance on a set of related tasks [23].

Modeling learning at the KC level involves two steps: task modeling and evidence modeling [3]. An important aspect of task modeling involves specifying the target KCs associated with a given set of learning activities (KC discovery) [37]. Prior work has approached KC discovery through expert-driven design [2], neural approaches that jointly learn latent abstract KCs and their associations with problems from student code while enforcing learning-curve–based constraints [37], and, more recently, Large Language Models (LLMs) that infer candidate KCs directly from instructional content or student work [39, 27, 10]. In evidence modeling, traces of students' problem-solving behavior should be mapped to the discovered KCs to enable effective tracing of their learning progress (KC attribution) [36]. The accuracy of this mapping directly influences the effectiveness of knowledge tracing and learning analytics, and consequently determines how well student knowledge can be modeled and personalized instruction can be delivered.

In domains such as mathematics and chemistry, problem-solving paths are relatively constrained, and KC granularity is often predefined [39]. In computer science (CS) education, however, KC discovery and attribution pose significant challenges due to the context-dependent nature of what constitutes a unit of knowledge, the vast state space of possible solutions to the same problem, the many programming

structures that can represent the same KC, and the interconnected nature of programming patterns [2]. First, there is no agreed-upon ground truth list of KCs for introductory programming courses [37]. Second, the open-ended nature of programming allows for numerous correct implementations; students may solve the same problem using very different structures and strategies, such as recursion vs. iteration or modular decomposition vs. inline logic [33]. Third, even when students use the same programming construct (e.g., a conditional or loop), that construct can serve different conceptual roles depending on context. such as boundary checking, control flow, or decision-making, making it difficult to reliably attribute a single KC. Fourth, the granularity of KCs in programming remains undefined; while low-level constructs like loops and conditionals can be considered fine-grained KCs, high-level KCs often emerge from combinations of these primitives, reflecting more complex problem-solving strategies or conceptual understanding, and can become relevant more over the course of time [22]. Finally, substantial variation in code structure and expression style across student submissions introduces noise into KC attribution, complicating standardized mappings between code and underlying knowledge components. These challenges render expert-driven KC discovery, common in cognitive tutors and intelligent tutoring systems for many fields, and manual KC allocation infeasible for programming education [32].

In response, recent approaches have sought to discover KCs as constructs that students commonly acquire during problem-solving, as reflected in their code submissions over time. Prior work has leveraged code parsers [32, 2], deep learning models [38, 37], as well as LLMs [29, 10] to infer KCs from student code. However, these methods face limitations, including a lack of explainability and limited granularity of discovered KCs, often focusing on programming constructs such as loops or conditionals rather than capturing more nuanced conceptual understanding.

To address these challenges, we propose a framework for automated KC discovery from student programming submissions. Unlike traditional KCs, which are often defined in terms of skill-tags or curriculum objectives, our approach defines KCs as structural patterns, which we introduce as *pattern-based KCs*. These pattern-based KCs are recurring and semantically meaningful subtree patterns extracted from the Abstract Syntax Trees (ASTs) of student code, reflecting programming patterns and algorithmic constructs necessary for correctly solving a programming problem.

Our framework begins with the Subtree-based Attention Neural Network (SANN) [16], trained to predict code correctness while assigning attention weights to relevant AST subtrees [18]. These subtrees are then normalized to remove literal and identifier-specific variance, encoded into a latent space using a Variational Autoencoder (VAE) to generate important patterns from the code and abstract them in a smooth latent space to capture similarities across student submissions. These patterns are then clustered using K-means to form the discovered pattern-based KCs.

To demonstrate how our KC structures can support modeling of student learning, we adapt the Deep Knowledge Tracing (DKT) model [30] to incorporate the discovered pattern-based KCs. Experimental results show that our approach improves student performance predictions when incorporated in the traditional DKT method and produces meaningful learning trajectories, providing a foundation for detecting mastery and guiding instructional decision-making. To further examine the alignment of pattern-based KCs with learning theory, we analyze student learning curves to track mastery development across KCs [32]. Finally, we validate the educational relevance and internal coherence of the discovered KCs through expert evaluation, and illustrate their explainability with a case study tracing a representative KC across multiple problems and student solutions.

The paper has four main contributions. First, we formalize *pattern-based KCs* as a new conceptual knowledge representation in CS education, grounded in recurring and semantically meaningful code structures. Second, we develop a scalable and explainable framework for automatically discovering these KCs from student code submissions via attention-based modeling and representation learning. Third, we demonstrate that integrating pattern-based KCs into a knowledge tracing model significantly improves predictive accuracy of student performance modeling compared to traditional approaches. Fourth, we provide empirical validation of the educational utility and coherence of the discovered KCs through learning curve analysis, expert evaluation, and explainability case studies. These contributions advance explainable knowledge modeling in programming education, enabling more reliable tracking of how students acquire programming skills and providing actionable insights for researchers and educators working on individualized instruction.

2. RELATED WORK

Understanding how learners acquire and apply specific skills is a central goal of educational research, and Knowledge Components (KCs) provide a structured way to represent this learning. We review prior work relating to the conceptualization of KCs and existing methods for their automated discovery, with a focus on programming education.

2.1 Knowledge Components

The concept of Knowledge Components (KCs) is foundational in educational research, particularly within cognitive science and intelligent tutoring systems. KCs are defined as discrete units of knowledge or skills that a learner must acquire to perform a task successfully. They encompass a range of cognitive elements, including concepts, principles, facts, and procedures [23, 32].

The Knowledge-Learning-Instruction Framework [23] further elaborates on KCs by linking them to instructional events and learning outcomes. According to this framework, KCs are the mental constructs that mediate between instructional activities and observable learning behaviors. This perspective underscores the importance of accurately discovering and modeling KCs to enhance instructional design, learning analytics, and develop Intelligent Tutoring Systems [9]. In programming education, defining KCs and determining their appropriate granularity is inherently challenging, as they can range from fine-grained concepts, such as loop initialization, to complex interactions involv-

ing constructs such as nested conditionals within loops, encompassing control structures, data manipulation, and algorithmic strategies. Accurately modeling these components is crucial for developing adaptive learning systems that can provide insights into student knowledge tracing [26], help provide personalized feedback, recommend worked examples, and support to learners [1].

2.2 Knowledge Component Discovery

Automated discovery of KCs from student code has gained attention, aiming to mitigate the labor-intensive nature of manual annotation and to scale personalized education.

Deep learning approaches have been used to model student learning and discover KCs by analyzing code submissions. These AST-based methods leverage the structural representation of code to assign KCs [38, 13, 32]. For instance, the KC-Finder model integrated learning curve theory into its architecture to discover KCs that align with students' learning progression [37]. Although these models can assign KCs, they often suffer from explainability issues, making it challenging to understand the rationale behind KC assignments. Moreover, these approaches may struggle with the semantic understanding of code, especially when different code structures achieve the same functionality, and understanding the granularity and hierarchical nature of different KCs could be challenging.

LLMs have recently been explored for KC discovery due to their ability to understand programming code and generate human-like text. Studies have demonstrated that LLMs can discover KCs from code and problem descriptions [29, 28, 10, 12]. Despite their potential, LLMs can produce overgeneralized KCs, i.e., different conditional structures can not be identified effectively due to a lack of KC taxonomy, or even irrelevant KCs. While manual KC tagging by experts might ensure high accuracy, it is impractical for large-scale applications due to the significant time and effort required [38]. Moreover, inconsistencies among experts can lead to variability in KC definitions, affecting the reliability of the annotations. While automated methods for KC discovery have advanced, challenges remain in balancing accuracy, explainability, and scalability. Our work aims to address these issues by introducing a pattern-based approach that leverages attention mechanisms and clustering techniques to discover meaningful KCs from student code submissions.

3. DATASET

We conduct our work using a publicly available dataset[1] obtained from the CodeWorkout platform [11]. The dataset contains Java programming submissions collected during an introductory CS course offered at a US public university in Spring 2019. In total, the dataset spans 50 programming problems distributed across 5 course assignments, along with students' normalized final exam scores (scaled between 0 and 1). No demographic or personally identifiable information is included in the dataset. Each submission is evaluated based on the number of passed test cases and is assigned a correctness score between 0 and 1. For the purpose of this work, we binarize these labels into correct (all test cases passed) and incorrect (one or more test cases failed) classes. The dataset comprised a total of 47,764 code submissions from 407 students, of which 18,787 were correct submissions and 28,977 were incorrect.

[1] `https://pslcdatashop.web.cmu.edu/Files?datasetId=3458`

4. METHODOLOGY

We aim to model students' learning progress by discovering KCs as structural patterns in code. We refer to these KCs as *pattern-based KCs*, representing actionable and explainable units that capture students' focal programming knowledge and skills. To discover pattern-based KCs from student code submissions, our approach combines (i) fine-grained signals that pinpoint the code segments most critical for program correctness and (ii) context-aware abstraction that groups structurally distinct yet semantically related implementations under the same conceptual pattern. First, we *extract relevant structural patterns from student programs* using SANN [18], an attention-based model designed for program analysis, which applies hierarchical AST processing and attention scores to reveal which subtrees drive predictions of program correctness. However, SANN is trained to separate programs by correctness and does not strictly learn a continuous latent space where structurally similar subtrees can be clustered together. To address this, we *generate context-aware representations of influential patterns* using a VAE. By reconstructing sequences of high-attention subtrees, the VAE (i) injects broader context, capturing how salient patterns interact within the complete program and across submissions, and (ii) regularizes them into a continuous latent manifold that preserves embedding similarity.

This latent representation provides the basis for *clustering related patterns into distinct pattern-based KCs*. To discover these KCs, we train the VAE to generate representative patterns corresponding to focal programming concepts and skills present in the input AST. Due to the randomness involved in generating programming patterns, multiple patterns might represent the same focal concept and skill. Thus, we apply K-means clustering to group similar patterns into clusters that represent distinct pattern-based KCs. K-means is suitable here since its Euclidean distance assumption aligns well with the smooth, isotropic structure of the VAE model's learned latent manifold. These clusters form the set of pattern-based KCs used in our framework. Finally, we *advance* student modeling by integrating the discovered pattern-based KCs into a Deep Knowledge Tracing model [30], enabling more accurate prediction of student performance. Furthermore, to assess educational relevance, we perform learning curve analysis to test whether the KCs align with established learning theory [32].

4.1 Extracting Relevant Structural Patterns from Student Programs

The first stage of the framework focuses on pinpointing the structural elements in student code that are most critical for program correctness. These elements correspond to programming patterns required to meet problem requirements and are often sources of student difficulty. To extract them, we leverage a modified Subtree-based Attention Neural Network (SANN) [18], an explainable architecture that learns from the hierarchical structure of code and identifies logical errors in incorrect implementations. SANN has been pre-

viously validated across different tasks, including program algorithm classification, detecting LLM-generated code, and recommending relevant worked examples [16, 19, 17].

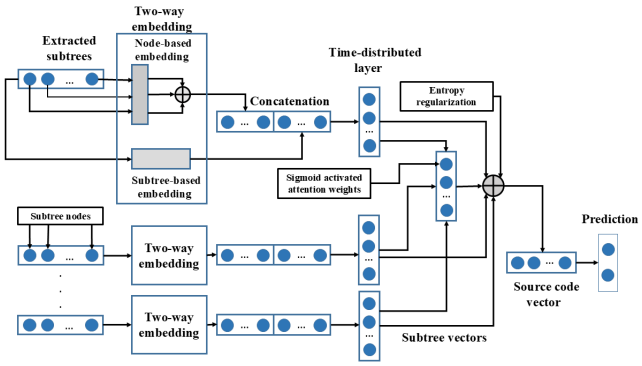


Figure 1: Architecture of the SANN model.

SANN operates on code ASTs by extracting subtrees at multiple depths, which may overlap due to the hierarchical structure of programs. This overlap allows the model to capture coarse-grained constructs (e.g., full conditionals) and fine-grained elements (e.g., comparison expressions) that can independently contribute to correctness [20]. Trained to classify student submissions as correct or incorrect, the model produces attention weights indicating which subtrees most influence predictions (Figure 1). In correct submissions, high-attention subtrees typically represent programming patterns necessary for solving the problem correctly, such as loops, conditionals, or combinations of multiple concepts. In incorrect submissions, high-attended subtrees often reflect logical errors or misconceptions, providing insight into why the submission failed [18].

Within our framework, these high-attention subtrees form the initial candidates for pattern-based KCs. They are carried forward into the next stage, where the VAE abstracts away surface-level variation and organizes them into a latent space suitable for clustering into representative KCs.

4.2 Latent Representation of Candidate KCs

The second stage of the framework uses a β-VAE model [14] to generate latent representations of the most influential programming patterns inferred from student code solutions. These representations form the basis for modeling students' learning over time.

As a first step, we abstract away surface-level syntactic differences, such as specific variable names, literal values, and function identifiers, while preserving the underlying semantic structure. Although such details can be useful for correctness classification, they introduce redundant variation when training the β-VAE and add noise to KC discovery [37]. This abstraction ensures the VAE focuses on the structural and semantic features of code rather than superficial syntactic differences. The normalized ASTs and their associated high-attention subtrees are then used to train the β-VAE model. During training, subtrees are encoded into continuous latent representations and reconstructed, with the model designed so that semantically similar subtrees occupy neighboring regions in latent space. This property enables grouping identified patterns into representative pattern-based KCs. A schematic of the β-VAE architecture is shown in Figure 2.

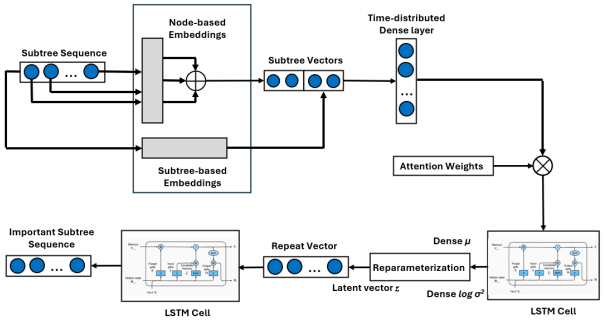


Figure 2: Architecture of the β-VAE model.

To encode subtrees, the VAE model embeds each along two dimensions: the sequence of AST nodes within the subtree and the subtree itself. Node sequences are passed through a time-distributed embedding layer with positional encoding to preserve the relative ordering of nodes within each subtree. This is essential for capturing nuanced structural differences, especially when similar logic is implemented with different nesting or sequencing. The embedded node sequences are then fed into a time-distributed LSTM layer, which compresses each node-level sequence into a fixed-length vector representation.

In parallel, subtree identifiers are embedded and concatenated with the output of the node-level LSTM to form composite embeddings. These combined vectors are then processed by a time-distributed dense layer with a non-linear activation to produce the final subtree representations. To emphasize the most relevant patterns, we scale each vector by the attention weights from SANN, giving greater influence to subtrees most strongly linked to correctness.

The weighted sequence of subtree vectors is passed through an LSTM to capture their sequential order, with dropout applied for regularization. The output is then mapped to two vectors representing the mean, $\boldsymbol{\mu}$, and log-variance, $\log \boldsymbol{\sigma}^2$, of the latent distribution. From these, we sample a latent vector, $\mathbf{z}$, using reparameterization from the distribution:

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, I).$$

To ensure the latent space remains smooth and structured, the KL divergence loss is added, encouraging the distribution to stay close to a standard Gaussian:

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2} \sum_{j=1}^{d} \left(1 + \log \sigma_j^2 - \mu_j^2 - \sigma_j^2\right).$$

The decoder reconstructs the original subtree sequence by repeating the latent vector across the sequence dimension and passing it through an LSTM, followed by a dense output layer with softmax activation. Reconstruction loss is calculated using categorical cross-entropy between the predicted and true subtree tokens:

$$\mathcal{L}_{\text{recon}} = -\sum_{t=1}^{T} \log p(\hat{y}_t \mid \mathbf{z}).$$

The total VAE loss is a weighted combination of reconstruction and KL divergence losses, with a tunable parameter β controlling the trade-off:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \beta \cdot \mathcal{L}_{\text{KL}}.$$

Within the framework, the trained VAE builds on SANN by reconstructing the sequences of high-attention subtrees, capturing broader program context that SANN alone cannot. In doing so, it acts as a denoising compressor, generating representative patterns of important subtrees from correct submissions that preserve the essential aspects of program structure. This design also enables the analysis of incorrect submissions: during inference, sequences of subtrees from incorrect code are passed through the model, and those with high attention weights, identified by SANN, are examined as sources of logical errors [18]. The VAE can then suggest corrected structural equivalents for these faulty segments by leveraging both the local context of the faulty subtrees and the global structure of the code. This mapping facilitates the construction of learning curves based on KC mastery, which in turn enables the evaluation of KC models [32]. In this way, the VAE produces structural KC representations that generalize across variation while preserving conceptual integrity, providing the foundation for defining *pattern-based KCs*. The next step is to consolidate these representations into stable and explainable units of knowledge.

4.3 Clustering Pattern-based KCs

The third stage consolidates the previously generated latent representations into representative pattern-based KCs. These KCs serve as abstractions of the minimal functional substructures required to solve a programming problem, independent of surface-level syntactic differences. Clustering is performed using the encoded subtree vectors from the VAE encoder, which embed normalized high-attention subtrees drawn from correct student submissions. Before clustering, we filter out anomalous student behaviors that could distort KC interpretation. Consistent with prior work [15, 37], we observe that some students exhibit suspicious learning patterns, such as frequent failures on simple problems followed by sudden, consistent success on more advanced problems without intermediate improvement. These cases, which may indicate cheating or code sharing, can skew the KC distribution and are therefore excluded to preserve the integrity of the discovery process.

After filtering, we cluster subtree vectors from correct code submissions, which are expected to reflect structurally sound and semantically valid constructs. K-means clustering is used to group subtrees based on their similarity in the learned latent space, under the assumption that subtrees with nearby encodings represent similar underlying knowledge. The appropriate number of clusters is determined via the elbow method, a commonly used heuristic that identifies the point of diminishing returns in the variance explained by additional clusters.

Each cluster defines a pattern-based KC, represented by its centroid as the canonical example of that KC. High-attention subtrees from both correct and incorrect submissions are then assigned to their nearest centroid, linking each to a representative KC. This enables our proposed framework, which provides a consistent and explainable way

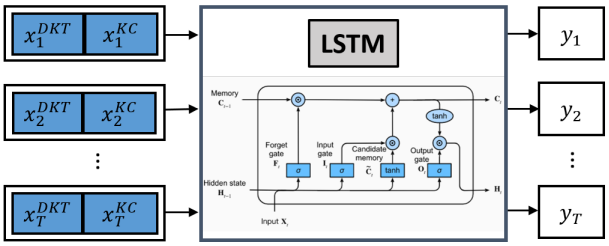


Figure 3: KC-DKT model with pattern-based KC vectors.

to track student learning through code submissions. The presence or incorrect student implementation of a pattern-based KC can be measured across attempts, and the underlying code pattern responsible for the resultant pattern-based KC can be traced back through the pipeline. In doing so, the framework scales automated knowledge modeling in programming education while remaining grounded in structurally meaningful code representations.

4.4 Improving Deep Knowledge Tracing with Pattern-based KCs

We incorporate the discovered pattern-based KCs into the original DKT framework [30] to advance student modeling by providing a more precise representation of learners' knowledge. In DKT, student mastery is modeled as a sequence of interaction events, with the model estimating the probability of success on future tasks. We hypothesize that incorporating more precise KC-level information will improve the performance of DKT and thereby support more accurate tracing of students' learning progress.

Formally, let $S = \{x_1, x_2, \ldots, x_T\}$ represent a sequence of T student interactions. Each interaction x_t corresponds to the student's attempt on a problem at time t, consisting of a problem identifier $q_t \in \{1, 2, \ldots, M\}$ and a correctness label $a_t \in \{0, 1\}$, where M is the total number of unique problems. Following the original DKT formulation, each interaction is encoded as a one-hot vector of size $2M$. Specifically, the element at index $q_t + M \cdot (1 - a_t)$ is set to 1, indicating whether the student answered the problem correctly or incorrectly, and all other elements are set to 0.

In our enhanced model (KC-DKT), the input is extended by incorporating the pattern-based KC mastery vector associated with each submission. This process results in a binary vector of length K, where K is the number of pattern-based KCs. For each submission, this vector encodes the presence (1) or absence/incorrect implementation (0) of each pattern-based KC in the student's solution. The input $\tilde{x}_t$ to the DKT model at time t is constructed by concatenating the original DKT problem-correctness encoding with the KC vector:

$$\tilde{x}_t = [x_t^{\text{DKT}} \parallel x_t^{\text{KC}}].$$

The KC-DKT architecture (see Figure 3), implemented as an LSTM, receives the input sequence $\{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_T\}$ and outputs predictions $\{y_1, y_2, \ldots, y_T\}$, where each $y_t \in \mathbb{R}^M$ gives the probability of successful implementation on each of the M problems. The predicted probability for the next attempted problem q_{t+1} is obtained from the corresponding

entry in y_t. The model is trained using binary cross-entropy loss and evaluated using the Area Under the Receiver Operating Characteristic Curve (AUC) metric to assess performance in distinguishing correct versus incorrect submissions.

4.5 Educational Utility of Pattern-based KCs

Along with improving predictive knowledge tracing, we assess whether the discovered pattern-based KCs are educationally meaningful and suitable for modeling student learning. We evaluate this along three dimensions: learning curve analysis to check alignment with learning theory, an explainability case study, and pedagogical coherence through expert evaluation of KC clusters.

4.5.1 Learning Curve Analysis

Learning curve analysis is a standard approach for evaluating whether inferred knowledge components align with the learning trajectory of student knowledge over time. Following conventions in programming education research [32, 37, 2], we define an *opportunity* as a student's first attempt on a problem that exercises a given KC. Restricting to first attempts reduces the confounding effects of feedback or guessing in later submissions. For each pattern-based KC, we compute the average error rates across increasing opportunity counts and analyze the resulting learning curves. KCs that display a downward error trend with increasing practice are categorized as *good learning*, while flat or erratic curves suggest *still-learning* or *no-learning* [32, 37]. These profiles indicate whether the discovered KCs align with the learning theory and provide a basis for instructional interpretation.

4.5.2 Evaluating Pattern-based KC clusters

To validate whether the discovered pattern-based KCs are pedagogically meaningful and internally coherent, we conducted a structured expert evaluation of the KC clusters. For each cluster, a representative medoid pattern was selected (the subtree closest to the cluster center in latent space) and sampled ten additional patterns from the same cluster. To ensure robustness, five patterns were selected based on proximity to the medoid and five based on proximity to the cluster centroid using cosine similarity.

Three expert evaluators in programming education assessed each cluster. Evaluators first judged whether the medoid represents an important programming pattern that students should learn in introductory programming courses (educational utility). They then compared each candidate pattern to the medoid, judging whether it reflected the same underlying programming idea (conceptual consistency) and how similar the pattern's operation on program state or control flow was to that of the medoid (action-level similarity) on a scale of 1 to 4 (1: Completely different/unrelated action, 2: Related action, with weak relationship, 3: Same action type, minor variation, and 4: Same action type and same role). Evaluators were instructed to focus on conceptual intent and control structure (e.g., branching, iteration, accumulation), while ignoring superficial differences such as variable names, formatting, or literal values.

To establish reliability, all evaluators independently annotated an initial 10% of the clusters. Inter-rater agreement on this subset was perfect for judgments of educational utility and conceptual consistency, and exceeded 0.92 for action-level similarity. After resolving minor disagreements and aligning interpretations, the remaining clusters were divided among evaluators and annotated independently. This protocol provides an external validation of both the educational relevance and coherence of the pattern-based KCs.

4.5.3 Explainability of Pattern-based KCs

To examine the explainability of the discovered pattern-based KCs, we conducted a case study tracing a representative KC across multiple problems and student submissions. Using the projection of attention-weighted subtrees back onto source code, we annotated the portions of students' code associated with the KC in both correct and incorrect attempts. This procedure enabled us to follow how a single KC manifested in successive student submissions and across different problems, providing a basis to evaluate whether the KCs capture consistent and reusable conceptual patterns.

5. EXPERIMENTS

We evaluated the proposed framework by training each component in sequence and examining its ability to discover meaningful pattern-based KCs from student code. The experiments proceeded in three stages: training the modified SANN model to identify code substructures most relevant to program correctness, learning a latent representation of these subtrees with a β-VAE, and clustering the resulting embeddings into representative KCs. The resulting set of KCs was then used for downstream analyses, including integration into a KC-enhanced DKT framework and learning curve evaluation. On average, each programming problem was associated with about 24 KCs (23.68, SD = 6.02), each student submission involved roughly 4 KCs (3.87, SD = 1.56), and each KC was observed across approximately 25 problems (25.36, SD = 17.28), indicating broad coverage of programming constructs throughout the dataset.

In the first stage, we trained the modified SANN model to classify Java submissions as correct or incorrect while assigning attention weights to AST subtrees. The full dataset, consisting of 18,787 correct and 28,977 incorrect submissions across 50 problems, was split into training, validation, and test sets with an 80:10:10 stratified split to preserve label proportions. The model was trained using a code embedding size of 128 with a maximum of 100 nodes per subtree and 100 subtrees per program, for up to 100 epochs with an early stopping patience of 20. We used the Adamax optimizer with a learning rate of 0.001 and minimum entropy regularization weight of 3×10^{-5} to guide the attention mechanism without compromising classification performance. The trained model achieved high scores across all evaluation metrics: accuracy (0.87), precision (0.86), recall (0.88), and F1-score (0.87), confirming its ability to capture correctness-related structural patterns. Attention weights and associated subtrees were extracted for the next stage.

In the second stage, we filtered subtrees with attention weights above 20%, following the original study of this model [18]. From this filtering, we obtained a total of $78,647$ important subtrees from correct submissions and $132,658$ important subtrees from incorrect submissions. Identifier tokens were normalized to reduce surface variation, and a β-VAE was trained exclusively on high-attention

subtrees from correct code to ensure that the latent space reflected valid and pedagogically meaningful structures. The VAE used a subtree embedding size of 64, latent dimension 128, intermediate dense layers matching the embedding size, dropout rate 0.2 (to prevent overfitting), and was trained for up to 30 epochs (patience 10) with Adamax optimizer (lr $= 0.001$) and $\beta = 1 \times 10^{-2}$ to balance reconstruction fidelity with latent space regularization. The VAE achieved 95% reconstruction accuracy across all tokens, though only 10% when considering important subtrees in isolation. This lower score was expected, since the objective was not perfect reconstruction but learning a latent embedding space where structurally similar subtrees are mapped to nearby regions [14]. In practice, the VAE compressed $11,953$ unique subtree patterns into 106 unique representative outputs, demonstrating its ability to compress and abstract fine-grained structural variations. To validate semantic alignment, we compared cosine similarity scores between predicted and input subtrees using CodeT5 (pretrained programming language model, 220 million parameters) embeddings [40], against a randomized baseline. Our proposed approach achieved a significantly higher mean similarity score of 0.96 (standard deviation 0.03) compared to the randomized baseline's mean similarity of 0.85 (standard deviation 0.04) between the input and predicted representative subtrees. A statistical test confirmed that the difference between the two distributions was significant with a *p-value* of 0.0001.

In the final stage, we clustered the VAE-generated subtree vectors into distinct pattern-based KCs using K-means. Only high-attention subtrees from correct submissions were considered, ensuring that clusters reflected valid programming patterns. The elbow method indicated an optimal cluster count of $k = 50$, balancing granularity with interpretability. Each subtree vector was assigned to its nearest cluster center, with the centroid serving as the representative pattern for that KC. Both correct and incorrect subtrees were then mapped to their closest KC, enabling downstream analyses of student performance at the KC level. This process produced a set of 50 structurally coherent and pedagogically grounded KCs extracted directly from student code, providing a foundation for the subsequent integration into our KC-DKT model and learning curve evaluation.

6. RESULTS

In this section, we evaluate the proposed pattern-based KC framework from three complementary perspectives. First, we examine whether the discovered KCs improve predictive student modeling through knowledge tracing. Second, we assess whether the KCs exhibit learning dynamics consistent with learning theory via learning curve analysis. Finally, we validate the educational relevance and internal coherence of the KC clusters through an expert evaluation.

6.1 Deep Knowledge Tracing

To demonstrate that our discovered pattern-based KCs improve on previous approaches to modeling and predicting student learning over time, we integrated them into the original DKT model [30]. Unlike previous approaches that used raw code vectors as input to the DKT model [18, 35], our *KC-DKT* model incorporated a mastery vector that encodes, for each of the 50 discovered pattern-based KCs, whether it was correctly demonstrated in a student's submission.

Table 1: Knowledge tracing AUC (%) averaged over 10 runs with different random splits. *($p < 0.01$), **($p < 0.001$) indicate significant improvements of KC-DKT over DKT.

Model	A1	A2	A3	A4	A5	All
BKT	61.6	57.5	54.1	56.3	53.2	56.5
DKT	71.2	73.1	76.8	69.2	75.1	65.9
Code-DKT	74.3	76.6	80.4	72.8	79.1	-
KC-DKT	**82.4****	**77.6***	**80.8***	**82.0****	**85.4****	**74.8***

Our KC-DKT model was trained to predict whether the student would succeed on the next attempt based on their KC state history. To make the evaluation fair, we split the dataset at the student level using an 80:10:10 ratio for training, validation, and testing, ensuring no overlap between students across splits. Hyperparameter optimization was performed using Keras Tuner. The best performing configuration used an LSTM layer with 512 hidden units, a dropout rate of 0.1, and a learning rate of 0.01. Our KC-DKT model using pattern-based KC vectors achieved an AUC of 74.77%, outperforming the original DKT baseline [30], which achieved an AUC of 65.87%. Assignment-wise breakdowns of AUC scores are provided in Table 1, along with a comparison with Bayesian Knowledge Tracing (BKT) [4] and Code-DKT [35], which uses AST-derived code vectors for single assignments.

These results highlight the strength of our pattern-based KC representation for knowledge tracing. The approximately 13% improvement over the original DKT represents a substantial gain relative to previously reported extensions of knowledge tracing models that incorporate code-level representations, which have typically yielded smaller or more incremental improvements in their respective evaluation settings [35, 18, 8]. Having demonstrated predictive advantages, we further examine whether the discovered KCs align with learning theory by analyzing student performance trajectories through learning curve analysis.

6.2 Learning Curve Analysis

We next evaluated the discovered pattern-based KCs using learning curve analysis to test whether the KCs align with expected trends from learning theory [37, 32, 12, 2]. A learning curve reflects improvement over repeated opportunities to practice a KC, typically showing a decline in error rate as mastery increases. Following prior work in knowledge tracing [34, 4], we considered only the first submission of each student per problem to avoid confounds from debugging or superficial edits. For each KC, we then computed the average error rate across sequential opportunities, where an opportunity was defined as a student's encounter with code involving that KC.

We observe that the learning curves showed three qualitatively distinct trends [32, 12] (Figure 4). The majority, 35 out of 50 pattern-based KCs, showed a steady decline in error rates, which we denote as *good learning*. The learning curves exhibit the expected behavior of a steady decline, indicating that students gradually acquire mastery over these

patterns through repeated exposure, with these KCs appearing in an average of 20 problems each (see Figure 4a).

A second group of clusters, 10 out of 50 pattern-based KCs, labeled *high error/still learning*, also trended downwards but showed fluctuations in later opportunities before eventually decreasing (see Figure 4b). This trend likely reflects the difficulty and recurrence of these patterns across problems. Many of these KCs involve complex conditional structures, such as nested conditionals and complex Boolean expressions, appearing in about 30 problems per KC.

Finally, 4 clusters fell into the *no learning* category, where error rates remained high across opportunities (see Figure 4c). These patterns often combined multiple structural elements or appeared in highly variable set of problems (mean number of problems per KC: 37). Such "no learning" curves are not unique to our approach but occur in KC modeling more generally, reflecting challenges in reliably measuring genuine learning progress amid noise [32]. Apart from these categories, one additional cluster lacked sufficient opportunities to generate a reliable curve.

Overall, the learning curve analysis demonstrates that our pattern-based KCs capture meaningful differences in how students acquire and apply programming patterns. Unlike traditional knowledge tracing over predefined skills, this approach tracks mastery of recurring structural patterns directly from student code, revealing whether students steadily improve, struggle intermittently, or persistently fail on specific programming patterns. These findings reinforce the validity of the KC model and motivate our ablation study presented in the following section.

6.3 KC Model Comparison and Ablation Study

To further evaluate the effectiveness of our proposed framework, we tested its ability to explain and predict student performance within the Additive Factors Model (AFM) [7, 24]. The AFM estimates the probability of a correct response as a function of KC-specific intercepts and learning rates by fitting logistic learning curves. Following prior research [37, 2], we compared the performance of our KC model against two common baselines: the single-KC model, which treats all submissions as practicing the same KC, and the item-KC model, which assigns each problem its own KC. Model performance was evaluated using three metrics: Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), and Root Mean Squared Error (RMSE). Lower AIC and BIC indicate better fit, with BIC applying a stronger penalty for model complexity, while RMSE reflects average prediction error.

We also conducted an ablation study to isolate the contribution of our combined approach, including the SANN and β-VAE pipeline. Two additional baselines were tested: (i) replacing the VAE model by using embeddings from a large programming language model, jina-code-embeddings (1.5b parameters) [25], (ii) replacing the VAE model with a fine-tuned CodeT5 model (pretrained programming language model, 220m parameters) [40], using its encoder output in the clustering step and (iii) removing the VAE entirely and clustering directly on the SANN-generated embeddings.

Table 2: KC model comparison (↓ = lower is better).

Model	AIC ↓	BIC ↓	RMSE ↓
Single KC	60,327.41	60,344.95	0.50
Item KC	57,496.26	58,372.99	0.48
jina-code-embeddings	60832.70	61704.36	0.48
CodeT5-based KC	57,300.44	58,155.97	0.47
SANN-based KC	47,941.18	48,737.84	0.40
Pattern-based KC	**39,319.73**	**40,196.47**	**0.35**

As shown in Table 2, our pattern-based KC model outperformed all baselines. It achieved lower AIC and BIC values, indicating better model fit with appropriate complexity, and a lower RMSE, demonstrating improved predictive accuracy of student performance. These results suggest that our framework provides a more accurate and effective representation of KCs compared to traditional skill-attribution or direct embedding strategies. We next examine the explainability of pattern-based KCs, focusing on how the framework surfaces the code structures underlying each discovered KC.

6.4 Pattern-based KC Cluster Evaluation

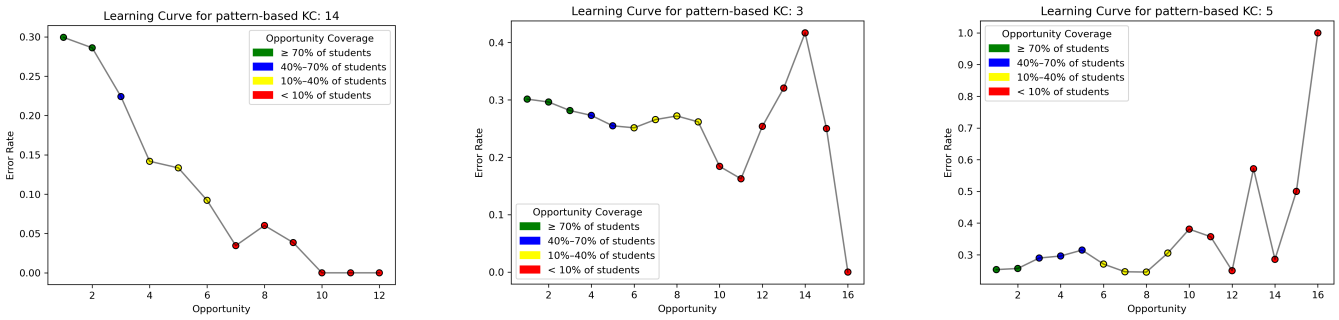
While predictive performance and learning curve behavior provide indirect evidence of KC quality, it is also important to verify that the discovered clusters correspond to educationally meaningful and internally coherent programming patterns. To this end, as described in Section 4.5.2, we conducted an expert evaluation of 35 pattern-based KC clusters, which showed a clear learning trend. The expert evaluation results indicate that the discovered pattern-based KCs are both educationally meaningful and internally consistent. All evaluated cluster medoids (100%) were judged to represent important programming patterns appropriate for introductory programming instruction. These medoids commonly captured reasoning patterns such as compound conditionals, guarded logic, accumulator updates, and loop–condition interactions, suggesting that the clustering process consistently surfaces instructionally relevant concepts rather than trivial or boilerplate code fragments.

Cluster consistency was also high. Across all clusters, 97.7% of candidate patterns were judged to express the same underlying programming idea as their corresponding medoid, despite syntactic variation and differences in problem context. In addition, candidate patterns showed strong similarity in how they manipulated program state or control flow, with an average action-level similarity score of 3.7 (std = 0.65) on a 1–4 scale. The high mean is expected, as candidate patterns were sampled from within-cluster neighborhoods around each medoid, and the goal of this evaluation was to assess cluster coherence.

Altogether, the proposed framework yields stable pattern-based KCs. The resulting clusters align with important programming constructs and form coherent units for downstream analyses, including knowledge tracing, learning curve evaluation, and explainable feedback generation.

6.5 Explainability of Pattern-based KCs

A central goal of our framework is not only to model performance but also to make visible the reasoning behind predictions, toward actionable feedback. To do so, we project



(a) Good learning (b) High error / still learning (c) No learning

Figure 4: Representative learning curves for 'good learning', 'high error/still learning', and 'no learning' trends. Node colors indicate opportunity coverage: the percentage of students who encountered the KC at a given opportunity.

```
public int sortaSum(int a, int b)
{
    int answer = 0;
    if (a + b < 10 && a + b > 19)
    {
        answer = a + b;
    }
    else
    {
        answer = 20;
    }
    return answer;
}
```

(a) Attempt 1 (incorrect, sortaSum)

```
public int sortaSum(int a, int b)
{
    int answer = 0;
    if ((a + b) < 10 && (a + b) > 19)
    {
        answer = a + b;
    }
    else
    {
        answer = 20;
    }
    return answer;
}
```

(b) Attempt 2 (incorrect, sortaSum)

```
public int sortaSum(int a, int b)
{
    int answer = 0;
    if ((a + b) < 10 || (a + b) > 19)
    {
        answer = a + b;
    }
    else
    {
        answer = 20;
    }
    return answer;
}
```

(c) Attempt 3 (correct, sortaSum)

```
public boolean in1To10(int n, boolean outsideMode)
{
    if (outsideMode)
        if (n <= 1 || n >= 10)
            return true;
        else
            return false;
    else
        if (n >=1 && n <= 10)
            return true;
    else
            return false;
}
```

(d) Correct, int1To10

Figure 5: KC 42 across problems. (a–c) show three sequential attempts on the *sortaSum* problem, where KC 42 is incorrectly (a–b) and then correctly (c) implemented. (d) shows a correct implementation of the same KC in the *int1To10* problem.

attention-weighted subtrees back onto source code and label each span with its associated pattern-based KC. This process reveals the syntactic constructs driving the model's decisions, allowing for precise diagnosis of misconceptions.

Figure 5 illustrates this process using pattern-based KC 42, which governs the correct use of compound comparison and logic operators in conditional expressions. In a sequence of three attempts on the *sortaSum* problem, a randomly selected student first mis-implements KC 42, misapplying `&&` instead of `||` (Figures 5a, 5b); the model highlights the faulty condition in red. In the third attempt (Figure 5c), the student corrects the logic, and the highlight persists, indicating the KC remains relevant while now being mastered. Figure 5d shows another instance of KC 42 in a correct submission for the *int1To10* problem, where students must reason about boundary conditions. Despite the different problem context, the conditional again reflects KC 42, validating its conceptual consistency across problems and representing how our framework discovers reusable conceptual patterns that extend beyond individual assignments.

These visualizations show how our framework links model predictions to code structures, making visible the constructs that drive performance outcomes. They also demonstrate the cross-problem consistency of pattern-based KCs, showing that the same conceptual pattern can be recognized across different assignments. In this way, the framework not only clarifies individual predictions but also represents reusable knowledge structures that make KC tracking more generalizable and pedagogically meaningful. Having established the validity and utility of pattern-based KCs, we discuss their implications for student modeling, programming education, and future research in the following section.

7. DISCUSSION

This work set out to address the challenge of defining and modeling knowledge components in introductory programming, a domain where solution paths are highly variable and traditional skill lists are difficult to specify. We introduced *pattern-based KCs*, defined as recurring structural patterns in student code that represent the programming constructs involved in solving problems.

To operationalize this definition, we developed a pipeline to discover pattern-based KCs from student code submissions. Using the SANN model, we identified AST substructures most relevant to code correctness. These substructures were then abstracted with a variational autoencoder (VAE) to capture structural similarities across submissions and clustered into representative KCs. The resulting set of pattern-based KCs advances student modeling by enabling

knowledge tracing at a more fine-grained level of learner behavior. Our proposed KC-DKT model improved predictive performance by more than 13% over the original DKT model, and learning curve analysis further confirmed that the discovered KCs align with established learning theory. In addition, an expert evaluation of the clusters found that all medoid patterns were judged educationally useful and that within-cluster samples showed high conceptual and action-level consistency, providing evidence that the extracted KCs correspond to coherent instructional units.

Our approach offers several advantages over traditional methods. The integration of SANN provides transparency by enabling visualization of attention weights on specific code structures, allowing educators to understand and inspect each KC. Moreover, each KC can be traced back to the original student code patterns, supporting explainable, transparent practices for AI in education. This stands in contrast with existing deep learning and LLM-enabled approaches for KC discovery, which often operate as black boxes for KC definition and attribution. Additionally, clustering of the VAE-generated AST substructures captures a wide range of programming patterns, from simple granular patterns to complex logic. This breadth reflects the diversity of student solution strategies and ensures that the resulting KCs capture the full spectrum of patterns students use in practice, enabling more precise modeling of student knowledge in CS education.

These findings have direct implications for adaptive learning systems. By accurately modeling student mastery at the KC level in submitted code, our approach facilitates real-time feedback on specific coding patterns, enabling personalized learning approaches in CS education [9]. For example, with the identification provided by our approach, if a student consistently struggles with a particular loop construct, an adaptive learning system can provide targeted exercises or resources associated with that pattern to address this gap [1]. Pattern-based KCs can also inform content recommendation engines, guiding the selection of problems [6] or worked examples [21] that align with their current knowledge and learning needs.

Beyond individual feedback, aggregated KC-level data can help instructors identify which patterns are most challenging across a cohort, informing curriculum adjustments and instructional focus. Over time, this integration of KC analytics into adaptive platforms can create a feedback loop between student performance data and instructional design, ensuring that learning environments adapt to address the most persistent knowledge gaps. In this way, our work connects automated KC discovery with practical instructional use, providing an explainable and scalable framework for modeling student learning in CS education.

8. LIMITATIONS AND FUTURE WORK

The results demonstrate the effectiveness of discovering pattern-based KCs from AST structures with attention-based modeling, although there are limitations that can be addressed through future research. First, because the dataset does not include demographic or background information, we are unable to analyze potential differences in KC distributions or learning trajectories across student subgroups; examining fairness and generalizability across populations is an important direction for future work. Second, our method relies on successful AST parsing, which excludes uncompilable submissions (15% of the dataset) and removes potentially valuable data related to misconceptions or syntax errors [37, 18]. Future work could incorporate error-tolerant parsers or automated syntax-error repair tools that can handle uncompilable code [31].

Moreover, our evaluation followed standard practice in knowledge tracing by focusing on first submission attempts to avoid noise from feedback-driven debugging. While this reduces bias, it also limits the ability to study repeated practice effects for KCs. Modeling post-first-attempt behavior, and distinguishing genuine skill development from surface-level fixes [5], is an important next step. Another limitation of the current work is that all code representations are learned solely from the CodeWorkout dataset, without leveraging large pretrained code models or large language models (LLMs) to warm-start token or node embeddings. While this design choice allows the discovered pattern-based KCs to remain tightly grounded in course-specific student behavior, it also constrains the representational capacity of the model, particularly given the relatively modest size of the dataset. Pretrained code embeddings have been shown to capture broad syntactic and semantic regularities across diverse programming contexts, which could improve generalization and stability of learned representations. An important direction for future work is therefore to initialize node- or subtree-level embeddings using pretrained code representations (e.g., from LLMs or large-scale code models), followed by fine-tuning on course-specific student data. More broadly, the framework could benefit from a two-stage training strategy in which models are pretrained on large, heterogeneous programming corpora and then adapted to individual courses or assignments, enabling the discovery of both general programming patterns and course-specific KCs. Hybrid pretraining–fine-tuning approaches may further improve the robustness, scalability, and transferability of pattern-based KC discovery.

Another important direction for future work should be the generation of natural language descriptions for the discovered pattern-based KCs. Currently, KC representations exist as structural subtrees, which may not be easily interpretable by novice learners or instructors. Using the natural language generation capabilities of large language models, future work could generate human-readable explanations or summaries of each KC. Further, integrating these KCs into a recommendation system for CS1 courses that provides targeted practice problems and worked examples could enable our framework to model student knowledge and actively support learning in real time. This would further enhance the accessibility, pedagogical utility, and feedback generation potential of our framework in educational settings.

9. CONCLUSION

In this work, we presented a framework for discovering pattern-based KCs from student programming submissions by combining the SANN model, a variational autoencoder, and clustering. We evaluated the resulting pattern-based KCs through integration with DKT, which produced significant improvements in predictive performance, and through

learning curve analysis, which confirmed their alignment with learning theory. Our approach advances knowledge modeling in CS education by providing scalable, structurally grounded representations of student understanding. Beyond predictive modeling, these KCs also create opportunities for targeted practice, adaptive feedback, and personalized instructional support in programming education.

10. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation (NSF) under Grants #2418658, #2418655, and #2418657. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of NSF.

11. REFERENCES

[1] G. Abdelrahman, Q. Wang, and B. Nunes. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11):1–37, 2023.

[2] R. S. d. Alencar, M. A. Demirtas, A. S. Saha, Y. Shi, and P. Brusilovsky. Integrating expert knowledge with automated knowledge component extraction for student modeling. In *Proceedings of the 33rd ACM International Conference on User Modeling, Adaptation and Personalization (UMAP)*, pages 307–312, 2025.

[3] M. Arieli-Attali, S. Ward, J. Thomas, B. Deonovic, and A. A. Von Davier. The expanded evidence-centered design (e-ecd) for learning and assessment systems: A framework for incorporating learning goals and processes within assessment design. *Frontiers in Psychology*, 10:1–17, 2019.

[4] A. Badrinath, F. Wang, and Z. Pardos. pybkt: An accessible python library of bayesian knowledge tracing models. In *Proceedings of the 14th International Conference on Educational Data Mining (EDM)*, pages 468–474, 2021.

[5] R. Baker. Gaming the system: A retrospective look. *Philippine Computing Journal*, 6(2):9–13, 2011.

[6] R. S. d. Baker, A. T. Corbett, K. R. Koedinger, S. Evenson, I. Roll, A. Z. Wagner, M. Naim, J. Raspat, D. J. Baker, and J. E. Beck. Adapting to when students game an intelligent tutoring system. In *Proceedings of the Intelligent Tutoring Systems: 8th International Conference, ITS*, pages 392–401, 2006.

[7] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 164–175, 2006.

[8] Y. Choi, Y. Lee, D. Shin, J. Cho, S. Park, S. Lee, J. Baek, C. Bae, B. Kim, and J. Heo. Ednet: A large-scale hierarchical dataset in education. In *Proceedings of the International Conference on Artificial Intelligence in Education*, pages 69–73, 2020.

[9] A. T. Corbett, K. R. Koedinger, and J. R. Anderson. Intelligent tutoring systems. In T. K. L. M. Helander and P. Prabhu, editors, *Handbook of Human-Computer Interaction*, pages 849–874. Elsevier Science B.V., Amsterdam, 1997.

[10] Z. Duan, N. Fernandez, S. Kanakadandi, B. Akram, and A. Lan. Automated knowledge component generation and knowledge tracing for coding problems. *arXiv preprint arXiv:2502.18632*, 2025.

[11] S. H. Edwards and K. P. Murali. Codeworkout: short programming exercises with built-in data collection. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, pages 188–193, 2017.

[12] J. Fan, T. Mihaylova, B. Akram, N. Norouzi, P. Brusilovsky, A. Hellas, and J. Leinonen. Adaptive learning curve analytics with llm-kc identifiers for knowledge component refinement. In *Proceedings of the 2025 Conference on UK and Ireland Computing Education Research*, pages 1–7, 2025.

[13] R. Filgueira and D. Garijo. Inspect4py: A knowledge extraction framework for python code repositories. In *Proceedings of the 19th International Conference on Mining Software Repositories*, pages 232–236, 2022.

[14] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–22, 2017.

[15] M. Hoq, P. Brusilovsky, and B. Akram. Explaining explainability: Early performance prediction with student programming pattern profiling. *Journal of Educational Data Mining*, 16(2):115–148, 2024.

[16] M. Hoq, S. R. Chilla, M. Ahmadi Ranjbar, P. Brusilovsky, and B. Akram. SANN: Programming code representation using attention neural network with optimized subtree extraction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 783–792, 2023.

[17] M. Hoq, A. Patil, K. Akhuseyinoglu, P. Brusilovsky, and B. Akram. An automated approach to recommending relevant worked examples for programming problems. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education (SIGCSE) V. 1*, pages 527–533, 2025.

[18] M. Hoq, A. Rao, R. Jaishankar, K. Piryani, N. Janapati, J. Vandenberg, B. Mott, N. Norouzi, J. Lester, and B. Akram. Automated identification of logical errors in programs: Advancing scalable analysis of student misconceptions. In *Proceedings of the 18th International Conference on Educational Data Mining (EDM)*, pages 90–103, 2025.

[19] M. Hoq, Y. Shi, J. Leinonen, D. Babalola, C. Lynch, T. Price, and B. Akram. Detecting chatgpt-generated code submissions in a cs1 course using machine learning models. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education*, page 526–532, 2024.

[20] M. Hoq, J. Vandenberg, B. Mott, J. Lester, N. Norouzi, and B. Akram. Towards attention-based automatic misconception identification in introductory programming courses. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2*, pages 1680–1681, 2024.

[21] R. Hosseini, K. Akhuseyinoglu, P. Brusilovsky, L. Malmi, K. Pollari-Malmi, C. Schunn, and T. Sirkiä. Improving engagement in program construction

examples for learning python programming. *International Journal of Artificial Intelligence in Education*, 30(2):299–336, 2020.

[22] Y. Huang, P. Brusilovsky, J. Guerra, K. Koedinger, and C. Schunn. Supporting skill integration in an intelligent tutoring system for code tracing. *Journal of Computer Assisted Learning*, 39(2):477–500, 2023.

[23] K. R. Koedinger, A. T. Corbett, and C. Perfetti. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, 36(5):757–798, 2012.

[24] K. R. Koedinger, E. A. McLaughlin, and J. C. Stamper. Automated student model improvement. In *Proceedings of the 5th International Conference on Educational Data Mining*, pages 17—-24, 2012.

[25] D. Kryvosheieva, S. Sturua, M. Günther, S. Martens, and H. Xiao. Efficient code embeddings from code generation models. *arXiv preprint arXiv:2508.21290*, 2025.

[26] Y. Liang, T. Peng, Y. Pu, and W. Wu. Help-dkt: an interpretable cognitive model of how students learn programming based on deep knowledge tracing. *Scientific Reports*, 12(1):4012–4013, 2022.

[27] S. Moore, R. Schmucker, T. Mitchell, and J. Stamper. Automated generation and tagging of knowledge components from multiple-choice questions. In *Proceedings of the eleventh ACM conference on learning@ scale*, pages 122–133, 2024.

[28] R. Niousha, M. Hoq, B. Akram, and N. Norouzi. Use of large language models for extracting knowledge components in cs1 programming exercises. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2*, pages 1762–1763, 2024.

[29] R. Niousha, A. O'Neill, E. Chen, V. Malhotra, B. Akram, and N. Norouzi. Llm-kci: Leveraging large language models to identify programming knowledge components. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2*, pages 1557–1558, 2025.

[30] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Proceedings of the 29th International Conference on Neural Information Processing Systems*, page 505–513, 2015.

[31] G. Pitts, A. Pandya, D. Rank, T. Bhatt, M. Hoq, and B. Akram. Automated program repair of uncompilable student code. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.2*, pages 1483–1484, 2026.

[32] K. Rivers, E. Harpstead, and K. R. Koedinger. Learning curve analysis for programming: Which concepts do students struggle with? In *Proceedings of the 12th ACM Conference on International Computing Education Research (ICER)*, volume 16, pages 143–151, 2016.

[33] K. Rivers and K. R. Koedinger. Data-driven hint generation in vast solution spaces: a self-improving python programming tutor. *International Journal of Artificial Intelligence in Education*, 27(1):37–64, 2017.

[34] D. Selent, T. Patikorn, and N. Heffernan. Assistments dataset from multiple randomized controlled experiments. In *Proceedings of the 3rd ACM Conference on Learning @ Scale*, pages 181–184, 2016.

[35] Y. Shi, M. Chi, T. Barnes, and T. Price. Code-dkt: A code-based knowledge tracing model for programming tasks. In *Proceedings of the 15th International Conference on Educational Data Mining*, pages 50–61, 2022.

[36] Y. Shi, M. Chi, T. Barnes, and T. Price. Evaluating multi-knowledge component interpretability of deep knowledge tracing models in programming. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 288–295, 2024.

[37] Y. Shi, R. Schmucker, M. Chi, T. Barnes, and T. Price. Kc-finder: Automated knowledge component discovery for programming problems. In *Proceedings of the International Conference on Educational Data Mining (EDM)*, pages 28–39, 2023.

[38] Y. Shi, R. Schmucker, K. Tran, J. Bacher, K. Koedinger, T. Price, M. Chi, and T. Barnes. The knowledge component attribution problem for programming: Methods and tradeoffs with limited labeled data. *Journal of Educational Data Mining*, 16(1):1–33, 2024.

[39] C. S. Tan and J.-J. Kim. Automated math word problem knowledge component labeling and recommendation. In *Proceedings of the International Conference in Methodologies and Intelligent Systems for Technology Enhanced Learning*, pages 338–348, 2024.

[40] Y. Wang, W. Wang, S. Joty, and S. C. Hoi. Codet5: Identifier-aware unified pre-trained encoder-decoder models for code understanding and generation. *arXiv preprint arXiv:2109.00859*, 2021.

Personalization in Educational Data Mining and Learning Analytics: A Systematic Review (2015 - 2025)

Sylvio Rüdian
Humboldt-Universität zu Berlin
Department of Computer Science & Education
Berlin, Germany
ruediasy@hu-berlin.de

ABSTRACT

Personalization is widely recognized as a path toward more effective digital learning; however, its conceptual and methodological treatment varies across different research traditions. This study systematically synthesizes a decade of work (2015 - 2025) in the Educational Data Mining (EDM) and Learning Analytics & Knowledge (LAK) communities employing the PRISMA framework to clarify how personalization is enacted and evaluated. We compile and screen 503 papers to isolate those that implement learner-level adaptation, coding what is personalized, the indicators that drive it, and how studies assess its impact. A mixed quantitative-qualitative analysis compares the two communities, highlighting distinct emphases, recurring methodological challenges, and complementary strengths. The review provides an integrated map of personalization research, identifies persistent gaps in validity, adoption, and causal evidence, and outlines opportunities for cross-fertilization between technically focused EDM models and human-centered LAK practices.

Keywords

Adaptive Systems, Intelligent Tutoring System, Review

1. INTRODUCTION

Personalization has long been one of the most compelling promises of digitally mediated learning: the idea that learning material, feedback, and support can be adapted to meet the needs of individual learners rather than delivered as one-size-fits-all instruction. Over the past decade, this promise has been intensively investigated by two closely related research communities: Educational Data Mining (EDM) and Learning Analytics and Knowledge (LAK). They share many data sources and venues but often differ in theoretical commitments, methodological preferences, and the artifacts they produce [13]. As algorithmic techniques for sequential decision making and representation learning matured, and as institutions deployed analytics tools to support teachers and students, the term "personalization" became diverse. Yet it is used to describe a wide variety of interventions, from reinforcement-learning policies that select the "next best" activity to dashboards that help instructors tailor feedback. Without a careful, comparative map of how EDM and LAK operationalize personalization, it is challenging to synthesize evidence, identify gaps, or chart a coherent course of action.

The similarities and differences between LAK and EDM have been extensively examined in the scholarly literature. Both fields operate within the educational domain, leveraging data collection and analysis to improve learning processes and student outcomes. They share a commitment to data-driven approaches for understanding and enhancing education, frequently utilizing overlapping datasets and tasks such as predicting student performance or identifying at-risk learners [22]. Both LAK and EDM exploit large-scale educational data, employ statistical and machine learning techniques, and aim to transform raw data into actionable insights for educators, students, and administrators. Core methodologies such as classification, clustering, and prediction are widely used in both areas, reflecting their shared origins in the computational analysis of educational environments [32].

Despite these commonalities, LAK and EDM differ in methodological orientation and epistemological emphasis. EDM has traditionally focused on the development of novel computational techniques to interrogate educational data, often emphasizing model-driven approaches such as association rule mining, clustering, and predictive modeling. In contrast, LAK integrates a broader range of methods, including visualization, social network analysis, and statistical modeling, while maintaining a strong emphasis on the practical application of analytics to enhance learning and teaching. LAK scholarship frequently highlights the human and organizational dimensions of education. The two fields are therefore proximate yet complementary: EDM has historically concentrated on model- and policy-centric questions - such as how to represent latent knowledge, predict academic success, or optimize instructional decisions under uncertainty - whereas LAK has invested heavily in human-centered analytics, focusing on making data interpretable and actionable for learners, teachers, and institutions through feedback, visualization, and participatory design. Both perspectives are indispensable for advancing credible personalization at scale. Algorithmic policies lacking usability, transparency, or alignment with pedagogical practice are unlikely to succeed in authentic educational contexts, while well-designed

Sylvio Rüdian. Personalization in Educational Data Mining and Learning Analytics: A Systematic Review (2015 - 2025). In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 229–242. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039837

dashboards that lack empirically validated adaptive mechanisms risk becoming well-intentioned but ineffective advice systems. A comparative perspective thus illuminates areas of conceptual convergence, points of productive methodological divergence, and opportunities for cross-fertilization that could accelerate progress in the field. Recent bibliometric studies further indicate that LAK typically pursues research with the explicit intent of improving learning actions and environments, whereas EDM is more strongly oriented toward understanding learning behaviors through algorithmic analysis [32]. However, a focused comparison of how the two communities conceptualize and operationalize personalization remains absent.

2. FROM ADAPTATION TO PERSONALIZATION

Personalization is widely invoked in digital learning, yet the term is used inconsistently across research traditions and is frequently conflated with adjacent concepts such as adaptation and customization. Because this paper compares personalization work across EDM and LAK, we require a definition that is conceptually defensible, broad enough to cover both communities' artifacts, and precise enough to support coherent interpretation of results.

Because our aim is to compare two communities whose personalization artifacts range from fully algorithmic decision policies (e.g., sequencing and recommendation) to teacher- and learner-facing tools (e.g., dashboards and feedback systems), we adopt a definition that is simultaneously inclusive and boundary-setting. Specifically, we require a framing that (a) is broad enough to cover both automated and human-in-the-loop instantiations of individualized support, (b) excludes changes that are applied uniformly at the cohort or course level (i.e., adaptation without differentiation between learners), (c) distinguishes system-driven personalization from user-initiated customization, and (d) is operationalizable for transparent screening and coding. This emphasis on codeability matters for a comparative review: without explicit decision rules, the term "personalization" risks collapsing into a rhetorical umbrella, making it difficult to interpret patterns across venues or to attribute differences to real methodological or conceptual divergence rather than to definitional drift.

We begin with the general notion of *adaptation*. In dictionary terms, adaptation denotes modifying an artefact, process, or environment so that it suits a new purpose or situation, and an *adaptive* system is one that can change in response to varying conditions [86]. These definitions are intentionally broad: adaptation can occur at multiple levels (individual, subgroup, course, or institution) and can be enacted by humans, algorithms, or both. In educational contexts, this breadth is reflected in uses of "adaptation" that range from optimizing the fit between learning requirements and course content [61] to broader forms of adjustment such as shifting instructional delivery modes or transferring courses across settings [34]. Such interventions involve change in response to context, but they are not necessarily individualized.

Personalization is a narrower, person-centered form of adaptation. Building on the Oxford framing, we use personalization to refer to tailoring content, processes, or interfaces to the needs of a particular individual [86]. The key differentiator is the *target of the adaptation*: personalization aims to differentiate the learning experience between learners rather than to adjust an experience uniformly for a cohort. This individualized focus is prominent across multiple strands of the literature. Work on adaptive hypermedia and web-based learning has long framed personalization as selecting and presenting information based on learner models [17, 18]. In online course settings, personalization is often instantiated as individualized sequencing or learning-path construction, where the system selects activities or resources based on learner prerequisites or performance [90, 8]. In the broader discourse of "personalized learning", personalization is frequently described as optimizing pace and approach to the learner, encompassing learning objectives, instructional strategies, and the sequencing of content [85, 8]. Related accounts emphasize motivational and competency dimensions [37] or characterize personalization along practical axes such as goals, path, time, place, and pace [10]. Although these formulations vary in granularity and pedagogical emphasis, they converge on the principle that instructional decisions should depend on learner-specific information rather than being uniform for all learners.

A second distinction concerns *locus of control*. Whereas personalization and adaptation are commonly enacted system-side (often implicitly through models or rules) *customization* refers to changes initiated by users based on explicitly specified preferences [86, 93]. Customization can contribute to individualized experiences, but it differs from personalization in who specifies the change and how it is triggered. This distinction is especially relevant for learning systems, where personalization is frequently implemented without explicit learner requests (e.g., through adaptive sequencing, recommendations, or tailored feedback).

This paper uses *personalization* to denote **individual-level, person-centered adaptation** of the learning experience, distinguished from broader adaptation by its learner-specific target and from customization by its system-side locus of control. This definition supports a consistent interpretation of the diverse mechanisms labeled as personalization in EDM and LAK, ranging from sequencing and recommendation to dashboards and feedback systems, while avoiding conflation with non-individualized course-level adjustments.

3. THIS REVIEW

This review provides a structured, side-by-side account of personalization research in EDM and LAK from 2015 to 2025. Rather than treating "personalization" as a rhetorical label, we treat it as an operational construct: an intervention that adapts some aspect of the learning experience to an individual learner (or to explicitly defined learner subgroups) using learner-specific information. The contribution of the paper is therefore integrative rather than algorithmic: it offers an auditable comparative map of how personalization is instantiated across two influential communities, the types of signals used to drive it, and the kinds of evidence used to justify impact. We do not aim to produce a pooled effect size of "whether personalization works"; instead, we clarify what counts as personalization in this literature and characterize where each community concentrates its efforts

and how strongly those efforts are tied to outcome-focused evaluation.

Building from the operational definition, we distinguish (i) what is personalized (e.g., task sequencing, recommendations, feedback, dashboards, assessment criteria), (ii) which indicators are used to drive adaptation (behavioral traces, knowledge-state estimates, text, demographics/context, affect/motivation), and (iii) which techniques implement it (reinforcement learning, BKT / IRT and related cognitive models, recommender algorithms, supervised ML, deep learning, visual analytics, rules, and qualitative / UX / co-design). Because personalization claims often risk conflating usage with learning, we also attend to evaluation designs and explicitly differentiate studies that measure effects on learning outcomes from those that report only process, engagement, or usability, treating process measures as outcomes only when a study provides an explicit validity argument that they are learning-proximal in context.

Two additional considerations motivate a careful synthesis at this time. First, the proliferation of powerful modeling techniques, contextual bandits and deep reinforcement learning, sequence-aware recommenders, and transformer-based approaches for text and multimodal traces, has increased the feasibility of fine-grained adaptation while raising the stakes around fairness, privacy, and explainability. Second, institutions and platforms are deploying analytics at scale, creating opportunities for rigorous evaluation (e.g., pragmatic trials, stepped-wedge designs) while also exposing persistent gaps in construct validity and generalizability across courses and populations. A review that connects what is being personalized to how it is decided and what evidence supports it can help the field move beyond isolated successes toward dependable practice.

Our contribution is threefold. First, we build a transparent corpus by identifying LAK and EDM papers that mention personalization (2015–2025), then screening for studies that are substantively about personalization rather than merely naming it. This keeps the review focused on instantiated mechanisms rather than aspirations. Second, we extract a common set of variables from the included studies, definitions, targets of personalization, indicators, techniques, contexts and scales, evaluation designs, and reported limitations/future work, and organize papers into interpretable clusters defined by the object of personalization, yielding community-specific profiles that can be compared directly. Third, we use those profiles to answer a compact set of research questions that address where each community focuses its personalization efforts, which signals it relies on, and what gaps remain most salient for future work.

We structure the inquiry around three research questions that bridge description and critique:

RQ1: In studying personalization, where do EDM and LAK place their main focus, and how do those foci differ?

RQ2: Which learner- and context-level indicators are used to drive personalization in EDM and LAK?

RQ3: What research gaps and future directions emerge across the two communities?

Answering these questions yields both a map and a mirror: a map of the conceptual and methodological terrain of personalization across two influential communities, and a mirror in which each community can see its strengths and blind spots. By anchoring personalization in concrete targets and signals, and by contrasting EDM and LAK along those axes, the review aims to move the conversation from whether personalization "works" in the abstract to how we can design effective, equitable, and adoptable personalization in practice.

4. METHODOLOGY

4.1 Review design and reporting framework

We conducted a structured literature review of personalization research in the LAK and EDM conference proceedings for the years 2015–2025. Our goal is to map how personalization is instantiated (RQ1), what signals are used to drive it (RQ2), and what limitations/future directions are reported (RQ3). In terms of review type, this work aligns most closely with a *scoping review* (i.e., aiming to characterize and map a body of work rather than estimate an effect size). To improve transparency and reproducibility, we report our search, screening, and selection steps in a PRISMA-compatible structure (flow counts and explicit eligibility criteria). Where PRISMA terminology is used, it is adopted as reporting guidance rather than implying a registered SLR protocol.

4.2 Information sources and search strategy (high-recall)

We searched the official proceedings and hosting digital libraries for LAK and EDM for each year in scope. To maximize recall while keeping the query auditable, we used a single high-recall lexical stem centered on `personali*`. This stem captures spelling variants and common morphological forms, including (non-exhaustively) *personalization*, *personalisation*, *personalized / personalised*, *personalizing / personalising*, and *personalizability / personalisability*. We executed the query separately for each venue-year and community. Overall, we record the authors, identifier / DOI, year, venue, title, and abstract, and exported all data to a CSV and concatenated into a raw "intake" file. Each row corresponds to one retrieved record prior to de-duplication and screening.

4.3 Eligibility criteria (inclusion/exclusion)

We predefined inclusion and exclusion criteria before screening and applied them consistently throughout.

Inclusion criteria - A record was eligible if it:

- is a paper in LAK or EDM proceedings within 2015 - 2025 (excluding workshop-only volumes if not part of the main proceedings, unless explicitly in scope);
- contains an instantiated personalization/adaptation mechanism (see operational definition below).

Exclusion criteria - A record was excluded if it:

- is not a LAK/EDM proceedings paper in scope (e.g., abstract-only items, workshops);
- mentions personalization only rhetorically (e.g., as motivation or aspiration) without an implemented adaptive mechanism.

We define personalization as an *intentional adaptation* of content, sequence/path, feedback, support, interface, or assessment to individual learners (or explicitly defined learner subgroups) using learner-specific information (data, models, or explicit rules). Mere parameter tuning at the population level, generic "one-size-fits-all" recommendations, or aspirational statements without a realized adaptive mechanism do not qualify.

4.4 Screening and selection process

Screening proceeded in two stages and was documented in a dedicated workbook sheet with validation rules.

Stage 1: about-personalization decision - We screened each unique record to determine whether it is substantively about personalization under the operational definition above. Records were labeled `Include`, `Exclude`, or `Unclear`. Unclear cases were resolved via discussion using decision notes and adjudicated conservatively (i.e., inclusion required evidence of an instantiated adaptive mechanism).

Stage 2: learning-outcomes evaluation decision - In Stage 2, we focus on outcome evidence because this review aims to synthesize support for learning impact claims, i.e., whether personalization changes what learners know or can do, rather than only whether an intervention is used, accepted, or engaging. Process characteristics (e.g., clicks, time-on-task, persistence, navigation patterns, or strategy indicators) are often meaningful as mediators or early warning signals, but they are not consistently identifiable with learning across contexts and can be driven by novelty, interface constraints, or compliance effects. Given the field's well-known risk of conflating "more activity" with "more learning", we treat process measures as learning outcomes only when a study provides an explicit validity argument that the process metric is learning-proximal in that context (e.g., validated linkage to performance/competency or a preregistered/justified surrogate endpoint). Otherwise, process measures are coded as indicators or intermediate outcomes rather than learning outcomes. For records included after Stage 1, we assessed whether the study measured effects on learning outcomes. We treated learning outcomes as changes in knowledge, skill, or performance evidenced by assessments (pre/post or delayed), mastery probabilities, grades explicitly tied to competencies, validated performance tasks, or completion when justified as competency-proximal. Engagement, usability, satisfaction, or time-on-task were not counted as learning outcomes unless explicitly validated within the study design as learning-proximal endpoints.

PRISMA-style accounting of records - To support reproducibility, we report record counts at each step:

- Records retrieved by searches in EDM and LAK: **503** (LAK: $n = 370$, EDM: $n = 133$)
- Records excluded in Stage 1 (not personalization): **269** (LAK: $n = 226$, EDM: $n = 43$)
- Records included after Stage 1: **234** (LAK: $n = 144$, EDM: $n = 90$)
- Records excluded in Stage 2 (no learning-outcome effects measured): **174** (LAK: $n = 117$, EDM: $n = 57$)
- Records included after Stage 2 (learning outcomes measured): **60** (LAK: $n = 27$, EDM: $n = 33$)

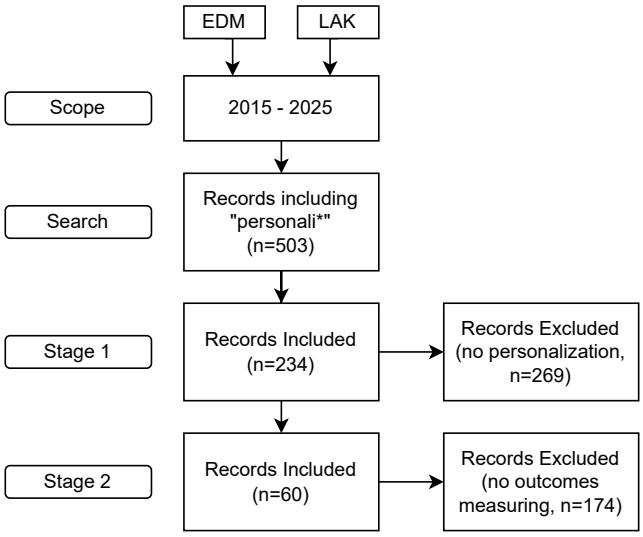


Figure 1: Inclusion and Exclusion of Records.

These counts correspond to the workbook filters and enable reconstruction of the selection process. Fig. 1 visualizes the accounting of records.

4.5 Data extraction, codebook, and coder reliability

Workbook and codebook - We maintained a multi-sheet Excel workbook as the study database. A `Dictionary` sheet defined each variable and its admissible values; `Metadata` stored bibliographic fields and provenance; `Screening` captured inclusion decisions with validation rules and inconsistency flags; and `Coding` contained analytical variables with controlled vocabularies (e.g., object of personalization, indicator families, techniques).

Coding variables - For all papers included after Stage 1, we extracted variables aligned with RQ1–RQ3. For RQ1, we coded the *object(s) of personalization* as non-exclusive categories (e.g., content, sequence/path, feedback, interface, support, assessment) and recorded domain and scale descriptors (participants and/or interaction records). For RQ2, we coded the indicators used to drive personalization (multi-select), such as knowledge-state estimates, performance history, demographics / learner profile, outcomes, behavioral traces, or social / context signals. For RQ3, we transcribed limitations and future-work statements as close paraphrases and coded them into thematic categories.

Coders and disagreement resolution - Coding was performed by **6** coders. To ensure consistency, we conducted a pilot on

30 papers to refine the codebook and decision rules. Each paper was coded by **two independent coders**. Disagreements were resolved through discussion. When ambiguous, both codings have been included. We assessed reliability on a double-coded subset of **40** papers using **Cohen's** $\kappa = .78$ for key categorical variables (Stage 1 decision, object of personalization, indicator families). Agreement was **Cohen's** $\kappa = .84$ in stage 2 and informed final refinements to the codebook prior to full coding.

4.6 Analysis

Analysis proceeded in three coordinated steps mapped to the research questions. For RQ1, we treated the object-of-personalization codes as the primary organizing axis and computed category distributions separately for EDM and LAK, yielding community-specific profiles of where personalization efforts concentrate. Because these object categories directly reflect how adaptation is instantiated, this approach supports straightforward validity checks and avoids the opacity of unsupervised topic modeling. To enrich the profiles, we examined within-category co-occurrence of indicators.

For RQ2, we compared the prevalence of indicator families across communities both overall and conditional on object category (e.g., whether knowledge-state estimates co-occur more frequently in EDM than in LAK, or whether LAK draws more heavily on learner profiles and social/context signals). For RQ3, we synthesized coded limitations and future-work themes to identify common gaps and directions, reported overall and stratified by venue and object category where informative.

In summary, this review identifies where EDM and LAK concentrate their personalization efforts (RQ1), which signals they rely on (RQ2), and which gaps are most salient for future research (RQ3).

5. RESULTS

Based on how each paper operationalizes personalization and the target goal to which personalization is optimized, we identified seven non-exclusive personalization types:

- **Learning Path (Learning Material)**: Adaptive selection and sequencing of course materials, activities, and difficulty to guide each learner's path.
- **Recommender System (Course, External Resources)**: Recommendations for courses/programs and external resources (videos, articles, apps) tailored to learners' goals and performance.
- **Dashboard**: Personalized analytics/visualizations that surface progress, risks, and suggested actions to support self-regulated learning.
- **Feedback/Support/Notifications**: Tailored hints, explanations, formative feedback, reminders, and intervention messages triggered by learner state.
- **Instruction**: Adaptation of instructional strategies and supports (e.g., tutoring policies, scaffolding, differentiated activities, teacher support allocation).
- **Scheduling**: Personalized timing and pacing - practice/test scheduling, spacing and review intervals, and ordering of assignments.
- **Collaboration/Forum**: Social personalization - peer / mentor recommendations, forum thread suggestions, group-based differentiation, and support for collaborative activities.

Papers were allocated to the personalization types as shown in the extract of Table 1. The entire table with all codings can be found at `https://doi.org/10.5281/zenodo.20023758`[79]. Six papers were excluded from type-based analyses because they could not be linked to any of the seven types ([92, 73, 51, 45, 48, 76]). Fig. 2 illustrates the absolute numbers of personalization-related papers clustered by personalization type, including all included papers (top) and the subset measuring learning outcomes (bottom). Absolute counts are shown to reflect overall research output, given the larger volume of LAK papers in stage 1, rather than emphasizing relative proportions.

5.1 RQ1: Where do EDM and LAK place their main focus, and how do those foci differ?

To test whether EDM and LAK differ in their emphasis across personalization types, we first applied Levene's tests to determine whether equal variances could be assumed. We then used independent samples t-tests when Levene's tests were non-significant and Welch tests otherwise.

Because RQ1 (and RQ2) involve comparing multiple categories, we interpret the p-values jointly rather than as independent tests. We therefore report exact p-values and effect sizes for transparency and interpret exploratory / descriptive. Given the review's descriptive / scoping aim and the non-independence of categories, we did not apply a strict family-wise correction (e.g., Bonferroni), which would be overly conservative and increase Type II error in an already power-limited setting, especially for the learning-outcomes subset ($n = 60$).

As a sensitivity check, we note that the key community differences in the full corpus (recommender systems, dashboards, and feedback / support with $p < .001$) would remain statistically significant under common false-discovery-rate control (e.g., Benjamini–Hochberg at $q = .05$), whereas marginal findings should be interpreted as exploratory and are not relied upon for central claims.

Considering all personalization-related papers ($n = 234$), Levene's tests were significant for learning path, recommender system, dashboard, and feedback/support ($p \leq .001$). Levene's tests for the remaining types were also significant ($p \leq .05$). The two-sided p-values of the Welch tests were below .001 for recommender system, dashboard, and feedback / support, indicating statistically significant differences between communities. For the remaining types, p-values were above .05, indicating no statistically significant differences at $p \leq .05$. Effect sizes were medium to large for feedback / support (Cohen's $d = -.616$), medium for recommender systems (Cohen's $d = .526$) and dashboards (Cohen's $d = -.477$), and small for learning path (Cohen's $d = .274$).

Table 1: Personalization types and indicators of LAK and EDM papers, which include personalization effects on the learning outcome (extract).

EDM/LAK		Year	Personalization Type							Indicators										
			Learning Path	Recommender System	Dashboard	Feedback/Support/Notifications	Instruction	Scheduling	Collaboration/Forum	Outcomes & Achievement	Knowledge/Mastery State	Assessment & Item Properties	Temporal Pacing & Workload	Behavioral Interaction	Strategy, Process & Help-Seeking	Learner Profile & Context	Social & Community Signals	Content/Curriculum	Model/Policy Signals & Diagnostics	Work Products & Artifact Features	Records & Administrative Data
EDM	[58]	2017	-	-	-	-	-	-	✓	-	-	-	✓	✓	-	✓	✓	-	-	-	-
EDM	[78]	2017	✓	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-
EDM	[69]	2017	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-
EDM	[92]	2018	-	-	-	-	-	-	-	-	-	-	-	✓	-	✓	-	-	-	-	-
EDM	[73]	2019	-	-	-	-	-	-	-	-	-	-	✓	✓	-	✓	-	-	-	-	-
EDM	[50]	2019	✓	-	-	-	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-
EDM	[87]	2019	✓	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-
EDM	[53]	2020	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	✓	-	-	-
EDM	[31]	2020	-	-	-	-	-	✓	-	-	✓	-	✓	-	-	-	-	-	-	✓	-
EDM	[56]	2020	-	✓	-	-	-	-	-	✓	✓	-	✓	-	-	✓	-	-	✓	-	-
EDM	[83]	2020	✓	-	-	-	-	-	-	✓	✓	-	-	-	-	✓	-	✓	-	-	-
EDM	[81]	2021	-	-	-	-	-	✓	-	✓	-	-	✓	✓	-	✓	✓	-	✓	-	-
EDM	[21]	2021	✓	-	-	-	-	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-
EDM	[66]	2021	-	-	-	✓	-	-	-	-	-	-	-	-	-	✓	-	-	✓	✓	-
EDM	[44]	2022	-	✓	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-
EDM	[74]	2022	✓	-	-	-	✓	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-
EDM	[80]	2023	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-
EDM	[60]	2023	-	-	-	✓	-	-	-	-	-	-	✓	✓	-	-	-	-	-	✓	-
EDM	[7]	2024	✓	-	-	-	-	-	-	✓	-	✓	✓	-	-	-	-	-	-	-	-
EDM	[67]	2025	-	-	✓	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
EDM	[71]	2025	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-
EDM	[89]	2025	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
EDM	[72]	2025	-	-	-	✓	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-
LAK	[26]	2017	-	-	✓	✓	-	-	-	-	-	-	✓	✓	-	✓	✓	-	-	-	-
LAK	[48]	2017	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-
LAK	[76]	2018	-	-	-	-	-	-	-	-	-	-	-	✓	-	✓	✓	-	-	-	-
LAK	[64]	2019	-	-	-	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
LAK	[59]	2020	-	-	✓	-	-	-	-	✓	✓	-	-	-	-	-	-	-	-	-	-
LAK	[68]	2020	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LAK	[54]	2021	-	-	-	✓	-	-	-	✓	-	-	-	✓	-	✓	-	-	-	-	✓
LAK	[57]	2021	-	✓	-	-	-	-	-	✓	-	-	-	✓	✓	✓	✓	-	-	-	✓
LAK	[70]	2021	-	-	-	✓	-	-	-	✓	✓	-	-	-	-	-	-	-	-	-	-
LAK	[91]	2022	-	-	-	✓	-	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-
LAK	[49]	2022	✓	-	-	-	-	-	-	✓	-	-	-	✓	-	✓	-	-	-	-	-
LAK	[65]	2022	-	-	-	✓	-	-	-	✓	-	-	-	-	-	✓	-	-	-	✓	-
LAK	[39]	2023	-	-	-	✓	-	-	-	✓	-	-	-	-	-	-	-	-	✓	-	-
LAK	[6]	2024	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-
LAK	[36]	2024	-	✓	-	-	-	-	-	-	-	-	-	-	-	✓	✓	✓	-	-	✓
LAK	[84]	2024	✓	-	-	-	-	-	-	✓	-	-	✓	-	-	-	-	-	✓	-	-
LAK	[45]	2025	-	-	-	-	-	-	-	-	✓	-	✓	✓	-	-	-	-	-	✓	-
LAK	[52]	2025	-	-	-	✓	-	-	-	-	-	✓	-	-	✓	-	-	-	-	-	-
LAK	[51]	2025	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LAK	[63]	2025	-	-	-	✓	-	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-

We repeated the same analysis for papers that measure personalization effects on learning outcomes ($n = 60$). Levene's tests were significant under $p \leq .05$ for learning path ($p = .002$), dashboard ($p = .012$), feedback/support ($p = .012$), instruction ($p \leq .001$), and collaboration ($p = .007$), indicating unequal variances. Neither the p-values of the two-sided Welch tests nor those of independent samples t-tests were significant under $p \leq .05$. Thus, while absolute counts suggest different emphases in the learning-outcomes subset, the subset is too small to detect statistically reliable differences.

In summary, this finding answers RQ1 and reveals that EDM and LAK papers published between 2015 and 2025 differ in terms of personalization type: LAK papers focus more prominently on feedback / support and dashboards, while EDM papers focus more often on recommender systems.

A minor difference is observed for learning-path personalization, where LAK papers occur slightly more frequently overall. For papers that also measure effects on learning outcomes, the number of included papers is too small to identify statistically significant differences. However, descriptively, 42% of EDM papers in the learning-outcomes subset focus on learning-path adaptations, while in LAK, only 22% focus on learning-path adaptations when learning outcomes are measured, although there is no remarkable difference when papers are not filtered to explicitly focus on learning outcomes.

5.2 RQ2: Which learner- and context-level indicators are used to drive personalization in EDM and LAK?

Across included papers, we identified 12 indicator categories to which each paper was linked:

- **Outcomes & Achievement**: Observable end results such as grades, test scores, pass/fail, completion, and proficiency.
- **Knowledge/Mastery State**: What the learner knows now and how it changes - KC mastery/ability estimates, probability of correctness, learning/retention.
- **Assessment & Item Properties**: Features of tasks/items - difficulty, discrimination, format, Q-matrix/skills mapping, steps/subparts.
- **Temporal Pacing & Workload**: When and how long - start/submit vs. deadlines, time-on-task, lags between sessions, workload over time.
- **Behavioral Interaction & Engagement**: Interaction traces - clicks, views, plays/pauses, navigation, resource use, attendance, dwell time.
- **Strategy, Process & Help-Seeking**: How learners work - attempt patterns, hints/scaffolds/JIT use, SRL signals (planning / monitoring), path / strategy metrics (e.g., TTS / CFA).
- **Learner Profile & Context**: Who the learner is - demographics, prior achievement / knowledge, goals / motivation, affect / anxiety, interests, availability.
- **Social & Community Signals**: Peer/community activity - posts, comments, messages, reactions, collaboration / co-authoring, centrality, participation roles.
- **Content / Curriculum**: Structure/semantics of learning materials and alignment - concepts / topics, tags, prerequisites, knowledge graphs, similarity / relevance.
- **Model / Policy Signals & Diagnostics**: System-side predictions and controls - predicted risk / performance, utilities / importance, bandit / adaptive policies, latent weights / biases, model diagnostics.
- **Work Products & Artifact Features**: Properties of learner-produced artifacts - writing (linguistic / structural features), code (tokens / AST / correctness), revisions / edits, talk / multimedia quality.
- **Records & Administrative Data**: Institutional / operational records - attendance / absences, mobility / enrollment, suspensions, assignment status, calendars / notifications, merged official logs.

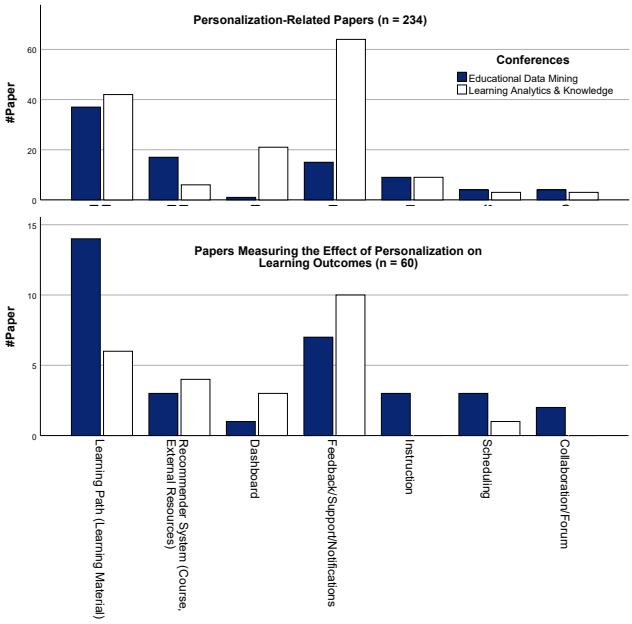


Figure 2: Personalization types in EDM and LAK papers.

To test indicator differences between EDM and LAK, we employed the same procedure as for RQ1. Levene's test was only significant for "Knowledge/Mastery State" ($p = .017$). However, across all indicators, there was no statistically significant difference in the independent samples t-tests or Welch tests when examining all personalization-related papers (Fig. 3).

Considering only papers that measured the effect of personalization on learning outcomes ($n = 60$), Levene's tests were significant for "Temporal Pacing & Engagement" only ($p \leq .001$). Again, no statistically significant differences were identified. This answers RQ2: there are no statistically reliable differences in the indicators used for personalization in EDM and LAK within the corpus, either overall or within the learning-outcomes subset.

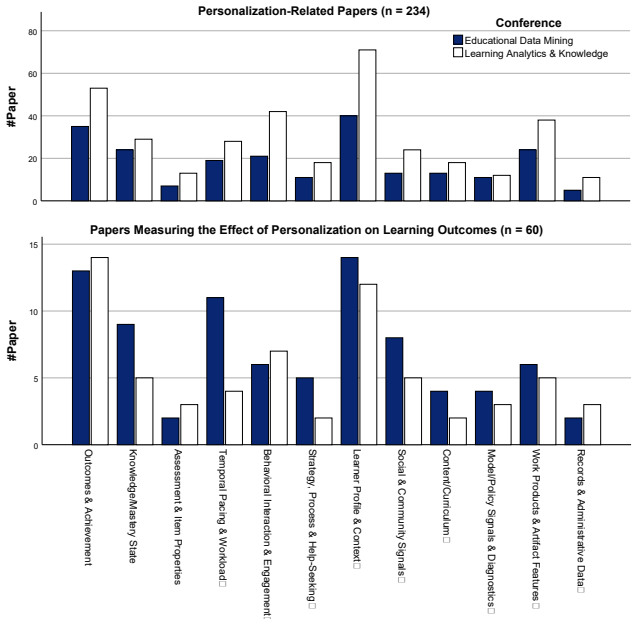


Figure 3: Personalization indicators in EDM and LAK papers.

5.3 RQ3: What research gaps and future directions emerge across the two communities?

To address RQ3, we analyzed the limitations and future-work statements reported in the included papers and synthesized them into recurrent themes. We report themes that appear across both communities, followed by community-leaning emphases.

Cross-cutting limitations (both communities) - Across EDM and LAK, reported limitations converged on threats to external validity and data quality. Many studies relied on small, context-bound samples or single-course deployments and noted limited generalizability to other settings and populations [1, 6, 27]. Cold-start and sparsity issues were frequently highlighted, especially for new learners, low-activity learners, or new content where personalization signals are weak [20, 12]. Studies also reported measurement noise, incomplete outcomes, and missing ground truth for learning constructs, which constrained robust evaluation and cross-study comparability [88, 1]. These recurring issues collectively limit confidence in transferability and reduce the feasibility of cumulative evidence building.

LAK-leaning limitations: adoption and ecological validity - Within LAK, limitations more often problematized ecological validity and adoption: platform- or course-specific interventions that do not generalize; institutional constraints on integration and data flows; instructor workload and interpretation burdens; and low voluntary engagement/compliance when interventions are deployed in authentic settings [1, 6, 15, 27, 49]. These limitations often framed personalization as a socio-technical intervention that depends on organizational fit, teacher workflow, and sustained user uptake rather than on predictive accuracy alone.

EDM-leaning limitations: model complexity and weak causal validation - Within EDM, limitations were predominantly model-centric. Papers emphasized parameter tuning difficulty, computational complexity, and overfitting risks (especially for deep or KT-style models), as well as performance degradation under distribution shift [88, 9, 25]. Another recurring theme was the gap between offline evaluation and causal impact: algorithmic improvements measured on logs or counterfactual simulations were often not validated through randomized/causal designs in authentic learning environments [38, 9].

Cross-cutting future directions (both communities) - Future-work statements commonly called for stronger evaluation and more realistic deployment contexts: online experimentation and adaptive policies embedded in courses; richer data beyond clickstream (e.g., multimodal traces); and clearer reporting standards to improve reproducibility and benchmarking [5, 41, 47, 42]. Ethical and governance themes also appeared, including fairness, privacy, and responsible use of personalization, alongside interoperability needs for cross-platform integration [33, 47, 43].

LAK-leaning future directions: human-centered orchestration and infrastructure - LAK papers more frequently emphasized moving from analytics to action through situated, human-centered interventions. Recurrent directions included teacher-in-the-loop orchestration (dashboards and workflow support), real-time adaptation in authentic contexts, interoperability and standards, and causal evaluation designs that move beyond correlational evidence [5, 11, 29, 35, 4, 15, 24, 28, 47]. Multimodal data and richer recommendation/sequencing support were also recurring motifs [41, 14, 3, 30, 46, 65].

EDM-leaning future directions: scalable policy optimization and benchmarking - EDM proposals concentrated on recommendation and curriculum sequencing, representation learning, and sequence / knowledge modeling (often extending KT) to improve personalization. Many papers also pointed to benchmarking, replicability, and robustness testing to strengthen external validity, and a growing emphasis on fairness / privacy and human-centeredness in adaptive systems [2, 23, 55, 75, 77, 9, 16, 53, 71, 82, 19, 40, 62, 69]. Overall, these results indicate that both communities articulate similar high-level needs (generalization, data quality, evaluation), but diverge in emphasis: LAK foregrounds adoption and socio-technical integration, while EDM foregrounds algorithmic generalization and policy/model validation.

6. DISCUSSION AND FUTURE WORK

The results paint a picture of two communities that are aligned in aspiration but differentiated in how personalization is *realized*, *justified*, and *validated*. This discussion interprets the three results in relation to the paper's central goal: clarifying how personalization is enacted and evaluated in EDM and LAK, and what that implies for credible progress.

6.1 Interpreting differences in personalization targets (RQ1)

The observed differences in focus can be read as a division of labor between *actionability* and *optimization*. Dashboards and feedback mechanisms are often designed around human interpretability and immediate use in learning/teaching workflows, which aligns with LAK's longstanding emphasis on making analytics actionable. Recommender systems and sequencing policies foreground algorithmic decision-making under uncertainty, fitting EDM's tradition of optimizing interventions given data and constraints. Importantly, these are not competing endpoints but complementary components of personalization at scale: optimized policies that are not interpretable or adoptable may fail in authentic settings, while actionable interfaces without validated adaptive mechanisms risk becoming advisory overlays.

A key nuance is that the learning-outcomes subset is small enough that statistical differences disappear even when descriptive differences remain. This suggests a practical constraint on the field's evidence base: personalization is frequently implemented and studied, but comparatively rarely evaluated with outcome measures in a way that supports stable comparisons across venues. Strengthening the outcome-evaluation base is therefore not merely a methodological preference; it is a prerequisite for cumulative knowledge.

6.2 Why indicators look similar across communities (RQ2)

Despite the differences in personalization *targets*, indicator usage does not differ statistically between EDM and LAK. One interpretation is that both communities draw from a shared measurement toolkit (performance, behavioral traces, context/profile, and model-derived signals), especially given overlapping platforms and datasets. Another interpretation is that the indicator taxonomy is broad enough to absorb community differences at a finer granularity: for example, both communities may use "behavioral interaction" indicators, but for different purposes (policy learning vs. teacher-facing explanation), or with different operationalizations (raw logs vs. engineered SRL constructs). In other words, the lack of indicator differences at category level should not be taken to mean that measurement practices are identical; rather, it highlights that divergence may lie in *how* signals are modeled, validated, and translated into adaptation.

This result also clarifies an opportunity: if both communities already rely on similar signal families, then shared reporting standards (feature definitions, preprocessing, missingness handling, construct validity arguments) could significantly improve reproducibility and cross-study comparability without requiring convergence on a single methodological paradigm.

6.3 From "gaps" to research programs (RQ3)

Research gaps are empirically grounded in what authors themselves repeatedly report. The discussion here focuses on what those recurring themes imply for a forward research program.

First, the cross-cutting limitations (context-bound samples, sparsity/cold-start, noisy and incomplete outcomes) point to a deeper structural issue: personalization research often advances through *bespoke* deployments and datasets, which hinders external validity. This suggests that progress depends on infrastructure-level changes: shared benchmarks where appropriate, multi-site studies, and more consistent definitions of what constitutes "personalization" and "learning outcomes" across contexts. Without these, even strong individual studies remain difficult to synthesize.

Second, the community-leaning emphases are complementary and could be integrated into more powerful evaluation and design patterns. LAK's adoption/ecological-validity concerns indicate that personalization should be evaluated as a socio-technical intervention embedded in workflows, constraints, and incentives. EDM's model-centric concerns indicate that personalization should be stress-tested for robustness, generalization, and policy validity beyond offline gains. A combined program would therefore evaluate personalization *end-to-end*: from signal quality and model robustness, through interface and orchestration, to causal impact on learning.

Third, the repeated call for stronger causal validation suggests that "better models" and "better dashboards" are insufficient if research designs cannot distinguish personalization effects from confounds. Pragmatic experimentation (e.g., online A/B tests, stepped-wedge rollouts, teacher-mediated randomization) and careful measurement design (validated outcomes, delayed post-tests, or competency-proximal assessments) are needed to turn personalization into reliable practice. The rarity of learning-outcomes evaluations in the corpus implies a field-level bottleneck: many studies stop at plausibility or engagement metrics, leaving learning impact under-evidenced.

Finally, ethics and governance themes (fairness, privacy, explainability) appear as an emergent but not yet uniformly operationalized agenda. A practical implication is that these considerations should be connected to the same objects / indicators / techniques framework used in this review. For example, fairness risks differ between a course recommender and a mastery-based sequencer; privacy risks differ between administrative records and multimodal sensing; explainability requirements differ between teacher-facing dashboards and learner-facing feedback. Treating ethics as design- and context-dependent, rather than as a generic checklist, is likely to make it more actionable across both communities.

6.4 Implications for cross-fertilization

Taken together, the results suggest a productive synthesis: EDM contributes scalable modeling and policy optimization methods, while LAK contributes human-centered orchestration, interpretability, and deployment realism. Cross-fertilization could be accelerated by (i) adopting shared reporting standards for personalization objects, signals, and evaluation designs; (ii) building evaluation pipelines that explicitly connect offline metrics to online impact; and (iii) designing systems where policy learning and human oversight are co-designed rather than bolted together. Such work would directly address the dominant limitations reported in both communities while preserving their strengths.

7. LIMITATIONS

This review offers a structured, comparative synthesis of personalization research in EDM and LAK between 2015 and 2025, but several limitations constrain the interpretation and generalization of the findings. The corpus was intentionally limited to papers published in the LAK and EDM conference proceedings to support direct comparisons between two influential communities. While this restriction improves internal coherence, it also narrows coverage by excluding relevant contributions published in journals and adjacent venues where personalization is frequently studied and where longer-term deployments, multi-institution evaluations, and theory-building work are often reported. Consequently, the patterns identified here should be understood as reflecting the proceedings-centered discourse of these two conference ecosystems rather than the full landscape of educational personalization research.

Our retrieval strategy employed a single high-recall lexical stem (`personali*`) to ensure auditability and capture common spelling and morphological variants. Although this choice strengthens reproducibility, it can lead to both false negatives and false positives. Relevant work may be missed when personalization is instantiated under alternative terminology (e.g., adaptive tutoring, individualized sequencing, targeted intervention, mastery-based progression, or recommendation framed without explicit "personalization" language), and records may be retrieved where personalization is invoked primarily as motivation rather than as an implemented mechanism. Accordingly, the resulting corpus depends not only on the search term but also on subsequent screening decisions.

The review further relies on an operational definition of personalization as system-driven, individual-level adaptation (or explicitly defined subgroups) based on learner-specific information. This definition provides conceptual clarity, but it may exclude studies in which personalization is enacted primarily through human mediation (e.g., instructor decision-making supported by analytics) or where personalization is realized as infrastructure, workflow integration, or institutional practice rather than as an explicit adaptive mechanism in software. Borderline cases are inevitable in a heterogeneous literature, and although adjudication procedures were used, some degree of classification error remains possible. Similarly, the second-stage distinction between studies that measure learning outcomes and those that report engagement, usability, or satisfaction requires interpretive judgment because papers do not always report outcomes with sufficient construct validity arguments or consistent operationalization. Proxies such as completion or time-on-task may be treated as learning-proximal in some contexts but not in others, and ambiguity in reporting can affect categorization.

The coding scheme necessarily abstracts across substantial within-category heterogeneity. Targets of personalization and indicator families were treated as non-exclusive categories, reflecting the fact that many systems combine mechanisms and signals. However, this design complicates inference because category frequencies do not correspond to mutually exclusive partitions, and coarse taxonomies can mask differences in how signals are operationalized and validated. For example, "behavioral interaction" may denote raw clickstream counts, engineered self-regulated learning constructs, or multimodal engagement indicators; "knowledge/mastery state" may arise from classical cognitive models, deep knowledge tracing variants, or hybrid approaches. As a result, small differences at the category level should not be interpreted as equivalence in measurement practice or modeling choices.

Quantitative comparisons are based on counts of papers coded into categories and employ standard significance tests. This approach treats papers as independent observations and does not explicitly account for dependencies such as recurring authorship, shared datasets, or coordinated research programs that may cluster publication patterns. Moreover, the subset of included studies that evaluate learning outcomes is comparatively small, limiting statistical power to detect stable differences between communities in that subset. Non-significant findings in the learning-outcomes analyses should therefore be interpreted cautiously as insufficient evidence for differences rather than evidence of no differences.

Finally, this work is not a meta-analysis and does not estimate pooled effect sizes. The included studies vary widely in context, intervention types, outcome measures, duration, and evaluation designs, ranging from offline analyses and observational studies to quasi-experiments and randomized evaluations. This heterogeneity limits the appropriateness of aggregating effects and complicates causal interpretation, particularly when reports omit information needed to assess internal validity (e.g., randomization fidelity, attrition, contamination, or multiple testing control). In addition, the synthesis of limitations and future-work statements depends on what authors choose to report and may be affected by reporting bias, page constraints, and selective emphasis, potentially underrepresenting negative results, failed deployments, or governance constraints such as privacy and fairness trade-offs. Taken together, these considerations indicate that the present review is best read as a transparent, reproducible map of how personalization is operationalized and evaluated in LAK and EDM proceedings during 2015–2025, rather than as an exhaustive census of all personalization research or a definitive causal estimate of personalization effectiveness.

8. CONCLUSION

EDM and LAK both pursue personalization, but they do so through different lenses, as this literature review has examined. This review compared how EDM and LAK operationalize personalization (2015 – 2025) by what is adapted, the indicators that guide personalization, and how effects are evaluated. We find distinct foci - LAK leans toward feedback and dashboards, EDM toward recommender/sequencing, but no systematic differences in indicator usage. Shared limitations include context-bound samples, data sparsity, and measurement noise; LAK stresses adoption/ecological validity, while EDM emphasizes model complexity and weak causal validation. Progress hinges on combining EDM's policy/model advances with LAK's human-centered orchestration, supported by practice-grounded trials, cross-platform benchmarks, and clear reporting of objects, indicators, and evaluation designs.

REFERENCES

[1] S. Abdi, H. Khosravi, S. Sadiq, and D. Gasevic. Complementing educational recommender systems with open learner models. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 360–365, 2020.

[2] K. Abhinav, V. Subramanian, A. Dubey, P. Bhat, and A. D. Venkat. Lecore: A framework for modeling learner's preference. In *Proceedings of the 11th International Conference on Educational Data Mining, Buffalo NY*, 2018.

[3] S. A. Adjei, A. F. Botelho, and N. T. Heffernan. Sequencing content in an adaptive testing system: the role of choice. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference*, pages 178–182, 2017.

[4] K. Aghaei, M. Hatala, and A. Mogharrab. How students' emotion and motivation changes after viewing dashboards with varied social comparison group: A qualitative study. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 663–669, 2023.

[5] A. P. Aguinalde and J. Shin. Talking in sync: How linguistic synchrony shapes teacher-student conversation in english as a second language tutoring environment. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 395–406, 2025.

[6] Q. U. Ain, M. A. Chatti, P. A. Meteng Kamdem, R. Alatrash, S. Joarder, and C. Siepmann. Learner modeling and recommendation of learning resources using personal knowledge graphs. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 273–283, 2024.

[7] N. Alam, B. Mostafavi, S. Tithi, M. Chi, and T. Barnes. How much training is needed? reducing training time using deep reinforcement learning in an intelligent tutor. In *Proceedings of the 17th International Conference on Educational Data Mining*, 2024.

[8] H. Alamri, V. Lowell, W. Watson, and S. L. Watson. Using personalized learning as an instructional approach to motivate learners in online higher education: Learner self-determination and intrinsic motivation. *Journal of Research on Technology in Education*, 52(3):322–352, 2020.

[9] H. Almoubayyed, S. Fancsali, and S. Ritter. Generalizing predictive models of reading ability in adaptive mathematics software. In *Proceedings of the 16th International Conference on Educational Data Mining*, 2023.

[10] K. T. Arnesen, C. R. Graham, C. R. Short, and D. Archibald. Experiences with personalized learning in a blended teaching course for preservice teachers. *Journal of online learning research*, 5(3):275–310, 2019.

[11] S. Aslan, E. Okur, N. Alyuz, S. E. Mete, E. Oktay, U. Genc, and A. A. Esme. Students' emotional self-labels for personalized models. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference*, pages 550–551, 2017.

[12] D. Azcona, P. Arora, I.-H. Hsiao, and A. Smeaton. user2code2vec: Embeddings for profiling students based on distributional representations of source code. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, pages 86–95, 2019.

[13] C. Baek and T. Doleck. Educational data mining versus learning analytics: A review of publications from 2015 to 2019. *Interactive Learning Environments*, 31(6):3828–3850, 2023.

[14] M. Bannert, I. Molenaar, R. Azevedo, S. Järvelä, and D. Gašević. Relevance of learning analytics to measure and support students' learning in adaptive educational technologies. In *proceedings of the seventh international learning analytics & knowledge conference*, pages 568–569, 2017.

[15] C. Borchers, J. Ooge, C. Peng, and V. Aleven. How learner control and explainable learning analytics about skill mastery shape student desires to finish and avoid loss in tutored practice. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 810–816, 2025.

[16] C. Botta, A. Segal, and K. Gal. Sequencing educational content using diversity aware bandits. In *Proceedings of the 16th International Conference on Educational Data Mining*, 2023.

[17] P. Brusilovsky, J. Eklund, and E. Schwarz. Web-based education for all: a tool for development adaptive courseware. *Computer networks and ISDN systems*, 30(1-7):291–300, 1998.

[18] P. Brusilovsky and C. Peylo. Adaptive and intelligent web-based educational systems. *International journal of artificial intelligence in education*, 13(2-4):159–172, 2003.

[19] S. Bulathwela, M. Verma, M. Pérez-Ortiz, E. Yilmaz, and J. Shawe-Taylor. Can population-based engagement improve personalisation? a novel dataset and experiments. *arXiv preprint arXiv:2207.01504*, 2022.

[20] H. Bydzovská. Course enrollment recommender system. In *Educational Data Mining*, 2016.

[21] N. Chakraborty, S. Roy, W. L. Leite, M. K. S. Faradonbeh, and G. Michailidis. The effects of a personalized recommendation system on students' high-stakes achievement scores: A field experiment. *International Educational Data Mining Society*, 2021.

[22] M. A. Chatti, A. L. Dyckhoff, U. Schroeder, and H. Thüs. A reference model for learning analytics. *International journal of Technology Enhanced learning*, 4(5-6):318–331, 2012.

[23] W. Chen, C. Joe-Wong, C. G. Brinton, L. Zheng, and D. Cao. Principles for assessing adaptive online courses. *International Educational Data Mining Society*, 2018.

[24] A. Claassen, N. Mirriahi, V. Kovanović, and S. Dawson. From data to design: Integrating learning analytics into educational design for effective decision-making: From data to design. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 558–567, 2025.

[25] B. Clement, P.-Y. Oudeyer, and M. Lopes. A comparison of automatic teaching strategies for heterogeneous student populations. In *EDM 16-9th international conference on educational data mining*,

2016.

[26] D. Davis, I. Jivet, R. F. Kizilcec, G. Chen, C. Hauff, and G.-J. Houben. Follow the successful crowd: raising mooc completion rates through social comparison at scale. In *Proceedings of the seventh international learning analytics & knowledge conference*, pages 454–463, 2017.

[27] S. Dawson, A. Pardo, F. Salehian Kia, and E. Panadero. An integrated model of feedback and assessment: From fine grained to holistic programmatic review. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 579–584, 2023.

[28] E. De Quincey, C. Briggs, T. Kyriacou, and R. Waller. Student centred design of a learning analytics system. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 353–362, 2019.

[29] H. Deininger, C. Parrisius, R. Lavelle-Hill, D. Meurers, U. Trautwein, B. Nagengast, and G. Kasneci. Who did what to succeed? individual differences in which learning behaviors are linked to achievement. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 771–782, 2025.

[30] D. Di Mitri, M. Scheffel, H. Drachsler, D. Börner, S. Ternier, and M. Specht. Learning pulse: A machine learning approach for predicting performance in self-regulated learning using multimodal data. In *Proceedings of the seventh international learning analytics & knowledge conference*, pages 188–197, 2017.

[31] K. C. Dieter, J. Studwell, and K. P. Vanacore. Differential responses to personalized learning recommendations revealed by event-related analysis. *International Educational Data Mining Society*, 2020.

[32] S. Dormezil, T. M. Khoshgoftaar, and F. Robinson-Bryant. Differentiating between educational data mining and learning analytics: A bibliometric approach. In *EDM (Workshops)*, pages 17–22, 2019.

[33] S. Doroudi and E. Brunskill. Fairer but not fair enough on the equitability of knowledge tracing. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 335–339, 2019.

[34] A. Downer, A. Shapoval, O. Vysotska, I. Yuryeva, and T. Bairachna. Us e-learning course adaptation to the ukrainian context: lessons learned and way forward. *BMC medical education*, 18(1):247, 2018.

[35] L. Faucon, J. K. Olsen, and P. Dillenbourg. A bayesian model of individual differences and flexibility in inductive reasoning for categorization of examples. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 285–294, 2020.

[36] J. Frej, N. Shah, M. Knezevic, T. Nazaretsky, and T. Käser. Finding paths for explainable mooc recommendation: A learner perspective. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 426–437, 2024.

[37] B. Garrick, D. Pendergast, and D. Geelan. Introduction to the philosophical arguments underpinning personalised education. In *Theorising Personalised Education: Electronically Mediated Higher Education*, pages 1–16. Springer, 2016.

[38] J. P. González-Brenes and Y. Huang. " your model is predictive–but is it useful?" theoretical and empirical considerations of a new paradigm for adaptive tutoring evaluation. *International Educational Data Mining Society*, 2015.

[39] A. Gurung, S. Baral, K. P. Vanacore, A. A. Mcreynolds, H. Kreisberg, A. F. Botelho, S. T. Shaw, and N. T. Heffernan. Identification, exploration, and remediation: Can teachers predict common wrong answers? In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 399–410, 2023.

[40] J. He-Yueya and A. Singla. Quizzing policy using reinforcement learning for inferring the student knowledge state. In *14th International Conference on Educational Data Mining*, pages 533–539. educationaldatamining. org, 2021.

[41] K. Holstein, G. Hong, M. Tegene, B. M. McLaren, and V. Aleven. The classroom as a dashboard: Co-designing wearable cognitive augmentation for k-12 teachers. In *Proceedings of the 8th international conference on learning Analytics and knowledge*, pages 79–88, 2018.

[42] Q. Hu and H. Rangwala. Reliable deep grade prediction with uncertainty estimation. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, pages 76–85, 2019.

[43] M. Huptych, M. Bohuslavek, M. Hlosta, and Z. Zdrahal. Measures for recommendations based on past students' activity. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference*, pages 404–408, 2017.

[44] P. Hur, H. Lee, S. Bhat, and N. Bosch. Using machine learning explainability methods to personalize interventions for students. *International Educational Data Mining Society*, 2022.

[45] G. D. Jaiyeola, A. Y. Wong, R. L. Bryck, C. Mills, and S. Hutt. One size does not fit all: Considerations when using webcam-based eye tracking to models of neurodivergent learners' attention and comprehension. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 24–35, 2025.

[46] W. Jiang, Z. A. Pardos, and Q. Wei. Goal-based course recommendation. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 36–45, 2019.

[47] K. Kitto, N. Sarathy, A. Gromov, M. Liu, K. Musial, and S. Buckingham Shum. Towards skills-based curriculum analytics: Can we automate the recognition of prior learning? In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 171–180, 2020.

[48] S. Kopeinik, E. Lex, P. Seitlinger, D. Albert, and T. Ley. Supporting collaborative learning with tag recommendations: a real-world study in an inquiry-based classroom project. In *Proceedings of the seventh international learning analytics & knowledge conference*, pages 409–418, 2017.

[49] W. L. Leite, S. Roy, N. Chakraborty, G. Michailidis, A. C. Huggins-Manley, S. D'Mello, M. K.

Shirani Faradonbeh, E. Jensen, H. Kuang, and Z. Jing. A novel video recommendation system for algebra: An effectiveness evaluation study. In *LAK22: 12th International learning analytics and knowledge conference*, pages 294–303, 2022.

[50] B. Levy, A. Hershkovitz, O. Tzayada, O. Ezra, A. Segal, K. Gal, A. Cohen, and M. Tabach. Teacher vs. algorithm: Double-blind experiment of content sequencing in mathematics. In *12th International Conference on Educational Data Mining, EDM 2019*, pages 603–606. International Educational Data Mining Society, 2019.

[51] H. Li, W. Xing, C. Li, W. Zhu, B. Lyu, F. Zhang, and Z. Liu. Who should be my tutor? analyzing the interactive effects of automated text personality styles between middle school students and a mathematics chatbot. In *Proceedings of the 15th international learning analytics and knowledge conference*, pages 910–917, 2025.

[52] T. Li, D. Nath, Y. Cheng, Y. Fan, X. Li, M. Raković, H. Khosravi, Z. Swiecki, Y.-S. Tsai, and D. Gašević. Turning real-time analytics into adaptive scaffolds for self-regulated learning using generative artificial intelligence. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 667–679, 2025.

[53] Z. Li, L. Yee, N. Sauerberg, I. Sakson, J. J. Williams, and A. N. Rafferty. Getting too personal (ized): The importance of feature choice in online adaptive algorithms. *International Educational Data Mining Society*, 2020.

[54] L.-A. Lim, D. Gasevic, W. Matcha, N. Ahmad Uzir, and S. Dawson. Impact of learning analytics feedback on self-regulated learning: Triangulating behavioural logs with students' recall. In *LAK21: 11th international learning analytics and knowledge conference*, pages 364–374, 2021.

[55] B. Ma, G. P. Hettiarachchi, and Y. Ando. Format-aware item response theory for predicting vocabulary proficiency. In *Proceedings of the 15th International Conference on Educational Data Mining*, page 695, 2022.

[56] B. Ma, Y. Taniguchi, and S. Konomi. Course recommendation for university environments. *International educational data mining society*, 2020.

[57] R. Matz, K. Schulz, E. Hanley, H. Derry, B. Hayward, B. Koester, C. Hayward, and T. McKay. Analyzing the efficacy of ecoach in supporting gateway course success through tailored support. In *LAK21: 11th international learning analytics and knowledge conference*, pages 216–225, 2021.

[58] F. Mi and B. Faltings. Adaptive sequential recommendation for discussion forums on moocs using context trees. *International Educational Data Mining Society*, 2017.

[59] I. Molenaar, A. Horvers, R. Dijkstra, and R. S. Baker. Personalized visualizations to promote young learners' srl: The learning path app. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 330–339, 2020.

[60] L. Mouchel, T. Wambsganss, P. Mejia-Domenzain, and T. Käser. Understanding revision behavior in adaptive writing support systems for education. 2023.

[61] A. Muhammad, Q. Zhou, G. Beydoun, D. Xu, and J. Shen. Learning path adaptation in online learning systems. In *2016 IEEE 20th international conference on computer supported cooperative work in design (CSCWD)*, pages 421–426. IEEE, 2016.

[62] R. Murali, C. Conati, and D. Poole. A comparison of real-time user classification methods using interaction data for open-ended learning. 2025.

[63] I. Musabirov, M. Reza, H. Song, S. Moore, P. Chen, H. Kumar, T. Li, J. Stamper, N. Bier, A. Rafferty, et al. Platform-based adaptive experimental research in education: Lessons learned from the digital learning challenge. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 13–23, 2025.

[64] T. NeCamp, J. Gardner, and C. Brooks. Beyond a/b testing: Sequential randomization for developing interventions in scaled digital learning environments. In *Proceedings of the 9th International Conference on learning analytics & knowledge*, pages 539–548, 2019.

[65] S. Nicoll, K. Douglas, and C. Brinton. Giving feedback on feedback: An assessment of grader feedback construction on student performance. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 239–249, 2022.

[66] J. W. Orr and N. Russell. Automatic assessment of the design quality of python programs with personalized feedback. *International Educational Data Mining Society*, 2021.

[67] A. Ozturk, R. Schmucker, T. Mitchell, and A. T. Kumtepe. Heterogeneous treatment effects of learning analytics dashboards: Do all learners benefit equally? In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 418–426, 2025.

[68] Z. A. Pardos and W. Jiang. Designing for serendipity in a university course recommendation system. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 350–359, 2020.

[69] T. Patikorn, N. T. Heffernan, and J. Zou. An offline evaluation method for individual treatment rules and how to find heterogeneous treatment effect. In *Workshop and Tutorials Chairs*, page 390. ERIC, 2017.

[70] J. Pesonen, O.-M. Palo-oja, and K. Ng. Personalized learning analytics-based feedback on a self-paced online course. In *Companion Proceedings of the 11th International Conference on Learning Analytics and Knowledge (LAK'21): The impact we make: The contributions of learning analytics to learning: April 12-16, 2021, Online, Everywhere*, pages 13–14. Society for Learning Analytics Research, 2021.

[71] T. Phung, N. Kotalwar, M. Liut, J. Leinonen, P. Denny, and A. Singla. Humanizing automated programming feedback: Fine-tuning generative models with student-written feedback. 2025.

[72] V. Prakash, R. Bansal, and S. Sharma. Quest-genius: An ai-driven personalised learning platform bridging educational gaps for secondary students - a quasi-experimental study on academic performance, engagement, and equity. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors,

Proceedings of the 18th International Conference on Educational Data Mining, pages 630–634, Palermo, Italy, July 2025. International Educational Data Mining Society.

[73] Praseeda, S. Srinivasa, and P. Ram. Validating the myth of average through evidences. In *Proceedings of the 11th International Conference on Educational Data Mining, Buffalo NY*, 2019.

[74] E. Prihar, M. Syed, K. Ostrow, S. Shaw, A. Sales, and N. Heffernan. Exploring common trends in online educational experiments. In *Proceedings of the 15th International Conference on Educational Data Mining*, 2022.

[75] A. N. Rafferty, R. Jansen, and T. L. Griffiths. Using inverse planning for personalized feedback. *EDM*, 16:472–477, 2016.

[76] M. J. Rodríguez-Triana, L. P. Prieto, A. Martínez-Monés, J. I. Asensio-Pérez, and Y. Dimitriadis. The teacher in the loop: Customizing multimodal learning analytics for blended learning. In *Proceedings of the 8th international conference on learning analytics and knowledge*, pages 417–426, 2018.

[77] J. Rollinson and E. Brunskill. From predictive models to instructional policies. *International Educational Data Mining Society*, 2015.

[78] I. Rushkin, Y. Rosen, A. M. Ang, C. Fredericks, D. Tingley, M. J. Blink, and G. Lopez. Adaptive assessment experiment in a harvardx mooc. In *EDM*, 2017.

[79] S. Rüdian. Personalization in EDM and LAK - Dataset, 2026.

[80] A. Schütt, T. Huber, I. Aslan, and E. André. Fast dynamic difficulty adjustment for intelligent tutoring systems with small datasets. 2023.

[81] F. Sense, M. Krusmark, J. Fiechter, M. G. Collins, L. Sanderson, J. Onia, and T. Jastrzembski. Combining cognitive and machine learning models to mine cpr training histories for personalized predictions. *International Educational Data Mining Society*, 2021.

[82] H. Shaikh, A. Modiri, J. J. Williams, and A. N. Rafferty. Balancing student success and inferring personalized effects in dynamic experiments. In *Proceedings of the 11th International Conference on Educational Data Mining, Buffalo NY*, 2019.

[83] K. Thaker, L. Zhang, D. He, and P. Brusilovsky. Recommending remedial readings using student knowledge state. *International Educational Data Mining Society*, 2020.

[84] D. R. Thomas, J. Lin, E. Gatz, A. Gurung, S. Gupta, K. Norberg, S. E. Fancsali, V. Aleven, L. Branstetter, E. Brunskill, et al. Improving student learning with hybrid human-ai tutoring: A three-study quasi-experimental investigation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 404–415, 2024.

[85] S. Thomas. Future ready learning: Reimagining the role of technology in education. 2016 national education technology plan. *Office of Educational Technology, US Department of Education*, 2016.

[86] J. Turnbull, D. Lea, D. Parkinson, P. Phillips, B. Francis, S. Webb, V. Bull, and M. Ashby. Oxford advanced learner's dictionary. *International Student's Edition*, 2010.

[87] O. Vainas, Y. Ben-David, R. Gilad-Bachrach, M. Ronen, O. Bar-Ilan, R. Shillo, G. Lukin, and D. Sitton. Staying in the zone: Sequencing content in classrooms based on the zone of proximal development. In *12th International Conference on Educational Data Mining, EDM 2019*, pages 659–662. International Educational Data Mining Society, 2019.

[88] J. Vassoyan, J.-J. Vie, and P. Lemberger. Towards scalable adaptive learning with graph neural networks and reinforcement learning. In M. Feng, T. Käser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 351–361, Bengaluru, India, July 2023. International Educational Data Mining Society.

[89] J. Woodrow, C. Piech, and S. Koyejo. Improving generative ai student feedback: Direct preference optimization with teachers in the loop. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 442–449, 2025.

[90] H. Yu, C. Miao, C. Leung, and T. J. White. Towards ai-powered personalization in mooc learning. *npj Science of Learning*, 2(1):15, 2017.

[91] A. Zavaleta Bernuy, Z. Han, H. Shaikh, Q. Y. Zheng, L.-A. Lim, A. Rafferty, A. Petersen, and J. J. Williams. How can email interventions increase students' completion of online homework? a case study using a/b comparisons. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 107–118, 2022.

[92] M. Zhang, J. Hao, P. Deane, and C. Li. Defining personalized writing burst measures of translation using keystroke logs. In *Proceedings of the 11th International Conference on Educational Data Mining, Buffalo NY*, pages 549–552, 2018.

[93] H. Zo. Personalization vs. customization: Which is more effective in e-services? 2003.

Real Enough to Matter? Implications of Synthetic Data for Reproducible Learning Analytics

Elena Tiukhova
KU Leuven, LIRIS, Leuven, Allianz Benelux, Brussels, Belgium
elena.tiukhova@kuleuven.be

Grzegorz Meller
KU Leuven, Dept. of Computer Science, Augment, an imec research group
Leuven, Belgium
grzegorz.meller@kuleuven.be

Dimitri Van Landuyt
KU Leuven, LIRIS
Leuven, Belgium
dimitri.vanlanduyt@kuleuven.be

Tinne De Laet
KU Leuven, LESEC
Leuven, Belgium
tinne.delaet@kuleuven.be

Bart Baesens
KU Leuven, LIRIS
Leuven, Belgium

Monique Snoeck
KU Leuven, LIRIS
Leuven, Belgium
monique.snoeck@kuleuven.be

ABSTRACT

Learning analytics (LA) and educational data mining offer powerful ways to study self-regulated learning (SRL), but privacy constraints often restrict access to authentic learner data, limiting reproducibility and collaboration. Synthetic data generation (SDG) provides a promising approach by preserving statistical properties of real data while mitigating privacy risks. This study evaluates synthetic SRL datasets along three dimensions: their resemblance to real data, their utility for predicting student success, and their capacity to reproduce prior explainable AI (XAI) findings. Results show that BayesNet-based SDG most effectively approximated real data and enabled predictive models that in some cases outperformed those trained on authentic data. Synthetic datasets also replicated XAI stability metrics from prior work, though contextual effects such as those linked to COVID-19 remained difficult to capture. Grounded in SRL theory, this study demonstrates both the promise and limitations of SDGs and contributes a publicly available multidimensional dataset for open LA research.

Keywords

Synthetic data, explainable AI, learning analytics

1. INTRODUCTION

Advancements in data collection and processing within the educational domain have given rise to the field of learning analytics (LA), which aims to optimize learning processes and environments in a data-driven manner [39]. Self-regulated learning (SRL), which has been shown to be important for student success [53, 54], can be studied by processing the data traces left by students interacting and engaging with online environments [49]. The potential of LA data to capture SRL is well-recognized [48]. Additionally, Educational Data Mining (EDM) can help to operationalize SRL constructs, to investigate the relationships between these constructs, to model and understand student learning trajectories, and to clarify how far we can trust conclusions and findings [49]. LA trace data has been shown to be most informative for predicting student success when higher level study indicators are generated from the highly granular trace data [17]. Dashboards and other LA tools that provide support for SRL aspects such as goal setting, monitoring, and self-reflection have been shown capable of positively influencing learning motivation and engagement, on condition that tools explicitly link their functionalities to SRL theories and constructs [55], and are grounded in educational theories [30, 46]. Extracting meaningful SRL indicators from trace data using EDM is an important prerequisite for providing effective LA tool support.

The use of actual, real-world trace data to support LA and evaluate SRL effectiveness is hindered by ethical and privacy concerns [43]. Access to real learning data is often restricted due to privacy regulations such as the GDPR in Europe, as well as institutional policies on processing student data – where data collection primarily serves the organization and administration of education. These restrictions complicate open science and collaborative research. One promising solution is the generation and use of synthetic data, which is a class of techniques that preserve the statistical properties of the original datasets while reducing the risks of information disclosure, re-identification or reconstruction [4]. This promise is amplified by recent advances in GenAI techniques capable of generating structured tabular datasets [18]. The synthetic data generation approach is especially valuable in large educational information systems composed of multiple services owned by different stakeholders, where data sharing is inherently challenging. By enabling safer data exchange, synthetic data can facilitate faster system development and testing, promote collaboration, and support broader adoption of LA [4].

Elena Tiukhova, Grzegorz Meller, Dimitri Van Landuyt, Tinne De Laet, Bart Baesens, and Monique Snoeck. Real Enough to Matter? Implications of Synthetic Data for Reproducible Learning Analytics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 243–255. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039709

In this paper, we examine the extent to which artificially generated data on learners' self-regulation (1) resembles real data, (2) can be leveraged to predict student success, and (3) enables the replication of prior studies on explainable AI (XAI) in LA. Our primary contribution is an in-depth comparative evaluation of the potential of synthetic datasets within the broader LA research domain, along with an analysis of their use in the context of XAI stability. In addition, we provide a synthetic dataset that addresses the scarcity of publicly available LA data. To support the link with SRL, we focus on a dataset constituted of SRL indicators, next to outcome data. This dataset is multi-dimensional, spanning multiple courses and academic years, and thus supports out-of-sample validation across both contexts and time.

This paper is organized as follows. Section 2 discusses related work, identifies research gaps and states the research questions. Section 3 details the study's methodology, including descriptions of the data and synthetic data generators, as well as the analysis setup. Section 4 reports the findings and Section 5 provides a broader discussion of the results, including the limitations and directions for future work. Finally, Section 6 summarizes the main contributions.

2. RELATED WORK

2.1 Synthetic Data Generation in LA

Privacy and confidentiality concerns have driven the adoption of synthetic data generators (SDGs). These tools are capable of learning the characteristics of real datasets to produce synthetic data with similar statistical properties [51]. Several publicly available SDG libraries exist, including Synthcity [35] and the Synthetic Data Vault (SDV) [33].

Statistical techniques such as Markov models or Bayesian networks (BayesNet) are widely used for synthetic data generation in education. Flanagan et al. [11] investigated the effectiveness of using synthetic educational data for real-world scenarios involving data sharing with third parties. The synthetic data was generated using a first-order Markov model trained on real interaction data from a digital material reading system, where learner actions were modeled as states and transition probabilities were learned from the original dataset. To evaluate the quality of the synthetic data, models were trained on it to predict academic achievement and then tested on real, private data. Several of these models showed significantly lower performance when applied to the non-synthetic datasets, indicating limitations in the utility of the synthetic data. On the other hand, the experiment of Dorodchi et al. [9] generated synthetic educational data using Bayesian networks and the models trained on synthetic data performed similarly to those trained on real data in benchmark tasks, suggesting synthetic data can be a viable alternative for algorithm development in LA.

In many machine learning (ML) studies, statistical methods are frequently benchmarked against more complex deep learning models (e.g., [14]), and SDG is no exception. Zhan et al. [51] evaluated four SDG techniques – GaussianCopula, CTGAN, CopulaGAN, and TVAE – for their ability to generate high-quality synthetic student data. Their evaluation followed a two-step process: (1) comparing the distributional similarity between synthetic and original datasets; and (2) assessing the utility of the synthetic data in LA tasks such as regression and classification. The results indicated that GaussianCopula and TVAE performed best, producing synthetic data that both closely mirrored the original distributions and enabled strong performance on downstream tasks. Expanding on this, Liu et al. [25] conducted a comparative analysis of SDGs in LA – including Gaussian Multivariate (GM), Gaussian Copula, and CTGAN – using metrics for similarity, utility, and privacy. While GM performed well in terms of utility, it fell short in privacy protection: the close resemblance between GM-generated synthetic data and the real data increases the risk of exposing sensitive information. In contrast, Gaussian Copula offered the most balanced performance across all three dimensions, making it the preferred choice for LA tasks.

The emergence of large language models (LLMs) has spurred their adoption in the domain of synthetic data generation. In addition to the widely used CTGAN model, Khalil et al. [18] employed three different LLMs to generate synthetic tabular data. The resulting datasets exhibited strong alignment with real-world data, particularly in terms of statistical similarity. Their findings showed that CTGAN achieved performance comparable to that of the LLM-based approaches, with both CTGAN and DialoGPT emerging as the top performers overall.

Liu et al. [24] investigated the trade-off between privacy and fairness in synthetic data generation, evaluating both GAN-based models and LLMs using standard ML algorithms as well as fairness-aware methods. In LA, privacy concerns center on obtaining learners' consent for data collection and preventing harm from data sharing, while fairness focuses on avoiding the reinforcement of existing societal biases against minority groups in LA solutions. Their study found that the DECAF model achieved the best overall balance between privacy and fairness, albeit at the cost of the lowest utility performance. For use cases where privacy is the primary concern, ADSGAN was identified as the most suitable option. Table 1 summarizes the abovementioned studies on investigating the performance reported for existing SDGs.

2.2 Stability Analysis in Learning Analytics

One of the key downstream tasks in LA is predicting student success using ML models trained on learning indicators derived from detailed study activity data, in alignment with SRL theory [17]. The insights generated by these models can support feedback provision, such as personalized study advice and timely interventions. However, real-time study data is not always accessible due to privacy regulations (e.g., GDPR), making models trained on historical data the default choice for many LA practitioners. Consequently, the usefulness of these models depends on their stability across academic years and their ability to adapt to changing contexts [31].

Tempelaar et al. [41] investigated the stability and sensitivity of LA prediction models across different student cohorts and instructional designs by using linear regression models. They find that models are stable when the instructional design remains unchanged, maintaining similar predictors and accuracy. However, when the course design changes, the models show sensitivity, hence affecting model stability. Saqr et al. [38] perform a single-paper meta-analysis

Table 1: Summary overview of the most prevalent related work.

Source	Type of techniques	Data source	Purpose	Performance effects
[11]	Statistical: Markov model	Digital material reading system	Data sharing	↓
[9]	Statistical: BayesNet	Demographics, academic performance	Open science, privacy	∼
[51]	Statistical: Gaussian Copula Deep learning: CTGAN, CopulaGAN and TVAE	Demographics, VLE data, outcome data	Privacy and ethics	TVAE, Gaussian Copula: ∼ CTGAN, CopulaGAN: ↓
[25]	Statistical: Gaussian Multivariate, Gaussian Copula; Deep Learning: CTGAN	Demographics, study performance, MOOC data	Privacy, open science, LA scalability	Gaussian Multivariate: ∼ Gaussian Copula: ↓ (for privacy) CTGAN: ↓ (for utility)
[18]	LLMs: GPT2, DistilGPT2, DialoGPT Deep learning: CTGAN	Demographics, learning activity (also from VLE), outcome data	Data sharing; reproducibility and collaboration; privacy	GPT2, DialoGPT, CTGAN: ∼ DistilGPT2: ↓
[24]	LLMs: DistilGPT2; Deep learning: CTGAN, ADSGAN, DECAF, PATEGAN	Demographics, learning activity, outcome data	Privacy and fairness	DECAF: ∼ for privacy and fairness ADSGAN: ∼ for privacy DistilGPT: ∼ for fairness PATEGAN & CTGAN: ↓

∼ Performance is on par with real data;
↓ Performance is worse than with real data

(using correlation analysis for each individual analysis) to identify which commonly used LA indicators reliably generalize across course runs, reporting total activity and forum participation as the most consistently portable predictors across similar course contexts.

Tiukhova et al. [44] extended prior work by evaluating the stability of LA models over time and across student cohorts, employing both statistical and ML techniques alongside XAI methods. By integrating XAI, they introduced an important explainability dimension into model stability analysis. XAI encompasses a variety of approaches that make AI models "more intelligible to humans by providing explanations" [15]. In educational settings, explainable AI (XAI-ED) has been conceptualized as a structured framework that addresses six interrelated dimensions – stakeholders, benefits, explanation methods, model types, human-centered interface design, and explanation-specific pitfalls – in order to meet the unique needs of educational contexts [19].

Building on this foundation, Tiukhova et al. [44] combined concept and prediction drift analysis with explainability as a diagnostic tool. They demonstrated how feature-importance rankings generated by SHAP [29] can be used to assess whether model explanations remain consistent across cohorts. They show that explainability can enrich stability analysis by complementing traditional techniques and uncovering new insights: SHAP not only clarified model predictions but also functioned as a diagnostic tool for evaluating global model stability across contexts. The authors conclude that LA models should be updated, as only general engagement metrics consistently remain reliable [44].

2.3 Research gap and research questions

Evaluating reproducibility is a key requirement for ensuring generalizable outcomes in both LA and SRL research. However, it has not been systematically explored yet in the context of SDGs. This paper addresses this gap by specifically examining the reproducibility of existing LA research findings on model stability across cohorts and over time, i.e., whether models and their explanations remain consistent across time and cohorts [44] when synthetic data is used for these models. Crucially, the artificially generated data is based on a dataset containing meaningful SRL indicators to ensure that its educational relevance and interpretability are preserved. In addition, it is essential to compare the reproducibility-oriented utility of SDGs with more conventional measures – statistical resemblance and predictive performance – to determine whether commonly used evaluation criteria truly reflect the capacity of synthetic data to support valid and transferable LA research conclusions. The research therefore first assesses the quality of the synthetic data along the state of the art dimensions of statistical resemblance (RQ1) and prediction performance (RQ2). It then advances the state of the art by investigating the reproducability of the stability analysis (RQ3) and verifying the alignment to the results of RQ1 and RQ2. The RQs are therefore as follows:

RQ1 Statistical resemblance To what extent does synthetic data resemble real data in terms of distributional similarity, and which SDG algorithms produce the most realistic distributions?

RQ2 Prediction utility How effectively can synthetic data be used for student success prediction, and which SDG algorithms yield a dataset that can be used to train the models with the highest prediction performance when applied to the real data?

RQ3 Replication utility To what extent can synthetic data be used to replicate existing research going beyond predictions – specifically, the stability analysis conducted by Tiukhova et al. [44] – and which SDG algorithms produce a dataset allowing for the highest degree of reproducibility of the stability analysis?

3. METHODOLOGY

We first discuss data collection and feature engineering activities, after which we detail the selected SDGs for this study. Then, we elaborate on the main methodology to assess (i) statistical resemblance, (ii) prediction utility, and (iii) replication utility to address the respective research questions (Figure 1).

3.1 Data and Feature Engineering

In the light of RQ3, it was decided to base the research on [44] and the original data was obtained from the authors.

It represents online study behavior and academic achievement for two compulsory courses (Accountancy and Global Economics) of first-year bachelor students from KU Leuven, for three consecutive academic years (AY): 2018-2019, 2019-2020, and 2020-2021. The Accountancy course was offered in the first semester while the Global Economics course is delivered during the second semester. When mapping this to the external contexts of education during the aforementioned AYs, we can see that the Accountancy course was affected by COVID-19 in AY 2020-2021, while the Global Economics course was affected by COVID-19 during both 2019-2020 and 2020-2021. In [26], this external change was found to have a significant effect on studying patterns.

The raw data represents student's activity on the course's LMS page, including granular interactions with course material and contribution and consumption of the discussion forum. To ground the analysis in the educational sciences and SRL theory in particular, the granular data is used to construct representative SRL indicators that represent motivated learning choice along different dimensions [17].

To mitigate an excessive loss of significance, the SDG was based on the featurized dataset developed in [44] instead of the raw data. The features selected by [44] build on prior work by [45], [17], and [40]. Specifically, [17] models study behavior along two dimensions: level of activity and study regularity. For each dimension, both overall patterns and learning-action–specific patterns are captured, yielding four feature categories: overall level of activity (OLA), learning-action–specific level of activity (LALA), overall regularity of study (ORS), and learning-action–specific regularity of study (LARS). For the action-specific features, four types of learning activities are further distinguished: forum contribution, forum consumption, access to learning materials, and access to the main course page. This feature set was further enriched with features inspired by financial consumption patterns that measure exploration, exploitation, and plasticity of human consumption [40]. Feature selection was performed using correlation analysis, resulting in a final set of 16 features (no categorical features) used in the analysis [44]. Study success is operationalized using the summative grade from the first exam attempt (scored on a 0–20 scale, with a passing threshold of 10). These grades are subsequently transformed into a binary pass/fail outcome variable [44]. More details on feature engineering process and dataset characteristics can be found in [44] and [45].

3.2 Synthetic Data Generators

We use three well-established SDG models (see Table 1), each representing a distinct approach: a statistics-based BayesNet, an autoencoder-based model (TVAE), and an adversarial learning-based model (CTGAN) (the generated datasets can be found on Zenodo [42]). We exclude LLM-based models due to their similar performance to other deep learning methods, allowing us to avoid the additional computational cost associated with LLMs (see Table 1). These SDGs are used to generate single-table synthetic data from the original dataset after applying feature engineering, an approach widely used in the literature [51, 25]. The feature engineering process follows the methodology described in [44], resulting in 16 predictors that capture student SRL behaviors, along with one target binary variable indicating whether the student passed the first exam attempt.

The Conditional Tabular Generative Adversarial Network (**CTGAN**) model uses the principle of the original GAN model [13] when generating synthetic data samples while simultaneously addressing its main shortcomings, i.e., working with mixed data types, non-Gaussian and multimodal distributions and highly imbalanced categorical columns [50]. Like the original GAN, CTGAN has a generator that learns to produce synthetic tabular data imitating real data and a discriminator that learns to distinguish between real and synthetic samples. We implement CTGAN using the *ctgan* plugin from the Synthcity library [35] and its default hyperparameters (max. 1000 encoder iterations; 3 hidden layers in both encoder and decoder, each with 500 hidden units; leaky ReLU activation function in encoder, with a dropout rate of 0.1; leaky ReLU activation function in decoder, with no dropout; learning rate of 0.001, L2 regularization of 0.00001; batch size of 200; early stopping after 100 epochs if no improvement during 5 epochs; gradient clipping of 1).

The Table Variational Autoencoder (**TVAE**) [50] is an adaptation of a variational autoencoder [20] for mixed-type tabular data generation. It consists of an encoder that maps data to a latent distribution and a decoder that can reconstruct the input by sampling from this learned latent distribution. We implement TVAE using the *tvae* plugin from the Synthcity library [35] and its default hyperparameters (max. 2000 generator iterations; 2 hidden layers in both generator and descriminator, each with 500 hidden units; ReLu activation function in generator, dropout rate of 0.1; beta tuple in the generator optimizer of (0.5, 0.999); leaky ReLU activation function in discriminator, dropout rate of 0.1; 1 iteration of discriminator per generator iteration; beta tuple in the discriminator optimizer of (0.5, 0.999); learning rate of 0.001 and L2 regularization of 0.001; batch size of 200; gradient clipping of 1 and gradient penalty of 10; early stopping after 100 epochs if no improvement during 5 epochs).

A Bayesian network (**BayesNet**) is capable of capturing the correlation structure between attributes, making it possible to further sample data from the learned distribution. This structure is represented via a directed acyclic graph (DAG), used to sample synthetic data points [3]. We implement synthetic data generation with BayesNet using the *bayesian_network* plugin from the Synthcity [35] and its default hyperparameters (max. 1000 learning iterations, tree-structured BayesNet; K2 score as a Bayesian scoring metric; up to 4 parent nodes for each node).

3.3 Statistical resemblance (RQ1)

To assess the statistical resemblance of the synthetic datasets, we employed the Kolmogorov–Smirnov (K-S) test [21] for the independent variables and the chi-squared test [34] for the dependent variable (Figure 1).To evaluate multivariate fidelity, we use the SynthEval tool [23] to compute (1) the differences between mixed correlation matrices, which quantifies the preservation of pairwise linear dependencies, and (2) the differences between mutual information matrices, which capture the extent to which general (including non-linear) statistical dependencies between variables are preserved.

The K-S test is a non-parametric statistical test used to compare the empirical distribution functions of two samples. It quantifies the maximum absolute difference between their cumulative distribution functions and tests the null hypothesis that both samples are drawn from the same distribution [21]. The main motivation for using the K-S test lies in its widespread adoption in the literature [25], as it is non-parametric and sensitive to differences in both location and shape – making it a natural fit for LA settings where concept drift and cohort changes are common. Recent LA work explicitly uses K-S to monitor year-over-year shifts in student-behavior features [44]. Beyond LA, K-S is one of the main metrics in the tabular data evaluation framework *SynthEval* of Lautrup et al. [23], which presents a robust implementation and reports summary diagnostics such as the average K-S distance and the fraction of attributes flagged as significantly different. K-S tests are also commonly listed among fidelity measures in recent various evaluations, reinforcing its status as a standard similarity test when judging whether synthetic data preserves key marginal distributions [28, 27]. We apply the K-S test feature-wise to compare the synthetic SRL features to their corresponding real features, allowing us to detect whether any statistically significant distributional drift occurred between them.

The chi-squared test is a statistical method for comparing categorical data across two or more groups [34]. It is commonly used in data drift detection for categorical features, where the null hypothesis states that the feature distribution in the current dataset matches that of the reference dataset. In this study, we use the test to assess differences in the distribution of the target variable (the binary pass/fail feature) between the synthetic and real datasets.

The mixed correlation matrix (MCM) difference is computed as the Frobenius norm of the difference between correlation matrices derived from the real and synthetic data. Specifically, Pearson correlation is used for numerical–numerical pairs, Cramér's V for categorical–categorical pairs, and the correlation ratio η for categorical–numerical pairs [52]. Lower values indicate greater similarity in pairwise linear and association structure. The mutual information matrix (MIM) difference is computed analogously as the Frobenius norm of the difference between pairwise normalized mutual information (NMI) matrices. Here, NMI quantifies general (including non-linear) dependencies between variables, with lower values indicating closer agreement between real and synthetic data.

3.4 Prediction utility (RQ2)

For prediction utility analysis, we used the "train on synthetic, test on real dataset" workflow, i.e., we trained classifiers on synthetic datasets and evaluated their performance on a held-out portion (30%) of the corresponding real dataset (Figure 1). This was compared against a baseline where the models were trained on 70% of the real data and tested on the same 30% real data holdout. We selected three classification algorithms that vary in complexity including a linear model – Logistic Regression (LogReg) [32], a linear probabilistic model – Gaussian Naive Bayes (GaussNB) [36], and a tree-based ensemble model – XGBoost [6]. These models are also widely used in the LA literature for student success prediction [1, 7]. Each model underwent hyperparameter tuning via grid search with 5-fold stratified cross-validation, using balanced accuracy as the evaluation metric. This metric was selected because many of the datasets – both real and synthetic – exhibit varying degrees of class imbalance between pass and fail outcomes (see Appendix A, Table 9). Balanced accuracy offers a more reliable measure of performance in such settings, as it reflects the ability to correctly classify both classes irrespective of their prevalence. The Wilcoxon signed-rank test, which assumes as its null hypothesis that two paired samples come from the same distribution [47], is used to assess whether the performance of models trained on synthetic data differs from that of models trained on real data. The entire pipeline is illustrated in Figure 1, which outlines the full evaluation process from data preparation to model training and evaluation.

3.5 Replication utility (RQ3)

To demonstrate the value of synthetic data for replication analysis, we replicate the model agreement evaluation study conducted in [44]. In this paper, we define replication (or reproducibility) utility as the extent to which findings concerning a model's stability across study cohorts remain consistent when synthetic data is used. In [44], model stability is examined in the context of using model-derived insights to support educational advising. For black-box models, these insights are obtained through XAI techniques, which enable educators to interpret model outputs and incorporate them into advising practices. Owing to ethical and privacy constraints, such advice is typically based on data from previous course runs. Within this setting, stability is characterized as the consistency of the model's "reasoning", operationalized as the consistency of global feature importance as revealed by the explainability method, specifically Kernel SHAP [29]. In student success prediction, these features correspond to students' SRL capabilities, and the relative importance of each SRL dimension is critical for informing effective interventions. Accordingly in [44], the degree to which a feature's importance remains stable across successive course runs constitutes the core notion of stability.

Specifically, we use synthetic data to train and evaluate a range of model configurations that vary both in the choice of algorithm and the data used for training and testing. Eight ML algorithms are considered: Naïve Bayes (NB) [37], Support Vector Machine (SVM) [16], Multi-Layer Perceptron (MLP) [37], Logistic Regression (LR) [32], K-Nearest Neighbors (KNN) [8], Random Forest (RF) [5], XGBoost (XGB) [12], and TabNet (TAB) [2] – see [44] for further details.

Model evaluation is conducted across four different pipelines, which differ primarily in the AY data used for training and testing the model with optimal hyperparameters (as determined by cross-validation): (1) the optimal model of AY T-1 data applied on AY T-1 test data, (2) the optimal model of AY T-1 applied on AY T test data, (3) the optimal model of AY T-1 retrained on AY T data applied on AY T test data and (4) the optimal model of AY T applied on AY T test data. For each pipeline, feature importance rankings are generated using the Kernel SHAP algorithm [29].

Kernel SHAP is an additive feature attribution method that decomposes a model's prediction into the contributions of each input feature. For every data sample, it assigns a score

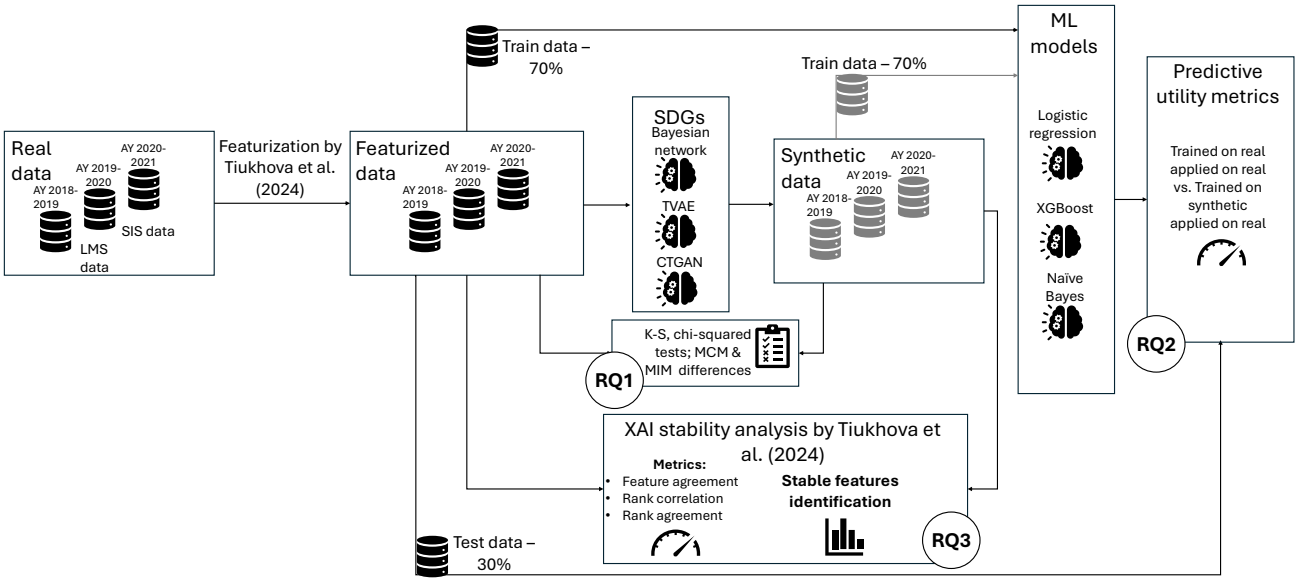


Figure 1: Analysis Pipeline

to each feature representing its importance in generating the prediction. By averaging the absolute SHAP values across all samples, one can estimate global feature importance and produce feature rankings. These rankings are then used to compute stability metrics widely used for investigating a general disagreement problem in the XAI domain [22]: feature agreement, rank agreement, and rank correlation. Feature agreement measures the overlap in the top-k important features between models, while rank agreement captures both the shared top-k features and their relative positions in the rankings. Rank correlation assesses the overall consistency in feature importance orderings across models. For detailed metric definitions and formulas, refer to [44].

In this study, we compute these stability metrics using synthetic data. Additionally, we replicate the stable feature identification analysis by determining which features consistently appear among the top eight across all four model configurations. Finally, we compare our results with those reported in [44] to assess whether the key findings hold when using synthetic data.

4. RESULTS

RQ1 - Statistical resemblance. Table 2 displays the K-S test results (p-values) evaluating how well the feature distributions in the synthetic datasets match those of the original data. Values below 0.05 indicate statistically significant drift between real and synthetic data distributions and are printed in bold in the tables. Based on this criterion, TVAE performs the worst, with the highest number of bolded entries across academic years and courses, suggesting substantial distributional drift. CTGAN shows a moderate level of fidelity, with fewer instances of drift than TVAE, though still not consistently preserving all features. Notably, the BayesNet outperforms both, demonstrating stronger distributional resemblance across most features, especially in more recent years, with comparatively few cases of drift. However, it is important to highlight that the binary Pass/Fail outcome consistently exhibits drift across all years and both subjects, with Bayesian Network models showing this most prominently. Notably, the imbalance ratio in the synthetic data does not systematically deviate in a single direction, but instead varies across settings, sometimes overestimating and sometimes underestimating the minority class (Appendix A, Table 9). This indicates that the model does not reliably preserve the class prior, but rather exhibits high variance in its estimation under class imbalance. The effect is exacerbated by the sparse representation of the minority class, which leads to unstable conditional probability estimates.

Table 3 shows that across both datasets and all academic years, BayesNet consistently achieves the lowest mixed correlation and mutual information matrix differences, indicating superior preservation of both linear and general dependency structures. CTGAN exhibits moderate performance, while TVAE consistently shows the largest deviations, suggesting weaker fidelity in capturing multivariate relationships. The consistency of these results across metrics and cohorts indicates that the observed performance differences are systematic rather than dataset-specific.

RQ2 - Prediction utility. Table 4 reports the differences in balanced accuracy scores on the test set for three algorithms (LogReg, GaussNB, and XGBoost) trained on real data and on synthetic data generated by BayesNet, TVAE, and CTGAN, across multiple courses and years. Negative values indicate that models trained on synthetic data outperformed their real-data counterparts, while positive values indicate the opposite. BayesNet produced 12 negative values, TVAE 8, and CTGAN 9. On average, BayesNet achieved the strongest prediction utility, with a mean balanced accuracy difference of –0.0287, reflecting superior generalization to the real-data holdout. CTGAN followed closely

Table 2: Drift detection results (p-values) for synthetic data generated with BayesNet, CTGAN, and TVAE. Chi-squared test was used for the binary Pass/Fail outcome and the K–S test for numerical features.

Technique		Accountancy			Global Economics		
		AY 18-19	**AY 19-20**	**AY 20-21**	**AY 18-19**	**AY 19-20**	**AY 20-21**
BayesNet	Pass/Fail outcome	**0**	**0**	**0.001**	**0**	**0**	**0**
	Bingeing of sessions	0.289	0.941	0.946	0.498	0.706	0.969
	Constancy of clicks	0.225	0.181	0.644	0.728	0.780	0.566
	Constancy of session length	0.611	0.762	0.648	0.676	0.861	0.929
	Median diff. Between active days	**0**	**0**	1	**0**	**0**	**0**
	Median number of actions per session	**0**	**0.002**	**0**	**0**	**0.001**	**0**
	Median number of active days per week	**0**	1	**0**	0.999	1	1
	Median session duration	0.199	0.745	0.902	0.236	0.761	0.503
	Proportion of active days	0.098	0.091	0.657	0.492	0.606	0.852
	Proportion of first-day-of-week activity	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of posts read	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of weeks with first-day activity	**0.049**	**0.001**	**0.042**	**0.020**	0.050	**0.047**
	Regularity of sessions	0.306	0.257	0.843	0.646	0.974	0.629
	Total number of created posts	0.936	0.135	0.054	0.931	1	1
	Total number of sessions	0.200	0.615	1	0.852	0.557	0.549
	Total sessions duration	0.742	0.451	0.893	0.249	0.823	0.831
	Uniformity of sessions	0.160	0.835	0.936	0.751	0.757	0.208
CTGAN	Pass/Fail outcome	**0**	**0**	0.052	**0**	**0**	0.653
	Bingeing of sessions	**0**	**0**	**0.001**	**0**	**0**	**0**
	Constancy of clicks	**0**	**0**	0.110	**0.002**	**0**	0.088
	Constancy of session length	**0.003**	**0**	**0**	**0**	0.051	**0**
	Median diff. Between active days	**0**	**0**	0.973	**0**	**0**	**0**
	Median number of actions per session	**0**	**0**	**0**	**0**	**0**	**0**
	Median number of active days per week	**0**	0.327	**0**	0.083	**0.011**	0.597
	Median session duration	**0.002**	**0**	**0**	**0**	**0**	**0**
	Proportion of active days	**0**	**0**	0.055	**0**	0.114	**0**
	Proportion of first-day-of-week activity	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of posts read	**0**	**0**	**0**	**0**	**0.008**	**0**
	Proportion of weeks with first-day activity	**0**	**0**	**0**	**0**	**0**	**0**
	Regularity of sessions	**0.001**	**0**	**0**	**0**	**0**	**0**
	Total number of created posts	0.459	0.080	0.520	0.808	1	0.703
	Total number of sessions	**0**	**0.016**	**0**	**0.032**	0.093	**0**
	Total sessions duration	**0**	**0**	**0**	**0**	**0**	**0**
	Uniformity of sessions	**0.017**	**0**	**0**	**0**	**0**	**0.007**
TVAE	Pass/Fail outcome	**0**	**0.007**	0.904	**0.019**	**0.023**	**0**
	Bingeing of sessions	**0**	**0**	**0**	**0**	**0**	**0**
	Constancy of clicks	**0**	**0**	**0**	**0**	**0**	**0**
	Constancy of session length	**0**	**0**	**0**	**0**	**0**	**0**
	Median diff. Between active days	**0**	**0**	**0.014**	**0**	**0**	**0**
	Median number of actions per session	**0**	**0**	**0**	**0**	**0**	**0**
	Median number of active days per week	**0**	0.701	**0**	0.118	0.128	0.696
	Median session duration	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of active days	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of first-day-of-week activity	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of posts read	**0**	**0**	**0**	**0**	**0**	**0**
	Proportion of weeks with first-day activity	**0**	**0**	**0**	**0**	**0**	**0**
	Regularity of sessions	**0**	**0**	**0**	**0**	**0**	**0**
	Total number of created posts	**0.027**	**0**	**0**	0.318	0.515	0.519
	Total number of sessions	**0**	**0**	**0**	**0**	**0**	**0**
	Total sessions duration	**0**	**0**	**0**	**0**	**0**	**0**
	Uniformity of sessions	**0**	**0**	**0**	**0**	**0**	**0**

(–0.0051), while TVAE was the only method with a positive mean difference (+0.0186), indicating weaker performance relative to the real-data baseline. Notably, this pattern suggests that higher statistical fidelity – as measured by the K-S test, mixed correlation matrix difference and pairwise mutual information difference – correlates with stronger generalization in this setting.

To assess the statistical significance of these differences, we applied the two-sided Wilcoxon signed-rank test to compare the distributions of metric values from models trained on real versus synthetic data. Results showed that BayesNet's scores differed significantly from those based on real data ($p = 0.027$), whereas neither TVAE ($p = 0.130$) nor CTGAN ($p = 0.899$) reached significance. To further test BayesNet's potential advantage, we conducted a one-sided Wilcoxon test with the alternative hypothesis H_1 : BayesNet > Real, which was also significant ($p = 0.013$). Taken together, these findings indicate that models trained on BayesNet-generated synthetic data can not only match, but in some cases surpass, the performance of models trained on real data.

RQ3 - Replication utility. Table 5 displays the agreement metrics calculated based on the SHAP global rankings built for the models trained and applied on synthetic data generated with BayesNet, CTGAN and TVAE models, respectively. Darker colors in the tables represent higher agreement. What can be seen immediately is that the strictness of the metrics (metric strictness refers to the degree to which a stability metric imposes additional conditions on agreement, with stricter metrics requiring not only feature overlap but also stronger alignment in ranking or relative ordering) that was demonstrated in the original paper [44] (as shown in Table 6) is preserved: rank agreement stays the strictest metric. Moreover, the pattern of decreasing agreement of updated models is preserved too: we see the highest agreement for the cases where the model is applied as-is, while

Table 3: Comparison of metrics across models and domains

Model	Mixed Correlation Matrix Difference						Pairwise Mutual Information Difference					
	Accountancy			Global Economics			Accountancy			Global Economics		
	AY1819	AY1920	AY2021	AY1819	AY1920	AY2021	AY1819	AY1920	AY2021	AY1819	AY1920	AY2021
BayesNet	0.8361	0.7919	0.8872	0.7803	0.7701	1.0073	5.6872	4.4731	4.4721	5.7675	5.2238	5.7349
CTGAN	1.4469	1.3459	1.8139	1.1738	1.2493	1.5497	5.4587	4.1415	4.5663	6.0585	5.4132	6.3879
TVAE	2.7583	2.8634	3.4395	2.5288	2.2621	2.7515	5.9134	4.6051	4.6336	6.0189	5.4282	5.7553

Table 4: Balanced accuracy scores for selected models trained on real and synthetic data across courses and years.

Course	Year	Algorithm	Difference (Real - Synthetic)		
			BayesNet	TVAE	CTGAN
Accountancy	1819	LogReg	-0.043	-0.031	-0.021
		GaussNB	-0.021	0.007	-0.017
		XGBoost	-0.025	0.062	-0.150
Accountancy	1920	LogReg	-0.114	0.026	-0.046
		GaussNB	-0.071	-0.021	-0.055
		XGBoost	-0.106	0.062	-0.057
Accountancy	2021	LogReg	0.004	-0.007	0.014
		GaussNB	0.002	-0.013	0.029
		XGBoost	-0.051	0.080	0.032
Global Economics	1819	LogReg	0.041	0.026	0.019
		GaussNB	0.007	-0.004	0.023
		XGBoost	-0.039	-0.009	0.014
Global Economics	1920	LogReg	-0.020	-0.014	-0.014
		GaussNB	0.040	0.070	-0.004
		XGBoost	-0.075	-0.001	0.190
Global Economics	2021	LogReg	0.003	0.038	0.038
		GaussNB	-0.002	0.004	-0.029
		XGBoost	-0.047	0.059	0.018
		AVERAGE	**-0.0287**	**0.0186**	**-0.0051**

the lowest agreement is achieved once the model's hyperparameters were updated and the model itself was retrained. Another observation that we make is the fact that the Naive Bayes model demonstrated the highest agreement, similarly to what was reported in [44].

Another finding of [44] concerns the differences in agreement metrics calculated in different external contexts. Table 7 illustrates this fact by presenting the differences in the stability metrics between the comparisons made for the AYs that share the same external context (COVID-19 vs. COVID-19 or non-COVID-19 vs. non-COVID-19), and the comparisons that do not share the same context (non-COVID-19 vs. COVID-19). In other words, for the Accountancy course offered in the first semester, the difference is calculated by subtracting the metrics of the comparison AY 2018–2019 vs. AY 2019–2020 (non-COVID-19 vs. non-COVID-19) from those of the comparison AY 2019–2020 vs. AY 2020–2021 (non-COVID-19 vs. COVID-19). For the Global Economics course offered in the second semester, the difference is calculated by substracting the metrics of the comparison AY 2019-2020 vs. AY 2020-2021 (COVID-19 vs. COVID-19) from those of the comparison AY 2018-2019 vs. AY 2019-2020 (non-COVID-19 vs. COVID-19).

These differences were found to be positive in [44], as a shared context was expected to result in higher agreement (bold values in Table 7). However, when examining stability across different academic years calculated using the data generated with SDGs, we do not observe the same pattern. Specifically, the stability metrics are not consistently higher for comparisons involving academic years with a shared external context (see red-highlighted values in Table 7).

Such behavior can be explained by the simplifying assumptions underlying the SDGs. In particular, SDGs based on probabilistic graphical models capture dependencies only among explicitly modeled variables (learning behavior in this paper); therefore, if the external context (e.g., a COVID-19 indicator) is not explicitly encoded as a conditioning variable, its indirect or latent influence on agreement patterns cannot be reproduced in the synthetic data. Second, the contextual signal itself may be relatively weak or diffuse in the feature space. The impact of COVID-19 is likely mediated through multiple behavioral changes (e.g., engagement patterns, assessment strategies), each contributing only partially to the overall effect. As a result, the signal is distributed across variables and may not be strongly identifiable by generative models optimizing global distributional fit. This leads to attenuation of context-specific effects in the synthetic data. Third, model-specific limitations further contribute to this behavior. For example, probabilistic graphical model-based SDGs (e.g., Bayesian Networks) encode dependencies through local conditional structures and may fail to capture higher-order or nonstationary effects spanning multiple variables and time periods. Neural generative models such as CTGAN can, in principle, model more complex interactions, which may explain their closer alignment with the findings of [44], but it still lacks explicit mechanisms to enforce consistency of context-dependent patterns across temporal slices.

Consequently, context-specific effects observed in real data are either attenuated or inconsistently represented in synthetic data, resulting in the absence of a systematic "shared context'' pattern. Indeed, across all SDGs, there is no consistent trend in the differences between the metric values for AY 2018–2019 vs. AY 2019–2020 and AY 2019–2020 vs. AY 2020–2021, reinforcing the conclusion that unmodeled contextual factors are not reliably preserved.

Another finding reported in [44] concerns stable learning indicators: these are the features that stay in the top important features irrespective of the context changes. They reported the overall level of activity features (Total number of sessions and Total session duration) as the most stable ones, followed by the features representing the regularity of study (Constancy of clicks and Constancy of session length). The replication using synthetic data reveals similar patterns, with overall level of activity features – Total number of sessions and Total session duration – emerging as the most stable, particularly for the CTGAN and TVAE models across SDGs. In contrast, the regularity of study features show less consistent alignment: Constancy of clicks is stable when data is generated using the BayesNet or TVAE, while Constancy of session length is stable with the CTGAN and TVAE models. Among all models, TVAE shows the closest alignment with the original findings of [44], particularly regarding stable features per course: for Accountancy, these include Total number of sessions, Total session duration, and Constancy of clicks; for Global Economics, the stable fea-

Table 5: Agreement metrics for BayesNet, CTGAN and TVAE. Darker colors represent higher agreement.

	AY 2018-2019 vs. AY 2019-2020									AY 2019-2020 vs. AY 2020-2021								
	Feature agreement			*Rank correlation*			*Rank agreement*			*Feature agreement*			*Rank correlation*			*Rank agreement*		
Model	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
BayesNet																		
Accountancy																		
NB	0.96	0.86	0.88	0.91	0.65	0.63	0.63	0.45	0.39	0.96	0.91	0.81	0.80	0.86	0.72	0.55	0.49	0.35
RF	0.93	0.76	0.75	0.84	-0.13	-0.10	0.50	0.09	0.10	0.91	0.64	0.61	0.51	0.08	0.03	0.21	0.08	0.11
LR	0.95	0.69	0.58	0.86	0.51	0.28	0.64	0.09	0.05	0.94	0.66	0.64	0.67	0.15	-0.09	0.31	0.08	0.05
SVM	0.84	0.73	0.69	0.62	0.21	0.16	0.38	0.13	0.14	0.76	0.73	0.70	0.74	0.34	0.36	0.34	0.19	0.18
KNN	0.85	0.73	0.73	0.69	0.24	0.24	0.31	0.05	0.05	0.89	0.79	0.80	0.68	0.65	0.68	0.28	0.18	0.18
MLP	0.86	0.80	0.61	0.71	0.38	0.04	0.29	0.14	0.05	0.90	0.81	0.63	0.31	0.37	0.21	0.19	0.16	0.15
XGB	0.89	0.75	0.74	0.62	-0.13	-0.14	0.34	0.13	0.10	0.93	0.64	0.63	0.59	0.10	0.04	0.30	0.06	0.10
TAB	0.89	0.66	0.63	0.69	0.03	0.01	0.31	0.08	0.06	0.83	0.63	0.63	0.39	0.06	-0.06	0.15	0.09	0.10
Global economics																		
NB	1.00	0.90	0.71	0.78	0.78	0.60	0.61	0.63	0.18	0.84	0.81	0.70	0.88	0.89	0.59	0.64	0.71	0.15
RF	0.91	0.61	0.60	0.81	0.58	0.50	0.55	0.26	0.25	0.89	0.69	0.64	0.86	0.39	0.36	0.59	0.28	0.28
LR	0.96	0.69	0.56	0.59	0.46	0.20	0.41	0.28	0.19	0.96	0.53	0.58	0.83	0.44	0.46	0.50	0.19	0.11
SVM	0.75	0.68	0.63	0.53	0.36	0.34	0.26	0.10	0.08	0.80	0.69	0.73	0.71	0.25	0.19	0.26	0.10	0.11
KNN	0.88	0.74	0.78	0.74	0.60	0.60	0.49	0.29	0.29	0.78	0.73	0.69	0.51	0.41	0.36	0.31	0.20	0.21
MLP	0.78	0.73	0.56	0.53	0.45	0.39	0.38	0.29	0.21	0.84	0.79	0.66	0.72	0.57	0.37	0.39	0.29	0.16
XGB	0.86	0.64	0.63	0.66	0.41	0.39	0.39	0.25	0.21	0.85	0.60	0.59	0.66	0.49	0.49	0.41	0.31	0.29
TAB	0.81	0.64	0.66	0.46	0.16	0.15	0.30	0.14	0.19	0.83	0.69	0.68	0.66	0.26	0.28	0.33	0.19	0.18
CTGAN																		
Accountancy																		
NB	0.99	0.99	0.99	0.84	0.84	0.84	0.55	0.66	0.66	0.98	0.96	0.98	0.82	0.62	0.65	0.48	0.20	0.28
RF	0.91	0.80	0.74	0.74	0.14	0.12	0.41	0.13	0.13	0.85	0.60	0.61	0.71	0.31	0.22	0.25	0.09	0.15
LR	0.98	0.80	0.70	0.89	0.16	0.22	0.70	0.05	0.09	0.98	0.83	0.55	0.78	0.56	0.26	0.48	0.33	0.16
SVM	0.93	0.76	0.71	0.74	0.14	0.16	0.36	0.11	0.04	0.88	0.71	0.59	0.46	0.20	-0.01	0.21	0.19	0.05
KNN	0.88	0.84	0.84	0.71	0.27	0.26	0.36	0.10	0.13	0.84	0.65	0.66	0.75	0.58	0.54	0.31	0.16	0.18
MLP	0.91	0.86	0.69	0.61	0.48	0.16	0.23	0.21	0.09	0.94	0.86	0.56	0.50	0.59	0.49	0.21	0.24	0.10
XGB	0.90	0.68	0.68	0.58	0.09	0.05	0.44	0.13	0.09	0.85	0.58	0.59	0.70	0.16	0.06	0.38	0.08	0.09
TAB	0.89	0.54	0.59	0.65	-0.03	-0.01	0.43	0.05	0.01	0.84	0.55	0.59	0.50	0.20	0.24	0.19	0.09	0.11
Global economics																		
NB	0.74	0.73	0.70	0.75	0.71	0.59	0.51	0.46	0.24	0.99	0.98	0.80	0.89	0.63	0.64	0.64	0.45	0.25
RF	0.84	0.70	0.66	0.74	0.23	0.29	0.34	0.04	0.06	0.90	0.54	0.53	0.54	0.32	0.29	0.25	0.13	0.11
LR	0.93	0.79	0.73	0.65	0.48	0.29	0.30	0.19	0.14	0.96	0.81	0.78	0.76	0.31	0.17	0.49	0.16	0.10
SVM	0.89	0.68	0.63	0.56	0.04	0.08	0.34	0.11	0.04	0.79	0.66	0.66	0.52	0.24	0.16	0.25	0.14	0.10
KNN	0.90	0.61	0.60	0.74	0.25	0.24	0.39	0.08	0.10	0.78	0.64	0.61	0.64	0.18	0.16	0.40	0.10	0.11
MLP	0.89	0.80	0.54	0.38	0.37	0.36	0.16	0.18	0.05	0.89	0.73	0.63	0.54	0.14	0.16	0.40	0.15	0.19
XGB	0.84	0.56	0.61	0.59	0.24	0.23	0.31	0.05	0.09	0.83	0.49	0.54	0.64	0.31	0.29	0.28	0.09	0.13
TAB	0.81	0.58	0.60	0.38	0.11	0.11	0.16	0.05	0.06	0.83	0.68	0.60	0.36	0.09	0.26	0.14	0.06	0.08
TVAE																		
Accountancy																		
NB	1.00	0.98	0.99	0.79	0.64	0.67	0.43	0.39	0.45	0.96	0.95	0.96	0.88	0.82	0.81	0.60	0.54	0.58
RF	0.90	0.68	0.63	0.63	0.10	0.21	0.28	0.09	0.09	0.81	0.63	0.64	0.43	0.26	0.25	0.16	0.06	0.09
LR	0.88	0.64	0.55	0.81	0.09	0.10	0.50	0.03	0.05	0.93	0.75	0.69	0.81	0.36	0.36	0.43	0.20	0.19
SVM	0.83	0.66	0.70	0.51	0.31	0.19	0.16	0.10	0.09	0.70	0.58	0.55	0.54	0.15	0.11	0.10	0.15	0.05
KNN	0.88	0.70	0.73	0.62	0.26	0.25	0.28	0.13	0.14	0.79	0.76	0.76	0.52	0.39	0.39	0.14	0.10	0.14
MLP	0.85	0.71	0.64	0.62	0.44	-0.01	0.19	0.19	0.13	0.79	0.68	0.55	0.54	0.29	0.21	0.18	0.10	0.09
XGB	0.88	0.64	0.71	0.55	0.22	0.12	0.29	0.04	0.05	0.75	0.64	0.65	0.45	0.29	0.24	0.18	0.05	0.04
TAB	0.84	0.63	0.61	0.48	0.09	0.05	0.18	0.09	0.08	0.78	0.64	0.65	0.31	0.17	0.09	0.15	0.05	0.04
Global economics																		
NB	0.95	0.93	0.94	0.82	0.67	0.67	0.59	0.44	0.44	0.99	0.95	0.91	0.85	0.62	0.56	0.58	0.31	0.28
RF	0.81	0.63	0.63	0.54	0.36	0.34	0.31	0.06	0.09	0.85	0.54	0.53	0.33	0.01	-0.01	0.10	0.09	0.06
LR	0.93	0.66	0.59	0.73	0.42	0.34	0.46	0.20	0.13	0.88	0.70	0.66	0.73	0.19	0.11	0.28	0.04	0.06
SVM	0.94	0.55	0.65	0.74	0.19	0.13	0.43	0.05	0.05	0.80	0.70	0.61	0.46	0.14	-0.02	0.13	0.14	0.11
KNN	0.93	0.86	0.86	0.72	0.33	0.14	0.45	0.18	0.13	0.90	0.75	0.71	0.44	0.19	0.19	0.14	0.09	0.06
MLP	0.95	0.86	0.71	0.62	0.66	0.23	0.26	0.21	0.11	0.78	0.76	0.65	0.49	0.19	0.10	0.18	0.09	0.15
XGB	0.80	0.58	0.59	0.57	0.39	0.31	0.19	0.11	0.05	0.84	0.55	0.53	0.48	0.15	0.09	0.21	0.04	0.08
TAB	0.84	0.63	0.73	0.47	0.31	0.19	0.18	0.13	0.06	0.84	0.65	0.65	0.15	0.09	0.11	0.11	0.11	0.10

Table 6: Agreement metrics: original paper by Tiukhova et al. [44]

	AY 18-19 vs. AY 19-20									AY 19-20 vs. AY 20-21								
	Feature agreement			*Rank correlation*			*Rank agreement*			*Feature agreement*			*Rank correlation*			*Rank agreement*		
Model	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
Accountancy																		
NB	1,00	0,91	0,91	0,84	0,69	0,66	0,58	0,40	0,38	0,93	0,89	0,85	0,74	0,69	0,69	0,28	0,25	0,28
RF	0,93	0,86	0,88	0,89	0,63	0,65	0,56	0,20	0,23	0,93	0,65	0,69	0,81	0,11	0,14	0,60	0,06	0,09
LR	0,95	0,78	0,68	0,85	0,65	0,22	0,59	0,24	0,11	0,94	0,70	0,68	0,76	0,39	0,31	0,40	0,18	0,13
SVM	0,95	0,93	0,78	0,73	0,35	0,28	0,31	0,18	0,15	0,93	0,59	0,63	0,79	0,31	0,25	0,45	0,15	0,10
KNN	0,94	0,86	0,84	0,74	0,49	0,49	0,46	0,19	0,20	0,90	0,81	0,81	0,79	0,29	0,29	0,41	0,13	0,15
MLP	0,86	0,80	0,69	0,73	0,66	0,36	0,40	0,29	0,19	0,93	0,81	0,60	0,82	0,68	0,54	0,46	0,35	0,20
XGB	0,91	0,79	0,80	0,67	0,32	0,36	0,33	0,15	0,16	0,90	0,54	0,56	0,81	0,18	0,21	0,44	0,10	0,11
TAB	0,90	0,61	0,61	0,80	0,36	0,34	0,46	0,05	0,10	0,85	0,59	0,58	0,61	0,11	0,22	0,24	0,16	0,10
Global Economics																		
NB	0,95	0,84	0,70	0,66	0,50	0,33	0,51	0,36	0,24	0,88	0,85	0,73	0,82	0,77	0,66	0,73	0,70	0,26
RF	0,93	0,81	0,78	0,89	0,47	0,35	0,64	0,26	0,23	0,91	0,81	0,76	0,83	0,66	0,66	0,60	0,38	0,38
LR	0,89	0,75	0,68	0,65	0,57	0,29	0,39	0,33	0,23	0,96	0,65	0,58	0,89	0,47	0,31	0,66	0,26	0,18
SVM	0,93	0,68	0,69	0,77	0,36	0,41	0,50	0,23	0,13	0,88	0,74	0,83	0,88	0,40	0,35	0,56	0,21	0,14
KNN	0,89	0,81	0,79	0,86	0,66	0,61	0,59	0,29	0,33	0,88	0,78	0,76	0,71	0,46	0,49	0,34	0,28	0,24
MLP	0,89	0,81	0,68	0,76	0,67	0,49	0,44	0,38	0,26	0,94	0,89	0,61	0,84	0,78	0,55	0,48	0,46	0,30
XGB	0,88	0,76	0,74	0,66	0,39	0,35	0,36	0,19	0,21	0,95	0,76	0,76	0,85	0,58	0,58	0,59	0,36	0,36
TAB	0,84	0,59	0,69	0,65	0,24	0,32	0,35	0,16	0,16	0,86	0,66	0,65	0,62	0,35	0,25	0,29	0,11	0,18

Table 7: Stability metrics differences for changing contexts: original vs. synthetic data.

Course	Data	*Feature agreement*			*Rank correlation*			*Rank agreement*		
		(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
Accountancy	*Original data*	**0.05**	**0.14**	**0.13**	**0.01**	**0.16**	**0.13**	**0.11**	**0.10**	**0.10**
	BayesNet	0.01	0.02	0.02	0.16	-0.11	-0.10	0.13	-0.02	-0.03
	CTGAN	0.03	0.07	0.10	0.07	-0.14	-0.08	0.12	0.01	0.01
	TVAE	0.07	0.00	0.01	0.06	-0.08	-0.11	0.05	-0.03	-0.02
Global Economics	*Original data*	**0.03**	**0.04**	**0.01**	**0.07**	**0.08**	**0.15**	**0.11**	**0.11**	**0.08**
	BayesNet	-0.02	-0.01	0.02	0.09	-0.01	-0.01	0.00	0.00	-0.01
	CTGAN	0.02	0.01	0.01	0.01	-0.03	-0.01	0.04	0.02	0.04
	TVAE	-0.03	-0.01	-0.05	-0.16	-0.22	-0.15	-0.14	-0.06	-0.02

tures are Total number of sessions, Total session duration, and Constancy of session length.

5. DISCUSSION

When evaluating the feature distributions in the data (RQ1) and the prediction utility of synthetic data (RQ2), we found that the BayesNet consistently outperformed other SDGs. This model was able not only to generate data that closely matched the distribution of the real dataset (both marginal distributions and multivariate dependencies), but also to produce synthetic data that could be effectively used to train ML models for student success prediction. While close resemblance to real data may increase the risk of exposing sensitive information when identifiable or sensitive attributes are present [25], such resemblance can be advantageous in settings where only behavioral data is used (as in this study), as it enhances the practical utility of synthetic data in scenarios where access to real data is limited or where data sharing and cross-institutional collaboration are required. These results suggest that simpler statistical models can perform just as well as, or even better than, more complex deep learning approaches for synthetic data generation in LA. Consistent with our findings, prior research has also shown that statistical models – such as Bayesian networks and Gaussian multivariate/copula methods – can achieve performance on par with models trained on real data [9, 51, 25]. Nevertheless, for categorical features, such as the binary study outcome, reweighting or resampling strategies may be necessary to stabilize the estimation of the outcome distribution and improve the fidelity of the synthetic data.

These insights extend beyond the field of LA. For example, Gunnarsson et al. [14] reported similar results when benchmarking deep learning methods for credit scoring, where simpler models were found to perform more reliably. Likewise, Dacrema et al. [10] questioned the added value of deep learning in recommender systems, highlighting cases where traditional methods offered more effective solutions.

Regarding the replication utility (RQ3), no single SDG consistently outperformed the others. Across the agreement metrics, all SDGs yielded results comparable to those reported in the original study [44]. The stable learning indicators identified with TVAE align most closely with the original findings, which contrasts with the conclusions from RQ1 and RQ2 suggesting the superiority of BayesNet. Nonetheless, the stable indicators derived from both the BayesNet and CTGAN also remain broadly consistent with the results of [44]. All the SDGs failed to replicate the COVID-19 effects reported by [44].

These findings suggest that even when an SDG performs well in terms of statistical resemblance and prediction utility, its replication utility may depend on the scope and depth of the replication being conducted. In our case, the differences in external COVID-19 effects across academic years that were evident in the real data did not fully emerge in the synthetic data. This may be attributable either to the effect sizes themselves – which were not explicitly examined in [44] – or to the fact that SDGs are unable to capture external and unpredictable factors such as COVID-19.

Limitations and Future work We recognize several limitations in this study. First, we employed a restricted set of SDGs, omitting potentially valuable models such as LLMs. We mitigated this limitation by focusing on the most widely used models in the literature with demonstrated performance – an aspect that LLMs currently lack in this context. Second, our analysis was limited in scope with respect to the range of ML models and statistical tests applied as well as two courses only, leaving a more comprehensive exploration for future research. Finally, we reproduced only the model agreement stability analysis of [44], excluding prediction drift and model performance stability analyses. We consider these to be promising directions for future investigation. Another future work direction is exploring more principled synthetic data generation strategies that explicitly account for heterogeneity across course contexts and learners, for example by modeling within-person behavioral patterns and hierarchical distributions (e.g., learners nested within courses) prior to aggregation. Such approaches may better preserve contextual effects and intra-learner structure, which have been shown to be critical for both predictability and explainability in educational data.

6. CONCLUSION

The fields of LA and EDM have shown how education can benefit from data-driven decision-making, particularly when aligned with pedagogical theories such as SRL. However, ethical and privacy concerns often limit access to authentic learning data, posing challenges for open science and collaborative research. Synthetic data – designed to statistically mirror real data – offers a promising alternative. Prior studies have primarily assessed the utility of synthetic data for its distributional similarity to real data and the predictive performance of ML models trained on it, compared with those trained on real data. This study extended prior research by evaluating the use of synthetic data for reproducing existing research, a crucial step toward reproducible LA. Grounded in SRL theory, our investigation highlighted how synthetic data can bridge pedagogy and the use of SDGs. To further support open LA, we provide a synthetic dataset that addresses the scarcity of publicly available LA data.

Table 8: Stable indicators across models

Accountancy		Global Economics	
AY 18-19 vs. AY 19-20	*AY 19-20 vs. AY 20-21*	*AY 18-19 vs. AY 19-20*	*AY 19-20 vs. AY 20-21*
BayesNet			
Prop. of weeks with first-day activity (7/8)	Total sessions duration (6/8)	Total sessions duration (8/8)	Total sessions duration (8/8)
Total sessions duration (6/8)	Prop. of weeks with first-day activity (5/8)	Uniformity of sessions (4/8)	Prop. of posts read (5/8)
Constancy of clicks (4/8)	Bingeing of sessions (4/8)	Constancy of clicks (2/8)	Constancy of clicks (2/8)
Bingeing of sessions (4/8)	Constancy of clicks (3/8)	Prop. of weeks with first-day activity (2/8)	Prop. of weeks with first-day activity (2/8)
Total number of sessions (2/8)	Median #active days per week (2/8)	Median #actions per session (1/8)	Median session duration (1/8)
CTGAN			
Constancy of session length (7/8)	Total sessions duration (7/8)	Total sessions duration (8/8)	Total sessions duration (8/8)
Median #active days per week (5/8)	Median #active days per week (6/8)	Uniformity of sessions (3/8)	Constancy of session length (4/8)
Total sessions duration (5/8)	Prop. of active days (3/8)	Constancy of session length (3/8)	Total number of sessions (2/8)
Total number of sessions (3/8)	Prop. of weeks with first-day activity (3/8)	Total number of sessions (2/8)	Median #actions per session (2/8)
Prop. of weeks with first-day activity (3/8)	Median #actions per session (2/8)	Median #actions per session (2/8)	Prop. of posts read (2/8)
TVAE			
Constancy of session length (5/8)	Total number of sessions (7/8)	Total sessions duration (7/8)	Total sessions duration (6/8)
Prop. of active days (5/8)	Prop. of active days (6/8)	Constancy of session length (6/8)	Total number of sessions (5/8)
Constancy of clicks (3/8)	Median #active days per week (5/8)	Total number of sessions (3/8)	Constancy of session length (3/8)
Total number of sessions (2/8)	Total sessions duration (3/8)	Prop. of active days (3/8)	Median #active days per week (2/8)
Total sessions duration (2/8)	Constancy of clicks (2/8)	Median #active days per week (2/8)	Regularity of sessions (2/8)
Original Paper [44]			
Total sessions duration (8/8)	Total sessions duration (8/8)	Total sessions duration (8/8)	Total sessions duration (8/8)
Total number of sessions (6/8)	Constancy of clicks (5/8)	Total number of sessions (6/8)	Prop. of posts read (7/8)
Prop. of weeks with first-day activity (6/8)	Median #active days per week (4/8)	Constancy of session length (3/8)	Total number of sessions (5/8)
Constancy of clicks (5/8)	Total number of sessions (3/8)	Prop. of posts read (3/8)	Prop. of active days (4/8)
Median #active days per week (4/8)	Prop. of active days (3/8)	Median diff. between active days (1/8)	Constancy of session length (3/8)

First, we demonstrated that synthetic data can effectively approximate the statistical properties of real data, with BayesNet achieving the strongest performance among the models tested. Second, student success prediction models trained on BayesNet-generated synthetic data achieved the highest predictive accuracy – at times even surpassing that of models trained on real data. Together, these findings highlight the advantage of statistical SDGs over deep learning–based approaches. Finally, all SDGs proved useful for reproducing the stability analysis conducted in [44], yielding stability metrics and stable features comparable to those in the original study. However, none of the SDGs successfully replicated the COVID-19 effects observed in the original research. This suggests that creating a versatile synthetic dataset capable of fully capturing external learning contexts remains highly challenging. Further research is needed to determine how best to select generative models that align with specific LA use cases. Nevertheless, the paper demonstrates how synthetic datasets can be used in the context of XAI stability, revealing aspects of their usefulness that are not captured by standard quality metrics.

Data availability The synthetic datasets used in this study have been made publicly available via Zenodo [42].

Acknowledgements This research was supported by KU Leuven's Special Research Fund (BOF) grant number IDN/24/004.

7. REFERENCES

[1] A. Abu Saa, M. Al-Emran, and K. Shaalan. Factors affecting students' performance in higher education: a systematic review of predictive data mining techniques. *Tech. Knowl. Learn.*, 24:567–598, 2019.

[2] S. O. Arik and T. Pfister. Tabnet: Attentive interpretable tabular learning. *Proc. AAAI Conference on Artificial Intelligence*, 35(8):6679–6687, May 2021.

[3] I. Ben-Gal. Bayesian networks. *Encyclopedia of statistics in quality and reliability*, 2008.

[4] A. M. Berg, S. T. Mol, G. Kismihók, and N. Sclater. The role of a reference synthetic data generator within the field of learning analytics. *J. Learn. Anal.*, 3(1):107–128, 2016.

[5] L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[6] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794. ACM, Aug. 2016.

[7] W. Chen, C. G. Brinton, D. Cao, A. Mason-Singh, C. Lu, and M. Chiang. Early detection prediction of learning outcomes in online short-courses via learning behaviors. *IEEE Trans. Learn. Technol.*, 12(1):44–58, 2018.

[8] T. Cover and P. Hart. Nearest neighbor pattern

classification. *IEEE Trans. Inf. Theory*, 13(1):21–27, 1967.

[9] M. Dorodchi, E. Al-Hossami, A. Benedict, and E. Demeter. Using synthetic data generators to promote open science in higher education learning analytics. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 4672–4675. IEEE, 2019.

[10] M. Ferrari Dacrema, P. Cremonesi, and D. Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM conference on recommender systems*, pages 101–109, 2019.

[11] B. Flanagan, R. Majumdar, and H. Ogata. Fine grain synthetic educational data: challenges and limitations of collaborative learning analytics. *IEEE Access*, 10:26230–26241, 2022.

[12] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[13] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[14] B. R. Gunnarsson, S. Vanden Broucke, B. Baesens, M. Óskarsdóttir, and W. Lemahieu. Deep learning for credit scoring: Do or don't? *Eur. J. Oper. Res.*, 295(1):292–305, 2021.

[15] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang. Xai—explainable artificial intelligence. *Sci. Robot.*, 4(37):eaay7120, 2019.

[16] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. Support vector machines. *IEEE Intell. Syst. Appl.*, 13(4):18–28, 1998.

[17] J. Jovanović, M. Saqr, S. Joksimović, and D. Gašević. Students matter the most in learning analytics: The effects of internal and instructional conditions in predicting academic success. *Comput. Educ.*, 172:104251, 2021.

[18] M. Khalil, F. Vadiee, R. Shakya, and Q. Liu. Creating artificial students that never existed: Leveraging large language models and CTGANs for synthetic data generation. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK '25, page 439–450, New York, NY, USA, 2025. Association for Computing Machinery.

[19] H. Khosravi, S. B. Shum, G. Chen, C. Conati, Y.-S. Tsai, J. Kay, S. Knight, R. Martinez-Maldonado, S. Sadiq, and D. Gašević. Explainable artificial intelligence in education. *Comput. Educ.: Artif. Intell.*, 3:100074, 2022.

[20] D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2022.

[21] A. Kolmogoroff. Confidence limits for an unknown distribution function. *Ann. Math. Stat.*, 12(4):461–463, 1941.

[22] S. Krishna, T. Han, A. Gu, S. Wu, S. Jabbari, and H. Lakkaraju. The disagreement problem in explainable machine learning: A practitioner's perspective. *Trans. Mach. Learn. Res.*, 2024.

[23] A. D. Lautrup, T. Hyrup, A. Zimek, and P. Schneider-Kamp. Syntheval: a framework for detailed utility and privacy evaluation of tabular synthetic data. *Data Min. Knowl. Discov.*, 39(1):6, Dec. 2024.

[24] Q. Liu, O. Deho, F. Vadiee, M. Khalil, S. Joksimovic, and G. Siemens. Can synthetic data be fair and private? a comparative study of synthetic data generation and fairness algorithms. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK '25, page 591–600, New York, NY, USA, 2025. Association for Computing Machinery.

[25] Q. Liu, M. Khalil, J. Jovanovic, and R. Shakya. Scaling while privacy preserving: A comprehensive synthetic tabular data generation and evaluation in learning analytics. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, LAK '24, page 620–631, New York, NY, USA, 2024. Association for Computing Machinery.

[26] N. G. López Flores, A. S. Islind, and M. Óskarsdóttir. Effects of the COVID-19 pandemic on learning and teaching: A case study from higher education. *arXiv preprint arXiv:2105.01432*, 2021.

[27] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang. Learning under concept drift: A review. *IEEE Trans. Knowl. Data Eng.*, 31(12):2346–2363, 2019.

[28] N. Lu, G. Zhang, and J. Lu. Concept drift detection via competence models. *Artif. Intell.*, 209:11–28, 2014.

[29] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[30] W. Matcha, N. A. Uzir, D. Gašević, and A. Pardo. A systematic review of empirical studies on learning analytics dashboards: A self-regulated learning perspective. *IEEE Trans. Learn. Technol.*, 13(2):226–245, Apr. 2020.

[31] A. Mathrani, T. Susnjak, G. Ramaswami, and A. Barczak. Perspectives on the challenges of generalizability, transparency and ethics in predictive learning analytics. *Comput. Educ. Open*, 2:100060, 2021.

[32] J. A. Nelder and R. W. M. Wedderburn. Generalized linear models. *J. R. Stat. Soc.*, 135(3):370–384, 1972.

[33] N. Patki, R. Wedge, and K. Veeramachaneni. The synthetic data vault. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 399–410, Oct 2016.

[34] K. Pearson. X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Lond. Edinb. Dubl. Phil. Mag.*, 50(302):157–175, 1900.

[35] Z. Qian, B.-C. Cebere, and M. van der Schaar. Synthcity: facilitating innovative use cases of synthetic data in different data modalities, 2023.

[36] S. Russell and P. Norvig. *Artificial Intelligence, Global Edition A Modern Approach*. Pearson Deutschland,

2021.

[37] C. Sammut and G. I. Webb. *Encyclopedia of machine learning*. Springer Science & Business Media, 2011.

[38] M. Saqr, J. Jovanovic, O. Viberg, and D. Gašević. Is there order in the mess? A single paper meta-analysis approach to identification of predictors of success in learning analytics. *Stud. High. Educ.*, 47(12):2370–2391, 2022.

[39] G. Siemens and P. Long. Penetrating the fog: Analytics in learning and education. *EDUCAUSE review*, 46(5):30, 2011.

[40] V. K. Singh, B. Bozkaya, and A. Pentland. Money walks: Implicit mobility behavior and financial well-being. *PLOS ONE*, 10(8):1–17, 08 2015.

[41] D. T. Tempelaar, B. Rienties, and B. Giesbers. Verifying the stability and sensitivity of learning analytics based prediction models: An extended case study. In S. Zvacek, M. T. Restivo, J. Uhomoibhi, and M. Helfert, editors, *Computer Supported Education*, pages 256–273, Cham, 2016. Springer International Publishing.

[42] E. Tiukhova, G. Meller, D. Van Landuyt, T. De Laet, B. Baesens, and M. Snoeck. Synthetic datasets: Real enough to matter? implications of synthetic data for reproducible learning analytics. `https://doi.org/10.5281/zenodo.17220489`, Sept. 2025.

[43] E. Tiukhova, D. Van Landuyt, B. Baesens, and M. Snoeck. Open data, private learners: a de-identified student activity and performance dataset for learning analytics. *Scientific Data*, 13(1):548, 2026.

[44] E. Tiukhova, P. Vemuri, N. L. Flores, A. S. Islind, M. Óskarsdóttir, S. Poelmans, B. Baesens, and M. Snoeck. Explainable learning analytics: Assessing the stability of student success prediction models by means of explainable AI. *Decis. Support Syst.*, 182:114229, 2024.

[45] E. Tiukhova, P. Vemuri, M. Óskarsdóttir, S. Poelmans, B. Baesens, and M. Snoeck. Discovering unusual study patterns using anomaly detection and XAI. In *Proc. HICSS*, pages 1427–1436, 2024.

[46] E. Tiukhova, C. Verbruggen, T. De Laet, B. Baesens, and M. Snoeck. Learning analytics dashboard with peer comparison for student feedback in conceptual modeling education. In *International Conference on Business Process Modeling, Development and Support*, pages 301–317. Springer, 2025.

[47] F. Wilcoxon. Individual comparisons by ranking methods. *Biometr. Bull.*, 1(6):80–83, 1945.

[48] P. Winne. Learning Analytics for Self-Regulated Learning. In C. Lang, G. Siemens, A. F. Wise, and D. Gaševic, editors, *The Handbook of Learning Analytics*, pages 241–249. Society for Learning Analytics Research (SoLAR), Alberta, Canada, 1 edition, 2017.

[49] P. H. Winne, R. S. Baker, et al. The potentials of educational data mining for researching metacognition, motivation and self-regulated learning. *J. Educ. Data Min.*, 5(1):1–8, 2013.

[50] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32, 2019.

[51] C. Zhan, O. B. Deho, X. Zhang, S. Joksimovic, and M. de Laat. Synthetic data generator for student data serving learning analytics: A comparative study. *Learn. Lett.*, 1:5–5, 2023.

[52] Y. Zhu, Z. Zhao, R. Birke, and L. Y. Chen. Permutation-invariant tabular data synthesis. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 5855–5864. IEEE, 2022.

[53] B. J. Zimmerman. Self-regulated learning and academic achievement: An overview. *Educ. Psychol.*, 25(1):3–17, 1990.

[54] B. J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory Pract.*, 41(2):64–70, 2002.

[55] R. P. Álvarez, I. Jivet, M. Pérez-Sanagustín, M. Scheffel, and K. Verbert. Tools designed to support self-regulated learning in online learning environments: A systematic review. *IEEE Trans. Learn. Technol.*, 15(4):508–522, 2022.

APPENDIX

A. TARGET FEATURE DISTRIBUTION

Table 9: Pass/fail distribution across real and synthetic datasets used in the experiments.

Course	Year	Dataset	%P	%F	N
Accountancy	1819	Real	70.24	29.76	756
		BayesNet	55.92	44.08	726
		CTGAN	51.52	48.48	726
		TVAE	53.17	46.83	726
Accountancy	1920	Real	67.79	32.21	711
		BayesNet	54.57	45.43	700
		CTGAN	57.86	42.14	700
		TVAE	60.86	39.14	700
Accountancy	2021	Real	70.07	29.93	725
		BayesNet	61.70	38.30	718
		CTGAN	74.65	25.35	718
		TVAE	69.78	30.22	718
Global Economics	1819	Real	56.26	43.74	743
		BayesNet	44.04	55.96	722
		CTGAN	42.80	57.20	722
		TVAE	50.14	49.86	722
Global Economics	1920	Real	75.48	24.52	681
		BayesNet	60.95	39.05	676
		CTGAN	65.09	34.91	676
		TVAE	69.97	30.03	676
Global Economics	2021	Real	56.51	43.49	676
		BayesNet	44.56	55.44	662
		CTGAN	55.29	44.71	662
		TVAE	42.30	57.70	662

Stability and Contextual Sensitivity of Keystroke Process Features in Longitudinal Student Writing Portfolios

Damilola Babalola
North Carolina State University
djbabalo@ncsu.edu

Naga Buddarapu
North Carolina State University
vbuddar@ncsu.edu

Piotr Mitros
Educational Testing Service
pmitros@ets.org

Paul Deane
Educational Testing Service
pdeane@ets.org

Collin Lynch
North Carolina State University
cflynch@ncsu.edu

ABSTRACT

This study examines the stability and instructional context sensitivity of keystroke writing process features across a full academic year of middle school writing. Using 8,576 in-class writing sessions collected from two U.S. middle schools, we analyze whether writing-process indicators remain consistent within students over time and how they vary across instructional contexts. We define stability as within-student consistency of writing features across tasks and sensitivity as systematic variation associated with grade level and school. A broad set of features capturing fluency, pausing, revision, and engagement was extracted from students' longitudinal writing portfolios. We quantified stability using intraclass correlation coefficients and examined contextual sensitivity using linear mixed-effects models with random student-level intercepts. Results show that most features exhibit low stability within School x Grade cohorts, indicating substantial within-student variability across tasks. Only a small subset of low-level timing features related to word-level text production demonstrated moderate stability, whereas many revision- and engagement-related features varied systematically across instructional contexts. These findings clarify the measurement properties of keystroke-derived writing indicators and inform their appropriate use in longitudinal educational data mining applications.

Keywords

Keystroke logging, Writing process analytics, Longitudinal study, Feature stability, Contextual sensitivity

1. INTRODUCTION

Writing is a complex cognitive activity that involves coordinating planning, text production, and revision [13]. In educational settings, understanding how students engage in these processes is essential for assessing writing development and supporting instruction. However, traditional writing research has largely relied on final texts or observational methods, which provide limited insight into how writing unfolds over time [23]. Recent advances in keystroke logging technologies enable fine-grained capture of writers' real-time behaviors by recording keystrokes, pauses, and edits during composition [2, 22]. This data offers direct access to writing processes that are invisible in finished texts. These processes however, have become increasingly important in educational data mining (EDM) for applications such as learner modeling, performance prediction, and process-based feedback [7, 14].

Prior work shows that keystroke-derived features are educationally meaningful [37]. More proficient writers tend to produce text more fluently and pause strategically, whereas less proficient writers pause more frequently and revise differently [1, 33]. At the same time, writing processes are highly context-sensitive. Features such as total writing time, revision frequency, and output vary substantially across tasks, genres, and instructional conditions, while only a subset of low-level timing measures exhibit moderate stability across occasions [8, 32].

These findings highlight a fundamental measurement challenge for longitudinal writing analytics: some process features reflect relatively persistent student-level tendencies, whereas others primarily capture contextual adaptation [8]. This distinction matters even within a single instructional setting. As students encounter different tasks, genres, and instructional demands across an academic year, observed changes in their writing process features may reflect shifts in task context rather than genuine developmental growth. Without first establishing which features are stable within students across tasks and which vary systematically with context, longitudinal analyses risk conflating developmental change with task-specific effects, leading to misleading inferences about writing growth [2]. This motivates the present study's focus on measurement before modeling, characterizing the stability and contextual sensitivity of keystroke-derived features as a foundation for future longitudinal work.

This challenge is particularly salient for educational applications that aim to model writing development over time or deploy process-based analytics in classrooms [7, 14]. Be-

Damilola Babalola, Naga Buddarapu, Piotr Mitros, Paul Deane, and Collin Lynch. Stability and Contextual Sensitivity of Keystroke Process Features in Longitudinal Student Writing Portfolios. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 256–266. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039819

fore modeling growth or generating feedback, it is necessary to establish whether keystroke-derived features are interpretable across writing samples [2, 32]. Accordingly, this study adopts a measurement-focused perspective that precedes growth modeling. Rather than modeling change directly, we examine the stability and contextual sensitivity of writing process indicators in authentic classroom settings.

We address this goal through two research questions:

- **RQ1:** To what extent do keystroke-derived writing process indicators exhibit within-student consistency across tasks in students' longitudinal writing portfolios?
- **RQ2:** To what extent does variability in these indicators reflect systematic sensitivity to instructional context, specifically grade level, incoming academic proficiency, and school membership?

To answer these questions, we analyze longitudinal keystroke data collected from everyday classroom writing across an entire school year in two middle schools. By characterizing which process features exhibit within-cohort stability and which vary systematically with context, this work provides a principled foundation for longitudinal learner modeling and process-aware educational data mining in real-world instructional environments [2, 36].

2. BACKGROUND

2.1 Writing process features

Research on writing processes has consistently shown that keystroke-derived features capture meaningful aspects of how students compose text, including productivity, fluency, pausing, and revision behavior [22, 33]. Across populations, more proficient writers tend to write more fluently, produce longer bursts of text, and pause less frequently, reflecting more efficient translation of ideas into language [15, 18]. Fluency-related measures, in particular, have been repeatedly linked to writing quality and performance in both classroom and assessment contexts [5].

Pausing behavior provides additional insight into cognitive processing during composition. Skilled writers typically exhibit shorter pauses at character, word, and sentence boundaries. This is consistent with more automatic lexical access and smoother coordination between planning and production [26, 35]. Burst length, the amount of text produced between pauses, has emerged as a robust indicator of writing fluency, with longer bursts associated with higher-quality writing across multiple studies [2, 5].

Revision behavior presents a more complex picture. Prior work suggests that stronger writers engage in more strategic, higher-level revisions that reorganize or clarify meaning, while less skilled writers tend to make frequent surface-level edits during drafting [11, 34]. Although revision features are informative, their interpretation often depends strongly on task demands and instructional context [9]. Together, this body of work demonstrates that keystroke-derived features capture meaningful variation in writing processes and motivate applications in learner modeling, performance prediction, and process-based feedback within educational data mining [7, 16].

2.2 Stability of writing process features

A central methodological question in writing process research concerns whether keystroke-derived features represent stable characteristics of writers or fluctuate substantially across tasks. Early work emphasized that many process measures reflect multiple interacting cognitive and situational factors and cautioned against assuming direct mappings to stable traits [16, 28]. However subsequent empirical studies have demonstrated that stability varies widely across features.

In assessment contexts for example, Deane and Zhang showed that only a limited subset of keystroke features exhibits moderate test– retest reliability across essay prompts, while many others display low consistency [37]. Crucially, these stable features were also the ones that generalized across tasks in predictive models, whereas highly variable features failed to support robust inference about writing ability [10]. This finding underscores that stability is not merely a psychometric concern but a prerequisite for longitudinal interpretation and modeling.

Longitudinal evidence from text-based writing research reinforces the importance of this distinction between stability and change. Using mixed-effects models over a large learner corpus, Lei, Wen, and Yang showed that development in syntactic complexity unfolds gradually and is characterized by substantial within-writer variability over time [21]. Their findings emphasize that longitudinal inference depends on features that are sufficiently stable to support growth modeling, while still allowing for uneven, context-dependent trajectories of change.

Related results have been reported in adult and second-language writing. Choi and Deane found that despite substantial within-writer variability, certain low-level timing features remained relatively stable across tasks and time points and were meaningfully related to writing quality [6]. Across studies, a consistent pattern emerges: low-level fluency measures (e.g., typing speed, within-word pause duration) tend to behave as writer-specific tendencies, whereas higher-level features related to planning, revision, or sentence-level pausing are more sensitive to task characteristics [1, 37].

2.3 Contextual sensitivity and task effects

Alongside work on stability, a substantial literature demonstrates that writing processes are highly sensitive to task and context. Writing behavior varies with prompt complexity, genre, topic familiarity, time constraints, and instructional conditions [4, 17]. Experimental studies manipulating task difficulty show that output-related features and revision frequency increase substantially for more demanding tasks, while core timing features such as average inter-key interval remain comparatively stable [1].

At the same time, longitudinal writing development is inherently context-sensitive. A study demonstrated that trajectories of syntactic complexity growth vary systematically with instructional and contextual factors, even when mod-

eled at scale [21]. This finding highlights a key challenge for longitudinal writing research: apparent change may reflect shifts in task demands or learning contexts rather than underlying development, underscoring the need to interpret longitudinal patterns in relation to both stability and context.

Genre effects further highlight contextual sensitivity. Argumentative writing typically elicits longer pauses and more revisions than narrative writing, reflecting increased planning and evaluation demands [4, 11]. In assessment settings, researchers have cautioned that process measures observed for one prompt may not generalize to another, limiting their interpretability across occasions [10, 36]. For longitudinal analyses, this presents a key challenge: observed changes in process features may reflect contextual variation rather than genuine development.

2.4 Measurement & modeling

Interest in writing process data has grown across research on learner modeling and instructional support. Keystroke-derived features have been incorporated into predictive models of writing quality and shown to add explanatory power beyond static text features [6, 7]. Process features have also been explored as signals for real-time feedback and formative analytics in writing support systems [22, 29].

Broader efforts in human-computer interaction have also examined the design of intelligent writing support systems. A study proposed a comprehensive design space for writing assistants organized around user, task, technology, interaction, and ecosystem dimensions, highlighting writing efficiency and proficiency as key user-facing considerations [20]. The measurement properties of keystroke-derived features examined in the present study are directly relevant to such systems, as reliable process signals are a prerequisite for effective real-time writing support.

However, a recurring concern in this line of work is the validity of process features across contexts. If the meaning of a feature varies substantially across genres or task conditions, its use in general-purpose models becomes problematic [1, 36]. Modeling applications therefore require a principled understanding of which features are robust across contexts and which require contextualized interpretation.

Prior research establishes that writing process features capture skill-related patterns, vary widely in stability, and are strongly influenced by task context. What remains underexplored is how these features behave across broader instructional contexts, such as grade level and school, within naturalistic longitudinal classroom data. This study addresses that gap by systematically examining the stability and contextual sensitivity of keystroke-derived writing process features in everyday classroom writing, providing a measurement foundation for longitudinal modeling and process-aware analysis.

3. DATASET

3.1 Data Collection

This study uses longitudinal keystroke data collected via Writing Observer, a research platform integrated into Google Docs and deployed in two U.S. public middle schools. Writing Observer is a suite of tools designed to support classroom orchestration in writing-to-learn tasks. The platform unobtrusively captures real-time writing data at the keystroke level, recording students' editing behaviors, reading activity, and document interactions from start to finish of each writing session. It is also designed to integrate into existing learning management systems to track class enrollments, assignments, and evaluations, enabling monitoring of writing practices within and across assignments. Since students contributed multiple writing sessions over an academic year, the resulting dataset supports analysis of within-student consistency and contextual variation in writing-process features across authentic classroom tasks.

All data collection was conducted in accordance with institutional research ethics requirements. The study operated under IRB approval, with students participating under appropriate consent and assent procedures. All records were pseudonymized prior to analysis, and no personally identifiable information is reported.

3.2 Participants

The study participants included 423 middle school students in Grades 7 and 8 from two U.S. public schools who participated during the 2023–2024 academic year. One school was a large urban middle school in Utah, and the other was a suburban middle school in Pennsylvania. Together, these schools provided a diverse instructional context while remaining comparable in grade span and curricular structure.

The writing activities were part of regular classroom instruction, primarily in English Language Arts (ELA) courses, with additional writing in subjects such as science and social studies. Assignments included narrative, argumentative, and informative writing, as well as text-dependent analysis, and non-ELA writing such as lab reports and class notes. Session durations varied and were computed from the first to the last logged keystroke event within each document. Writing prompts were not standardized and varied naturally by grade, teacher, genre, and topic, reflecting authentic classroom practice rather than controlled assessment conditions.

We collected a total of 9,831 writing-session logs. After excluding documents with fewer than 50 words, the final corpus consisted of 8,576 sessions (3,641 from School 1 and 4,935 from School 2). All data were collected under appropriate institutional approvals, and logs were de-identified prior to analysis. Table 1 summarizes sample sizes by school, grade, and dataset. Figures 1 and 2 show the distribution for both schools.

Also, we collected the incoming proficiency scores of all the students from the 2022-2023 academic year which were binned into four categories (1 = Below Basic, 2 = Basic, 3 = Proficient, 4 = Advanced). Given the absence of Below-Basic students in School 2 and their low numbers in School 1, Below Basic group was excluded to maintain comparability across schools.

Gender distribution across cohorts was as follows: School 1 Grade 7 had 70 female and 40 male students; School 1 Grade

8 had 47 female and 51 male students; School 2 Grade 7 had 31 female and 30 male students; and School 2 Grade 8 had 75 female and 79 male students.

Table 1: Sample sizes by school, grade, and number of documents

School	Grade 7		Grade 8	
	Students	Docs	Students	Docs
1	110	1840	98	1801
2	61	989	154	3946

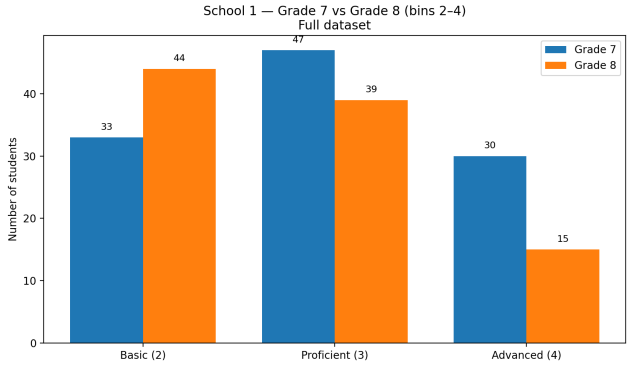


Figure 1: School 1 (Grade 7 and 8) Full State Assessment Scores

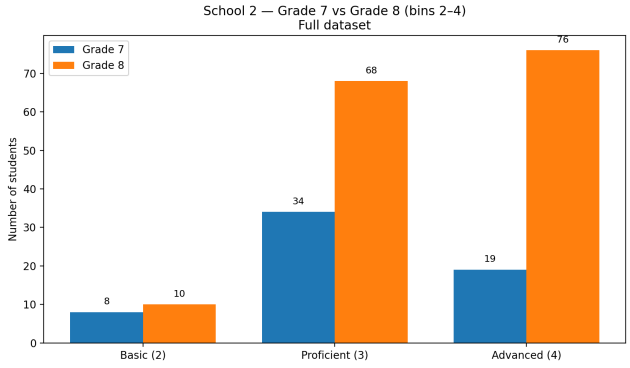


Figure 2: School 2 (Grade 7 and 8) Full State Assessment Scores

3.3 Feature Extraction

Each keystroke log consists of a timestamped sequence of writing events, including the character insertions, deletions, pauses, and navigation actions. Pauses were inferred from inter-keystroke intervals (IKIs), and revision behavior was identified through insertion–deletion patterns in the event stream [2, 22].

From these logs, we derived a set of keystroke-based process features describing how students generated, revised, and interacted with text. Following prior writing-process research (Section 2), features were organized into four behavioral dimensions: fluency, revision, complexity, and engagement [1, 8, 10, 36]. In total, ten frequency-based features (capturing how often actions occurred) and fourteen time-based features (capturing pause durations and temporal dynamics) were computed. We summarize the feature families:

1. **Fluency:** Features capturing transcription efficiency and writing flow, including the number of inserts, in-word inserts, between-word inserts, before-word inserts, word-start inserts, and pause measures such as IKI between inserts, within-word IKI, between-word IKI, before-word IKI, and word-start IKI [1, 22].
2. **Revision:** Features describing editing and text modification behavior, including before-word inserts, total edits (insertions + deletions), starting and continuing backspaces, edit duration, and pause measures associated with deletion activity (IKI starting backspace, IKI continuing backspace) [2, 10].
3. **Complexity:** Features reflecting structural and syntactic elaboration during writing, including clause-break inserts, minor punctuation inserts (e.g., commas, parentheses), and associated pause measures at clause boundaries and minor punctuation (IKI at clause break, IKI at minor punctuation) [1, 4].
4. **Engagement:** Features that capture non-typing interactions with the document, including overall menu time, reading time (inactive but focused time), navigation time (cursor, scrolling, arrow keys), and total active writing time [35].

Time-based features were log-transformed using $\log(1+x)$ to mitigate heavy-tailed pause distributions, and pauses longer than 120 seconds were truncated to exclude off-task intervals [1, 22]. For each applicable time-based feature, the mean, median, standard deviation, and variance were computed, yielding a total of 44 features used in analysis [2, 10]. Although a broader feature set was available, this study focuses on core indicators of fluency, revision, complexity, and engagement that are most relevant to examining longitudinal stability and contextual sensitivity [1, 8].

4. METHODS

Our analysis addresses two research questions using complementary statistical approaches. RQ1 examines the stability of keystroke-derived writing process features within cohorts using variance decomposition via intraclass correlation, while RQ2 examines sensitivity to contextual factors using linear mixed-effects modeling [12, 27].

4.1 Data Preparation and Cohort Definition

Writing process features were extracted from document-level writing logs, with multiple writing samples available per student. Because proficiency-level information was not available at the document level, student proficiency-level labels were obtained from a separate roster summary and merged onto document-level records using anonymized student identifiers.

Writing samples without an associated proficiency-level label were excluded from further analysis. Analyses were conducted separately for each school, and within each setting,

data were further stratified by grade level. All results reported for RQ1, therefore, reflect within-cohort analyses, defined by a specific combination of school and grade. To ensure reliable estimation of student-level effects, analyses were restricted to students who contributed at least two writing sessions within a given cohort, and only features present and numerically valid within that cohort were included.

4.2 RQ1: Intra-Class Correlation Analysis for Feature Stability

RQ1 investigates the extent to which keystroke-derived writing process features exhibit stable, student-specific structure within cohorts, as opposed to variability across writing samples produced by the same student. Stability is quantified using the intraclass correlation coefficient (ICC), which decomposes total variance into between-student and within-student components [12, 24]. In the context of writing-process analysis, stability indicates whether a feature reflects a persistent student-level tendency rather than transient task-specific behavior.

For each writing process feature and for each school × grade cohort, we estimated the intraclass correlation coefficient (ICC) using a random-intercept formulation:

$$y_{ij} = \mu + u_i + \varepsilon_{ij}, \tag{1}$$

where y_{ij} denotes the value of a given writing-process feature for student i on writing task j, μ is the overall mean, $u_i \sim \mathcal{N}(0, \sigma^2_{\text{student}})$ represents a student-specific random intercept capturing stable between-student differences, and $\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2_{\text{within}})$ captures within-student variability across tasks, , where $\sigma^2_{\text{student}}$ and σ^2_{within} denote the between-student and within-student variance components, respectively.

We computed the intraclass correlation coefficient (ICC) as:

$$\text{ICC} = \frac{\sigma^2_{\text{student}}}{\sigma^2_{\text{student}} + \sigma^2_{\text{within}}}, \tag{2}$$

which represents the proportion of total variance attributable to stable between-student differences. Under this random-intercept formulation, total variance is fully decomposed into between-student and within-student components, such that their sum represents the total variability in the feature. An ICC close to 1 indicates that a feature is highly stable within students across writing tasks, whereas an ICC near 0 indicates that most variability arises from task-to-task fluctuations within students.

For each feature, point estimates of the ICC were obtained using a random-intercept mixed-effects model estimated via restricted maximum likelihood (REML) [31]. When model estimation was unstable, point estimates were computed using an ICC estimator based on a one-way analysis of variance [30]. To quantify uncertainty, we computed ICCs and corresponding 95% confidence intervals for each feature using all available writing sessions. Following established conventions, ICC values were categorized as indicating poor (< 0.20), fair (0.20–0.40), moderate (0.40–0.60), or high (≥ 0.60) stability [19].

This analysis enables ranking writing-process features by their degree of longitudinal consistency and identifying which features function as persistent student-level signatures versus those dominated by task-specific variation. Establishing this distinction is a prerequisite for meaningful longitudinal modeling of writing development, as features that lack sufficient stability cannot be reliably interpreted as indicators of student growth.

4.3 RQ2: Modeling Contextual Sensitivity

RQ2 examines how writing-process features vary systematically with school context, grade level, and academic proficiency, while accounting for repeated observations from the same student [31, 3]. Specifically, RQ2 asks whether a portion of observed feature variance can be explained by known contextual factors rather than treated as undifferentiated task noise. Each feature was modeled using a linear mixed-effects framework with fixed effects for contextual variables and a random student-level intercept. Although writing-process features may exhibit complex or nonlinear patterns over time, the present analysis uses a linear mixed-effects framework because contextual predictors are modeled as categorical fixed effects, allowing for flexible, non-monotonic differences across groups without imposing a specific functional form.

For each writing-process feature, we fit the following model:

$$y_{ij} = \beta_0 + \beta_1 \text{School}_i + \beta_2 \text{Grade}_i + \beta_3 \text{StateScore}_i + b_i + \varepsilon_{ij},$$

where y_{ij} is the feature value for student i on writing task j, β_0 is the intercept, and β_1, β_2, and β_3 represent fixed effects of school, grade level, and incoming academic proficiency, respectively. The term $b_i \sim \mathcal{N}(0, \sigma^2_{\text{student}})$ captures student-specific deviations from the population mean, and $\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2_{\text{residual}})$ represents residual variability [31, 3].

We treated school, grade level, and incoming academic proficiency as categorical fixed effects to allow for non-linear and non-monotonic differences across groups [31]. Academic proficiency was operationalized using a categorical state assessment score with four ordered levels.

To identify school effects, we fit models across schools and grade levels, ensuring that the school predictor varied within each subset. We estimated all models using restricted maximum likelihood and included only students who had at least two writing sessions [31]. Subsets with insufficient numbers of students or insufficient variation in predictors were excluded.

For each fitted model, we report fixed-effect estimates, standard errors, z-statistics, and approximate two-sided p-values based on normal approximations [29]. We also report variance components for the student-level random intercept and the residual term to contextualize the relative contribution of between- and within-student variability.

5. RESULTS

5.1 RQ1: Feature Stability Within Cohorts

This section reports results for RQ1, which examines the stability of keystroke-derived writing process features within School × Grade cohorts. We quantify stability using intraclass correlation coefficients (ICCs), where higher values

Table 2: Feature-Level ICCs for School 1 – Grade 7

Feature	ICC	Stability
meanLogWordStartInsertTime	0.448	Moderate
meanLogBeforeWordInsertTime	0.288	Fair
stdLogBetweenWordInsertTime	0.184	Poor
varLogContinuingBackspaceTime	0.104	Poor
varLogEditTime	0.094	Poor
varLogBeforeWordInsertTime	0.082	Poor
stdLogWordStartInsertTime	0.074	Poor
overallNavigationTime	0.056	Poor

indicate greater consistency of a feature across multiple writing samples produced by the same student, relative to variability between students within the same cohort.

Across all cohorts, a substantial proportion of features yielded ICC values of 0 (or effectively 0), indicating no detectable between-student variance after accounting for within-student variability. Tables 2–5 report only features with non-zero ICC estimates; all omitted features had ICC = 0 and are therefore not interpretable as stable individual-level measures in these contexts. Overall, feature-level stability varied markedly across schools, grade levels, and feature types. The majority of features exhibited Poor stability, with only a small subset, primarily low-level insertion timing features, demonstrating Moderate or High stability, most consistently in School 2.

5.1.1 School 1 – Grade 7

In School 1, Grade 7, the writing process features demonstrated overwhelmingly low stability (Table 2). Of the 41 features examined, most produced ICC values of 0, indicating that variability was almost entirely within students across writing tasks rather than between students. Among the features with non-zero ICCs, only meanLogWordStartInsertTime reached Moderate stability (ICC = 0.448). meanLogBeforeWordInsertTime showed Fair stability (ICC = 0.288), while all remaining reported features, including measures of timing variability, backspacing behavior, and navigation time, exhibited Poor stability (ICC < 0.20).

These results suggest that, in this cohort, most writing-process features do not function as stable student-level characteristics and instead fluctuate substantially across writing tasks.

5.1.2 School 1 – Grade 8

In School 1, Grade 8, stability patterns remained limited, though a small number of features demonstrated modest increases relative to Grade 7 (Table 3). As in Grade 7, the majority of the features analyzed yielded ICCs of 0 and are not shown. Two within-word insertion timing features, meanLogInWordInsertTime (ICC = 0.494) and medianLogInWordInsertTime (ICC = 0.484), demonstrated Moderate stability, while medianLogStartingBackspaceTime reached Fair stability (ICC = 0.386). All other reported features exhibited Poor stability, and no features achieved High stability.

Overall, while some fine-grained timing features became more consistent in Grade 8, feature-level stability in School 1 remained sparse and highly selective.

Table 3: Feature-Level ICCs for School 1 – Grade 8

Feature	ICC	Stability
meanLogInWordInsertTime	0.494	Moderate
medianLogInWordInsertTime	0.484	Moderate
medianLogStartingBackspaceTime	0.386	Fair
meanLogClauseBreakInsertTime	0.175	Poor
varLogContinuingBackspaceTime	0.144	Poor

Table 4: Feature-Level ICCs for School 2 – Grade 7

Feature	ICC	Stability
medianLogInWordInsertTime	0.681	High
meanLogInWordInsertTime	0.679	High
meanLogBetweenWordInsertTime	0.620	High
medianLogBetweenWordInsertTime	0.594	Moderate
medianLogWordStartInsertTime	0.589	Moderate
meanLogWordStartInsertTime	0.567	Moderate
medianLogStartingBackspaceTime	0.504	Moderate
meanLogStartingBackspaceTime	0.462	Moderate
meanLogBeforeWordInsertTime	0.266	Fair
stdLogMinorPunctInsertTime	0.145	Poor
varLogContinuingBackspaceTime	0.142	Poor
varLogBeforeWordInsertTime	0.042	Poor
stdLogClauseBreakInsertTime	0.032	Poor

5.1.3 School 2 – Grade 7

In contrast, School 2 Grade 7 exhibited markedly higher stability across several features (Table 4). Although many of the 41 features still yielded ICC values of 0, a larger subset demonstrated meaningful between-student consistency. Three features, medianLogInWordInsertTime (ICC = 0.681), meanLogInWordInsertTime (ICC = 0.679), and meanLogBetweenWordInsertTime (ICC = 0.620), reached High stability, indicating strong within-student consistency across writing sessions. Several additional features related to word-start insertion and starting backspace timing demonstrated Moderate stability (ICC ≈ 0.50–0.59). Despite this, higher-order variability measures (e.g., variance-based timing features and punctuation timing variability) continued to show Poor stability or zero ICCs.

These results suggest that, in this context, micro-temporal production behaviors may reflect relatively stable individual writing signatures, while broader variability-based features do not.

5.1.4 School 2 – Grade 8

In School 2, Grade 8, overall stability declined relative to Grade 7, though it remained higher than in School 1 (Table 5). As in other cohorts, many of the 41 features analyzed had ICCs of 0 and are omitted. Two within-word insertion timing features, meanLogInWordInsertTime (ICC = 0.589) and medianLogInWordInsertTime (ICC = 0.568), retained Moderate stability, while meanLogClauseBreakInsertTime demonstrated Fair stability (ICC = 0.315). All remaining reported features exhibited Poor stability, and no features reached High stability.

These findings indicate that even within the same school context, feature stability can attenuate with grade level and

Table 5: Feature-Level ICCs for School 2 – Grade 8

Feature	ICC	Stability
meanLogInWordInsertTime	0.589	Moderate
medianLogInWordInsertTime	0.568	Moderate
meanLogClauseBreakInsertTime	0.315	Fair
stdLogWordStartInsertTime	0.124	Poor
stdLogContinuingBackspaceTime	0.106	Poor
stdLogClauseBreakInsertTime	0.059	Poor
stdLogStartingBackspaceTime	0.054	Poor
overallActiveWritingTime	0.015	Poor

that stable features become increasingly restricted to specific low-level timing measures.

5.1.5 Summary of RQ1 Findings

Across all School × Grade cohorts, there are three key patterns that emerged:

- Zero ICCs are the norm: In every cohort, a substantial number of the 41 analyzed features yielded ICC values of 0, indicating that most writing-process features do not exhibit stable between-student differences.
- Stability is highly feature-specific: When stability emerged, it was largely confined to within-word and word-initial insertion timing features, rather than variability based or aggregate measures.
- Stability is context-dependent: School 2, particularly Grade 7, exhibited systematically higher stability than School 1, suggesting that instructional or institutional context may shape the consistency of writing processes.

Taken together, these results indicate that most keystroke-derived writing process features should not be treated as stable individual traits. Instead, stability appears rare, selective, and context-dependent, motivating the mixed-effects analyses in RQ2, which explicitly examine how contextual factors explain systematic variation in writing process behavior.

5.2 RQ2: Sensitivity to Contextual Factors

RQ2 examined whether variability in keystroke-derived writing process features can be systematically explained by contextual factors, specifically, school, grade level, and incoming academic proficiency, after accounting for repeated observations from the same student. To address this question, we fit a series of linear mixed-effects models, one per feature, including fixed effects for school, grade, and proficiency level, and a random intercept for student. All models were fit on the Full dataset, pooling Grades 7 and 8 across both schools.

5.2.1 Overall Patterns of Contextual Sensitivity

Across features, contextual sensitivity varied substantially by feature type. In general, incoming academic proficiency exhibited more consistent associations with writing process behavior than grade level and school membership. This pattern suggests that observed variation in writing processes reflects proficiency-related differences more than developmental or school-specific artifacts. Random intercept variances for students were consistently non-zero across models, confirming substantial between-student variability and justifying the use of mixed-effects modeling to account for repeated writing samples per student. Throughout the following subsections, fixed-effect coefficients are reported alongside back-transformed percentage differences (computed as $e^{\hat{\beta}} - 1$), which reflect multiplicative change in the original pause durations given that features were log-transformed prior to modeling.

5.2.2 Grade-Level Effects

Grade-level effects were observed for several timing-based features. In particular, Grade 8 students often exhibited shorter within-word insertion times than Grade 7 students. These effects were typically negative in direction, indicating increased efficiency or fluency with advancing grade level. However, grade effects were not universal across all features and were generally modest in magnitude, suggesting that developmental differences manifest more strongly in fine-grained timing behaviors than in global duration measures. Table 6 summarizes features with significant grade-level effects.

Table 6: Significant Grade-Level Effects (Grades 8 vs. 7)

Feature	β	Interpretation
Median In-word Pause	−0.03	3% shorter
Mean In-word Pause	−0.03	3% shorter
Std. In-word Pause	−0.02	2% less varying
Median Clause-break Pause	−0.05	5% shorter
Mean Clause-break Pause	−0.05	5% shorter
Var. Clause-break Pause	−0.21	19% less varying
Median Edit Pause	+0.02	2% longer

5.2.3 Proficiency Effects

Academic proficiency demonstrated clearer and more interpretable associations with writing process features than grade level. For multiple fine-grained timing features, the highest proficiency level (Level 4) was associated with systematically faster within-word insertion behaviors relative to the Basic proficiency group, while Level 3 effects were generally small and non-significant. These patterns indicate that keystroke-level timing measures capture meaningful differences in writing fluency linked to incoming academic proficiency. Importantly, proficiency effects were not uniformly incremental across adjacent levels. In several cases, only the highest proficiency group differed significantly from the reference group, indicating potential threshold effects rather than linear progression across proficiency bands. Table 7 summarizes features with significant proficiency effects.

Table 7: Significant Proficiency Effects (Level 4 vs. Level 2)

Feature	β	Interpretation
Median In-word Pause	−0.04	4% shorter
Mean In-word Pause	−0.05	5% shorter
Std. In-word Pause	−0.02	2% less varying
Median Clause-break Pause	−0.08	8% shorter
Mean Clause-break Pause	−0.08	8% shorter
Var. Clause-break Pause	+0.19	21% more varying
Mean Minor Punct. Pause	−0.080	7.7% shorter

5.2.4 School Effects

School membership explained relatively little variance in writing process features after controlling for grade and proficiency. Most school-related coefficients were small and non-significant, indicating that differences in writing processes were largely consistent across institutional contexts. This finding supports the generalizability of keystroke-based process measures across schools and suggests that observed behavioral differences are not driven by local instructional or institutional idiosyncrasies. Table 8 summarizes the two features with significant school effects.

Table 8: Significant School Effects (School 2 vs. School 1)

Feature	β	Interpretation
Std. In-word Pause	-0.02	2% less varying
Mean Minor Punct. Pause	-0.09	8.6% shorter

5.2.5 Summary of RQ2 Findings

Taken together, the RQ2 results demonstrate that while many writing process features exhibit substantial within-student variability, a subset shows systematic sensitivity to meaningful contextual factors, particularly incoming academic proficiency and, to a lesser extent, grade level. Fine-grained timing features appear especially informative for distinguishing writing behaviors associated with higher proficiency, whereas global duration measures reflect engagement-related differences. These findings complement the stability analyses in RQ1 by showing that writing process features may be individually variable yet still contextually interpretable when modeled appropriately.

6. DISCUSSION

This study examined keystroke-derived writing-process features from two complementary perspectives: their stability across students and tasks (RQ1) and their systematic sensitivity to contextual factors such as grade level, school, and incoming academic proficiency (RQ2). Taken together, the findings reveal a nuanced picture in which most writing process features are highly variable at the individual level, yet a subset remains meaningfully interpretable when contextualized appropriately.

6.1 Limited Stability of Writing Process Features

The RQ1 analyses demonstrate that zero or near-zero ICCs were the norm across all School × Grade cohorts. For a large proportion of the 41 features examined, between-student variance was effectively absent once within-student variability across writing tasks was accounted for. This finding challenges the common assumption, implicit in many learning analytics applications, that keystroke-derived features can be treated as stable, student-specific traits.

When stability did emerge, it was highly feature-specific, largely confined to low-level temporal production measures, such as within-word and word-initial insertion times. These features plausibly reflect motor–cognitive coordination processes that are less sensitive to task content and genre, making them more likely to persist across writing contexts. In contrast, higher-order variability measures, revision timing dispersion, and aggregate duration features consistently exhibited poor or zero stability, suggesting that these measures are strongly task-dependent rather than student-specific.

Importantly, stability was also context-dependent. School 2, particularly Grade 7, exhibited substantially higher stability than School 1, indicating that instructional context, curriculum alignment, or writing task design may shape the consistency with which students express their writing processes. This variability cautions against generalizing stability findings from a single institutional setting and underscores the importance of reporting stability estimates alongside proposed writing-process indicators. This finding does not contradict the RQ2 result showing no significant school effect. The two analyses ask different questions about the data: RQ2 examines whether students in different schools write differently on average, while RQ1 examines whether individual students write consistently across tasks. A school context can produce similar average writing behaviors across institutions while still shaping how consistently individual students express those behaviors from task to task.

6.2 Reconciling Instability With Contextual Sensitivity

At first glance, the widespread instability observed in RQ1 may appear to undermine the usefulness of keystroke-derived writing process features. However, the RQ2 results clarify that lack of individual-level stability does not imply lack of interpretability. While many features fluctuate substantially across students within tasks, a subset shows systematic sensitivity to meaningful contextual factors, particularly incoming academic proficiency and grade level.

Fine-grained timing features, especially within-word and word-initial insertion measures, showed consistent associations with both grade and proficiency. Higher-grade and higher proficiency students tended to exhibit faster insertion behaviors, aligning with theoretical accounts of increased writing fluency and automatization with development and skill acquisition. These findings suggest that although such features may not function as stable individual signatures, they nonetheless capture reliable group-level differences tied to developmental and proficiency-related factors.

By contrast, macro-level duration features, such as overall active writing time, followed a different pattern. These features showed limited grade sensitivity but were associated with proficiency in ways that suggest differences in engagement or persistence rather than efficiency. Notably, higher-proficiency students spent more time actively writing, indicating that longer writing durations should not be interpreted simplistically as inefficiency, but rather as potentially reflecting sustained effort or deeper engagement.

This finding resonates with and extends prior work on revision behavior in writing support systems. A study demonstrated that revision patterns captured from writing tool logs are meaningful and vary systematically across learners, including differences tied to gender and feedback condition [25]. Our results complement this by showing that revision features are among the least stable across tasks at the individual level, suggesting that longitudinal pipelines relying on revision indicators should account for substantial within-

student variability rather than treating revision behavior as a fixed learner characteristic.

6.3 Implications for Writing Analytics and Educational Dashboards

Together, these findings have important implications for the design and interpretation of writing analytics systems. First, they suggest that keystroke features should not be universally treated as stable learner traits, especially when used for longitudinal profiling or individual diagnosis. Most features are better understood as context-sensitive behavioral indicators that respond to task demands, instructional conditions, and developmental factors.

Second, the results highlight the value of mixed-effects modeling frameworks for research on the writing process. By explicitly modeling both within-student variability and contextual predictors, such approaches allow researchers and practitioners to extract meaningful patterns from features that would otherwise appear noisy or unreliable at the individual level.

Finally, the differential behavior of fine-grained timing features versus aggregate duration measures suggests that feature families serve distinct interpretive roles. Low-level timing features may be well-suited for examining fluency and automatization, while duration-based features may provide insight into engagement, persistence, or strategic effort. Analytics systems and dashboards should therefore avoid aggregating heterogeneous features into undifferentiated indices and instead support feature-specific interpretation.

7. CONCLUSIONS

This paper examined the measurement properties of keystroke-derived writing process features in authentic classroom settings, focusing on their within-cohort stability and sensitivity to contextual factors. Using a large corpus of middle school writing sessions, we showed that most writing process features exhibit low or zero stability within School × Grade cohorts, reflecting substantial within-student variability across writing tasks. Only a small subset of fine-grained timing features related to word-level text production demonstrated moderate stability, and high stability was rare and strongly context-dependent.

In contrast, many writing process features varied considerably across writing sessions. Mixed-effects analyses revealed that, for a limited set of features, this variability aligns systematically with incoming academic proficiency and grade level, rather than reflecting undifferentiated noise. In particular, within-word insertion timing features showed consistent associations with proficiency, with higher-proficiency students exhibiting faster text production. Grade-level effects were present but modest, suggesting that writing process development during middle school unfolds gradually and that much observed variability reflects situational adaptation rather than rapid developmental change. After controlling for grade and proficiency, school-level effects were small and largely non-significant, supporting the generalizability of keystroke-based process measures across institutional contexts.

Taken together, these findings clarify an important measurement distinction: keystroke-derived writing-process features do not uniformly serve as stable student-level indicators. Some features provide relatively consistent signals within cohorts, while others primarily capture context-responsive behavior. This distinction is critical for longitudinal research, as features dominated by within-student variability cannot be interpreted as reliable indicators of enduring writing characteristics when analyzed in isolation.

The primary contribution of this study is methodological. Rather than modeling writing growth directly, we establish which classes of process features are sufficiently stable to support longitudinal interpretation and which require explicit contextual modeling. This groundwork enables more principled feature selection for future growth modeling, evaluation, and intervention studies in educational data mining.

From a practical perspective, the results suggest complementary roles for different feature types in educational tools. Features with higher within-cohort stability may support longitudinal monitoring of foundational writing processes, whereas context-sensitive features may serve as indicators of task-specific difficulty, instructional demands, or adaptive strategy use. Interpreting these signals jointly allows educators and researchers to distinguish enduring writing patterns from situational challenges.

While future work is needed to link process features to writing quality, extend analyses across longer time scales, and replicate findings in broader populations, this study provides a focused and principled analysis of writing-process measurement in real classroom contexts. Overall, the results demonstrate that keystroke-derived writing data contain both reliable structure and meaningful variability. Advancing longitudinal writing analytics will depend on leveraging the former while modeling the latter with appropriate attention to context.

7.1 Limitations and Future Work

While this study provides a systematic characterization of the stability and contextual sensitivity of keystroke-derived writing process features, several limitations should be acknowledged, along with directions for future research.

7.1.1 Absence of Writing Outcome Measures

A primary limitation of this study is the absence of direct measures of writing quality or learning outcomes. Although we identify features that exhibit greater stability within School × Grade cohorts (RQ1) and others that vary systematically with contextual factors such as grade level and incoming academic proficiency (RQ2), we do not evaluate whether these patterns correspond to improved writing performance or desirable writing behaviors. Stability alone does not imply educational value, and context sensitivity does not necessarily indicate instructional relevance.

Without outcome measures such as essay scores, rubric-based assessments, or teacher feedback, it is not possible to determine whether stable features reflect productive writing habits, emergent fluency, or merely idiosyncratic behaviors. Future work will integrate outcome measures to examine how stable and context-sensitive process features relate to

writing quality and development. In particular, longitudinal growth models linking changes in writing process features to changes in writing performance will be critical for validating the educational significance of the process indicators identified in this study.

7.1.2 Limited Temporal Scope

The data analyzed in this study span approximately one academic year, which constrains conclusions about long-term stability and developmental change. Features that appear stable over this time frame may still evolve over longer periods, particularly across major instructional transitions or shifts in curricular emphasis. As a result, stability within a single school year should not be interpreted as permanence or trait-like invariance.

Longer-term longitudinal studies, especially those tracking students across multiple grade levels or school transitions, would help determine how writing process features change over extended time scales. Such studies could reveal developmental trajectories, periods of consolidation, or phases of heightened sensitivity to instruction that are not observable within a single academic year.

7.1.3 Unmodeled Task-Level Variation

Writing data were collected in authentic classroom settings, resulting in substantial variation in task topics, genres, instructional goals, and scaffolding. While this ecological validity strengthens the relevance of the findings, it introduces task-level variability that cannot be fully disentangled from student-level variability. In the absence of repeated or standardized prompts, it remains difficult to separate task-specific effects from within-student fluctuation, particularly for features that exhibited low or zero stability in RQ1.

Additionally, temporal distribution data for writing sessions across the academic year was not recorded, limiting our ability to assess whether uneven clustering of sessions could have introduced bias in ICC estimation. Task-level metadata, including genre classification and prompt type, was also not fully available at the time of analysis, constraining interpretation of school-level effects and task-driven variability. Future work will incorporate richer task annotation as labeling efforts are completed.

A related limitation concerns the inclusion of non-ELA writing tasks, such as lab reports and class notes, alongside ELA assignments. While this reflects the authentic range of classroom writing students engage in, it introduces additional heterogeneity that may affect certain feature families disproportionately. For instance, complexity features designed to capture syntactic elaboration may be less interpretable for highly formulaic or note-taking tasks.

Future research could adopt hybrid study designs that combine naturalistic classroom writing with repeated benchmark tasks, and examine whether stability patterns hold when analyses are restricted to ELA writing. Such designs would allow more direct estimation of test–retest reliability while preserving ecological validity. Additionally, richer annotation of task characteristics, such as genre, prompt complexity, revision expectations, and instructional supports, would enable more precise modeling of context-sensitive writing behaviors across subject areas.

7.1.4 Future Research Directions

Building on the findings of this study, several directions for future research are planned. First, we will integrate writing quality outcomes to model how stable and context-sensitive process features relate to writing development over time. Second, stability estimates from RQ1 can be leveraged to design process-aware educational tools, such as dashboards that emphasize meaningful deviations from a student's typical writing behavior rather than raw feature values.

Finally, future work will investigate the sources of stability and variability through richer contextual modeling and qualitative inquiry, and will extend the feature set to include linguistic, syntactic, and higher-level process indicators. Together, these efforts aim to advance a more nuanced understanding of writing processes as both persistent and adaptive, supporting longitudinal modeling and actionable educational data mining applications.

8. REFERENCES

[1] V. M. Baaijen and D. Galbraith. Discovery through writing: Relationships with writing processes and text quality. *Cognition and Instruction*, 36(3):199–223, 2018.

[2] V. M. Baaijen, D. Galbraith, and K. De Glopper. Keystroke analysis: Reflections on procedures and measures. *Written Communication*, 29(3):246–277, 2012.

[3] D. J. Barr, R. Levy, C. Scheepers, and H. J. Tily. Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of memory and language*, 68(3):255–278, 2013.

[4] C. Beauvais, T. Olive, and J.-M. Passerault. Why are some texts good and others not? relationship between text quality and management of the writing processes. *Journal of Educational Psychology*, 103(2):415–428, 2011.

[5] N. A. Chenoweth and J. R. Hayes. Fluency in writing: Generating text in l1 and l2. *Written communication*, 18(1):80–98, 2001.

[6] I. Choi and P. Deane. Evaluating writing process features in an adult efl writing assessment context: A keystroke logging study. *Language Assessment Quarterly*, 18(2):107–132, 2021.

[7] R. Conijn, C. Cook, M. van Zaanen, and L. Van Waes. Early prediction of writing quality using keystroke logging. *International Journal of Artificial Intelligence in Education*, 32(4):835–866, 2022.

[8] R. Conijn, J. Roeser, and M. Van Zaanen. Understanding the keystroke log: The effect of writing task on keystroke features. *Reading and Writing*, 32(9):2353–2374, 2019.

[9] P. Deane. Writing assessment and cognition. *ETS Research Report Series*, 2011(1):1–60, 2011.

[10] P. Deane and M. Zhang. Exploring the feasibility of using writing process features to assess text production skills. *ETS Research Report Series*, 2015(2):1–16, 2015.

[11] L. Faigley and S. Witte. Analyzing revision. *College Composition & Communication*, 32(4):400–414, 1981.

[12] J. L. Fleiss and J. Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619, 1973.

[13] L. Flower and J. R. Hayes. A cognitive process theory of writing. *College Composition & Communication*, 32(4):365–387, 1981.

[14] H. Guo, P. D. Deane, P. W. van Rijn, M. Zhang, and R. E. Bennett. Modeling basic writing processes from keystroke logs. *Journal of Educational Measurement*, 55(2):194–216, 2018.

[15] J. R. Hayes. Modeling and remodeling writing. *Written communication*, 29(3):369–388, 2012.

[16] J. R. Hayes and L. S. Flower. Identifying the organization of writing processes. In L. W. Gregg and E. R. Steinberg, editors, *Cognitive processes in writing*, pages 3–30. Routledge, New York, NY, 2016.

[17] R. T. Kellogg. Training writing skills: A cognitive developmental perspective. *Journal of writing research*, 1(1):1–26, 2008.

[18] R. T. Kellogg. A model of working memory in writing. In C. M. Levy and S. Ransdell, editors, *The science of writing*, pages 57–71. Routledge, New York, NY, 2013.

[19] T. K. Koo and M. Y. Li. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of chiropractic medicine*, 15(2):155–163, 2016.

[20] M. Lee, K. I. Gero, J. J. Y. Chung, S. B. Shum, V. Raheja, H. Shen, S. Venugopalan, T. Wambsganss, D. Zhou, E. A. Alghamdi, T. August, A. Bhat, M. Z. Choksi, S. Dutta, J. L. Guo, M. N. Hoque, Y. Kim, S. Knight, S. P. Neshaei, A. Shibani, D. Shrivastava, L. Shroff, A. Sergeyuk, J. Stark, S. Sterman, S. Wang, A. Bosselut, D. Buschek, J. C. Chang, S. Chen, M. Kreminski, J. Park, R. Pea, E. H. R. Rho, Z. Shen, and P. Siangliulue. A design space for intelligent and interactive writing assistants. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, pages 1054:1–1054:35, New York, NY, USA, 2024. Association for Computing Machinery.

[21] L. Lei, J. Wen, and X. Yang. A large-scale longitudinal study of syntactic complexity development in efl writing: A mixed-effects model approach. *Journal of Second Language Writing*, 59:100962, 2023. Article 100962.

[22] M. Leijten and L. Van Waes. Keystroke logging in writing research: Using inputlog to analyze and visualize writing processes. *Written Communication*, 30(3):358–392, 2013.

[23] C. M. Levy and S. Ransdell. *The science of writing: Theories, methods, individual differences and applications*. Routledge, 2013.

[24] K. O. McGraw and S. P. Wong. Forming inferences about some intraclass correlation coefficients. *Psychological methods*, 1(1):30–46, 1996.

[25] L. Mouchel, T. Wambsganss, P. Mejia-Domenzain, and T. Käser. Understanding revision behavior in adaptive writing support systems for education. 2023.

[26] T. Olive and R. T. Kellogg. Concurrent activation of high-and low-level production processes in written composition. *Memory & Cognition*, 30(4):594–600, 2002.

[27] J. C. Pinheiro and D. M. Bates. *Mixed-effects models in S and S-PLUS*. Springer, 2000.

[28] G. Rijlaarsdam and H. Van den Bergh. Writing process theory. In C. A. MacArthur, S. Graham, and J. Fitzgerald, editors, *Handbook of writing research*, pages 41–53. Guilford Press, New York, NY, 2006.

[29] A. Shibani, S. Knight, and S. Buckingham Shum. Contextualizable learning analytics design: A generic model and writing analytics evaluations. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 210–219, 2019.

[30] P. E. Shrout and J. L. Fleiss. Intraclass correlations: uses in assessing rater reliability. *Psychological bulletin*, 86(2):420–428, 1979.

[31] T. A. Snijders and R. Bosker. Multilevel analysis: An introduction to basic and advanced multilevel modeling. 2011.

[32] M. A. Tabari, X. Lu, and Y. Wang. The effects of task complexity on lexical complexity in l2 writing: An exploratory study. *System*, 114:103021, 2023.

[33] L. Van Waes and M. Leijten. Fluency in writing: A multidimensional perspective on writing fluency applied to l1 and l2. *Computers and Composition*, 38:79–95, 2015.

[34] L. Van Waes and P. J. Schellens. Writing profiles: The effect of the writing mode on pausing and revision patterns of experienced writers. *Journal of pragmatics*, 35(6):829–853, 2003.

[35] Å. Wengelin. Examining pauses in writing: Theory, methods and empirical data. In K. P. H. Sullivan and E. Lindgren, editors, *Computer key-stroke logging and writing*, pages 107–130. Elsevier, Oxford, 2006.

[36] M. Zhang and P. Deane. Process features in writing: Internal structure and incremental value over product features. *ETS Research Report Series*, 2015(2):1–12, 2015.

[37] M. Zhu, M. Zhang, and P. Deane. Analysis of keystroke sequences in writing logs. *ETS Research Report Series*, 2019(1):1–16, 2019.

Stable and Privacy-Preserving Synthetic Educational Data with Empirical Marginals: A Copula-Based Approach

Gabriel Diaz Ramos
Rice University
gd27@rice.edu

Lorenzo Luzi
Rice University, OpenStax
enzo@rice.edu

Debshila Basu Mallick
OpenStax
debshila@rice.edu

Richard G. Baraniuk
Rice University, OpenStax
richb@rice.edu

ABSTRACT

To advance Educational Data Mining (EDM) within strict privacy-protecting regulatory frameworks, researchers must develop methods that enable data-driven analysis while protecting sensitive student information. Synthetic data generation is one such approach, enabling the release of statistically generated samples instead of real student records; however, existing deep learning and parametric generators often distort marginal distributions and degrade under iterative regeneration, leading to distribution drift and progressive loss of distributional support that compromise reliability. In response, we introduce the Non-Parametric Gaussian Copula (NPGC), a plug-and-play synthesis method that replaces deep learning and parametric optimization with empirical statistical anchoring to preserve the observed marginal distributions while modeling dependencies through a copula framework. NPGC integrates Differential Privacy (DP) at both the marginal and correlation levels, supports heterogeneous variable types, and treats missing data as an explicit state to retain informative absence patterns. We evaluate NPGC against deep learning and parametric baselines on five benchmark datasets and demonstrate that it remains stable across multiple regeneration cycles and achieves competitive downstream performance at substantially lower computational cost. We further validate NPGC through deployment in a real-world online learning platform, demonstrating its practicality for privacy-preserving research.

Keywords

Synthetic data generation, differential privacy, educational data, marginal distribution preservation, model collapse

1. INTRODUCTION

Educational Data Mining (EDM) and Learning Analytics (LA) rely on student-level records such as interaction logs, assessment outcomes, and contextual variables, governed by strict privacy and data-protection regulations. Even after removing direct identifiers, education datasets may remain disclosive because fine-grained behavioral traces and contextual variables enable re-identification or harm [24]. This constraint is evident in the field's open-science record: A reproducibility audit of LAK proceedings (11th–12th conferences) found that 5% of papers shared raw datasets and 2% reported data available "on request" [18]. A parallel audit of EDM proceedings (14th–15th conferences) found that 15% of papers used or shared open datasets, 5% reported availability "on request," and most provided no accessible pathway to datasets [17]. Together, these findings indicate a structural constraint on cumulative progress in EDM and LA: The data that power state-of-the-art models are often the least shareable, limiting reproducibility, benchmarking, and cross-institution comparison.

The impact of this constraint is increasing with the rapid uptake of generative AI (genAI), particularly large language models (LLMs) [51], which increases demand for high-quality student datasets, including open-ended responses, code, and dialogue. A recent systematic review of empirical LLM-in-education studies (Nov 2022–Mar 2025) documents rapid

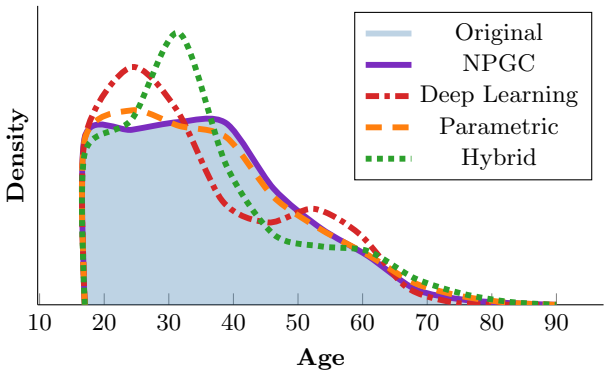


Figure 1: Marginal density of the *Age* variable in the Adult [2] dataset. The shaded region denotes the empirical density of the real data. Curves correspond to synthetic samples generated by a deep learning model, a Parametric Gaussian Copula (PGC), a Copula-GAN hybrid, and our proposed Non-Parametric Gaussian Copula (NPGC). Existing methods exhibit visible deviations from the empirical marginal, whereas NPGC aligns perfectly with the real distribution.

Gabriel Diaz Ramos, Lorenzo Luzi, Debshila Basu Mallick, and Richard Baraniuk. Stable and Privacy-Preserving Synthetic Educational Data with Empirical Marginals: A Copula-Based Approach. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 267–279. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040131

deployment while highlighting recurring privacy and governance challenges [40]. Many LLM-based learning supports (e.g., tutor-style feedback and hint generation) are validated on real student work and interaction traces that are difficult to release safely at scale due to re-identification risk [35]. The model-development pipeline introduces additional exposure: Fine-tuning and adaptation rely on sensitive datasets and admit privacy attacks (e.g., membership inference and data extraction), motivating stronger privacy-preserving practices [10]. This tension between data-hungry models and privacy-constrained records makes synthetic data generation compelling for EDM and LA, enabling method development and sharing while reducing reliance on protected student records and motivating the methodology proposed in this paper.

Synthetic data generation for tabular data (SDG-T) provides a practical approach for addressing this constraint. By combining statistical modeling with Differential Privacy (DP) [13] and established privacy evaluation protocols [15], SDG-T constructs artificial datasets that preserve statistical structure without exposing individual records. SDG-T enables privacy-compliant analysis and model development in educational settings. Unlike conventional deidentification, i.e., removing direct identifiers such as names and addresses, which offers no guarantees against reidentification through linkage with auxiliary data, DP is a formal framework that limits how much any single individual's record can influence released statistics, governed by a privacy parameter ϵ, where smaller values indicate stronger privacy. A useful property of DP is that any further processing of a DP output remains differentially private at the same ϵ.

Despite its promise, SDG-T faces a central technical challenge: preserving the empirical marginal distributions of heterogeneous variables. We refer to deviations between real and synthetic feature distributions as *marginal mismatch*. Educational attributes, such as test scores, attendance rates, or financial aid counts, rarely follow standard parametric families and often exhibit skewness, multi-modality, and discrete mass points.

Deep learning generators, including TVAE [36] and CTGAN [53], optimize joint objectives that prioritize inter-column dependencies, often causing systematic errors like mode collapse [25]. These models can distort individual marginals even when aggregate utility metrics appear competitive. Parametric approaches, such as the Gaussian Copula [44], assume predefined marginal distributions that may not reflect the real data shape. Hybrid methods, including Copula-GAN [34], apply Gaussianization before adversarial training, but still rely on fitted marginal models and do not guarantee exact empirical recovery.

As shown in Fig. 1, existing generators often distort individual feature marginals relative to the empirical data. Deep learning models lack explicit marginal constraints, whereas parametric approaches impose distributional forms that may not reflect the true distribution of the variables. In contrast, NPGC preserves empirical marginals through non-parametric anchoring. Marginal drift can bias downstream analysis; this example reflects behavior observed consistently across datasets in our experiments.

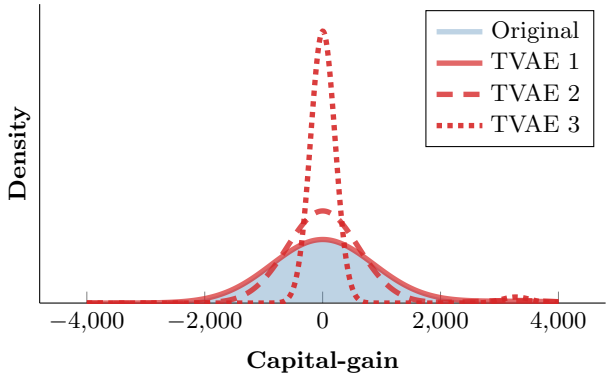


Figure 2: Marginal density of the *Capital-gain* variable in the Adult [2] dataset across three sequential regeneration iterations of a Tabular Variational Autoencoder (TVAE). The shaded region denotes the empirical density of the original data, and red curves (iterations 1–3) represent synthetic samples generated under repeated synthetic feedback. The progressive concentration of probability mass indicates variance collapse and loss of distributional support across iterations.

Marginal distortion is further amplified under iterative regeneration. Because synthetic data can be produced at scale [52], subsequent models may be trained on previously generated samples rather than on the original dataset [41]. We refer to this recursive setting as a Synthetic Feedback Loop (SFL). To quantify robustness under SFL, we define regeneration stability as the preservation of statistical structure across iterations. Under repeated regeneration, SFL can produce progressive variance shrinkage and loss of distributional support, consistent with *model collapse* [1].

We examine this SFL phenomenon in tabular synthesis. As shown in Fig. 2, unconstrained regeneration leads to systematic deviation from the original distribution across iterations. Such degradation is particularly consequential in educational datasets, where sub-representation of low-frequency groups can bias subgroup analysis and downstream prediction.

To address both marginal mismatch and regeneration instability, we introduce the Non-Parametric Gaussian Copula (NPGC). NPGC enforces exact empirical marginal preservation through non-parametric anchoring while modeling inter-variable dependencies in a Gaussian copula framework. NPGC supports continuous, integer, and categorical variables, and treats missing values as informative signals when estimating marginals and correlations. Beyond private synthesis, NPGC supports data augmentation, where empirical marginal anchoring ensures augmented samples preserve feature distributions, preventing statistical drift and erasure of rare subgroups.

NPGC integrates DP by injecting calibrated noise into both the marginal distributions and the correlation structure, providing formal privacy guarantees. By constraining synthetic generation to respect empirical marginals, NPGC mitigates variance degradation under SFL while maintaining competitive downstream utility and substantially lower computa-

tional cost than deep learning–based baselines. While NPGC improves marginal fidelity and regeneration stability, it models dependencies through a Gaussian copula on rank-transformed data. The Gaussian copula is parameterized solely by a correlation matrix and thus cannot represent interactions beyond linear rank correlation; as a result, complex nonlinear or higher-order dependencies may not be fully preserved.

We evaluate NPGC with respect to fidelity, privacy, regeneration stability, and downstream predictive utility, each formally defined in Section 4. Fidelity measures statistical agreement between real and synthetic data, privacy quantifies resistance to disclosure and inference attacks, and regeneration stability evaluates robustness under iterative synthesis.

Experiments are conducted on five datasets from the UCI Machine Learning Repository [11], a widely used benchmark collection of heterogeneous tabular datasets, including two education-focused datasets (*Student Dropout* [30], and *Student Performance* [6]). Figures 1 and 2 use the *Adult* [2] dataset due to its larger sample size, which enables clear visualization of marginal behavior and regeneration effects.

Across benchmarks, NPGC achieves higher fidelity and privacy scores than baseline methods while maintaining competitive utility and substantially lower computational cost. By enforcing empirical marginals, NPGC stabilizes regeneration under SFL and mitigates variance degradation.

We further evaluate NPGC on the OpenStax learning platform using engineer-generated test interaction logs from the platform's quality assurance system. The dataset exhibits severe categorical imbalance, a common characteristic of educational activity data. NPGC preserves empirical category proportions under this imbalance while satisfying privacy constraints. We demonstrate that the NPGC supports practical data sharing for instructional analytics without exposing individual-level records. We implement NPGC in Python as a synthesizer wrapper designed for straightforward integration into existing data workflows, enabling practical plug-and-play deployment.[1]

This paper is organized as follows. Section 2 reviews prior approaches to tabular data generation and related instability phenomena. Section 3 presents the formal formulation of NPGC. Section 4 describes the benchmark datasets and the evaluation protocols. Section 5 reports comparative results on fidelity, privacy, and regeneration stability. Section 6 details an applied case study on the OpenStax learning platform. Sections 7 and 8 discuss, conclude, and outline future directions.

2. RELATED WORK

Synthetic tabular data generation has long been studied as a mechanism for releasing useful data while controlling disclosure risk [37, 49]. In education, the tension between utility, fidelity, privacy, and validity constraints is particularly pronounced: Student-level datasets are high-dimensional and heterogeneous, combining continuous, integer/count, categorical, and ordinal variables within a single table, yet institutional and regulatory constraints often prevent broad sharing for benchmarking and reproducibility. Recent surveys and benchmark studies show that reported rankings of tabular generators can depend strongly on evaluation protocols, hyperparameter tuning effort, and available compute resources [19, 27, 23, 48]. These findings motivate synthesis methods that are not only competitive in utility but also stable, reproducible, and computationally accessible.

Following the requirement-oriented taxonomy of [49], we summarize representative model families and position our approach relative to their design trade-offs.

2.1 Deep Learning for Tabular Data

Deep generative models are widely used for tabular synthesis to capture complex cross-feature dependencies. GAN-based approaches such as TGAN [55], CTGAN [54], and TVAE [36], and CTAB-GAN/CTAB-GAN+ [57, 58] adapt the adversarial framework of [16] to mixed continuous and categorical data through conditional training and tailored normalization schemes. Deep learning models often achieve strong downstream predictive performance. However, because they optimize global or reconstruction objectives, marginal distributions are not explicitly constrained; univariate shapes can deviate from empirical distributions [25], particularly for skewed or multi-modal educational attributes (Fig. 1).

Diffusion- and score-based extensions, including TabDDPM [26], STaSy [22], CoDi [28], AutoDiff [50], FinDiff [39], and TabSyn [56], increase modeling flexibility through multi-step denoising objectives [45, 20]. LLM-based generators such as GReaT [4], built on Transformer architectures [51], provide flexible conditioning at the cost of substantial computational resources. Empirical studies further indicate that deep learning does not consistently outperform simpler approaches on tabular data and typically requires greater tuning effort [42, 23].

Given our objective of providing a lightweight, stable, and reproducible synthesizer for privacy-sensitive educational datasets, we restrict our deep learning baselines to representative GAN/VAE models (CTGAN and TVAE), enabling controlled comparison against a statistical alternative.

2.2 Parametric and Copula-Based Statistical Methods

Copulas provide a classical framework for modeling multivariate dependence by separating marginals from the joint structure under Sklar's theorem [44]. Gaussian and vine copulas are widely used in finance, marketing, survival analysis, and applied statistics [46, 9, 33, 7, 8, 21]. Their decoupled formulation enables efficient sampling and interpretability, motivating synthetic tabular systems such as the Gaussian-copula model implemented in the Synthetic Data Vault (SDV) [34], a Python library for generating synthetic tabular data, and related copula-based generators for complex data [3].

Most copula-based synthesizers assume parametric marginal

[1]The synthetic data generation toolkit is available: `https://github.com/gdiaz95/Synthetic_data_generation`

families. When educational attributes are skewed, discrete, or multi-modal, such assumptions can introduce systematic distortion. Differentially private extensions such as DPCopula [29] inject calibrated noise into marginal and correlation estimates to satisfy formal DP guarantees, but privacy noise may exacerbate misspecification effects. Nonparametric copula generators based on empirical CDFs relax distributional assumptions [38], yet they do not explicitly address DP accounting, regeneration stability, or reproducible low-overhead benchmarking in privacy-restricted educational settings.

NPGC retains the copula decomposition while replacing parametric marginals with privatized empirical anchors and enforcing positive semi-definite correction of the noisy correlation matrix. This design prioritizes exact marginal fidelity, computational efficiency, and reproducibility under constrained compute, positioning NPGC as a lightweight statistical alternative rather than a high-capacity hybrid or deep generator.

2.3 Recursive Training and Synthetic Feedback Effects

Researchers increasingly reuse synthetically generated data for downstream training, distillation, and iterative regeneration. When models train on data generated by earlier models, distributional drift can arise. Recent work documents instability under recursive self-training, including Model Autophagic Disorder (MAD) [1], model collapse [41], degradation of diversity when scaling with synthetic data alone [14], and progressive loss of variance and tail behavior in large models trained on recursively generated corpora [41]. In GAN settings, researchers attribute such behavior to the concentration of probability mass on limited modes and propose mitigation strategies such as VEEGAN [47]. Related analyses in LLM fine-tuning show that recursive adaptation can amplify distributional bias [31].

Although this literature focuses primarily on vision and language models, the underlying mechanism, accumulation of approximation error under repeated self-consumption, extends directly to tabular synthesis. In tabular data, even small marginal distortions can compound across generations, attenuating rare categories or low-frequency subgroups. In educational settings, such drift is particularly consequential because subgroup underrepresentation may bias downstream analytics, fairness assessments, and policy decisions.

To the best of our knowledge, model collapse and regeneration stability have not been systematically analyzed in privacy-sensitive tabular synthesis. We therefore introduce an explicit empirical evaluation of synthetic feedback effects in tabular data. In Section 5, we examine how marginal preservation influences recursive stability and whether constraining marginals mitigates variance shrinkage under synthetic feedback.

3. NON-PARAMETRIC GAUSSIAN COPULA (NPGC)

We formulate NPGC as a three-stage generative process grounded in Sklar's Theorem [44], which decomposes a joint distribution into univariate marginal distributions and a copula-based dependency structure. To provide formal privacy guarantees, differential privacy (DP) [13] is incorporated directly into both marginal estimation and correlation modeling. The procedure is defined as follows:

3.1 Marginal Transformation: $X \to Z$

Intuitively, this step reorders each variable into a common Gaussian scale while storing its observed distribution, allowing dependencies to be modeled separately from marginal shapes. NPGC maps the original data matrix $X \in \mathbb{R}^{n\times p}$ into a latent Gaussian space $Z \in \mathbb{R}^{n\times p}$ column-wise, where n denotes the number of records and p the number of features. Privacy is enforced during marginal estimation prior to Gaussian projection. For each feature column X_j ($j = 1, \ldots, p$), the transformation proceeds in two steps:

1. Mapping of X_j to a variable $U_j \in (0, 1)$ via a column-specific empirical probability integral transform; under consistency of $\widehat{F}_j$, $U_j \Rightarrow \mathcal{U}(0, 1)$.
2. Projection to the standard Gaussian space via $Z_j = \Phi^{-1}(U_j)$, where Φ denotes the cumulative distribution function of the standard normal distribution.

For continuous variables, we map observed values to the unit interval using a histogram-based estimate of the marginal CDF. Let $\widehat{F}_j$ denote the empirical CDF constructed from the non-missing entries of column j. We set $U_j = \widehat{F}_j(X_j)$ using linear interpolation, so that $U_j \in (0, 1)$.

For integer-valued variables with empirical probability mass function p_j and CDF $\widehat{F}_j$, we apply dequantization $U_j = \widehat{F}_j(X_j) - r\, p_j(X_j)$ with $r \sim \mathcal{U}(0, 1)$, which distributes probability mass uniformly over $(\widehat{F}_j(x^-), \widehat{F}_j(x)]$ for each observed value x, where x^- denotes the immediate predecessor of x in the ordered discrete support.

Categorical variables are treated analogously by estimating empirical category probabilities and sampling uniformly within each category's CDF interval. This embeds continuous, integer, and categorical variables into a common continuous domain $(0, 1)$.

Let f_{nan} denote the empirical fraction of missing values in column j. Observed numeric values are scaled to occupy $(0, 1 - f_{\text{nan}})$, while the upper interval $(1 - f_{\text{nan}}, 1)$ is reserved for missingness. This produces a disjoint partition of $(0, 1)$ in which observed variability and missingness are encoded separately. This representation also preserves correlations between missingness and observed variables. As a result, missingness patterns can be reproduced when they are reflected in the data (e.g., missing values occurring more frequently within specific subgroups). However, if missingness depends on unobserved factors, these dependencies cannot be fully recovered, since the copula captures only associations among observed variables. After uniformization, we apply the Gaussian projection $Z_j = \Phi^{-1}(U_j)$, where Φ is the standard normal CDF, yielding latent variables with approximately standard normal marginals for correlation estimation.

Concretely, DP is enforced by adding calibrated random noise to the statistics NPGC fits (the histogram bin counts

for each marginal and entries of the correlation matrix), scaled according to the sensitivity of each statistic and the privacy parameter ϵ. NPGC adds DP noise to two quantities, namely, the marginals and the correlation matrix, so we split the ϵ between them. This split is called a *privacy budget* $\epsilon = \epsilon_m + \epsilon_c$ with $\epsilon_m = \epsilon_c = \epsilon/2$, where ϵ_m is applied to marginal estimation and ϵ_c to correlation estimation. Marginal distributions are privatized using the Laplace mechanism under ϵ_m-differential privacy per column, with $\epsilon = 1$ by default and configurable by the user. For histogram bin counts h_k, we construct $\tilde{h}_k = \max(0, h_k + \eta_k)$ with $\eta_k \sim \text{Lap}(1/\epsilon_m)$. For categorical or integer counts c_ℓ, we construct $\tilde{c}_\ell = \max(0, c_\ell + \eta_\ell)$ with $\eta_\ell \sim \text{Lap}(1/\epsilon_m)$. The empirical CDF $\widehat{F}_j$ is then formed from the normalized perturbed counts and stored for inverse sampling (Section 3.3). Under sequential composition across all p columns, the total privacy cost of the marginal step is $p \cdot \epsilon_m$.

The resulting matrix $Z \in \mathbb{R}^{n \times p}$ has standard normal marginals across columns. Specifically, let Z_j denote the j-th column of Z. Then $Z_j \sim \mathcal{N}(0, 1)$, and the columns are suitable for correlation estimation in the latent Gaussian space.

3.2 Correlation and Gaussian Sampling

In simple terms, this step captures how variables co-vary after removing their individual distributions, enabling joint dependence to be reproduced independently of individual form. Once the data is transformed into the Gaussian space Z, we estimate and replicate the dependency structure. We compute the empirical correlation matrix $\widehat{R} = \frac{1}{n-1} Z^\top Z$. The matrix is privatized by adding symmetric Laplace noise to each entry. Since $Z_{ij} = \Phi^{-1}(\widehat{F}_j(X_{ij}))$ and the empirical CDF maps observations to ranks, the maximum value any entry of Z can take is $\Phi^{-1}(n/(n+1))$, yielding a data-independent per-entry sensitivity of $[\Phi^{-1}(n/(n+1))]^2/(n-1)$. Noise is added with scale $b = [\Phi^{-1}(n/(n+1))]^2/((n-1)\epsilon_c)$ per entry, so that each entry satisfies ϵ_c-differential privacy. Under sequential composition across all $p(p-1)$ off-diagonal entries, the total privacy cost of the correlation step is $p(p-1)\cdot\epsilon_c$. Let $\tilde{R}$ denote the noisy matrix. To enforce positive semi-definiteness, we compute the eigen-decomposition $\tilde{R} = V \Lambda V^\top$ and replace Λ with $\Lambda^+ = \text{diag}(\max(\lambda_k, \delta))$, where we use $\delta = 10^{-8}$ in practice. The matrix is then reconstructed as $\widehat{R} = V \Lambda^+ V^\top$ and normalized to have a unit diagonal. The resulting $\widehat{R}$ is stored for subsequent sampling.

We generate $Z_{\text{iid}} \in \mathbb{R}^{m \times p}$, where m denotes the number of generated samples, with rows sampled independently as $\mathcal{N}(0, I_p)$. Let $\widehat{R} = LL^\top$ denote the Cholesky factorization of the learned correlation matrix. Correlated samples are obtained via $Z_{\text{syn}} = Z_{\text{iid}} L^\top$. Each row of Z_{syn} then follows $\mathcal{N}(0, \widehat{R})$, yielding a synthetic Gaussian matrix that preserves the learned dependency structure.

3.3 Marginal Reconstruction: $Z_{\text{syn}} \to X_{\text{syn}}$

This step converts the synthetic data back into the original format of the dataset using the empirical marginal anchors, preserving the original distributions. Since sampling depends only on the privatized marginals $\widehat{F}_j$ and correlation matrix $\widehat{R}$, the inverse stage is a post-processing operation. We now map the Gaussian samples Z_{syn} back to the original feature space to obtain X_{syn} by reversing the forward transformation:

1. Map to uniform domain via the standard normal CDF: $U_{\text{syn},j} = \Phi(Z_{\text{syn},j})$.
2. Apply inverse empirical CDF, $X_{\text{syn},j} = \widehat{F}_j^{-1}(U_{\text{syn},j})$, using interpolation for continuous variables and bin assignment for discrete types.

The generative process yields $X_{\text{syn}} \in \mathbb{R}^{m \times p}$. Marginals are generated using empirical anchors stored during fitting: sorted empirical values for numeric columns and (noisy) category masses for categorical columns, applied through inverse empirical CDF sampling. In contrast to standard Gaussian copula models that assume parametric marginal families, NPGC relies directly on empirical (and privatized) marginal distributions while modeling dependencies solely through the Gaussian copula correlation structure.

To simplify usage, NPGC is implemented as a synthesizer class following a standard fit–sample paradigm. As shown in Listing 1, the user fits the model to a tabular dataset. The fit step estimates the empirical marginal distributions and the Gaussian copula correlation matrix. After fitting, the model can generate synthetic samples of arbitrary size, and the trained instance can be saved to or loaded from disk for reproducible generation. In practice, this follows a straightforward workflow: fit the model to real data, then generate synthetic samples from the learned distributions.

Listing 1: Example usage of the Non-Parametric Gaussian Copula (NPGC) synthesizer.

```
# Instantiate the NPGC synthesizer
synthesizer = NPGC()

# Fit the model to the original dataset (pandas
    DataFrame input)
synthesizer.fit(original_dataframe)

# Alternatively, load a previously trained model
synthesizer.load("models/synthetic_model.pkl")

# Generate synthetic samples (returned as a
    pandas DataFrame)
synthetic_data = synthesizer.sample(num_rows
    =1000)

# Save the fitted model for future reuse
synthesizer.save("models/synthetic_model.pkl")
```

An overview of the NPGC fit and sample procedures is shown in Fig. 3. The model consists of two components: privatized empirical marginals and a Gaussian copula correlation matrix. The fit phase estimates these quantities in a latent Gaussian space, and the sample phase generates correlated Gaussian draws followed by inverse marginal reconstruction to produce synthetic data. Details of the computational complexity for the fit and sample procedures are provided in Appendix B.1.

4. EXPERIMENTAL SETUP

The datasets reflect heterogeneous tabular structure, varying in sample size from $n = 649$ to $n > 45{,}000$ and in dimensionality from $p = 4$ to $p = 36$, with mixtures of continuous,

Table 1: Benchmark datasets used for evaluation, showing the number of instances and features for each dataset.

Dataset	Instances (n)	Features (p)
Adult [2]	48,842	14
Balance Scale [43]	625	4
Nursery [32]	12,960	8
Student Dropout [30]	4,424	36
Student Performance [6]	649	33

discrete, and categorical variables and missing values in several cases. We do not impose a theoretical minimum sample size. Robustness to dataset size is assessed empirically across benchmarks ranging from $n = 625$ to $n > 45{,}000$ (Table 1). NPGC maintains stable performance across this range, including the smallest dataset ($n = 649$), without dataset-specific tuning. For each dataset, we generate a synthetic dataset of size equal to the training partition ($m = 0.8n$). Each dataset is split once into fixed 80/20 train–holdout partitions, following standard practice, for fidelity and privacy evaluation. For utility, a separate 70/30 split is used for Train-on-Synthetic, Test-on-Real (TSTR) evaluation, as downstream classifier evaluation benefits from a larger test partition; both partitions use a predefined random seed. Synthetic samples are generated from the training split only. The holdout set is never used for model fitting and remains fixed across regeneration iterations.

We evaluate NPGC against four baseline models implemented using the Synthetic Data Vault (SDV) single-table synthesizers [34], selected to represent the main classes in tabular data synthesis: parametric copula models (PGC) [44], deep generative models (CTGAN, TVAE) [54, 53], and hybrid approaches (Copula-GAN). Baselines provide a consistent basis for comparison within the SDV framework, covering the main modeling paradigms used in tabular synthesis and enabling evaluation across parametric, deep learning, and hybrid approaches. Future work may evaluate NPGC against more recent tabular synthesis methods. Experiments are conducted on the benchmark datasets summarized in Table 1, obtained from the UCI Machine Learning Repository [11].

We evaluate synthetic data quality using three standard criteria,fidelity, utility, and privacy, following established practice [49]. In addition, we introduce a regeneration stability protocol that measures robustness under iterative synthetic feedback.

Fidelity measures agreement between the statistical structure of real and synthetic data. It is quantified using the SD Metrics single-table Quality Report [34], summarized by the Overall Score, defined as the arithmetic mean of the report's univariate and bivariate components: Column Shapes (univariate distribution agreement, including Kolmogorov–Smirnov comparisons for numerical variables) and Column Pair Trends (pairwise relationship preservation). These scores lie in $[0, 1]$, with 1 indicating perfect agreement under the SDMetrics definitions.

Privacy refers to protecting sensitive information in the original dataset and preventing the synthetic output from enabling re-identification attacks (e.g., linkage, inference, or membership attacks). In this work, privacy is evaluated using Discriminator AUC and Distance to Closest Record (DCR) Share [34]. Discriminator AUC is computed by training a binary classifier solely for evaluation to distinguish real from synthetic samples and measuring the area under the receiver operating characteristic curve. DCR Share quantifies proximity between synthetic and real records based on nearest-neighbor distances. Both metrics are bounded in $[0, 1]$, and values closer to 0.5 indicate lower separability and reduced memorization risk.

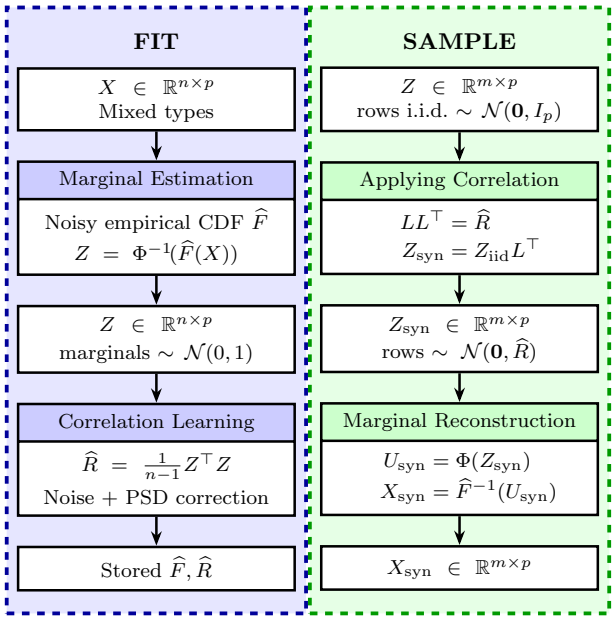


Figure 3: Overview of NPGC. FIT: Empirical marginal estimation via empirical Cumulative Distribution Functions (CDFs) with Gaussian projection, followed by correlation estimation $\widehat{R}$ with positive semi-definite (PSD) correction; the collection of marginals $\{\widehat{F}_j\}_{j=1}^{p}$, where p denotes the number of features (shown as $\widehat{F}$ in the figure for clarity), and $\widehat{R}$ are stored. SAMPLE: Independent Gaussian samples are correlated using Cholesky factorization $LL^\top$ of $\widehat{R}$ and mapped back through inverse empirical CDF reconstruction.

Utility refers to how fit for use the synthetic dataset is for the intended analytic task. In this work, utility is evaluated using the Train-on-Synthetic, Test-on-Real (TSTR) protocol together with runtime measurements. An XGBoost classifier [5] (fixed random seed, default hyperparameters) is trained on synthetic data and evaluated on real data using classification accuracy. We select XGBOOST as a baseline for TSTR due to its strong performance, robustness to heterogeneous feature types, and widespread use in educational data mining. Future work may evaluate utility across a broader set of predictive models. Accuracy is bounded in $[0, 1]$, with higher values indicating better predictive performance. Performance is compared to a classifier trained on real data under the same split, and relative degradation (Accuracy Drop) is reported. Training time denotes synthesizer fitting time, and evaluation time denotes synthetic sample

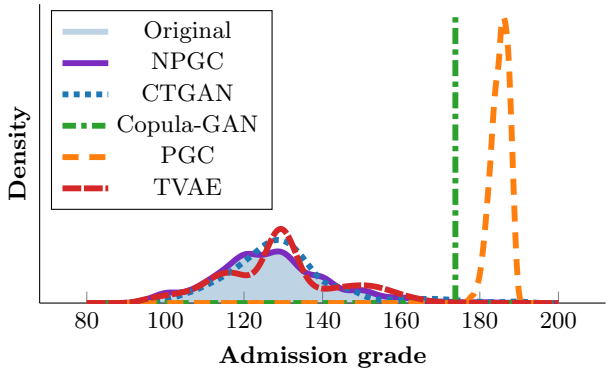


Figure 4: Marginal density of the Admission grade variable in the Student Dropout Success [30] dataset. The shaded region denotes the empirical density of the real data. Curves correspond to TGAN, TVAE, PGC, Copula-GAN, and the proposed NPGC. Deep learning and parametric baselines distort the marginal shape, while NPGC preserves it through empirical anchoring.

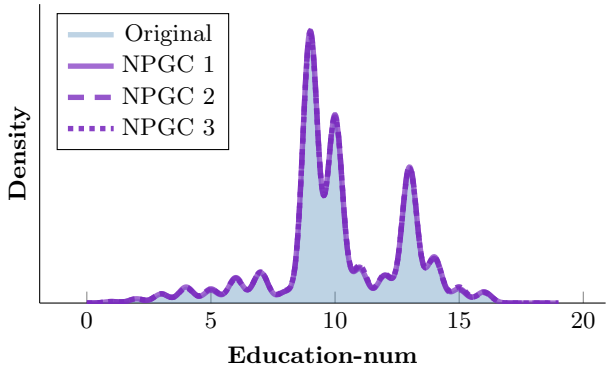


Figure 5: Marginal density of the *Education-num* variable in the Adult [2] dataset across three sequential regeneration iterations of NPGC. The shaded region denotes the empirical density of the original data, and purple curves (iterations 1–3) represent synthetic samples generated under repeated regeneration. The overlap across iterations demonstrates stability of the marginal distribution.

generation time.

Regeneration stability measures preservation of fidelity under repeated Synthetic Feedback Loop (SFL) iterations. It is evaluated using a 10-step synthetic feedback protocol on the Adult dataset [2]. At each iteration, the model is reinitialized and retrained on the synthetic dataset generated at the previous generation. Let $S^{(k)}$ denote the synthetic dataset generated at iteration k under SFL. Regeneration stability is defined as the preservation of the Overall Score between $S^{(k)}$ and the original real dataset across $k = 1, \ldots, 10$.

5. RESULTS

We evaluate NPGC against four baseline models across the five benchmark datasets described in Section 4. Results are reported for fidelity, utility, privacy, and regeneration stability and are averaged across all datasets (one aggregate score per dataset).

5.1 Fidelity

Fidelity measures how closely the synthetic data matches the statistical structure of the real data. In practice, higher fidelity indicates that summary statistics and distributions remain consistent between real and synthetic datasets. As shown in Table 2, NPGC achieves the highest Overall Score (0.962), as well as the highest Column Shapes score (0.985) and Column Pair Trends score (0.939). These results indicate that NPGC consistently outperforms all baselines in fidelity across both univariate and pairwise correlation preservation.

Figure 4 compares the marginal distribution of the *Admission grade* variable in the Student Dropout Success [30] dataset. The Copula-GAN collapses the distribution into a narrow spike near $x \approx 177.8$, indicating incorrect assumptions. PGC produces a unimodal approximation that does not capture the non-standard structure of the real data. NPGC consistently reproduces the empirical marginal.

5.2 Privacy

Privacy measures the extent to which individual records from the real dataset are protected. In practice, stronger privacy indicates a lower risk of distinguishing or linking synthetic samples to real individuals. As shown in Table 3, NPGC achieves the lowest Discriminator AUC (0.801) and DCR Share (0.527). Since values closer to 0.5 indicate lower separability and memorization, NPGC provides the strongest privacy performance among all methods.

5.3 Utility

Utility measures how useful the synthetic data is for downstream tasks. In practice, this reflects whether models trained on synthetic data behave similarly when applied to real data. Tables 4 and 5 show that TVAE attains the lowest accuracy drop (8.66%), while NPGC achieves a competitive drop of 16.99%, outperforming CTGAN and Copula-GAN and providing stronger privacy guarantees than TVAE.

Training times differ substantially: CTGAN and Copula-GAN exceed 790 seconds, TVAE requires 268 seconds, PGC 11.86 seconds, and NPGC 0.65 seconds. Sampling times are below 1.2 seconds for all methods. NPGC, as a non-deep-learning model, achieves orders-of-magnitude speedups over deep learning baselines and remains substantially faster than the PGC baseline.

Table 2: Average fidelity metrics across datasets. Overall Score reflects univariate (Column Shapes) and bivariate (Pair Trends) agreement between synthetic and real data. NPGC attains the highest fidelity across all metrics.

Model	Overall Score	Column Shapes	Pair Trends
NPGC	**0.962**	**0.985**	**0.939**
CTGAN	0.895	0.918	0.872
Copula-GAN	0.888	0.914	0.863
PGC	0.897	0.913	0.881
TVAE	0.832	0.883	0.781

Table 3: **Average privacy metrics across datasets. Lower values indicate reduced real–synthetic separability and lower memorization risk. NPGC achieves the strongest privacy performance.**

Model	Discriminator AUC	DCR Share
NPGC	**0.801**	**0.527**
CTGAN	0.853	0.567
Copula-GAN	0.871	0.558
PGC	0.850	0.549
TVAE	0.910	0.545

5.4 Regeneration Stability

Regeneration stability measures how well the synthetic data preserves its fidelity under repeated generation. In practice, this indicates whether the data maintains its variability and distribution across iterations. Regeneration stability is evaluated only on the Adult [2] dataset using the 10-step synthetic feedback protocol described in Section 4. As shown in Fig. 5, we observe the marginal distribution of the *Education-num* variable across successive NPGC iterations. NPGC correctly preserves the multi-modal empirical marginal distribution across iterations with minimal drift by construction, illustrating the absence of model collapse under repeated regeneration.

This marginal stability is reflected in the aggregate fidelity metric. As shown in Fig. 6, NPGC Overall Score decreases from 96.43% at iteration 1 to 95.27% at iteration 10 (1.16 percentage points). Copula-GAN decreases from 88.93% to 49.25%, CTGAN from 86.31% to 57.38%, TVAE from 89.33% to 65.56%, and PGC from 84.53% to 78.69%. These results show that NPGC maintains marginal and bivariate structure (as captured by the Overall Score) under repeated regeneration, while alternative methods exhibit substantial degradation.

6. EDUCATIONAL RESEARCH APPLICATION

We evaluate NPGC in an applied setting using engineer-generated test interaction logs from Assignable, the OpenStax homework platform that records student activity using the Experience API (xAPI) standard. The platform organizes student work hierarchically: each *assignment* contains several *activities*, and *assessment* activities contain individual *assessment-questions*. We analyze the activity layer only.

Table 4: **Average Train-on-Synthetic, Test-on-Real (TSTR) utility metrics across datasets using XGBoost. Synthetic Score denotes classification accuracy when training on synthetic data and testing on real data, and Accuracy Drop measures the relative decrease compared to training on real data. NPGC achieves the second-highest utility.**

Model	Synthetic Score	Accuracy Drop (%)
NPGC	**0.739**	**16.99**
CTGAN	0.619	25.73
Copula-GAN	0.647	22.80
PGC	0.675	17.04
TVAE	0.797	8.66

Table 5: **Average computational runtimes across datasets. Train Time denotes model fitting time, and Eval Time denotes sampling and evaluation time. NPGC achieves the lowest training time.**

Model	Train Time (s)	Eval Time (s)
NPGC	**0.650**	**1.19**
CTGAN	821.01	0.68
Copula-GAN	791.90	0.75
PGC	11.860	1.03
TVAE	268.46	0.34

Each Assessment is therefore counted as a single event, regardless of how many questions it contains. This yields 1,509 test interactions spanning four activity types: Reading, Assessment, Ancillary, and AI Feature (see Appendix A for definitions).

The empirical distribution is severely imbalanced: Assessment events account for 72.1% of records, Reading for 26.0%, Ancillary for 1.2%, and AI Feature for 0.7%. Such an imbalance is characteristic of educational interaction data, where dominant activity types vastly outnumber emerging or supplementary ones. Standard synthesis methods often collapse low-frequency categories under this regime, precisely the ones most relevant for studying new or rare student behaviors. We apply NPGC to generate 1,982 augmented synthetic records.

Figure 7 compares activity-type proportions between the test and synthetic datasets. NPGC preserves all four proportions to within roughly one percentage point, including AI Feature at 0.81% in the synthetic data versus 0.73% in the test data. The 11 test AI-Feature events are reflected as 16 synthetic events at matched relative frequency, demonstrating that empirical anchoring preserves rare-class structure even under aggressive class imbalance.

7. DISCUSSION

Our results carry several implications for synthetic data practice in educational research. We organize this discussion around three themes: what NPGC's marginal preservation means for analytic validity, what its limitations imply for specific research designs, and what directions would most benefit the EDM community.

Safeguarding Rare Educational Signals. When synthesizers smooth multi-modal distributions or collapse low-frequency categories, as Copula-GAN and CTGAN did across benchmarks, the distortion is not evenly distributed. Rare subgroups are disproportionately affected: sparsely observed demographic categories, low-frequency event types, and tail behaviors are attenuated first. In educational datasets, these are often the populations of greatest analytic interest: underrepresented students, rare help-seeking behaviors, or atypical learning trajectories. NPGC's empirical anchoring preserves these distributional features by construction, reducing the risk that synthetic data silently erases the students/researchers who most need to study. Our case study (Section 6) illustrates this concretely: NPGC maintained the categorical proportions of a severely imbalanced activity-type distribution, including AI-feature events occurring in

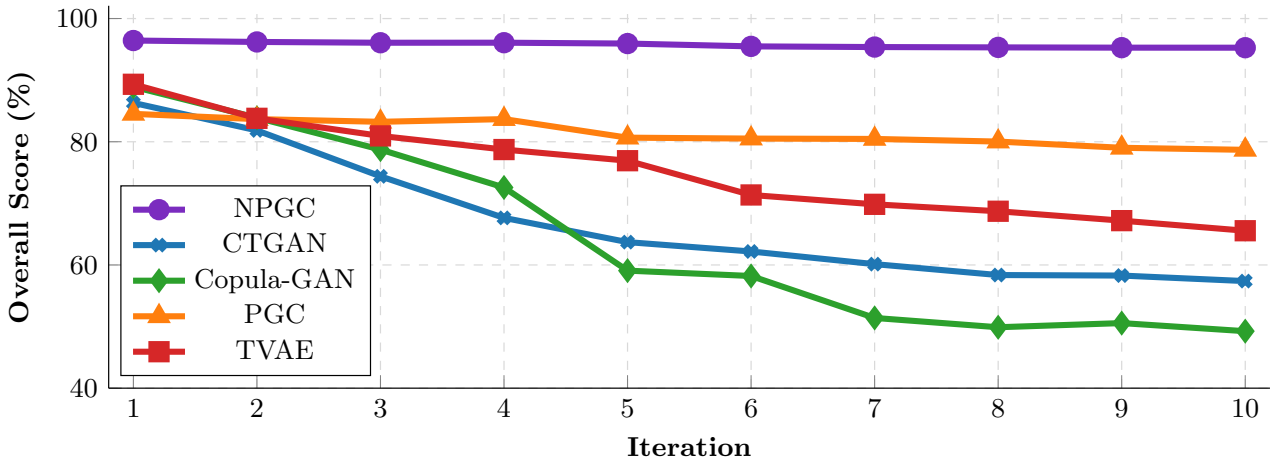


Figure 6: Overall Score (%) across 10 synthetic regeneration iterations on the Adult [2] dataset. The Overall Score aggregates univariate and bivariate distributional agreement between synthetic and real data. Curves correspond to CTGAN, Copula-GAN, PGC, TVAE, and NPGC.

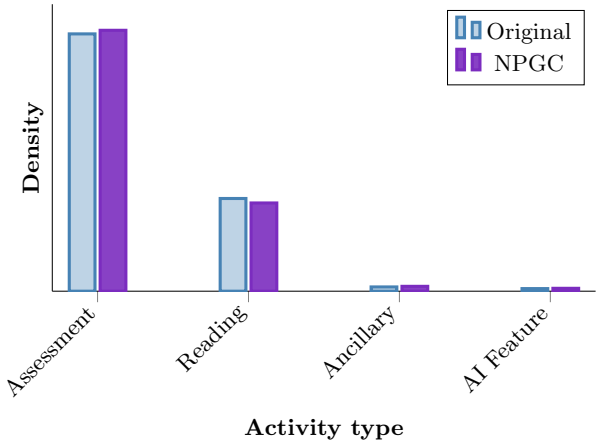


Figure 7: Proportion of activity types in the test and synthetic interaction dataset. The shaded bars represent empirical proportions from 1,509 test interaction records from the OpenStax platform. Synthetic proportions from 1,982 samples generated with NPGC closely match the empirical distribution across the four activity types, including low-frequency categories such as Ancillary and AI Feature. NPGC illustrates stable marginal preservation under deployment.

fewer than 1% of records.

The utility-fidelity tradeoff. NPGC achieves the highest fidelity but not the highest downstream classification accuracy; TVAE's accuracy drop (8.66%) is lower than NPGC's (16.99%). This gap likely reflects a fundamental tension: strict marginal anchoring constrains the joint distribution in ways that can sacrifice cross-feature predictive signal, particularly nonlinear interactions that deep generators capture through flexible function approximation. However, for many EDM use cases such as population characterization, distributional auditing, subgroup representation analysis, and descriptive reporting, marginal fidelity matters more than predictive utility. These differences have direct implications for common EDM tasks. In descriptive analyses, higher fidelity means that distributions of variables such as grades, activity counts, or demographic proportions remain consistent, so reported statistics do not change when using synthetic data. In predictive settings, comparable TSTR performance means that models trained on synthetic data behave similarly when evaluated on real data. For tasks such as dropout prediction or early warning systems, this reduces the risk that model outputs or intervention thresholds shift due to artifacts introduced during synthesis. Researchers should select their synthesizer based on their analytic objective.

Limitations and scope conditions. The Gaussian copula captures linear dependence structure in the latent space. Educational data frequently exhibits threshold effects, ceiling and floor artifacts, and nonlinear interactions (e.g., between prior knowledge and intervention response) that this assumption may not preserve. Additionally, our regeneration stability analysis was conducted on a single dataset; whether these properties generalize to high-cardinality educational features (e.g., course IDs or learning objective codes) remains an open question.

8. FUTURE DIRECTIONS

The most impactful extensions would address structures common in educational research: longitudinal panel data with temporal dependencies, multi-table relational schemas (students × courses × assessments), and conditional distributions relevant to causal inference in platform A/B experiments. Integrating flexible dependency models, such as vine copulas [8] or neural spline flows [12], while retaining empirical marginal anchoring could extend NPGC's applicability to these richer data structures without sacrificing its core marginal stability and reproducibility guarantees. Future work may also provide a more precise characterization of the dependency structures captured by the model, evaluate NPGC against more recent tabular synthesizers, and assess utility across a broader range of predictive models.

ACKNOWLEDGMENTS

This research is supported by the National Science Foundation under Award No. 2153481. We thank Dr. Jack Strelich and Dante Soares for their support in understanding the platform data.

APPENDIX

A. OPENSTAX ONLINE INSTRUCTIONAL SYSTEM

The OpenStax platform delivers digital instructional content and assessments to middle-school and post-secondary learners, recording fine-grained student interactions as time-stamped Experience API (xAPI) statements.

This paper uses full xAPI logs of example data generated during quality assurance tests of the system containing 7,506 records at three levels: *assignment* (the top-level container), *activity* (what the student engages with inside an assignment), and *assessment-question* (an individual question inside an assessment). We analyze the activity level only, yielding 1,509 records across four *activity_type* categories: **Reading** (`reading`), interaction with instructional content such as textbook pages; **Assessment** (`assessment`), initiation of a practice or formative evaluation; **Ancillary** (`ancillary`), access to supplemental materials such as videos or external references; and **AI Feature** (`promptly:preset`), an AI chatbot feature currently under development. Each Assessment is counted as a single event; the questions inside it are not counted separately. The empirical distribution is highly imbalanced. Assessment accounts for ∼72% of records and AI-feature for under 1%, which poses challenges for generative methods that rely on parametric density assumptions or adversarial training.

Using the dataset's training split, we apply NPGC to generate 1,982 synthetic interaction records: 1,449 Assessment, 490 Reading, 27 Ancillary, and 16 AI-feature events, with proportional agreement to the original distribution and preservation of both dominant and low-frequency categories. Because NPGC operates on an aggregated tabular structure rather than reproducing individual-level sequences, the released synthetic data preserves categorical frequency structure while mitigating identity disclosure risk, supporting release for educational research under privacy constraints.

B. BASELINE HYPERPARAMETERS

All baseline models were implemented using the SDV single-table synthesizers. Unless otherwise specified, default SDV hyperparameters were used without additional tuning.

Parametric Gaussian Copula (PGC) models each numeric column using a specified parametric family and couples variables through a Gaussian copula correlation structure. The settings below indicate the default numeric marginal family and two post-processing constraints: whether generated values are clipped to the observed range and whether numeric outputs are rounded to match discrete-valued columns.

Copula-GAN applies a copula-based transformation and then trains a GAN to model dependencies in the transformed space. The hyperparameters below define the training budget (epochs, batch size), representation size (embedding dimension), network capacity (hidden layer sizes), optimizer step sizes (learning rates), discriminator update policy (steps), and the PAC setting (number of samples packed per discriminator input). The last two options enforce range clipping and rounding during post-processing to better match the original data domain.

Table 6: Parametric Gaussian Copula (PGC) Hyperparameters

Parameter	Value
Default numerical distribution	beta
Enforce min/max values	True
Enforce rounding	True

Table 7: Copula-GAN Hyperparameters

Parameter	Value
Epochs	300
Batch size	500
Embedding dimension	128
Generator hidden dimensions	(256, 256)
Discriminator hidden dimensions	(256, 256)
Generator learning rate	2×10^{-4}
Discriminator learning rate	2×10^{-4}
Discriminator steps	1
PAC	10
Enforce min/max values	True
Enforce rounding	True

CTGAN is a conditional GAN designed for mixed continuous and categorical tabular data. The hyperparameters below specify the training budget (epochs, batch size), embedding size for conditional vectors and categorical representations, generator/discriminator capacity (hidden dimensions), optimizer step sizes (learning rates), discriminator update policy (steps per generator update), and PAC packing. As above, min/max enforcement clips generated values to the observed range and rounding enforces discrete-valued outputs where applicable.

TVAE is a variational autoencoder for tabular data. The hyperparameters below specify the training budget (epochs), latent/embedding size, encoder (compression) and decoder (decompression) layer widths, the strength of L2 regularization, and the loss weighting factor used in SDV's implementation. Min/max enforcement clips outputs to the observed range, and rounding enforces discrete-valued outputs where

Table 8: CTGAN Hyperparameters

Parameter	Value
Epochs	300
Batch size	500
Embedding dimension	128
Generator hidden dimensions	(256, 256)
Discriminator hidden dimensions	(256, 256)
Generator learning rate	2×10^{-4}
Discriminator learning rate	2×10^{-4}
Discriminator steps	1
PAC	10
Enforce min/max values	True
Enforce rounding	True

Table 9: TVAE Hyperparameters

Parameter	Value
Epochs	300
Embedding dimension	128
Compression layers	(128, 128)
Decompression layers	(128, 128)
L2 regularization scale	1×10^{-5}
Loss factor	2
Enforce min/max values	True
Enforce rounding	True

applicable.

NPGC is a nonparametric Gaussian copula synthesizer. The entries below summarize the privacy level used in experiments (the DP budget ϵ) and the modeling choices: no neural optimization, empirical (nonparametric) marginal anchoring, and dependence captured through a Gaussian copula correlation matrix.

B.1 Computational Complexity

Let n denote the number of rows in the training data and p the number of columns. For sampling, let m denote the number of generated rows. The computational cost of the non-parametric Gaussian copula synthesizer can be separated into the fitting stage and the sampling stage.

Fitting complexity. The algorithm learns the marginal distributions for each column and then estimates the latent Gaussian correlation matrix. For each of the p columns, the marginal-learning step requires sorting or equivalent ranking/uniqueness operations on up to n observations, yielding a cost of order $O(n \log n)$ per column. Across all columns, this contributes $O(pn \log n)$. After transforming the data into the latent Gaussian space, the empirical correlation matrix is computed from a matrix of size $n \times p$, which costs $O(np^2)$. When differential privacy is enabled, the noisy correlation matrix is projected onto the cone of valid correlation matrices through an eigen-decomposition, which costs $O(p^3)$. Therefore, the overall complexity of `fit()` is

$$O\big(pn \log n + np^2 + p^3\big)$$

when privacy is enabled, and $O\big(pn \log n + np^2\big)$ when the correlation-repair step is skipped.

Sampling complexity. The algorithm first generates an independent Gaussian matrix of size $m \times p$, which costs $O(mp)$. It then applies the learned correlation structure using a Cholesky factorization of the $p \times p$ correlation matrix. Computing the factorization costs $O(p^3)$, and multiplying the sampled Gaussian matrix by the Cholesky factor $O(mp^2)$. Finally, each column is transformed back through its inverse empirical CDF. For continuous and categorical variables, this step is typically linear in the number of generated samples, contributing approximately $O(mp)$ overall. Hence, the typical complexity of `sample()` is

$$O\big(mp^2 + p^3\big)\,.$$

There is, however, an important exception for integer-valued variables when minimum and maximum values are enforced. In that case, the inverse transformation compares each generated value against all unique observed integer levels. If a column has u_j unique integer values, then this step costs $O(mu_j)$. In the worst case, $u_j = O(n)$, so a single integer-valued column may require $O(mn)$. If $m = n$, this becomes $O(n^2)$. Thus, while sampling is usually cheaper than fitting, high-cardinality integer columns can make the sampling stage substantially more expensive in the worst case.

Table 10: NPGC Hyperparameters

Parameter	Value
Differential privacy parameter (ϵ)	1.0
Neural optimization	None
Marginal modeling	Empirical
Dependency modeling	Gaussian copula relation

Memory complexity. The memory requirements can also be separated into the fitting and sampling stages. During fitting, the algorithm stores the latent Gaussian matrix $Z \in \mathbb{R}^{n \times p}$, which requires $O(np)$ memory. It also stores the correlation matrix, which requires $O(p^2)$, and the learned marginal information. For numeric variables, the fitted model stores the sorted observed values for each column; across all columns, this requires $O(np)$ memory in the worst case. Categorical counts and metadata contribute at most lower-order terms relative to this. Therefore, the overall memory complexity of `fit()` is

$$O(np + p^2).$$

Sampling memory. During sampling, the algorithm stores the independently generated Gaussian samples, the correlated Gaussian samples, and the final synthetic table, each of size $m \times p$. Although these objects may coexist temporarily, their combined memory usage remains $O(mp)$ up to constant factors. The learned correlation matrix again requires $O(p^2)$ memory. Hence, the typical memory complexity of `sample()` is $O(mp + p^2)$.

Integer-valued columns. As with runtime complexity, integer-valued columns can increase memory usage. When minimum and maximum values are enforced, the inverse mapping step forms a temporary pairwise difference matrix between the generated values and all unique observed integer levels. If column j has u_j unique integer values, this temporary array requires $O(mu_j)$ memory. In the worst case, $u_j = O(n)$, so one such column may require $O(mn)$ additional memory. Therefore, the worst-case memory complexity of `sample()` becomes

$$O(mp + p^2 + mn).$$

REFERENCES

[1] S. Alemohammad, J. Casco-Rodriguez, L. Luzi, A. I. Humayun, H. Babaei, D. LeJeune, A. Siahkoohi, and R. Baraniuk. Self-consuming generative models go MAD. In *International Conference on Learning Representations*, 2024.

[2] B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996.

[3] F. Benali, D. Bodénès, N. Labroche, and C. de Runz. MTCopula: Synthetic complex data generation using copula. In *International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data (DOLAP)*, pages 51–60, 2021.

[4] V. Borisov, K. Sessler, T. Leemann, M. Pawelczyk, and G. Kasneci. Language models are realistic tabular data generators. In *International Conference on Learning Representations*, 2023.

[5] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[6] P. Cortez. Student Performance. UCI Machine Learning Repository, 2008.

[7] C. Czado. Pair-copula constructions of multivariate copulas. In *Copula Theory and Its Applications*, pages 93–109. Springer, 2010.

[8] C. Czado and T. Nagler. Vine copula based modeling. *Annual Review of Statistics and Its Application*, pages 453–477, 2022.

[9] P. J. Danaher and M. S. Smith. Modeling multivariate distributions using copulas: Applications in marketing. *Marketing Science*, pages 4–21, 2010.

[10] H. Du, S. Liu, L. Zheng, Y. Cao, A. Nakamura, and L. Chen. Privacy in fine-tuning large language models: Attacks, defenses, and future directions. In *Advances in Knowledge Discovery and Data Mining*, pages 326–344. Springer, 2025.

[11] D. Dua and C. Graff. Uci machine learning repository, 2017. University of California, Irvine, School of Information and Computer Sciences.

[12] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, 2019.

[13] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, pages 265–284. Springer, 2006.

[14] Y. Feng, E. Dohmatob, P. Yang, F. Charton, and J. Kempe. Beyond model collapse: Scaling up with synthesized data requires reinforcement. *arXiv:2406.07515*, 2024.

[15] A. Goldfarb and C. Tucker. Privacy and innovation. *Innovation Policy and the Economy*, 12:65–90, 2012.

[16] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

[17] A. Haim, R. Gyurcsan, C. Baxter, S. T. Shaw, and N. T. Heffernan. How to open science: Debugging reproducibility within the Educational Data Mining conference. In *International Conference on Educational Data Mining*, pages 114–124, 2023.

[18] A. Haim, S. Shaw, and N. Heffernan. How to open science: A principle and reproducibility review of the learning analytics and knowledge conference. In *International Learning Analytics and Knowledge Conference*, pages 156–164, 2023.

[19] L. Hansen, N. Seedat, M. van der Schaar, and A. Petrovic. Reimagining synthetic tabular data generation through data-centric AI: A comprehensive benchmark. In *Advances in Neural Information Processing Systems*, pages 33781–33823, 2023.

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

[21] H. Joe. *Dependence modeling with copulas*. Chapman and Hall/CRC, 2014.

[22] J. Kim, C. Lee, and N. Park. STaSy: Score-based tabular data synthesis. In *International Conference on Learning Representations*, 2023.

[23] G. C. N. Kindji, L. M. Rojas-Barahona, E. Fromont, and T. Urvoy. Tabular data generation models: An in-depth survey and performance benchmarks with extensive tuning. *Neurocomputing*, 2025.

[24] M. Klose, V. Desai, Y. Song, and E. Gehringer. EDM and privacy: Ethics and legalities of data collection, usage, and storage. In *International Conference on Educational Data Mining*, 2020.

[25] Y. Kossale, M. Airaj, and A. Darouichi. Mode collapse in generative adversarial networks: An overview. In *International Conference on Optimization and Applications*, pages 1–6, 2022.

[26] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko. TabDDPM: Modelling tabular data with diffusion models. In *International Conference on Machine Learning*, 2023.

[27] A. D. Lautrup, T. Hyrup, A. Zimek, and P. Schneider-Kamp. Systematic review of generative modelling tools and utility metrics for fully synthetic tabular data. *ACM Computing Surveys*, 2024.

[28] C. Lee, J. Kim, and N. Park. CoDi: Co-evolving contrastive diffusion models for mixed-type tabular synthesis. In *International Conference on Machine Learning*, 2023.

[29] H. Li, L. Xiong, and X. Jiang. Differentially private synthesization of multi-dimensional data using copula functions. In *International Conference on Extending Database Technology*, pages 475–486, 2014.

[30] M. V. Martins, D. Tolledo, J. Machado, L. M. T. Baptista, and V. Realinho. Early prediction of student's performance in higher education: A case study. In *Trends and Applications in Information Systems and Technologies*, pages 166–175, 2021.

[31] T. Meng, N. Mehrabi, P. Goyal, A. Ramakrishna, A. Galstyan, R. Zemel, K.-W. Chang, R. Gupta, and C. Peris. Attribute controlled fine-tuning for large language models: A case study on detoxification. *Association for Computational Linguistics*, pages 13329–13341, 2024.

[32] M. Olave, V. Rajkovic, and M. Bohanec. Nursery. UCI Machine Learning Repository, 1989.

[33] M. Othus and Y. Li. A Gaussian copula model for

multivariate survival data. *Statistics in Biosciences*, pages 154–179, 2010.

[34] N. Patki, R. Wedge, and K. Veeramachaneni. The synthetic data vault. In *International Conference on Data Science and Advanced Analytics*, pages 399–410, 2016.

[35] T. Phung, V.-A. Pădurean, A. Singh, C. Brooks, J. P. Cambronero, S. Gulwani, A. K. Singla, and G. Soares. Automating human tutor-style programming feedback: Leveraging GPT-4 tutor model for hint generation and GPT-3.5 student model for hint validation. In *International Learning Analytics and Knowledge Conference*, pages 12–23, 2024.

[36] Z. Qian, B.-C. Cebere, and M. van der Schaar. Synthcity: Facilitating innovative use cases of synthetic data in different data modalities. *arXiv:2301.07573*, 2023.

[37] J. P. Reiter. Releasing multiply imputed, synthetic public use microdata: An illustration and empirical study. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, pages 185–205, 2004.

[38] J. P. Restrepo, J. C. Rivera, H. Laniado, P. Osorio, and O. A. Becerra. Nonparametric generation of synthetic data using copulas. *Electronics*, page 1601, 2023.

[39] T. Sattarov, M. Schreyer, and D. Borth. FinDiff: Diffusion models for financial tabular data generation. In *International Conference on AI in Finance*, pages 64–72, 2023.

[40] Y. Shi, K. Yu, Y. Dong, and F. Chen. Large language models in education: A systematic review of empirical applications, benefits, and challenges. *Computers and Education: Artificial Intelligence*, 2026.

[41] I. Shumailov, Z. Shumaylov, Y. Zhao, N. Papernot, R. Anderson, and Y. Gal. AI models collapse when trained on recursively generated data. *Nature*, pages 755–759, 2024.

[42] R. Shwartz-Ziv and A. Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, pages 84–90, 2022.

[43] R. Siegler. Balance Scale. UCI Machine Learning Repository, 1976.

[44] A. Sklar. Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris*, 8:229–231, 1959.

[45] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2015.

[46] X.-K. P. Song. Multivariate dispersion models generated from Gaussian copula. *Scandinavian Journal of Statistics*, pages 305–320, 2000.

[47] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton. VEEGAN: Reducing mode collapse in GANs using implicit variational learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[48] M. C. Stoian, S. Dyrmishi, M. Cordy, T. Lukasiewicz, and E. Giunchiglia. How realistic is your synthetic data? constraining deep generative models for tabular data. In *International Conference on Learning Representations*, 2024.

[49] M. C. Stoian, E. Giunchiglia, and T. Lukasiewicz. A survey on tabular data generation: Utility, alignment, fidelity, privacy, and beyond. *arXiv:2503.05954*, 2025.

[50] N. Suh, X. Lin, D.-Y. Hsieh, M. Honarkhah, and G. Cheng. AutoDiff: Combining auto-encoder and diffusion model for tabular data synthesizing. *arXiv:2310.15479*, 2023.

[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[52] T. Wiehn. Synthetic data: From data scarcity to data pollution. *Surveillance & Society*, 22(4):472–476, 2024.

[53] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional GAN. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[54] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional GAN. In *Advances in Neural Information Processing Systems*, 2019.

[55] L. Xu and K. Veeramachaneni. Synthesizing tabular data using generative adversarial networks. *arXiv:1811.11264*, 2018.

[56] H. Zhang, J. Zhang, Z. Shen, B. Srinivasan, X. Qin, C. Faloutsos, H. Rangwala, and G. Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *International Conference on Learning Representations*, 2024.

[57] Z. Zhao, A. Kunar, R. Birke, and L. Y. Chen. CTAB-GAN: Effective table data synthesizing. In *Asian Conference on Machine Learning*, pages 97–112, 2021.

[58] Z. Zhao, A. Kunar, R. Birke, H. Van der Scheer, and L. Y. Chen. CTAB-GAN+: Enhancing tabular data synthesis. *Frontiers in Big Data*, 2024.

Temperature and Persona Shape LLM Agent Consensus With Minimal Accuracy Gains in Qualitative Coding

Conrad Borchers
Carnegie Mellon University
cborcher@cs.cmu.edu

Bahar Shahrokhian
Arizona State University
bshahrok@asu.edu

Francesco Balzan
University of Bologna
francesco.balzan3@unibo.it

Elham Tajik
University at Albany
etajik@albany.edu

Sreecharan Sankaranarayanan
Extuitive Inc.
sreecharan.primary@gmail.com

Sebastian Simon
University of Copenhagen
sas@psy.ku.dk

ABSTRACT

Large Language Models (LLMs) enable new possibilities for qualitative research at scale, including annotation and qualitative coding of educational data. While LLM-based multi-agent systems (MAS) can emulate human coding workflows, their benefits over single LLM agents for coding remain poorly understood. To that end, we conducted an experimental study of how persona and temperature of component agents of a MAS shapes consensus-building and coding accuracy for dialog segments. LLMs were prompted to code these segments deductively using a mature codebook with 8 codes and high inter-rater reliability derived from prior research. Our open-source MAS mirrors deductive human coding through structured agent discussion and consensus arbitration. Using six open-source LLMs (with 3 to 32 billion parameters) and 18 experimental configurations, we analyze over 77,000 coding decisions against a gold-standard dataset of human-annotated transcripts from online math tutoring sessions facilitated by educational software. Temperature significantly impacted whether and when consensus was reached across all six LLMs. MAS with multiple personas (including neutral, assertive, or empathetic) significantly delayed consensus in four out of six LLMs compared to uniform personas. In three of those LLMs, higher temperatures significantly diminished the effects of multiple personas on consensus. However, neither temperature nor persona pairing led to robust improvements in coding accuracy. Single agents matched or outperformed MAS consensus in most conditions. Qualitative analysis of MAS collaboration and coding disagreement may, however, improve codebook design and human-AI coding.

Keywords

large language models, qualitative coding, multi-agent systems, personas, temperature

Conrad Borchers, Bahar Shahrokhian, Francesco Balzan, Elham Tajik, Sreecharan Sankaranarayanan, and Sebastian Simon. Temperature and Persona Shape LLM Agent Consensus With Minimal Accuracy Gains in Qualitative Coding. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 280–292. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039879

1. INTRODUCTION

Educational research has seen a recent surge of multimodal data produced by increasingly rich forms of user-system interaction [3, 41, 8, 4]. One illustrative example is remote tutoring, where humans and students converse in the process of using educational technology [55, 61]. These contexts produce transcripts and logs of chat interactions that promise rich insights into the learning process. The surge in rich data stands in contrast with the process of analyzing it: qualitative analysis methods for such data, for example, thematic analysis (TA), requires human skills and domain knowledge and scale poorly with the size of the analyzed dataset. Novel methods of drawing insights from this data are needed, and educational data mining can contribute to the analysis and evaluation of these methods.

To that end, the potential of LLMs for qualitative research has sparked interest among the educational research community. Prior work has demonstrated the promise and limitations of using LLMs for inductive and deductive coding, raising questions about reliability, interpretability, and the degree of human oversight needed. Recent developments in LLM-based multi-agent systems (MAS) offer new possibilities for enhancing the robustness of LLM-based coding, mimicking the collaborative nature of human analysis by distributing roles [48, 44, 42]. One key benefit of MAS is that they can implement LLM self-consistency through repeated stochastic generation and aggregation across multiple agents (e.g., re-prompting and majority voting), improving robustness to sampling variance [60]. This study builds on that foundation to comprehensively study how temperature and consensus tuning can improve thematic coding accuracy and consensus in educational research settings, if at all.

The contributions from this research are three-fold. First, we introduce a system-centered contribution by developing and empirically evaluating a multi-agent framework for LLM-based deductive coding. The system supports integration of open-source language models and parameter configurations, enabling comparative analyses of their effects on coding reliability, accuracy, and interpretability. Leveraging this framework, we provide a novel empirical comparison of single-agent and multi-agent coding performance aligned with human-coded data, revealing that consensus-making reliably improves coding accuracy only within our sample for a single LLM and code at low temperatures. Second, we

systematically evaluate how different LLMs, agent persona congruency, and temperature settings influence consensus-building outcomes using mixed-effects modeling across six models of varying size (3 to 32 billion parameters). This addresses a critical gap in the literature; while persona traits and temperature are theorized to capture multiple perspectives [64, 53] and affect the degree to which agents converge on a shared interpretation or consensus [48], their specific impacts on coding dimensions have lacked rigorous evaluation in prior work [29, 50, 44, 42]. Third, we present a qualitative analysis of agent deliberation traces to examine how consensus emerges through design choices that closely parallel human-led thematic analysis practices [47]. By identifying how and when agents revise or reinforce codes during deliberation, this analysis offers deeper insights into the interpretability and epistemic limits of LLM-driven qualitative coding.

To build a methodology for investigating this, we draw from [4], who employed GPT-4 in a hybrid collaboration with human coders to co-construct codebooks for analyzing tutoring transcripts in math learning facilitated by technology. We extend this work to evaluate how varying system configurations, including different LLMs, persona prompts, and temperature values, shape the thematic outcomes and influence multi-agent coding, focusing on consensus-making and accuracy against human ground truth. Our findings advance theoretical understanding and practical implementation of LLM-assisted qualitative methods by identifying how, when, and why MAS may improve qualitative coding accuracy. Our open-source MAS is compatible with any open-source LLM via Ollama [37] and produces deliberation traces between agents that are scrutable for researchers and practitioners. This work contributes to ongoing efforts in the field to optimize LLM-mediated interpretive tasks through deliberate system design and structured variation in agent behavior. The research questions guiding this study are as follows:

- **RQ1:** How do agent persona and temperature influence the frequency of consensus-building in a multi-agent LLM system for thematic coding?
- **RQ2:** Does multi-agent coding with different agent personalities and temperature settings improve upon single-agent coding compared to the human baseline?
- **RQ3:** What qualitative patterns of successful consensus-making arise in multi-agent thematic coding?

2. BACKGROUND

This section reviews relevant literature on thematic analysis, the automation of qualitative coding with LLMs, and the emergence of MAS to enhance reliability and interpretability in LLM-driven analysis. It outlines the core processes of thematic analysis (foundational to qualitative coding and established research methods), discusses recent studies that apply LLMs to automate coding, and highlights how multi-agent approaches aim to simulate human collaboration to improve consensus and accuracy.

2.1 Thematic Analysis

Qualitative methods like TA offer the potential to move beyond quantitative metrics, which often fail to capture the nuanced dimensions that shape learning processes. TA addresses this limitation by uncovering latent themes and patterns through an interpretive and negotiated process among researchers, ultimately leading to the emergence of meaning [10, 11]. TA can be conducted using either an inductive or deductive approach. Inductive analysis follows a bottom-up approach, in which themes emerge from the data rather than being shaped by pre-existing theories or frameworks. In contrast, deductive analysis takes a top-down approach, guided by predefined theoretical constructs [10].

There are different types of thematic analysis, including reflexive thematic analysis, grounded theory, and content analysis. While these approaches are foundational for conducting qualitative research, translating their underlying theoretical frameworks into consistent practice can introduce variability, depending on a range of contextual factors [4, 32]. To mitigate these challenges, researchers often adopt strategies such as: (1) ensuring transparency in the analytic process and thoroughly documenting procedures, (2) iteratively refining the codebook, and (3) establishing inter-coder reliability [4]. Although these practices enhance the trustworthiness of the findings, they are also time-intensive, particularly when working with large datasets. As a result, there is growing interest in leveraging LLMs to streamline and support the TA process. The following section explores this emerging area in greater detail.

2.2 Automated Thematic Analysis

Although semi-automated, qualitative data analysis tools such as *NVivo* [21], *ATLAS.ti* [51], and *MAXQDA* [25] support coding efficiency; they rely on manual coding and interpretation, limiting scalability. Recent work has explored the potential of LLMs to automate or semi-automate TA. For instance, [4] compared four coding workflows—manual, automated, and two hybrid GPT-4-assisted approaches—on Algebra I tutoring transcripts. Hybrid methods yielded codebooks with greater clarity, mutual exclusivity between categories, and inter-coder agreement than either fully manual or fully automated approaches. Similarly, [33] evaluated GPT-4 across three educational datasets, testing various prompting strategies (e.g., zero-shot, few-shot, contextual). GPT-4 achieved substantial agreement with human coders ($\kappa > 0.70$ for 25 of 34 constructs), although it struggled with abstract or ill-defined categories. These findings highlight the potential of LLMs for inductive analysis, though a continued need for human oversight remains.

LLMs have also been studied for deductive coding. [63] found GPT-4 outperformed nCoder in recall and provided helpful feedback for construct refinement, though nCoder achieved higher accuracy. Likewise, [43] proposed GROPROE, an iterative prompting method for deductive TA using theory-based codebooks. After 60 prompt refinements, GPT-4 matched human inter-rater agreement (76%) on student reflections but remained prone to sycophantic bias. As [17] highlights, interpretive nuance and contextual understanding still require human judgment.

While promising, these approaches often depend on human feedback and small datasets. In contrast, recent work using MAS for TA, such as Thematic-LM [42], suggests that distributed LLM agents may support more autonomous and

scalable analysis. The following section introduces this and similar MAS-based approaches to thematic analysis.

2.3 Multi-Agent Thematic Analysis

Most prior work on LLM-assisted qualitative analysis has relied on single-agent models, meaning LLMs are applied to code data and no other LLM prompt processes or interacts with intermediate LLM output [33, 63, 17]. Past work has identified several limitations of this approach. [42] highlights three key issues: (1) the need for human oversight to monitor and contextualize LLM outputs; (2) limited perspective diversity, as single agents often mirror the coder's viewpoint; and (3) the absence of iterative refinement, hindering the model's ability to revise earlier codes.

In response, recent studies have explored multi-agent LLM systems to address these challenges (e.g., [48]). For instance, [42] proposed Thematic-LM, a multi-agent framework assigning distinct roles—such as coder agents, reviewers, and aggregators—to reduce thematic homogenization and enhance representativeness. Extending this approach, [44] demonstrated the effectiveness of MAS across various qualitative methods (e.g., grounded theory, content analysis), reporting an 87% practitioner satisfaction rate and improvements in both efficiency and scalability.

However, none of these studies have systematically explored how system parameters, such as temperature or agent persona, shape the effectiveness of consensus-building or coding accuracy in MAS. Moreover, while previous work has proposed architectural frameworks for role assignment among agents, there has been little empirical analysis of how these configurations influence the interpretability or consistency of LLM-generated codes. This leaves open critical questions about whether MAS approaches yield more reliable or representative qualitative insights than single-agent models and under what conditions such improvements occur, a gap we address through systematic prompt experimentation.

2.4 Consensus in Multi-Agent LLM Systems

Consensus seeking is a fundamental challenge in MAS, where agents with divergent initial states must converge toward shared solutions. In LLM-based MAS, agents naturally adopt an "average strategy" when no explicit consensus instructions are provided, computing the average of all agents' current states [14]. [14] have identified several critical factors that affect consensus dynamics in MAS-LLM systems, including the number of agents, agent persona, and temperature settings. For example, they demonstrated that suggestible agents in two-agent configurations can exhibit oscillatory behavior due to mutual influence without resolution. However, increasing the number of agents stabilizes the system through majority-supported states. In two prior studies, mixed-persona systems with heterogeneous agent groups achieved superior performance in complex reasoning and consensus tasks compared to homogeneous groups [14, 64]. However, we are not aware of any study that has demonstrated similar effects in qualitative coding. Moreover, these findings reveal that consensus achievement in MAS depends on agent characteristics, system parameters, and task requirements rather than purely on algorithmic design.

2.5 LLM Persona Research

Persona and Personality Traits in LLMs. With the advent of LLM agents and their ability to mimic human linguistic behavior, many researchers used the same questionnaires designed to understand human traits (stable differences between people), such as the Big Five personality model, with LLM agents [22]. In this literature, personality is defined as a stable set of dispositions. In contrast, a persona refers to a situationally induced role or style specified through prompting, instructions, or contextual framing (potentially aligned with a certain personality trait). The results of persona research in LLMs have been mixed. Some research has shown that LLM agents can exhibit intrinsic and extrinsic personality traits both lexically and in decision-making or downstream tasks (e.g., [49], [22], and [56]). For example, [22] found that GPT-3.5 and GPT-4 responses matched designated Big Five personality profiles when appropriately prompted. On the other hand, growing evidence points to significant instability in LLM persona expressions. For example, [20] demonstrated that trivial changes to prompt phrasing or answer ordering can dramatically alter LLM personality tests. These findings reveal that although LLM "personalities" color dialogue style and behavioral patterns, the mechanisms underlying these effects remain poorly understood. Chittem et al. [16] further argues that reliably inducing traits in LLMs requires continuous, nuanced control beyond simple instructions. This uncertainty raises critical questions about the extent to which induced personalities impact LLM agent performance and decision-making in real-world applications, such as deductive coding, a gap that our work aims to address.

Multi-Agent Systems and Personality. More directly relevant to multi-agent collaboration, research by [64] and [14] demonstrated that systems composed of agents with distinct personalities, such as "stubborn" and "suggestible" agents, outperform homogeneous agent groups in complex reasoning and consensus tasks. In particular, [14] showed how personality traits interact with network topology to influence convergence dynamics: stubborn agents tended to lead to consensus outcomes, sometimes producing hierarchical structures or clustering effects, especially in partially connected networks. Therefore, embedding diverse, interacting personalities in multi-agent LLM systems could also shape the consistency, agreement, and accuracy of multi-agent TA systems, though broader evidence of such effects, especially for qualitative coding, is lacking in the literature. In contrast, when most or all agents are stubborn, consensus becomes significantly harder to achieve regardless of the number of agents, since such agents rarely revise their positions and the likelihood of deadlock increases [14]. Our study builds on this foundation by contrasting dominant and permissive LLM personas, and their combinations, in a rigorous experimental setup. [64] suggested that odd-numbered agent groups outperformed even-numbered ones by avoiding tie situations during consensus. Therefore, each of our experiments introduces a third agent whenever consensus between the first two agents is not achieved.

3. SYSTEM DESIGN

3.1 Emulating Human Coding

Our system design follows established qualitative research practices for deductive coding with a predefined codebook.

In standard approaches, at least two trained coders independently assign codes to data segments, assess inter-coder reliability, and iteratively refine codes through discussion, with a third coder often serving as an adjudicator in cases of disagreement [4, 63, 9, 13]. We replicate this process by deploying multiple independent AI agents to initially code identical data segments without access to one another's coding decisions, thereby preserving analytical independence. For each coding turn, agents were provided with the current text, the LLM conversation history for the data point being coded (up to the current turn), the instructional prompt, and the codebook. Whereas human coders typically receive explicit training to establish a shared understanding of the coding procedure, AI agents were guided by a system prompt that provided the necessary codebook, contextual information, and procedural instructions. Disagreements were resolved by introducing a third, neutral agent who adjudicated them.

Inter-Coder Reliability (ICR) assessment Agreement between coders is assessed using statistical measures to evaluate consistency [15, 12]. For instance, according to the literature, a Cohen's kappa of 0.61 or higher indicates substantial agreement between the two coders [28]. Until such an agreement is reached, individual inconsistencies are revisited. This step is approximated in our system by determining whether the two agents fully align on their coding to assess consensus among AI agents over individual data segments.

Disagreement resolution. When coding discrepancies arise, human coders typically engage in structured discussions to reach consensus. This process is often facilitated by a third party or formal consensus process, especially if the discussion between agents does not resolve the disagreement. The new party can activate the majority rule by being the odd addition, or might be a more experienced researcher who can overrule the other coders and make the final decision [38]. To replicate this critical step, we implemented a consensus agent with a balanced, reflective personality that, upon detecting repeated disagreement (in the previous two turns) between the two coders, makes final coding decisions by considering the original coding prompt, each agent's initial classification, and each agent's discussion.

3.2 Injecting Personality

Methodological Foundation and Trait Selection. Building on the foundations established in Sections 2.4 and 2.5, we follow recent research that demonstrates that LLMs can be effectively prompted to exhibit distinct persona profiles [36]. Prior work has established that certain dimensions of the Five-Factor Model exert greater influence in group settings than others. [5] found that while conscientiousness showed no relationship to processes and outcomes at individual or group levels, the proportion of extraverted members correlated with enhanced task focus and group performance. Similarly, [35] demonstrated that agreeableness predicted superior interpersonal cooperation and reduced conflict in team environments. Based on these empirical findings, our study focuses on agents with either high extraversion or high agreeableness, alongside one agent maintaining neutral levels across all traits. This design allows systematic examination of how assertiveness versus cooperation influences consensus formation in multi-agent qualitative coding tasks [58]. The selection of these specific traits is further justified by research showing that prompt politeness and assertiveness significantly influence LLM compliance, including in the generation of false information [59]. In consensus-making scenarios during qualitative coding, dominance or compliance behaviors may yield different outcomes depending on the accuracy of the initial codes. To investigate these dynamics, each personality trait was implemented through prompt engineering and evaluated for its impact on consensus formation, alignment with human codes, and fallback frequency. Here are the personality profiles used in this study:

Bold and Dominant but Elaborative: This personality profile, used for coder agents, reflects assertiveness and extraversion while maintaining a willingness to engage in dialogue. It corresponds to the "bold" category in personality type and was used in multi-agent scenarios. **Empathetic and Open-Minded:** A permissive and collaborative profile emphasizing agreeableness and openness. Labeled as "empathetic" in personality type, this profile was also used in multi-agent settings to encourage broad discussion and interpretation. **Balanced and Reflective:** A neutral profile was applied to the single-agent (control condition) and consensus-agent.

3.3 Large Language Models

Another challenge in prompting LLM agents to behave in a certain way is that some LLMs tend to have an inherent personality as reported by [39] and [26]. These inherent traits can lead to variability in how models respond to induced personality prompts; some may resist the induced traits and revert to their default behavior, while others may behave inconsistently or unpredictably. [49] and [39] demonstrated that **instruction fine-tuned models** exhibit more reliable and consistent personality traits compared to their base (non-fine-tuned) counterparts. Given our focus on understanding the effect of personality injection and practical resource constraints, we prioritized LLM prompt instruction over task- or persona-specific fine-tuning. To investigate how personality and behavioral consistency vary across different architectures, we evaluated a diverse set of open-source language models spanning parameter sizes and development lineages. The models included in our analysis are:

LLaMA 3.2 3B [57], developed by Meta, is a lightweight decoder-only transformer model designed for efficient inference in resource-constrained settings, while still performing well on reasoning and multilingual tasks. This model is not instruction-fine-tuned by default. **OpenHermes V2 (7B)** [54] is a community-developed model based on the Mistral architecture, fine-tuned for instruction-following. It demonstrates strong performance on multi-turn dialogue and reasoning benchmarks such as MT-Bench and ARC, ranking among the top-performing 7B open models in humanaligned chat tasks [65]. **WizardLM 2 (7B)** [62] is jointly developed by Microsoft and Alibaba, optimized for complex multi-turn instruction tuning and is particularly effective in multi-step reasoning and contextual dialogue. **Phi 3 (14B)** [2] is released by Microsoft, utilizes a dense transformer architecture, and is instruction fine-tuned. This model is trained on a compact, highly curated dataset of synthetic educational material; it performs competitively on benchmarks involving reasoning, math, and safety. **Mistral-small 3.2 (24B)** [34] is an upgrade of their Small-3.1 24B model, op-

timized for instruction-following, reducing repetition, and improving function calling. This high-capacity model uses grouped-query attention for efficient inference, excelling at multilingual and factual question-answering tasks. **Deepseek R1 (32B)** [19] is a large decoder-only model developed by DeepSeek AI, trained on both code and natural language using a mixture-of-data approach. It achieves strong performance across a variety of benchmarks, particularly excelling in reasoning-heavy tasks and long-context understanding.

Our system and experimental evaluation code are fully open-source (repo: [1]) and compatible with the Ollama open-source framework for LLMs [37]. Codebooks and data can be requested from [4] to reproduce our analysis results using our code. The same applies to the LLM collaboration data, which often contains verbatim copies of the input data and is therefore under the same data-sharing constraints.

4. METHODS

4.1 Study Context

4.1.1 Data Set

The data for this study were originally collected and annotated by [4] and shared with the research team of the present study. The mentioned study was conducted using transcripts from high-dosage math tutoring sessions across two schools, in which trained tutors provided personalized, small-group instruction to support students' mathematics learning. The transcripts were drawn from four 60-minute virtual tutoring sessions. The participants were 9th-grade students enrolled in Algebra I, attending high-poverty schools in an urban region of the northeastern United States. The sessions took place between 2022 and 2023. For this study, we analyzed 579 tutor-student dialog segments from three session transcripts. Each segment is treated as an individual data point for model evaluation.

4.1.2 Codebook Development

Our annotation scheme builds on the hybrid human-AI codebook developed in [4], which was rated more favorably by human reviewers than exclusively LLM-built codebooks and found to support more reliable hybrid coding. The codebook includes eight high-level categories (see also study repository [1]): **Greetings, Instructions, Guiding Feedback, Aligning to Prior Knowledge, Understanding/Engagement Tutor, Technical or Logistics, Encouragement** and **Time Management**.

4.2 Multi-Agent System

We developed a multi-agent system to simulate and evaluate AI-assisted qualitative deductive coding based on a codebook. The system includes four components: (1) a single-agent coder, (2) a dual-agent discussion module, (3) a consensus agent, and (4) a hybrid human-AI dataset for benchmarking. All agents operate on the same codebook (see Section 4.1.2), implemented as a labeled dictionary of binary flags for each codebook category. (e.g., "Tutor: What's good, XX? {'Greeting': 1, 'Instruction': 0, ...}"). Each utterance includes a human-generated reference code derived from double-coding and resolution by trained annotators. The codebook is available in the digital appendix [1]).

The *SingleAgentCoding* module simulates individual annotation. It instantiates a single language model with a fixed persona (e.g., neutral, empathetic, or bold; see Section 3.2) and instructs it to assign codes to a given utterance based on the codebook definitions. The dialog segment and the full codebook are included in the user prompt. To simulate inter-annotator discussion, the *DualAgentDiscussion* module employs two distinct agents with personalities. Agents take turns critiquing each other's code using textual reasoning and are prompted to reflect on discrepancies across multiple rounds. In a post-processing routine, the discussion is checked for completeness, and annotations are extracted. The *ConsensusAgent* module finalizes coding by reviewing both agent responses. It is designed to explicitly identify disagreement and generate a resolved label. The consensus agent maintains a balanced, reflective, neutral persona (similar to the single-coding agent in the control condition), emphasizing justification and adherence to the codebook. The agent receives the data segment utterance and both prior agent responses for reference. The full MAS setup and prompts are available in the digital appendix [1].

4.3 Experimental Conditions

We conducted a 3×2 factorial experiment varying *decoding temperature* (0.0, 0.5, 1.0; ranging from low to high) and *persona pairing* (congruent vs. incongruent). Persona pairing was defined based on whether the two simulated agent personalities (*persona_a* and *persona_b*) matched (congruent) or differed (incongruent). The six pairings included three congruent pairs (balanced-balanced, bold-bold, empathetic-empathetic) and three incongruent pairs (bold-empathetic, balancedbold, balancedempathetic). The rationale for this design was to investigate whether multiperspectivity, operationalized as persona diversity, influences coding accuracy and consensus formation in LLMs. Each of the 18 configurations was run on a shared dataset of 579 utterances. Agents were instantiated with systematically varied personality traits drawn from three archetypes: *neutral*, *dominant*, and *permissive*. We chose these three traits as the politeness or assertiveness of prompts has been shown to influence LLM compliance [59] (see Section 3.2 for more detail).

Implementation Details. The system uses the Ollama API [37] to interface with local large language models (e.g., WizardLM 2 7B). Personalities and roles are injected via structured system prompts that emphasize justification and brevity to keep context windows manageable. Experiments are orchestrated in Python. A single experiment took about 2 hours on a standard MacBook Pro with an M1 CPU (except for DeepSeek R1, which was run on larger machinery). Running the MAS with the DeepSeek R1 model for a single data point on an NVIDIA A100 GPU took, on average, about 60 seconds. We computed the % **Agreement Rate** for each experiment, label, and condition, which measures the proportion of utterances in which the final system-assigned code matched the human-coded ground-truth label. The system stores outputs from all modules, including raw and expanded codes, intermediate discussion content, and final consensus decisions. The output is a structured CSV file used for analysis and comparison with human-coded data.

4.4 Data Analysis Methods for RQ1 and RQ2

To address RQ1, which examines the impact of persona and temperature on consensus-making, we analyzed how agent

persona pairings and decoding temperature influence the type of consensus achieved during dual-agent discussions. We categorized consensus outcomes into three types: (1) first consensus, where both agents agreed from the start; (2) resolved consensus, where initial disagreement was successfully resolved during discussion; and (3) no consensus, where agents failed to converge on a shared label and the consensus agent needed to enforce consensus. Our focus here lies in examining the likelihood of first consensus and no consensus across persona and temperature conditions.

We estimated generalized linear hierarchical mixed-effects modeling with a binomial link function using the lme4 package in R [6], predicting each outcome type as a function of agent persona congruency (i.e., matched or mismatched) and temperature (0.0, 0.5, or 1.0; see implementation details in Section 4.2), while controlling for random intercepts by text unit. This adjustment for the text unit is necessary because the same data segments are coded multiple times in our experiments and could systematically vary in difficulty, making it harder to reach consensus (e.g., because some texts are more ambiguous than others). We estimated three generalized linear mixed-effects models with a binomial link function to predict the likelihood of three consensus outcomes: initial, delayed, and no consensus, all following the same formula structure (but with different outcomes):

$$\begin{aligned}\text{logit}\left(\Pr(\text{Consensus}_i = 1)\right) = \\ \beta_0 + \beta_1 \cdot \text{Congruency}_i + \beta_2 \cdot \text{Temperature}_i + \\ \beta_3 \cdot (\text{Congruency}_i \times \text{Temperature}_i) + u_{\text{text}[i]}\end{aligned} \quad (1)$$

In both models, Congruency_i indicates whether the two agents in condition i had matched personality traits, and Temperature_i refers to the decoding randomness used by the model during generation. The interaction term captures how the effect of congruency varies across temperatures. The term $u_{\text{text}[i]}$ is a random intercept for each text unit, accounting for potential variation in the difficulty or ambiguity of individual utterances. We highlight that we separately repeated this procedure for each language model.

To evaluate whether MAS coding leads to greater alignment with human annotations than single-agent outputs (RQ2), we analyzed 77,273 individual coding decisions (treating assignment of each codebook category as a distinct binary decision). Each decision represented a binary classification across a fixed codebook, annotated by a human coder, a single-agent model, and a final consensus model. For each decision, we computed whether the model's label matched the human label, both for the baseline single-agent in the control condition and for the MAS's final output. We, again, organized experimental conditions along two dimensions: decoding *temperature* (0.0, 0.5, or 1.0) and *agent persona pairing congruency* (incongruent vs. congruent; see Section 3.2 for full details on persona pairing). We fit six models separately by LLM and fit a generalized linear mixed-effects model predicting whether the final consensus matched the human label, using temperature, persona group, and their interaction as fixed effects, and including random intercepts for each utterance and code category (since binary coding decisions are treated as separate observations in this model, as opposed to RQ1):

$$\begin{aligned}\text{logit}\left(\Pr(\text{FinalAgreement}_i = 1)\right) = \\ \beta_0 + \beta_1 \cdot \text{Temperature}_i + \beta_2 \cdot \text{Congruency}_i + \\ \beta_3 \cdot (\text{Temperature}_i \times \text{Congruency}_i) + \\ u_{\text{utterance}[i]} + u_{\text{code}[i]}\end{aligned} \quad (2)$$

As we did not find any significant and robust trends in how our experimentally manipulated dimensions impacted coding accuracy, we conducted an exploratory analysis. For this analysis, we recoded the persona factor into three levels (includes assertive agent, includes empathic agent, or both), ignoring the neutral-neutral pairing. We then tested model experiments individually (6 LLMs x 8 codes x 3 persona pairings x 3 temperatures = 432 contrasts). To evaluate whether final consensus codes are more closely aligned with human annotations than those produced by a single agent, we performed a paired samples t-test. This test aims to compare the percentage of correct code between the single agent and MAS across all contrasts. The Benjamini–Hochberg (BH) method was used to control the false discovery rate [7], given the large number of comparisons. We aimed to determine whether any condition, combination of LLMs, code, or experimental parameter would show reliable MAS improvements relative to the single-coder baseline, and whether any patterns emerged across conditions that produced MAS advantages in qualitative coding accuracy.

4.5 Qualitative Analysis Methods for RQ3

Our system design features the negotiation of codes through conversational turns, similar to how codes would evolve in a human conversation. Beyond this similarity, our system records all conversational turns, which could be considered recordings of code negotiation in the human analogy. Hence, the system offers a level of transparency not typically found in human coder teams, enabling analysis of the rationale behind each piece of code. We took advantage of this level of transparency to analyze *why* MAS configurations could achieve or fail to achieve the quality of human coding.

To understand the mechanisms behind coding shifts that can speak to patterns of successful MAS collaboration in our context (RQ3), we conducted an ad-hoc qualitative analysis of the *Guiding Feedback* category for OpenHermes, which showed statistically reliable improvements from single- to multi-agent coding, aligning more closely with human consensus. Among 9,735 total coded turns where no initial consensus was reached between the two agents, 71 cases involved *Guiding Feedback* being correctly removed during consensus (according to human ground-truth codes), and 79 cases involved it being correctly added. Two independent human coders analyzed these instances in a spreadsheet. The question guiding the coders' analysis was whether any notable patterns of the agents' consensus-making could explain the observed performance improvement. Coders analyzed: (a) how agents justified their coding decisions, (b) the internal consistency between rationale and codes of agents, (c) the external consistency between turns of agents, and (d) whether certain personalities tend to dominate or concede during discussions. Afterwards, they held a meeting to discuss their findings. In cases of disagreement, they reached consensus through discussion and organized such instances under shared themes, such as Hallucination.

5. RESULTS

Figure 1 shows the distribution of consensus outcomes across LLMs. Consensus frequencies substantially varied. DeepSeek-R1:32B reached no coding consensus in fewer than 0.1% of cases, whereas Llama:3b did not reach consensus in 21.7% of cases. Notably, for all LLMs, immediate consensus in the first round was the most common outcome.

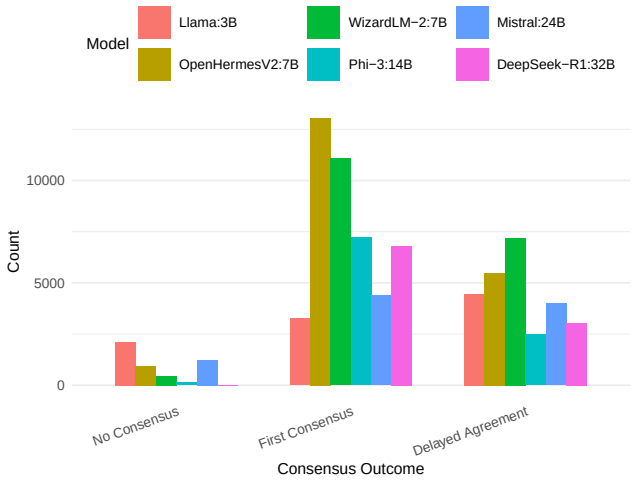


Figure 1: Distribution of consensus outcomes over total number of dialog segments and experiments (No Consensus, First Consensus, Delayed Agreement) across the six LLMs.

The distribution of codes, expressed as percent base rates from human coding, included: Aligning to prior knowledge (6.44%), Encouragement (5.58%), Greeting (5.15%), Guiding feedback (6.44%), Instruction (9.01%), Technical or Logistics (0.43%), Time Management (1.29%), and Understanding / Engagement–Tutor (26.2%).

To investigate how agent persona congruency and temperature influenced consensus-building, we fit a series of generalized linear mixed-effects models with random intercepts for each text. Models predicted three outcomes: (a) first round agreement, (b) no consensus at all, and (c) delayed consensus (agreement only after a second round). Tables in the digital appendix [1] report odds ratios (OR) and 95% confidence intervals (CI) for each predictor across the three language models. We summarize key findings below.

5.1 RQ1a: Consensus Making Effects

Immediate Consensus. Across all models, higher temperature significantly reduced the likelihood of reaching consensus in the first round ($p's < .001$), with odds ratios ranging from $OR = 0.07$ (OpenHermes-V2:7B) to $OR = 0.59$ (DeepSeek-R1:32B). Incongruent personas significantly reduced consensus likelihood in Llama:3B ($OR = 0.54$, $p < .001$), WizardLM-2:7B ($OR = 0.80$, $p = .010$), Phi-3:14B ($OR = 0.67$, $p < .001$), and Mistral-24B ($OR = 0.79$, $p = .002$), but not in OpenHermes-V2:7B or DeepSeek-R1:32B. Notably, Llama:3B ($OR = 1.96$, $p < .001$) and Phi-3:14B ($OR = 1.76$, $p < .001$) showed significant positive interactions between temperature and incongruency. *Full model results are reported in the digital appendix [1].*

No Consensus. Failure to reach consensus was consistently more likely under high temperature across all LLMs ($p's < .01$), with significantly increased odds in Llama:3B ($OR = 2.06$, $p < .001$), WizardLM-2:7B ($OR = 2.23$, $p = .003$), OpenHermes-V2:7B ($OR = 4.15$, $p < .001$), Phi-3:14B ($OR = 2.58$, $p = .003$), and Mistral-24B ($OR = 3.93$, $p < .001$). In contrast, congruency and its interaction with temperature did not significantly predict no-consensus outcomes in any model (p's $> .450$). DeepSeek-R1:32B was excluded from this analysis due to an insufficient number of non-consensus cases (less than 0.1%). *Full model results are reported in Appendix A in the digital appendix (see digital appendix [1]).*

Delayed Agreement. Higher temperature significantly increased the odds of delayed consensus across all models ($p's < .001$), with odds ratios ranging from $OR = 1.57$ (WizardLM-2:7B) to $OR = 10.28$ (OpenHermes-V2:7B). Incongruent personas were significantly associated with increased delayed agreement in Llama:3B ($OR = 1.69$, $p < .001$), WizardLM-2:7B ($OR = 1.24$, $p = .011$), Phi-3:14B ($OR = 1.52$, $p < .001$), and Mistral-24B ($OR = 1.21$, $p = .008$), but not in OpenHermes-V2:7B or DeepSeek-R1:32B (p's $> .328$). Significant negative interactions between congruency and temperature emerged in Llama:3B ($OR = 0.52$, $p < .001$), Phi-3:14B ($OR = 0.58$, $p < .001$), and Mistral-24B ($OR = 0.80$, $p = .040$), suggesting that when generation is more stochastic, the disruptive effects of persona mismatch on delayed agreement are attenuated in some LLMs. *Full model results are reported in the digital appendix [1]).*

Taken together, these findings show that **temperature robustly delays consensus across all LLMs**, while **persona congruency exerts more selective effects**. Interaction effects were similarly model-specific, but **generally showed that higher temperatures attenuate the effects of persona on consensus.** *See digital appendix for detailed model coefficients [1].*

5.2 RQ2: Human Alignment

Similarly to RQ1, we modeled the likelihood of final alignment with human judgments as a function of temperature, persona diversity (congruent vs. incongruent), and their interaction, using mixed-effects logistic regression with random intercepts for item and code (e.g., greeting vs. praise). Across all six models, persona congruency and its interaction with temperature showed no significant effects on final alignment ($p's > .149$). Temperature showed a small but significant negative association with alignment in OpenHermes-V2:7B ($OR = 0.88$, $p = .003$) and Mistral:24B ($OR = 0.85$, $p < .001$), but not in other models ($p's > .158$). Overall, final alignment with human judgments remained largely unaffected by the persona pairing or temperature manipulation, and no single parameter produced reliable improvements in coding accuracy. *See Appendix B for detailed model coefficients in the digital appendix [1].*

To further explain these findings, Table 1 summarizes the overall percentage of agreement (i.e., alignment with human coders) for each model under both single-agent and multi-agent (final consensus) coding conditions, including 95% confidence intervals.

Table 1: Percentage agreement (aligned responses) by model with 95% binomial confidence intervals.

Model	Final Consensus (%)	Single Agent (%)
DeepSeek-R1:32b	90.0 [89.8, 90.3]	89.9 [89.6, 90.1]
Llama:3B	81.6 [81.3, 81.8]	85.5 [85.2, 85.7]
Mistral:24B	89.3 [89.1, 89.6]	89.7 [89.4, 89.9]
OpenHermes-V2:7B	84.9 [84.7, 85.2]	85.8 [85.5, 86.0]
Phi-3 14B	89.5 [89.3, 89.8]	89.6 [89.4, 89.8]
WizardLM 2:7B	85.6 [85.4, 85.9]	87.2 [87.0, 87.5]

Surprisingly, following Table 1, across all six models, multi-agent coding (i.e., extended deliberation among agents) did not yield substantially better alignment than single-agent coding. In all but one model (DeepSeek-R1:32B), it reduced accuracy, averaged across all experiments. To further examine code-specific effects, we compared alignment between the two coding conditions across different code categories. As shown in Figure 2, some models outperformed others on particular codes, but the pattern of improvement across codes was not consistent across models. Further, the magnitude of performance changes (MAS vs. single agent) also differs across models (notice the difference in y-axis scaling in Figure 2 to improve plot readability). For instance, OpenHermes-V2:7B showed improvements of up to 3% but also performance reductions of about 7% for the MAS on another code. In contrast, DeepSeek-R1:32B did not show average performance changes greater than 0.5%. Beyond OpenHermesV2:7B, no model showed an average performance increase of more than 1%.

The heterogeneity observed in alignment differences raised two questions. First, do comparative differences between multi-agent and single-agent coding vary systematically by item category, temperature, or assigned persona? Second, are the observed differences statistically reliable, or are they false positives due to the large number of comparisons? To address this as part of an exploratory analysis, we applied a modular paired-sample comparison procedure using the Benjamini–Hochberg (BH) method to control the false discovery rate [7]. Specifically, we ran paired t-tests on agreement scores across coding conditions (multi-agent vs. single-agent), grouped by model, persona, temperature, and code category. We adjusted raw p-values using the BH correction across all 432 comparisons (see Section 4.4 for more details).

There were 70 statistically significant contrasts after FDR correction. Of these, a minority (5) showed higher agreement in the multi-agent condition, while 65 showed higher agreement in the single-agent condition. All five positive and reliable improvements were on the code "Guiding Feedback" with the OpenHermesV2:7B model. In other words, we have no evidence that any other positive gains in Figure 2 were due to systematic improvements as opposed to random chance. All of those five included temperatures of 0.5 or lower, and four of those five included bold personalities. Therefore, we conclude that MAS improvements in our dataset were only reliable for a specific model and code. We have no further evidence that consensus-making improves the accuracy of qualitative deductive coding with LLMs.

5.3 RQ3: Qualitative Insights

As detailed in the previous section, the OpenHermesV2-7B model showed significantly better alignment with human coders in its multi-agent configuration for the "Guiding Feedback" code. Aligned with RQ3 ("What qualitative patterns of successful consensus making arise in multi-agent thematic coding?"), We ran an investigation into the process of 150 coding conversations for the MAS on temperature and persona settings, showing statistically reliable MAS improvements (see Section 5.2). These coding conversations were characterized by either removing the "Guiding Feedback" during the conversation, aligning with human coding, or adding the "Guiding Feedback" during the conversation, again aligning with the human coding at the end of the MAS conversation. Given that all turns were recorded, we could investigate these successful turns in an exploratory analysis. Two coders each coded 75 conversations in an open coding process, for a total of 900 conversational turns. Notably, we observed that the LLM provided a rationale in only about 80 of 900 conversational turns. In other cases, the LLM explicitly stated that it had no explanation, referenced only a specific data point, or provided no explanation at all. For guiding feedback, the LLM rationale was rare, too, accounting for only 5 of the 80 rationales provided by the agents. Nevertheless, we were able to make several notable qualitative observations about model behavior for this specific subset of the data.

5.3.1 Consistency

Beyond the LLM's difficulty in providing a rationale in every turn as instructed, we also observed issues within the reported codes. For instance, agents sometimes incorrectly reported previous coding when introducing it with the phrase "the previous turn said," even though the previous turn's codes did not align with those statements. Other notable conversational turns involved the complete output of the provided codebook without further explication. Rare hallucinations concerned data points. In these cases, two agents discussed non-existent data points.

5.3.2 New Codes

In four conversations, the agents proposed new categories. In one example, an agent introduced the new code "Understanding / Engagement-Tutor" to their output, explicitly stating "The tutor is providing instruction [in this datapoint] on the next step in the math problem while also addressing potential confusion around variables or notation. However, I would like to add subcode under "Understanding/Engagement-Tutor" as the tutor is checking for understanding by asking, 'What should we do next?"'.

5.3.3 Hallucinated Datapoints

By design, the system analyzes only one data point from the transcript at a time. This means that agents did not have visibility into other codes. Agents sometimes would nevertheless invent and then discuss such data beyond the provided data point. In one instance, in a discussion about the data point "We found them," the second agent mentions in its reasoning that "the tutor asks if the student remembers what a factor is", which is not what the data point contains.

6. DISCUSSION

This study examined the behavior and performance of a multi-agent LLM system for deductive thematic analysis in

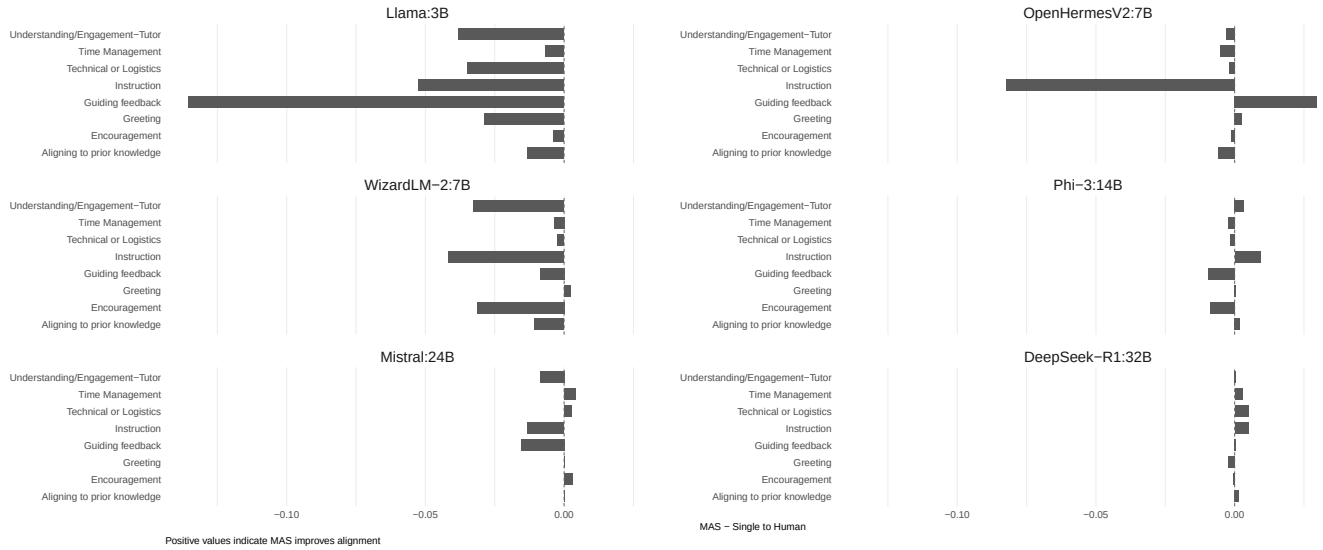


Figure 2: Mean difference in alignment between the MAS and single LLM coding with the human labels. Positive values indicate better MAS performance than single LLM coding.

an educational setting. Building on prior work on multi-agent qualitative coding, we introduced persona-infused agents. We systematically varied temperature to assess their effects on consensus formation and alignment with human-coded data. The resulting open-source system closely aligns with established qualitative coding practices [63, 4, 9].

Although existing studies highlight the potential of multi-agent architectures to emulate human collaboration [42, 64, 44], to our knowledge, no prior work has experimentally evaluated the roles of agent personality, stochasticity, and code-level variation on real-world educational datasets. Beyond empirical evaluation, we considered the implications of these factors for the viability of LLM-based thematic analysis and human–AI collaboration, informed by a qualitative inspection of agent deliberation traces (i.e., justifications for assigning a specific code) to identify conditions under which multi-agent interaction succeeds or breaks down.

6.1 RQ1: Consensus Building

RQ1 results showed that decoding temperature had a consistent, statistically significant effect across all models: higher temperatures reduced the likelihood of immediate consensus and increased the likelihood of delayed agreement and no-consensus outcomes. This aligns with prior findings in LLM generation behavior showing that temperature amplifies response diversity and unpredictability [14, 36].

We predicted that multiple personalities would lead to a more comprehensive consideration of data during coding. The influence of persona congruency, however, was model-specific. While mismatched (i.e., incongruent) personalities reduced immediate consensus across half of the LLMs we investigated, and delayed consensus in four out of six LLMs. These findings suggest that persona-driven heterogeneity may interact differently with LLM architectures. This finding echoes recent studies arguing that the inclusion of personas to induce multiperspectivity in LLMs is a path full of obstacles [29]. In particular, findings highlight that the specific relationships between personas and model behavior remain elusive: they may work in some contexts but not others. Therefore, while they might help in surfacing more diverse model behavior in some cases, we have no evidence that they can be systematically and consistently exploited across settings.

6.2 RQ2: Human Alignment

RQ2 assessed whether multi-agent deliberation improves alignment with human-coded ground truth. Overall, multi-agent systems did not consistently outperform single-agent models. Of 432 condition-level comparisons, only five showed statistically significant improvements. In most cases, consensus coding matched or underperformed single-agent performance. These findings align with prior work [30], which similarly reported higher agreement for a single-agent few-shot approach (82.8%), compared to a multi-agent setup (79.5%). A practical significance of this finding is that the MAS requires substantially more computational resources (see Section 4.3), thereby being substantially less resource-efficient for qualitative coding in our context.

This suggests that extended deliberation among same-model agents does not necessarily yield better accuracy, particularly when the codebook is already well-constructed and disambiguated, as was the case here using a codebook from [4]. This finding challenges the assumption that multi-agent deliberation inherently yields greater interpretive robustness and depth, especially when the underlying language model remains constant and deliberative rounds amount to interacting with variants of the same probabilistic system. Despite strong convergent evidence across several experiments and LLMs showing little tangible accuracy improvement, it is possible that better gains for MAS collaboration could be seen in cases where codebooks are less well-developed or when multiple LLMs collaborate. For instance, a resource-efficient approach could be to use smaller models for initial coding, and larger models only when no consensus is reached.

Such setups, however, are beyond the scope of the present study. Moreover, it may be that the task and the breadth of coding (i.e., assigning multiple codes per turn) are simply too complex for the LLMs studied here [27].

6.3 RQ3: Code-Level Variability and Limits

RQ3 sought to understand where and why MAS may provide additional value in thematic analysis, and what patterns of successful consensus-making emerged. Our observations reveal that, despite the potential of automated systems to assist with qualitative coding, human oversight remains essential for several reasons. First, it could correct technical and formatting inconsistencies, such as invalid code values or schema violations, that undermine data integrity (and can sometimes only be partially mitigated through automated means). In our study, we observed that the only model that reliably improved its own coding in a MAS setup, provided a rationale for coding a data segment in a way that was inconsistent with the instruction in less than 10% of cases. Second, human coders are needed to identify and rectify logical contradictions, hallucinated justifications, or unsupported code assignments [45], all of which compromise the quality of automated annotations. This also echoes findings by [27], who demonstrate that LLM performance often degrades in multi-turn conversations, showing increased unreliability and an inability to reproduce earlier instructions compared to fully specified single-turn instructions. While we re-prompted agents up to once when their formatting was incorrect to mitigate this issue, it is possible that performance losses would be smaller if reprompting were performed more often. However, such reprompting also increases computation time and cost.

Further, our analysis revealed a limited boundary condition under which multi-agent consensus provided its clearest benefit: cases of code ambiguity. Specifically, in the "Guiding Feedback" category, consensus agents helped refine overly broad applications of the code and, in some instances, introduced justified reassignments based on context-sensitive deliberation. However, even in these favorable instances, our qualitative inspection revealed significant transparency issues: agent dialogues lacked consistent internal logic, veered erratically, and often failed to provide reasoning that a human could follow or trust. For example, some decisions appeared arbitrary, with agents deferring to previous turns without clear justification or revising positions without explanation. Similar to how recent research has argued that LLM "reasoning" is merely an artifact of intermediate token prediction [23], so could agentic collaboration for coding be more akin to improving self-consistency via joint prediction [31] rather than a collaborative sensemaking process.

Notably, we observed that coding agents in our experiments sometimes independently proposed new codes, highlighting that consensus mechanisms could benefit human-AI hybrid coding teams [18] in different ways. Rather than maximizing coding accuracy, these systems may be better suited to enhance coding accuracy, codebook development, and challenging existing codes where human coding teams may benefit from new interpretations of data. Our findings thus echo broader calls in the field to retain human expertise in the applications of LLMs for applying expert-level judgment and grading [46, 18].

6.4 Implications for Human-AI Coding

While our system reproduces procedural elements of human workflows (multiple coders, deliberative dialogue, consensus arbitration), it may not reproduce the epistemic stance, contextual grounding, or reflexivity that characterize interpretive qualitative work [11]. Indeed, our qualitative analysis for RQ3 indicated that agents would often overwrite or change their codes without offering explanations (including, at times, in a contradictory manner), even though they were instructed to provide reasoning to the other agent. This behavior highlights that deductive coding with a codebook is a meaning-making process grounded in the researcher's perspective and iterative engagement with data. Our results suggest that MAS cannot reproduce these epistemic dimensions; nor are they designed to. Despite being framed as reasoning systems, they function as complex retrieval and prediction models that infer the most likely next token as argued by [23].

Findings in this study suggest that the value of LLMs in TA may lie less in reproducing consensus than in disrupting it by introducing "alternative perspectives." Prior work shows that combining human and synthetic data can enrich training outcomes, such as in open-response assessment [8]. Our findings similarly support calls for hybrid systems that pair human insight with machine generativity, enhancing qualitative inquiry rather than fully automating interpretive labor [24, 4]. Beyond the manual inspection of agents' consensus processes by humans, data mining methods could also help curate relevant moments in which LLMs question or struggle to disambiguate individual data segments or codes, helping surface analytically ambiguous cases for focused human review [52]. Furthermore, detailed analysis of agents' deliberation traces might give better insights into the coding process [30]. In addition, research has shown that LLM outputs can vary significantly depending on how prompts are phrased (e.g., [20]), opening yet another perspective for future work.

6.5 Limitations and Future Work

Several limitations contextualize these conclusions. First, we relied on a single dataset and a highly refined codebook, previously validated through hybrid human–AI processes [4]. The strong performance of the single-agent condition may reflect the maturity and clarity of this codebook as well as characteristics of the coding task. Future work could analyze the impact of codebook size, along with parameters such as the type of training dataset and domain. Second, the system uses multiple instances of the same base model, prompted differently. While personalities were carefully engineered and experimentally compared (as well as theoretically justified; see Section 3.2), the agents were instantiated solely through prompt-level instructions and all rely on the same underlying architecture. Although some research has demonstrated that LLMs tend to prefer the output generated by their own [40], it remains an open question whether multi-agent diversity derived from different model families (e.g., mixing Claude, GPT-4, LLaMA), as well as more extended and direct interactions between agents, would lead to even richer or more robust consensus processes. Third, while we focused on deductive coding tasks, future studies might examine inductive coding or mixed-methods settings where codebook refinement and creation are explicitly part of the analytic process [4].Finally, given that DeepSeek-R1, the most reasoning-intensive model

in our study, did not yield accuracy gains over single-agent coding, future work could examine whether the deliberation traces produced by such models offer utility beyond accuracy, particularly for codebook refinement [52].

7. CONCLUSION

As educational technologies generate increasingly rich textual data, such as chat and audio transcripts, methods for annotation require careful validation. Large language models have drawn attention for their potential to automate qualitative coding at scale. This study examines how multi-agent configurations of LLMs affect deductive coding processes and outcomes. Although multi-agent systems are often assumed to improve reliability and interpretive depth, our results challenge this assumption. Across experiments with six open-source LLMs, multiple persona pairings, and temperature settings, we find that multi-agent structures strongly shape consensus dynamics, influencing when agents converge or diverge, with persona effects varying across models. However, these changes yielded minimal and highly conditional accuracy gains.

Multi-agent deliberation only marginally outperforms single-agent predictions, and only under narrow conditions involving a specific model, code category, low temperature, and an assertive agent. This calls into question the reliability of agent collaboration as a general-purpose method for improving accuracy or diversity in deductive qualitative coding. Instead, our qualitative analyses suggest that MAS are better suited to exposing ambiguity, prompting codebook revision, and surfacing interpretations that human-only teams might miss. By showing when multi-agent systems do not improve coding, this work helps constrain overgeneralized claims about agentic LLM collaboration.

8. REFERENCES

[1] Supplementary Repository for: Temperature and Persona Shape LLM Agent Consensus With Minimal Accuracy Gains in Qualitative Coding. `https://github.com/conradborchers/llm-ta-consensus/`, 2026.

[2] M. Abdin, J. Aneja, H. Awadalla, A. Awadallah, A. A. Awan, N. Bach, A. Bahree, A. Bakhtiari, J. Bao, H. Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[3] Y. Asano, S. Sankaranarayanan, C. Bogart, et al. A thematic summarization dashboard for navigating student reflections at scale. In *International Conference on Computers in Education*, 2021.

[4] A. Barany, N. Nasiar, C. Porter, A. F. Zambrano, A. L. Andres, D. Bright, M. Shah, X. Liu, S. Gao, J. Zhang, et al. Chatgpt for education research: exploring the potential of large language models for qualitative codebook development. In *International conference on artificial intelligence in education*, pages 134–149. Springer, 2024.

[5] B. Barry and G. L. Stewart. Composition, process, and performance in self-managed groups: the role of personality. *Journal of Applied psychology*, 82(1):62, 1997.

[6] D. Bates, M. Mächler, B. Bolker, and S. Walker. Fitting linear mixed-effects models using lme4. *Journal of statistical software*, 67:1–48, 2015.

[7] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[8] C. Borchers, D. R. Thomas, J. Lin, R. Abboud, and K. R. Koedinger. Augmenting human-annotated training data with large language model generation and distillation in open-response assessment. *arXiv preprint arXiv:2501.09126*, 2025.

[9] C. Borchers, J. Zhang, R. S. Baker, and V. Aleven. Using think-aloud data to understand relations between self-regulation cycle characteristics and student performance in intelligent tutoring systems. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 529–539, 2024.

[10] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[11] V. Braun and V. Clarke. One size fits all? what counts as quality practice in (reflexive) thematic analysis? *Qualitative research in psychology*, 18(3):328–352, 2021.

[12] C. Chandler, T. Breideband, J. G. Reitman, M. Chitwood, J. B. Bush, A. Howard, S. Leonhart, P. W. Foltz, W. R. Penuel, and S. K. D'Mello. Computational modeling of collaborative discourse to enable feedback and reflection in middle school classrooms. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 576–586, 2024.

[13] G. Chen, C. K. Chan, K. K. Chan, S. N. Clarke, and L. B. Resnick. Efficacy of video-based teacher professional development for increasing classroom discourse and student learning. *Journal of the Learning Sciences*, 29(4-5):642–680, 2020.

[14] H. Chen, W. Ji, L. Xu, and S. Zhao. Multi-agent consensus seeking via large language models. *arXiv preprint arXiv:2310.20151*, 2023.

[15] K. K. C. Cheung and K. W. Tai. The use of intercoder reliability in qualitative interview data analysis in science education. *Research in Science & Technological Education*, 41(3):1155–1175, 2023.

[16] A. Chittem, A. Shrivastava, S. T. Pendela, J. S. Challa, and D. Kumar. Sac: A framework for measuring and inducing personality traits in llms with dynamic intensity control. *arXiv preprint arXiv:2506.20993*, 2025.

[17] S. De Paoli. Performing an inductive thematic analysis of semi-structured interviews with a large language model: An exploration and provocation on the limits of the approach. *Social Science Computer Review*, 42(4):997–1019, 2024.

[18] Y. Gao, Y. Chen, M. Wang, J. Wu, Y. Kim, K. Zhou, M. Li, X. Liu, X. Fu, J. Wu, et al. Optimising the paradigms of human ai collaborative clinical coding. *npj Digital Medicine*, 7(1):368, 2024.

[19] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[20] A. Gupta, X. Song, and G. Anumanchipalli. Self-assessment tests are unreliable measures of llm personality. *arXiv preprint arXiv:2309.08163*, 2023.

[21] A. H. Hilal and S. S. Alabri. Using nvivo for data analysis in qualitative research. *International interdisciplinary journal of education*, 2(2):181–186, 2013.

[22] H. Jiang, X. Zhang, X. Cao, C. Breazeal, D. Roy, and J. Kabbara. Personallm: Investigating the ability of large language models to express personality traits. *arXiv preprint arXiv:2305.02547*, 2023.

[23] S. Kambhampati, K. Stechly, K. Valmeekam, L. Saldyt, S. Bhambri, V. Palod, A. Gundawar, S. R. Samineni, D. Kalwar, and U. Biswas. Stop anthropomorphizing intermediate tokens as reasoning/thinking traces! *arXiv preprint arXiv:2504.09762*, 2025.

[24] A. H. Khan, H. Kegalle, R. D'Silva, N. Watt, D. Whelan-Shamy, L. Ghahremanlou, and L. Magee. Automating thematic analysis: How llms analyse controversial topics. *arXiv preprint arXiv:2405.06919*, 2024.

[25] U. Kuckartz and S. Rädiker. *Analyzing qualitative data with MAXQDA*. Springer, 2019.

[26] L. La Cava and A. Tagarelli. Open models, closed minds? on agents capabilities in mimicking human personalities through open large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1355–1363, 2025.

[27] P. Laban, H. Hayashi, Y. Zhou, and J. Neville. Llms get lost in multi-turn conversation. *arXiv preprint arXiv:2505.06120*, 2025.

[28] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[29] A. Li, H. Chen, H. Namkoong, and T. Peng. Llm generated persona is a promise with a catch. *arXiv preprint arXiv:2503.16527*, 2025.

[30] G. Li, L. Chen, C. Tang, V. Švábenskỳ, D. Deguchi, T. Yamashita, and A. Shimada. Single-agent vs. multi-agent llm strategies for automated student reflection assessment. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 300–311. Springer, 2025.

[31] X. Liang, S. Song, Z. Zheng, H. Wang, Q. Yu, X. Li, R.-H. Li, Y. Wang, Z. Wang, F. Xiong, et al. Internal consistency and self-feedback in large language models: A survey. *arXiv preprint arXiv:2407.14507*, 2024.

[32] L. Liu. Using generic inductive approach in qualitative educational research: A case study analysis. *Journal of Education and Learning*, 5(2):129–135, 2016.

[33] X. Liu, A. F. Zambrano, R. S. Baker, A. Barany, J. Ocumpaugh, J. Zhang, M. Pankiewicz, N. Nasiar, and Z. Wei. Qualitative coding with gpt-4: Where it works better. *Journal of Learning Analytics*, pages 1–17, 2025.

[34] Mistral-AI. Mistral-small-3.2-24b-instruct-2506, 2024. [Mistral-Small] Hugging Face.

[35] M. K. Mount, M. R. Barrick, and G. L. Stewart. Five-factor model of personality and performance in jobs involving interpersonal interactions. *Human performance*, 11(2-3):145–165, 1998.

[36] L. Newsham and D. Prince. Personality-driven decision-making in llm-based autonomous agents. *arXiv preprint arXiv:2504.00727*, 2025.

[37] Ollama. Ollama: Run large language models locally. `https://ollama.com`, 2023. Accessed: 2025-07-09.

[38] C. O'Connor and H. Joffe. Intercoder reliability in qualitative research: debates and practical guidelines. *International journal of qualitative methods*, 19:1609406919899220, 2020.

[39] K. Pan and Y. Zeng. Do llms possess a personality? making the mbti test an amazing evaluation for large language models. *arXiv preprint arXiv:2307.16180*, 2023.

[40] A. Panickssery, S. Bowman, and S. Feng. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802, 2024.

[41] Z. A. Pardos, L. F. Rosenbaum, and D. Abrahamson. Characterizing learner behavior from touchscreen data. *International Journal of Child-Computer Interaction*, 31:100357, 2022.

[42] T. Qiao, C. Walker, C. W. Cunningham, and Y. S. Koh. Thematic-lm: a llm-based multi-agent system for large-scale thematic analysis. In *Proceedings of the ACM on Web Conference 2025*, pages 649–658, 2025.

[43] S. Ramanathan, L.-A. Lim, N. R. Mottaghi, and S. Buckingham Shum. When the prompt becomes the codebook: Grounded prompt engineering (groproe) and its application to belonging analytics. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 713–725, 2025.

[44] Z. Rasheed, M. Waseem, A. Ahmad, K.-K. Kemell, W. Xiaofeng, A. N. Duc, and P. Abrahamsson. Can large language models serve as data analysts? a multi-agent assisted approach for qualitative data analysis. *arXiv preprint arXiv:2402.01386*, 2024.

[45] V. Rawte, A. Sheth, and A. Das. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*, 2023.

[46] M. Reza, I. Anastasopoulos, S. Bhandari, and Z. A. Pardos. Prompthive: Bringing subject matter experts back to the forefront with collaborative prompt engineering for educational content creation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–22, 2025.

[47] K. Richards and M. Hemphill. A practical guide to collaborative qualitative data analysis. *Journal of Teaching in Physical Education*, 37:1–20, 2017.

[48] S. Sankaranarayanan, C. Borchers, S. Simon, E. Tajik, A. H. Ataş, B. Celik, F. Balzan, and B. Shahrokhian. Automating thematic analysis with multi-agent llm systems. *EdArXiv Preprints (https://doi.org/10.35542/osf.io/kq8zh_v1)*, 2025.

[49] G. Serapio-García, M. Safdari, C. Crepy, L. Sun, S. Fitz, P. Romero, M. Abdulhai, A. Faust, and M. Matarić. Personality Traits in Large Language Models. *arXiv e-prints*, page arXiv:2307.00184, June 2023.

[50] S. Simon, S. Sankaranarayanan, E. Tajik, C. Borchers, s. Bahar, F. Balzan, S. Strauß, S. Viswanathan, A. Ataş, M. Čarapina, L. Liang, and B. Celik. Comparing human and llm-generated inductive

thematic analyses: Assessing agreement in coding consistency and interpretative accuracy. *Proceedings of 26th International Conference on Artificial Intelligence in Education (AIED)*, 2025.

[51] B. Smit. Atlas. ti for qualitative data analysis. *Perspectives in education*, 20(3):65–75, 2002.

[52] E. Tajik, B. Shahrokhian, C. Borchers, S. Simon, A. Keramati, S. Pal, and S. Sankaranarayanan. Disagreement as data: Reasoning trace analytics in multi-agent systems. In *Proceedings of the 16th International Learning Analytics and Knowledge Conference (LAK)*, Bergen, Norway, in press. ACM.

[53] R. Takata, A. Masumori, and T. Ikegami. Spontaneous emergence of agent individuality through social interactions in llm-based communities. *arXiv preprint arXiv:2411.03252*, 2024.

[54] Teknium. Openhermes-2-mistral-7b, 2023. [Openhermes2-7B] Hugging Face.

[55] D. R. Thomas, J. Lin, E. Gatz, A. Gurung, S. Gupta, K. Norberg, S. E. Fancsali, V. Aleven, L. Branstetter, E. Brunskill, et al. Improving student learning with hybrid human-ai tutoring: A three-study quasi-experimental investigation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 404–415, 2024.

[56] T. Tommaso, M. Hegazy, D. Lemay, M. Abukalam, I. Rish, and G. Dumas. Llms and personalities: Inconsistencies across scales. In *NeurIPS 2024 Workshop on Behavioral Machine Learning*, 2024.

[57] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[58] K.-T. Tran, D. Dao, M.-D. Nguyen, Q.-V. Pham, B. O'Sullivan, and H. D. Nguyen. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint arXiv:2501.06322*, 2025.

[59] R. Vinay, G. Spitale, N. Biller-Andorno, and F. Germani. Emotional prompting amplifies disinformation generation in ai large language models. *Frontiers in Artificial Intelligence*, 8:1543603, 2025.

[60] K. Wang, G. Zhang, M. Ye, X. Deng, D. Wang, X. Hu, J. Guo, Y. Liu, and Y. Guo. Mas2: Self-generative, self-configuring, self-rectifying multi-agent systems. *arXiv preprint arXiv:2509.24323*, 2025.

[61] D. Weitekamp, M. N. Siddiqui, and C. J. MacLellan. Tutorgym: A testbed for evaluating ai agents as tutors and students. In *International Conference on Artificial Intelligence in Education*, pages 361–376. Springer, 2025.

[62] C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, C. Tao, and D. Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*, 2023.

[63] A. F. Zambrano, X. Liu, A. Barany, R. S. Baker, J. Kim, and N. Nasiar. From ncoder to chatgpt: From automated coding to refining human coding. In *International conference on quantitative ethnography*, pages 470–485. Springer, 2023.

[64] J. Zhang, X. Xu, N. Zhang, R. Liu, B. Hooi, and S. Deng. Exploring collaboration mechanisms for llm agents: A social psychology view. *arXiv preprint arXiv:2310.02124*, 2023.

[65] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

The Cognitive Cost of Poor Item Design: How Item-Writing Flaws Shape Response Time and Difficulty Across Assessment Domains

Steven Moore
Information Sciences and Technology
George Mason University
StevenJamesMoore@gmail.com

ABSTRACT

Item-writing flaws (IWFs), violations of established guidelines for constructing multiple-choice questions, are known to compromise assessment quality. Their effects on item difficulty have proved inconsistent, with flawed items sometimes appearing easier, sometimes harder, and often no different from well-constructed items. We propose that this reflects a measurement problem, as different IWFs impose distinct cognitive demands that cancel or dilute when aggregated. Drawing on cognitive load theory, we classify 19 automatically detected IWFs into three categories of cue-giving, confusion-inducing, and structural complexity based on their expected cognitive function. Analyzing 1,230 items across three domains (medical licensure, K-12 mathematics, and university computer science), we find that each domain exhibits a different mix of IWFs rather than simply more or fewer flaws. Flaws categorized as confusion-inducing consistently increase response time by 14-29% across all three datasets, while cue-giving and structural flaws show no significant effects. This pattern holds across all three datasets, with no significant flaw-by-dataset interactions, suggesting the effect is robust to the differences in subject matter. Bootstrap mediation analyses reveal that confusion-inducing flaws increase processing time, which in turn associates with greater difficulty. This indirect effect is 2.5 times the magnitude of the aggregate indirect effect. These findings suggest that item quality improvement efforts should prioritize confusion-inducing flaws, which are the primary source of unnecessary cognitive burden on examinees.

Keywords

Assessment Quality, Response Time, Question Difficulty

1. INTRODUCTION

Multiple-choice questions (MCQs) remain the most widely used assessment format in education, certification, and professional licensure, where their quality directly affects what we can validly infer about the knowledge and skills of students [58]. With generative AI now accelerating item writing, quality is both more necessary and more possible [4]. Clear standards govern how to create high quality questions, but violations of those standards, item-writing flaws (IWFs), continue to appear even in commercial e-learning instruction [9, 36]. This compromises construct validity, fairness, and the interpretability of learning analytics derived from these assessments [12, 56]. IWFs impose unnecessary cognitive load as students interpret unclear wording rather than demonstrate knowledge, artificially inflating time-on-task through extraneous processing [44, 26, 53]. Not all flaws are expected to impose the same kind of cognitive load, some may even reduce it by cueing the answer, while others introduce confusion that slows examinees down.

When researchers have tested whether IWFs actually affect item performance, the results have been equivocal. Studies examining whether IWFs predict item difficulty or discrimination have reported weak, inconsistent, or null associations [39, 41, 46, 49]. This matters practically as if flaws do not measurably affect how items perform, it becomes hard to justify the cost of detecting and remediating them, and automated detection tools lose their rationale. The problem may not be that flaws are inconsequential but that the standard approach of counting them, treating all 19 criteria as interchangeable contributors to a single "flaw burden" score, masks the signal. Consider that some flaws cue the correct answer, potentially making items faster and easier, while others introduce confusion that slows examinees down and makes items harder. Aggregating these into one count pits opposing effects against each other, producing the attenuated relationships that prior work has observed. The same logic applies to response time, because if cue-giving flaws speed responses and confusion-inducing flaws slow them, a composite IWF count should show a diluted association with processing time, even though specific flaw types carry real cognitive cost. To better guide revision efforts, what is needed is a categorization that separates flaws by what they do to the examinee's response process, so that the distinct (and potentially opposing) mechanisms can be estimated rather than canceled.

We test this hypothesis by categorizing 19 automatically detected item-writing flaws into three cognitive-function categories of cue-giving, confusion-inducing, and structural complexity, and then testing whether this categorization recovers

Steven Moore. The Cognitive Cost of Poor Item Design: How Item-Writing Flaws Shape Response Time and Difficulty Across Assessment Domains. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 293–305. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040093

flaw-outcome relationships that are obscured in aggregate analyses. We analyze 1,230 MCQs bridging the three assessment domains of medical licensure (USMLE, n=653), K-12 mathematics (XES3G5M, n=365), and university computer science (DBE-KT22, n= 212). Each dataset provides item-level mean response time and a difficulty index, allowing us to trace the pathway from flaw presence through processing time to operational difficulty. The cross-domain design tests whether any observed effects reflect a general cognitive mechanism or are specific to a particular testing context. We investigate three research questions:

RQ1: How do item-writing flaw profiles differ across assessment domains?

RQ2: Do the effects of item-writing flaws on response time differ by cognitive-function category and are these effects consistent across domains?

RQ3: Does response time mediate the relationship between item-writing flaws and item difficulty and is this mediation strengthened when flaws are categorized by cognitive-function rather than treated in aggregate?

This study contributes (1) a cognitive-function taxonomy that classifies 19 IWFs into cue-giving, confusion-inducing, and structural complexity categories, revealing that only confusion-inducing flaws impose measurable cognitive cost on examinees, (2) evidence that this confusion effect replicates across three heterogeneous assessment contexts, increasing response time by 14–29% across medical, mathematics, and computer science assessments with no significant domain interactions, and (3) an explanation for why prior aggregate analyses found weak flaw–difficulty relationships. When flaws are categorized, the indirect effect of confusion-inducing flaws through slower response times to higher difficulty is 2.5× the magnitude of the aggregate effect, showing that lumping flaw types together masks the strongest signal.

2. RELATED WORK

2.1 Item-Writing Flaws

Item-writing flaws (IWFs) are features of multiple-choice questions that introduce testwiseness cues, compromise validity, and create unfair advantages for test-takers who recognize these patterns regardless of their actual knowledge [20, 56]. The fundamental concern is that IWFs shift what is being measured from domain knowledge to test-taking skill, introducing construct-irrelevant variance that distorts inferences about examinee ability [12]. Established taxonomies typically use a 19-criterion rubric applicable across subject areas [55, 9]. These capture grammatical or length cues that signal the correct answer, option-set pathologies such as K-type structures, and clarity violations including unfocused or vague question stems [36]. Despite awareness of these standards, IWFs remain prevalent even in professionally developed assessments, with studies finding 36–65% of items flawed in medical education [13, 8, 5] and similar rates in nursing [55], e-learning [9], and STEM disciplines [46, 49]. This persistence has motivated automated detection approaches, including rule-based systems [16] and hybrid systems like the Scalable Automatic Question Usability Evaluation Toolkit (SAQUET) [35]. This system operationalizes the full rubric and has achieved strong agreement with expert raters across diverse domains.

The empirical case that IWFs actually affect item performance is surprisingly weak. Studies in medical education found no consistent relationship between specific flaw types and difficulty or discrimination [41], with similar null results in veterinary [46] and broader STEM assessments [49]. One exception found that flawed items were easier in nursing assessments, but only for specific IWFs rather than the overall count [56]. If IWFs do not measurably affect how items perform, it becomes difficult to justify the cost of detecting and remediating them. However, the problem lies not with the flaws themselves but with how they have been measured.

2.2 The Aggregation Problem

Prior work treats all 19 IWF criteria as interchangeable contributors to a single score, but the rubric encompasses flaws whose cognitive mechanisms theory predicts should operate in opposite directions [8]. Cognitive load theory (CLT) distinguishes intrinsic load, inherent to the assessed content, from extraneous load imposed by how material is presented [10, 44]. In a well-constructed MCQ, the dominant load should be intrinsic as examinees expend effort retrieving and applying domain knowledge. IWFs distort this balance, but the direction of distortion depends on what the flaw does to the response process. Downing [12] drew a parallel distinction at the score level, noting that flawed items produce either construct-irrelevant easiness, which inflates scores, or construct-irrelevant difficulty, which deflates them. For instance, a word repeating between the MCQ's text and answer provides a lexical shortcut that lets test-wise examinees bypass retrieval (construct-irrelevant easiness) [20, 56]. In contrast, an ambiguous stem or gratuitous information forces examinees to infer what is being asked before retrieval can begin (construct-irrelevant difficulty), adding extraneous load [44] on features unrelated to the assessed content. These two classes of flaws should push response time and difficulty in opposite directions.

Aggregating them into a single count pits these opposing effects against each other. An item with one cue-giving flaw and one confusion-inducing flaw registers the same count as an item with two flaws of the same type, despite imposing very different cognitive demands [44]. Across a dataset, this cancellation systematically attenuates observed flaw-outcome relationships, producing exactly the weak or null associations that prior work has reported [41, 46, 39]. What is needed is a categorization that separates flaws by their cognitive function, so that opposing mechanisms can be estimated rather than canceled. This study tests this hypothesis by categorizing the 19 IWFs into three cognitive-function categories and examining whether this categorization recovers flaw-outcome relationships that aggregate analyses obscure.

2.3 Response Time and Difficulty

Response time (RT) has been established as a valuable process measure that captures cognitive processing demands during assessment tasks [40, 26]. Often grounded in CLT [44], RT can help distinguish between productive struggle with

content that supports learning and extraneous cognitive load imposed by irrelevant features that burden working memory [10]. Unlike accuracy alone, which provides only a binary outcome, RT reveals how students interact with items and the degree of cognitive effort required to formulate a response [50, 47]. This temporal signature is particularly valuable for understanding processing burdens imposed by poor item design. Theoretically, questions that introduce unnecessary confusion or complexity should produce characteristically longer processing times [61].

Previous research consistently demonstrates associations between longer RT and higher item difficulty, though the relationship is complex and multifaceted [22]. Multiple factors are known to influence RT beyond content difficulty itself, including the cognitive complexity of required reasoning, linguistic features such as readability and syntactic complexity, and structural characteristics including stem length and number of options [42, 29]. While RT relates to item structure and operational difficulty, it is unclear if RT mediates the relationship between specific quality issues (IWFs) and difficulty outcomes. Existing RT research has focused primarily on prediction rather than diagnosis, modeling overall time-on-task or completion rates without connecting temporal patterns to specific quality deficits that could guide authors in creating higher quality questions [61, 54].

2.4 Automated Item Diagnostics

Traditional psychometric quality control relies on post-hoc approaches such as Item Response Theory (IRT) and Classical Test Theory (CTT) to evaluate item performance [15, 52]. These methods require substantial response data, often hundreds of examinees per item, to produce stable parameter estimates [48]. This means that flawed items have already been deployed and consumed learner time before problematic questions are identified. Operationally, many assessments rely on simpler metrics such as proportion-correct or transformed accuracy indices rather than full IRT parameterization [15]. These have been shown to correlate highly with IRT difficulty estimates and require less extensive calibration data [62]. More fundamentally, both IRT and CTT characterize *what* an item's difficulty is, not *why* it is difficult. Neither framework traces the process pathway from specific design features through cognitive processing to observed outcomes, which is the question motivating the present study. Recent advances in machine learning and artificial intelligence have introduced pre-deployment difficulty prediction methods that leverage linguistic features [7], readability metrics [33], and large language models [3, 14] to estimate item difficulty from text alone. These approaches move closer to identifying difficulty sources before deployment, but they still treat items as isolated text rather than as interactive cognitive tasks, focusing on surface features without modeling the processing demands that items impose [17]. Also, they typically identify which items are likely to be problematic without providing actionable guidance about what specifically should be revised, limiting their utility for iterative quality assurance.

Parallel developments in automated time-on-task estimation and IWF detection have operated in similar isolation. RT estimation methods typically rely on shallow features or aggregate completion patterns without connecting temporal signatures to specific quality issues that create processing burdens [51]. Automated IWF detectors, while effective at flagging rule violations, function as diagnostic checklists rather than risk assessors [16, 35]. They identify the presence of flaws but do not indicate whether those flaws impose meaningful processing costs or distort difficulty in practice. The fundamental gap across these lines of work is the absence of an integrated framework that links flaw patterns to process signatures to operational outcomes. This disconnect becomes especially critical as AI-accelerated item generation scales assessment development, yet still produces MCQs that contain flaws [23, 28], even in standardized exams reviewed by multiple experts [34]. Practitioners need quality signals that are both predictive enough to enable pre-deployment screening and diagnostic enough to guide specific, actionable improvements.

Table 1: Overview characteristics of the three datasets.

	USMLE	DBE-KT22	XES3G5M
Domain	Medicine	Comp. Sci.	Math
N items	653	212	365
N options	4–8	2–5	2–5
Stakes	High	Low	Low
Resp./item	∼300+	458–1,049	5–3,237
RT: M (SD) s	85.6 (29.2)	78.5 (40.8)	197.2 (116.1)
Difficulty	Continuous	Ordinal	Continuous

3. METHODS

3.1 Datasets

We draw on three MCQ datasets spanning different disciplines, examinee populations, and testing contexts to assess whether the relationships between item-writing flaws and item performance generalize beyond a single setting. Our cross-setting comparisons therefore test whether the flaw–RT mechanism replicates across heterogeneous assessment contexts rather than isolating a pure domain effect. Table 1 summarizes the key characteristics of each dataset.

3.1.1 USMLE

The first dataset comprises retired clinical MCQs released by the National Board of Medical Examiners for a shared task on predicting item difficulty and response time [63]. The items are drawn from Steps 1, 2 CK, and 3 of the United States Medical Licensing Examination (USMLE) [37] and were authored by experienced subject-matter experts following guidelines that emphasize a standard MCQ structure. They discourage construct-irrelevant cues such as extraneous material, misleading wording, or correct options that are considerably longer than distractors. We excluded 14 of the 667 MCQs in this dataset for data-quality and format issues, primarily invalid Unicode in the source file or having more than 8 answer options, which the automated IWF evaluator we utilized does not support [35]. This yielded an analytic sample of 653 medical licensure MCQs. Each item was administered to approximately 300 or more examinees on a live high-stakes exam. For every item we observe the mean response time (RT) in seconds, computed from on-screen presentation until the examinee advances past the item, aggregated across all test-takers who encountered the item. We also observe a difficulty index defined as a linearly

transformed proportion correct scaled so that higher values indicate more difficult items.

3.1.2 DBE-KT22

The second dataset is drawn from DBE-KT22, a knowledge-tracing dataset collected between 2018 and 2021 on the CodeBench platform at the Australian National University [1]. It contains MCQs from an undergraduate Relational Databases course attempted by students across disciplines including computer science, engineering, business, and arts. We use all 212 items in the dataset. Each item was answered by between 458 and 1,049 students in a self-paced, low-stakes practice environment. Response time for each item is the mean time in seconds taken by students to submit an answer. Unlike the USMLE's continuous difficulty scale, difficulty in DBE-KT22 is an ordinal rating assigned by the course instructors (1=easy, 2=medium, 3=hard), which we retain as an ordinal outcome in downstream analyses.

3.1.3 XES3G5M

The third dataset originates from XES3G5M, a large-scale knowledge-tracing benchmark of third-grade mathematics exercises collected from an online learning platform operated by TAL Education Group in China [30]. The original items are in Chinese, so we use the English translation produced by Ozyurt et al. [38], who translated the question text via the deep-translate Python library for a related study. Because these items were machine-translated rather than originally authored in English, some detected flaws may partly reflect translation artifacts. However, the flaw–RT relationships in XES3G5M are consistent in direction and magnitude with those observed in the two English-native datasets and the pooled model finds no significant flaw-by-domain interactions (Section 4.2). This suggests that any such artifacts do not differentially drive the results.

From the translated item bank we selected the subset of MCQs for which at least five student responses and their associated timestamps were available, yielding 365 items. We adopt this inclusive threshold to maximize the analytic sample and models for XES3G5M use WLS with $\sqrt{n}$ weights to down-weight less precisely estimated item-level means. Robustness checks excluding items with fewer than 30 and 50 responses confirm that results are not driven by low-n items (Section 4.2). Because XES3G5M does not ship with precomputed item statistics, we derived both variables from the student-level interaction logs. Difficulty is computed as 1 minus the proportion of correct first-attempt responses, so that higher values indicate harder items. Response time for each student-item pair is estimated as the elapsed time between the start timestamp of the given question and the start timestamp of the next question in that student's sequence. These per-student times are then averaged across all students who attempted the item.

3.2 Automated Item-Writing Flaw Detection

We detect item-writing flaws (Table 2) with SAQUET (Scalable Automatic Question Usability Evaluation Toolkit) [1], an open-source and domain-agnostic system that operationalizes the standard 19-criterion IWF rubric to produce both per-flaw binary indicators and a total IWF count per item [35]. SAQUET was designed specifically as a tool to prescreen assessments before they are used with students, to capture testwiseness cues and clarity/alignment violations that introduce construct-irrelevant variance. This includes grammatical or length cues to the key, K-type option structures, negative or vague stems, non-parallel options, overlapping alternatives, and more. Prior work reports that such automated IWF application attains strong agreement with expert raters across diverse domains and that it outperforms readability-style proxies for question quality [35]. In this study we use SAQUET as the automated IWF engine and consume its outputs directly in our analyses. Following SAQUET's pipeline, we normalize Unicode and whitespace, lowercase, and tokenize stems and options before applying a layered set of detectors for the 19 IWF criteria.

We selected SAQUET because it is, to our knowledge, the only open-source tool that operationalizes the full 19-criterion IWF rubric in a domain-agnostic manner and has been used in previous research [49, 6]. Alternative MCQ evaluation systems either cover fewer criteria or are not publicly available [16]. Because SAQUET detects structural patterns rather than directly measuring cognitive disruption, its flags should be understood as indicators of flaw *presence* rather than flaw *impact*. This distinction bears on the interpretation of null results, particularly for cue-giving flaws (see Section 6). SAQUET returns a 19-dimensional binary vector per MCQ and derived aggregates (any-flaw and total IWF count). Full implementation details and previously reported accuracy (exact-match, Hamming loss, per-criterion F1) for this method are documented in prior work [35].

3.3 Flaw Category Taxonomy

The standard 19-criterion rubric [55] encompasses IWFs that should theoretically push examinee processing in opposite directions. We address this by mapping each of the 19 IWFs (described in Table 2) into one of three categories based on the type of cognitive demand the flaw is expected to impose. The mapping, summarized in Table 3, was fixed a priori before any outcome analyses were conducted based on the previous literature review. As outlined in Section 2.2, different IWFs are expected to distort the balance between intrinsic and extraneous cognitive load in opposing directions. We operationalize this distinction by mapping each flaw to one of three categories based on the direction of its expected effect as indicated in prior research.

We adopt three categories because the relevant theoretical distinction for our research questions is the direction of the predicted effect. Confusion-inducing flaws impose extraneous load with a clear directional prediction (↑ RT, ↑ difficulty), grounded in well-replicated effects of ambiguity, negation, and gratuitous information on processing demands [20, 44]. Structural-complexity flaws also alter processing, but the direction is not as predictable, as some (e.g. K-type with one obviously plausible combination) may simplify the task, others (e.g. "all of the above") expand it, and some operate primarily on guessing probabilities rather than RT. Collapsing these into a single extraneous load category would conflate flaws with directional predictions and flaws without, weakening the test the taxonomy is designed to perform. Cue-giving flaws form the third category because they carry the opposite directional prediction (↓ RT, ↓ diffi-

[1] `https://saquet.io/`

Table 2: The 19 Item-Writing Flaw (IWF) rubric criteria used to evaluate multiple-choice questions [55, 35].

Item-Writing Flaw	Description of Flaw
Ambiguous/Unclear	The question text or options contain unclear or ambiguous language
Implausible Distractors	One or more distractors are not plausible to be selected, reducing the effectiveness of the item
None of the Above	A "none of the above" option is included, which primarily tests the ability to detect incorrect answers
Longest Option Correct	The correct option is noticeably longer or more detailed than the distractors, cueing students
Gratuitous Information	The stem contains unnecessary information not required to answer the question
True/False Question	The options are structured as a series of true/false statements
Convergence Cues	Options contain overlapping combinations of components that hint at the correct answer
Logical Cues	The wording contains deductive shortcuts that allow the answer to be derived without domain knowledge
All of the Above	An "all of the above" option is included, allowing students to guess based on partial knowledge
Fill-in-the-Blank	Words are omitted from the middle of the stem for students to insert from the options
Absolute Terms	Absolute terms (e.g. always, never) appear in the options, which students recognize as typically false
Word Repeats	Words or phrases are repeated between the stem and the correct option but not in the distractors
Unfocused Stem	The stem does not present a clear, focused question that can be understood without reading the options
Complex or K-Type	The question requires selecting from combinations of responses rather than a single best answer
Grammatical Cues	Options are not grammatically consistent with the stem or parallel in style and form
Lost Sequence	Options are not arranged in a logical (e.g. chronological or numerical) order
Vague Terms	Vague terms (e.g. frequently, occasionally) are used in the options, whose meaning is subjective
More Than One Correct	The question does not have a single best answer, multiple options could be considered correct
Negative Wording	The stem uses negative wording, which can confuse students and obscure the intended learning outcome

culty), making a three-way structure necessary to represent the full range of theorized effects.

Category A (5 flaws): Cue-giving flaws reduce the effective cognitive work required to reach the correct answer. When a stem repeats a key phrase that appears only in the correct option, or when the correct option is conspicuously longer than the distractors, test-wise examinees can exploit these patterns to bypass the intended retrieval process [56, 20]. The five flaws in this category all share the property of providing a shortcut, such as word repeats creating lexical overlap between stem and key. The longest-answer-correct introduces a length asymmetry and grammatical cues allow elimination based on grammatical inconsistencies between an MCQ's text and options [59]. Convergence cues let overlapping options point toward the key and logical cues embed deductive pathways in the stem or option structure [18]. If these cues function as intended by the IWF literature, they should reduce both response time and difficulty [56].

Category B (6 flaws): Confusion-inducing flaws add processing demands that are extraneous to the assessed construct. An unfocused stem forces the examinee to infer what is being asked before retrieval can begin and ambiguous or unclear language requires re-reading and disambiguation [18]. Relatedly, gratuitous information consumes working memory without contributing to the answer [2]. Lost sequencing disrupts the natural scan order of options and negative wording introduces a reversal demand that is well documented to increase errors and slow responses [20]. Vague terms similarly force examinees to interpret subjective qualifiers rather than evaluate content, such as "usually" or "some". These six flaws share a common mechanism because they impose extraneous cognitive load in the sense of Puma et al. [44], consuming time-limited processing resources on construct-irrelevant features of the MCQ. We therefore predict that confusion-inducing flaws will increase response time and, through this added processing burden, associate with greater difficulty.

Category C (8 flaws): Structural complexity flaws alter the response format or option-set architecture in ways that change the task demands, but without a clear directional prediction. Complex K-type structures (e.g. "A and C", "B, C, and D") require combinatorial evaluation rather than single-best-answer selection, which may increase processing time but could also simplify the task if only one combination is plausible [60]. "None of the above" and "all of the above" options shift the response strategy from selecting the best answer to exhaustively verifying each option [11]. True/false formatting and fill-in-the-blank structures change the fundamental task from recognition to verification or recall [19]. Implausible distractors reduce the effective number of options, which could speed responses by shrinking the choice set or could paradoxically slow them if examinees second-guess why an obviously wrong option was included [4]. Absolute terms ("always", "never") occupy a genuinely ambiguous position as item-writing guidelines flag them because test-wise examinees learn to treat absolutes as likely incorrect [20]. This would place them in Category A, but the terms may also confuse less experienced examinees who take them at face value. We assign absolute terms to Category C to acknowledge this ambiguity. The eight flaws in this category share the property of restructuring the response task itself and we treat their effect direction as an empirical question rather than an a priori prediction.

The opposing predictions for Categories A and B are central to this study. If the taxonomy captures meaningful cognitive distinctions, an aggregate IWF count that collapses across categories should show a diluted effect, while category-level analyses should recover the underlying signals.

3.4 Analysis

For each item, we computed three structural control variables: stem length (word count of the question text), number of options (count of non-null answer choices, ranging from 2–8), and mean option length (average word count across options). From the 19 IWFs, we derived binary category variables indicating the presence of at least one cue-giving, confusion-inducing, or structural complexity flaw, as well as corresponding count variables for dose–response analyses. Response time (the item-level mean across all respondents)

Table 3: Mapping of 19 item-writing flaws to cognitive-function categories based on the expected effect on the examinee response process.

Item-Writing Flaw	Rationale
A: Cue-Giving (predicted: ↓ RT, ↓ difficulty)	
Word Repeats	Shared wording between stem and key lets examinees match rather than retrieve
Longest Ans. Correct	Length asymmetry singles out the key without content evaluation
Grammatical Cues	Agreement mismatches allow elimination of options on form alone
Convergence Cues	Overlapping option content narrows the viable set toward the key
Logical Cues	Deductive structure in the stem or options reveals the answer without domain knowledge
B: Confusion-Inducing (predicted: ↑ RT, ↑ difficulty)	
Ambiguous/Unclear	Unclear language forces examinees to infer intent before engaging content
Unfocused Stem	Absence of a clear question expands the space of plausible interpretations
Gratuitous Info	Irrelevant details load working memory without aiding the response
Lost Sequence	Unordered options disrupt systematic comparison across alternatives
Negative Stem	Negation imposes an additional reversal step during option evaluation
Vague Terms	Subjective qualifiers (e.g., "frequently") require interpretation beyond content
C: Structural Complexity (predicted: direction uncertain)	
Complex/K-Type	Combinatorial format may increase evaluation load or simplify when only one combination is plausible
None of the Above	Shifts task from selecting the best option to exhaustively verifying all options
All of the Above	Shifts task from selecting the best option to exhaustively verifying all options
Fill-in-the-Blank	Recall-oriented format changes the cognitive task from recognition to generation
True/False	Binary format alters baseline difficulty and guessing probability
Absolute Terms	May cue test-wise examinees ("always" = likely false) or confuse naïve ones
Implausible Distractors	Reduces effective choice set, but may also prompt second-guessing
More Than One Correct	Multiple defensible keys change elimination strategy unpredictably

was winsorized at the 1st and 99th percentiles within each dataset and log-transformed to approximate normality [32]. For within-dataset models, log(RT) serves as the outcome, enabling interpretation of coefficients as approximate percent changes in response time. For pooled cross-domain models, both RT and difficulty were z-scored within each dataset prior to concatenation, so that one unit represents one within-dataset standard deviation and cross-domain comparisons reflect relative rather than absolute effects. Difficulty is treated as continuous for USMLE and XES3G5M and as ordinal (1 = easy, 2 = medium, 3 = hard) for DBE-KT22. Because USMLE items were each answered by approximately 300+ examinees (roughly constant), those models use ordinary least squares with HC3-robust standard errors [31]. XES3G5M and DBE-KT22 have variable respondent counts per item (5–3,237 and 458–1,049, respectively), so models for these datasets use weighted least squares with $\sqrt{n}$ weights and HC3-robust standard errors to down-weight less precisely estimated item-level means [24].

To address RQ1, we compared the prevalence of each of the 19 IWFs across datasets using chi-square tests (Fisher's exact where expected cell counts fell below 5), with Benjamini-Hochberg false discovery rate correction ($q = 0.05$) applied across the 19 tests and Cramér's V quantified effect sizes at both the individual-flaw and category levels [21, 57]. For RQ2, we fit within-dataset regressions of log(RT) on the three category indicators and structural controls, followed by a pooled model with domain fixed effects and flaw-category-by-domain interactions to test cross-domain consistency. A non-significant joint Wald test for the interaction terms would support a domain-general mechanism [25]. For RQ3, we used bootstrap mediation [43] (5,000 iterations, bias-corrected and accelerated 95% confidence intervals) to estimate indirect effects of each flaw category on difficulty through response time. The aggregate single-indicator model serves as the baseline against which categorized indirect effects are compared. Mediation was conducted separately for USMLE and XES3G5M (continuous difficulty), pooled with domain fixed effects, and corroborated in DBE-KT22 using ordinal logistic regression in which coefficient attenuation upon adding RT indicates consistency with the mediation pathway. All analyses were conducted in Python using `statsmodels` (OLS/WLS, ordinal logistic, Wald tests) and `scipy` (chi-square, Spearman correlations, bootstrap implementations for mediation inference).

4. RESULTS

The final samples comprised 653 items (USMLE, medical), 365 items (XES3G5M, math), and 212 items (DBE-KT22, computer science), for a pooled 1,230 MCQs. All 19 IWFs were evaluated and detailed in the taxonomy in Table 2. Note that the more-than-one-correct IWF had zero instances across all three datasets and was excluded from all models.

4.1 IWF Profiles Across Domains

The three domains exhibited different flaw profiles in both their composition and prevalence (Table 4). Across all three datasets USMLE was the least flawed, with 64.8% of the MCQs being free of IWFs. DBE-KT22 sat at the opposite extreme, with 93.4% of the MCQs containing at least one flaw and XES3G5M occupied an intermediate position with 58.6% flawed. The datasets also differed in which flaw

Table 4: Flaw prevalence, category distribution, and mutual exclusion group sizes by domain.

	USMLE (Medical)	XES3G5M (Math)	DBE-KT22 (CS)
N items	653	365	212
% with any flaw	35.2%	58.6%	93.4%
Mean IWF count (SD)	0.47 (0.75)	0.99 (1.11)	2.02 (1.22)
% with cue-giving flaw	29.9%	21.1%	39.6%
% with confusion flaw	4.6%	39.2%	41.5%
% with structural flaw	6.1%	17.5%	65.1%
Mutual exclusion groups			
No cue/confusion	440 (67.4%)	178 (48.8%)	70 (33.0%)
Cue only	183 (28.0%)	44 (12.1%)	54 (25.5%)
Confusion only	18 (2.8%)	110 (30.1%)	58 (27.4%)
Both	12 (1.8%)	33 (9.0%)	30 (14.2%)

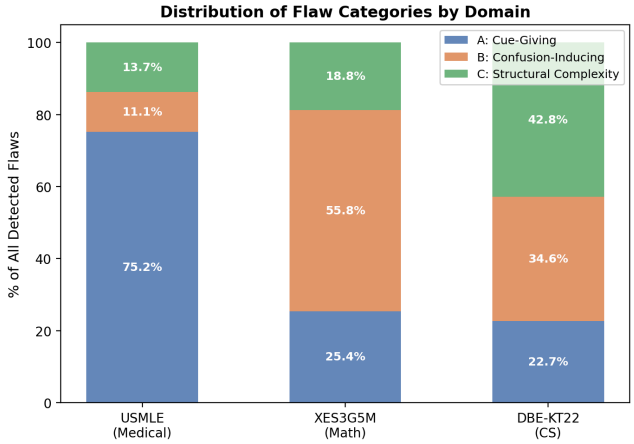


Figure 1: Distribution of detected flaws by cognitive-function category within each domain.

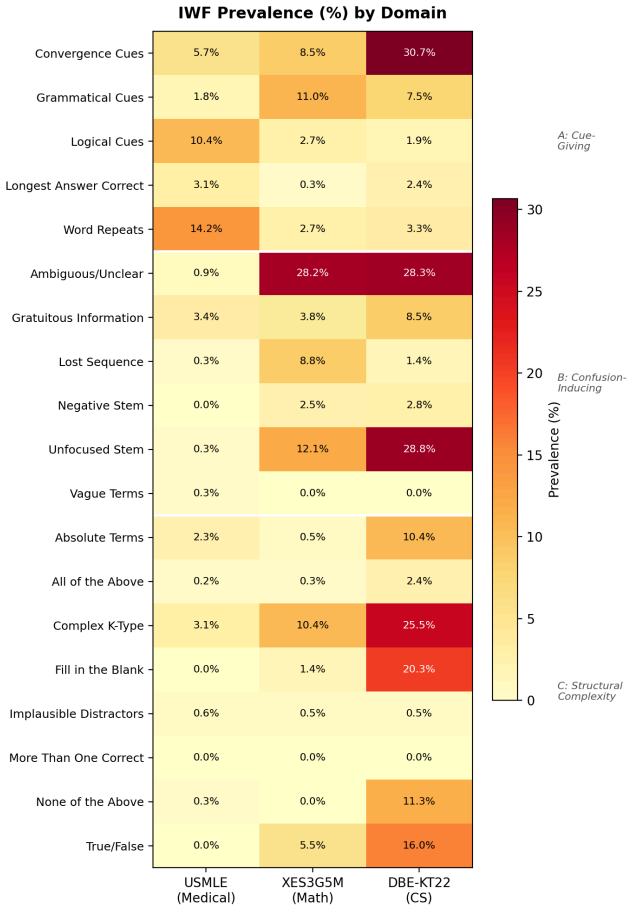


Figure 2: Prevalence of individual item-writing flaws by dataset.

categories dominated. Cue-giving flaws were most common in USMLE, driven by word repeats between stem and key (14.2%) and logical cues (10.4%). Confusion-inducing flaws were rare in USMLE but prevalent in both XES3G5M (39.2%) and DBE-KT22 (41.5%), driven primarily by ambiguous or unclear information (28% in both datasets) and unfocused stems (12.1% in XES3G5M, 28.8% in DBE-KT22). Structural complexity flaws were concentrated in DBE-KT22 (65.1%), where complex K-type items (25.5%), fill-in-the-blank formats (20.3%), and true/false items (16.0%) were common. As shown in Figure 1, when expressed as a share of all detected flaws the category composition of each domain is uniquely different. A majority of USMLE flaws were cue-giving, XES3G5M flaws were confusion-inducing, and DBE-KT22 was more evenly split between confusion and structural complexity.

Chi-square tests comparing each of the 19 IWFs across the three domains, with Benjamini-Hochberg correction for multiple comparisons ($q = 0.05$), found that 16 differed significantly in prevalence (Cramér's $V = 0.085$-0.396), with per-flaw prevalence shown in Figure 2. The three non-significant IWFs, vague terms, implausible distractors, and more-than-one-correct, all had near-zero prevalence in every dataset. At the category level, all three flaw categories differed significantly across domains (all $p < .0001$). The largest effect was for structural complexity ($V = 0.536$), followed by confusion-inducing flaws ($V = 0.433$) and cue-giving flaws ($V = 0.137$). These differences are not merely quantitative, as each domain has a distinct flaw *signature*. USMLE is dominated by cue-giving flaws, XES3G5M by confusion-inducing flaws, and DBE-KT22 by structural complexity. This heterogeneity suggests that treating IWFs as a single aggregate construct risks obscuring domain-specific patterns that may have consequences for examinee performance.

4.2 Item-Writing Flaws and Response Time

Before examining category-level effects, we established the aggregate association between IWF presence and response time. Spearman correlations between total IWF count and mean response time were significant and positive in all three datasets (Table 5). The OLS slopes translate this association into a practical metric of cognitive cost, where each additional detected IWF was associated with roughly 5-16 additional seconds of response time depending on the dataset. That this association holds across all three domains, despite differences in item format, examinee population, and absolute response time scales, establishes a consistent baseline that flawed items take longer to answer. This motivates the subsequent category-level analysis to determine whether this

Table 5: Aggregate association between flaw count and response time. Spearman ρ and OLS slope with 95% CI, all with $p < .001$.

	DBE-KT22	USMLE	XES3G5M
ρ	0.306	0.139	0.214
sec/flaw	+10.45	+4.86	+16.20
95% CI	[6.14, 14.76]	[1.89, 7.82]	[5.54, 26.86]

aggregate signal reflects a uniform contribution from all flaw types or is driven by a specific subset of them.

Within-dataset regression models predicting log(RT) from flaw-category indicators and structural controls revealed a clear asymmetry among flaw categories (Table 6). All models control for z-scored stem length, number of options, and z-scored mean option length, with USMLE estimated via OLS and XES3G5M and DBE-KT22 via WLS ($\sqrt{n}$ weights). Additionally, all use HC3-robust standard errors and coefficients are reported as percent change in RT ($\%\Delta = 100 \times (\exp(\beta) - 1)$). Confusion-inducing flaws were the only category to significantly increase RT, doing so in all three domains (+14-29%, all $p < .05$). A dose-response sensitivity analysis using IWF counts rather than binary indicators confirmed these patterns with the confusion count significant in all three domains ($p = .003$, .010, $< .001$). Cue-giving flaws showed no significant effect on RT in any dataset (all $p > .15$), with coefficients that were positive rather than negative, contrary to the a priori prediction that cue-bearing items would be answered faster. Structural complexity flaws were also non-significant everywhere (all $p > .52$).

A mutual exclusion grouping reinforced these findings as items with only confusion flaws took 16.5-27.0% longer than items free of IWFs. MCQs containing both cue-giving and confusion flaws showed nearly identical RT to the confusion-only group ($\Delta < 2\%$, both $p > .88$ in USMLE and DBE-KT22), indicating that the confusion component drives the effect. These results held across multiple robustness checks that included median RT (confusion significant in XES3G5M and DBE-KT22), unweighted models (all conclusions unchanged), exclusion of low-n items at thresholds of $n \geq 30$ and $n \geq 50$ (confusion $p = .009$ in both), and variance inflation factors below conventional thresholds (all VIFs < 1.5).

The pooled model with IWF-category-by-domain interactions ($R^2 = 0.146$, $N = 1{,}230$) provided a direct test of cross-domain consistency for IWFs increasing RT. The main effect of confusion-inducing flaws was significant ($\beta = +0.516$, $SE = 0.169$, $p = .002$), while the joint Wald test for all six interaction terms was non-significant ($F(6, 1215) = 0.956$, $p = .454$). Per-category interaction tests were also non-significant (cue $\times$ domain: $p = .853$, confusion $\times$ domain: $p = .167$, and structural $\times$ domain: $p = .275$). This pattern of a significant main effect with non-significant interactions supports the interpretation that confusion-inducing flaws impose a processing cost that replicates across these three heterogeneous assessment settings. We found 83-100% of estimable confusion-inducing flaws showed positive RT coefficients across datasets, while most cue-giving flaws did not show the predicted negative direction (20-60% alignment).

Table 6: Flaw category effects on log(RT) from within-dataset regressions, reported as percent change in RT.

Dataset	Cue-giving %Δ (p)	Confusion %Δ (p)	Structural %Δ (p)
USMLE ($N = 653$)	+3.7% (.151)	+14.0% (.016*)	−2.3% (.590)
XES3G5M ($N = 365$)	+15.0% (.233)	+29.4% (.005**)	+7.1% (.524)
DBE-KT22 ($N = 212$)	+8.1% (.295)	+18.9% (.040*)	−3.3% (.676)

4.3 Response Time Mediation of the Flaw-Difficulty Relationship

The aggregate flaw-RT association did not extend to a direct flaw-difficulty link. Spearman correlations between total IWF count and difficulty were non-significant in USMLE and XES3G5M, with only DBE-KT22 showing a reliable association ($\rho = 0.331$, $p < .001$). Meanwhile, the RT-difficulty link varied sharply across datasets, as it was strong in USMLE ($\rho = 0.525$, $p < .001$), moderate in DBE-KT22 ($\rho = 0.245$, $p < .001$), and near-zero in XES3G5M ($\rho = 0.060$, $p = .250$). Total effects of IWFs on difficulty, the direct association without accounting for RT, were non-significant in both datasets, for both the aggregate indicator and all three category-specific indicators (all $p > .18$). With flaws reliably increasing RT, but rarely associating directly with difficulty, mediation analysis is needed to test whether the flaw signal transmits *through* response time even when the direct path is negligible.

Formal mediation was conducted on USMLE and XES3G5M, which have continuous difficulty measures (Figure 3). Using 5,000 bootstrap iterations with bias-corrected confidence intervals, the aggregate indirect effect was modest in USMLE (+0.093) and non-significant in XES3G5M (+0.019). Categorizing by cognitive function revealed that the USMLE aggregate was driven almost entirely by the confusion pathway, as the confusion-inducing indirect effect (+0.231) was the only category to reach significance and was 2.5 times the aggregate magnitude. Cue-giving (+0.064) and structural (−0.040) indirect effects were non-significant. In XES3G5M, all category-specific indirect effects were non-significant, consistent with the weak RT-difficulty link in that dataset. The USMLE results also exhibited a suppression pattern as the positive indirect effect through RT coexisted with a near-zero total effect (total effect = −0.024) because the direct effect was negative (direct effect = −0.243 for confusion). This specifically indicates that the indirect and direct paths operate in opposing directions. The pooled model ($N = 1{,}018$, USMLE + XES3G5M with domain fixed effects) strengthened these findings. Both confusion (+0.117) and cue-giving (+0.051) indirect effects reached significance, with confusion roughly 2.3 times the cue-giving effect. Domain interaction terms were non-significant on both the flaw-to-RT and flaw-to-difficulty equations (both $p > .41$), supporting domain-general mediation.

In DBE-KT22 ($N = 212$), where difficulty is an instructor-assigned ordinal label, formal mediation is not applicable. Instead we tested whether adding RT to an ordinal logistic regression attenuated flaw-difficulty associations. Without

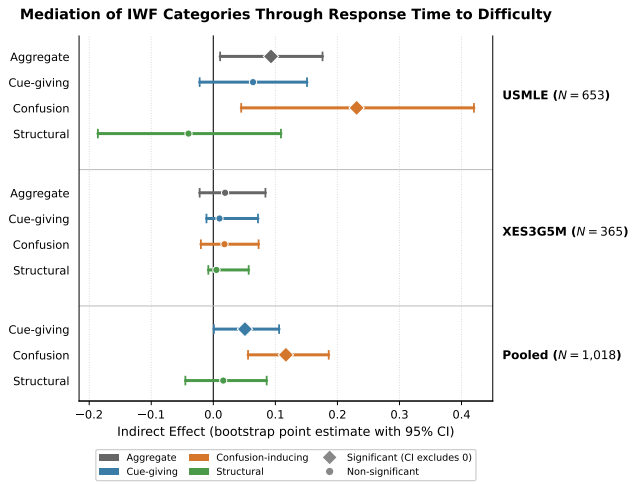


Figure 3: Bootstrap indirect effects (point estimates with 95% bias-corrected CIs) for the flaw category → RT → difficulty mediation pathway. All models control for stem length, option length, and number of options.

RT, both confusion ($\beta = +1.418$, $p < .001$) and structural ($\beta = +1.536$, $p < .001$) flaws predicted higher difficulty ratings. Adding log(RT) improved model fit (LR $\chi^2 = 12.85$, $p = .0003$) and attenuated the confusion coefficient by 10.3%, consistent with partial mediation through RT. Structural flaw coefficients showed no attenuation, suggesting their link to instructor-judged difficulty operates through other channels.

5. DISCUSSION

This study set out to test whether the inconsistent relationships between item-writing flaws with response time and difficulty reported in prior work [41, 46, 49, 39] reflect a measurement problem rather than a null effect. Across 1,230 items from three assessment domains we found that IWF profiles differ qualitatively, with each exhibiting a distinct signature rather than simply more or fewer flaws. When these flaws are categorized by their expected cognitive function, confusion-inducing flaws emerge as the sole category that imposes measurable processing cost, increasing response time by 14-29% across all three domains with no significant domain interactions. This processing cost transmits to difficulty where the RT-difficulty link is strong. In the medical MCQ dataset from the USMLE, the confusion-mediated indirect effect is 2.5 times the aggregate and the pooled model confirms confusion as the dominant pathway. The aggregate flaw-difficulty relationship is weak not because flaws do not matter, but because lumping cue-giving, confusion-inducing, and structural flaws into a single count dilutes the one category that carries the signal.

The domain specific IWF profiles are interpretable in light of each assessment's development context. USMLE's low flaw rate and cue-giving dominance is consistent with a professionally developed high-stakes exam where item writers follow established guidelines, but occasionally allow subtler cue-giving flaws (word repeats, logical cues) to persist [5]. DBE-KT22's saturation with structural complexity and confusion inducing flaws reflects the reality of instructor-authored items in a university CS course, where formal item-writing training is likely minimal [18]. XES3G5M's distinctive confusion profile, 39% of items flagged, driven by ambiguous stems and unfocused questions, may reflect the particular challenge of writing unambiguous items for younger learners in mathematics [27]. These domain specific signatures matter because they determine which category level effects will dominate in any given assessment context. A medical MCQ with one cue-giving flaw and a computer science MCQ with one confusion flaw both register identically under an aggregate count. However, our results show they carry fundamentally different cognitive consequences. Treating flaws as a monolithic construct, as prior work has largely done, obscures this heterogeneity and motivates the category-level analyses we conducted.

Our central finding is that confusion-inducing IWFs (ambiguous or unclear information, unfocused stems, gratuitous details, lost sequencing, and negative wording) are the active ingredient behind the aggregate flaw-RT association. Adding 5 to 16 additional seconds per IWF is carried entirely by the confusion-inducing category. Cue-giving flaws did not reduce RT in any dataset, contrary to the prediction from cognitive load theory and the IWF literature that they should speed responses by signaling the correct answer [56, 20]. Several explanations may account for this null result. SAQUET may detect structural patterns such as word overlap or length asymmetry that do not function as usable cues for examinees under time pressure. Cue-giving flaws may also co-occur with item properties that increase complexity, such as longer stems or more technical language, offsetting any facilitation. Alternatively, the cueing effect may operate on accuracy rather than speed, where examinees who would otherwise guess may benefit from the cue but not necessarily arrive at the answer faster [45]. This may be particularly if recognizing and exploiting the cue requires its own processing effort. Regardless of the cue-giving null, the confusion effect itself is robust across domains. Non-significant interaction terms (Wald $p = .454$) indicate that confusion-inducing flaws impose a comparable relative processing cost across these three datasets. However, since each domain is represented by a single dataset, we cannot disentangle domain from these co-varying factors. This consistency of the effect across such heterogeneous contexts is consistent with a construct-irrelevant processing burden, extraneous processing load in the language of cognitive load theory [44], that is independent of the domain knowledge being assessed.

The mediation results complete the chain from IWF presence through processing time to operational difficulty, while also revealing an informative boundary condition. In USMLE, where RT and difficulty are strongly linked ($\rho = .525$), the confusion-inducing indirect effect was 2.5 times the aggregate, revealing a mediation pathway that aggregation dilutes. In XES3G5M, confusion flaws still increased RT, but RT did not associate with difficulty ($\rho = .060$) and mediation was accordingly null. This divergence is informative rather than problematic as it suggests that what "difficulty" captures varies across assessment contexts. In USMLE, difficulty tracks closely with processing time where harder items take longer. In XES3G5M, difficulty may instead reflect conceptual barriers, such as misunderstanding the underly-

ing mathematics, that do not manifest as additional processing time. A student who lacks the requisite knowledge may answer quickly and incorrectly, producing high difficulty without elevated RT. This dataset dependence helps explain why prior work treating flaws in aggregate across single datasets has found weak or null flaw–difficulty relationships [41, 46]. Those studies conflated flaw types with different (or null) indirect effects and did not account for variation in how strongly aspects like RT track to difficulty. DBE-KT22 provides corroborative evidence from a third operationalization of difficulty. The confusion coefficients attenuated 10.3% when RT was added to an ordinal logistic model of instructor-assigned difficulty ratings, while structural flaw coefficients showed no attenuation. The differential attenuation pattern reinforces the category distinction where confusion flaws operate partly through response time and structural flaws reach difficulty through other channels. This may be through visible format features that instructors associate with complexity when assigning ratings. Together, these results indicate that categorization by cognitive function can help provide insight into the mechanism through which IWFs affect examinee performance.

There are interpretive caveats related to resposne time, as longer resposne time can reflect productive engagement, extraneous load, or sources unrelated to either. This makes it challenging to directly observe cognitive load at the item level. In this work, we treat the additional 14–29% response time under confusion-inducing flaws as cognitive cost because the flaws are by construction construct-irrelevant, but this is an inference from design rather than direct measurement. Additionally, the association of response time translating to performance harm depends on assessment context. In timed high-stakes tests, examinee time is bounded and per-item delays compound into real consequences for completion. In low-stakes practice the throughput cost is less acute, and the primary harm of confusion-inducing flaws may be to the construct validity rather than to performance directly. The XES3G5M boundary case, where confusion flaws increased RT but RT did not associate with difficulty, is consistent with this distinction.

The practical implication is that item quality improvement efforts should prioritize confusion-inducing flaws that create ambiguity, unfocused stems, and gratuitous information because these are the flaws that impose measurable cognitive cost on examinees. Cue-giving flaws remain a concern for validity if they inflate scores, and structural flaws may matter for format standardization, but neither category affects response time in our data. Because the confusion effect is domain general, this recommendation does not require adaptation for specific domains. The guidance to eliminate ambiguity and sharpen stem focus applies equally to medical licensure exams, K-12 math assessments, and university CS courses. For automated detection tools like SAQUET [35] or related ones [16], these findings suggest that not all flags deserve equal weight. A confusion-inducing IWF should trigger revision priority over a cue-giving flag when the goal is reducing construct-irrelevant cognitive load and saving examinee time. More broadly, the results argue for moving beyond binary "flawed or not" classifications toward taxonomies that distinguish flaws by their cognitive consequences. This approach may prove fruitful for both item quality assurance and for understanding how assessment design shapes examinee behavior. The 5-16 additional seconds per flaw documented in our analysis is a real cost, but attributing it equally to all IWFs misidentifies the source and therefore misdirects the remedy.

6. LIMITATIONS AND FUTURE WORK

Several design constraints temper the conclusions drawn here. All analyses operate at the item level using mean response times aggregated across respondents, which precludes examining within-person effects or distinguishing processing patterns, such as fast-correct versus slow-incorrect responses. The flaw indicators themselves are structural pattern matches produced by an automated detector, not ground-truth judgments of cognitive disruption. This distinction may partly account for the cue-giving null, if SAQUET flags formal patterns (e.g. word overlap) that do not function as usable cues under real testing conditions. DBE-KT22's small zero-flaw baseline ($n = 14$) increases estimator variance for that dataset, though the regression framework does not require a large joint-zero group in the same way ANOVA would. USMLE lacks median response time data, so the median-RT robustness check could not be applied to the dataset with the strongest mediation finding. While the three datasets used spanned unique domains, they also varied in examinee population, authoring practices, and more, which also attribute to the observed differences. Finally, although we use the language of mediation throughout, the design is observational and cross-sectional at the item level. These indirect effects should be interpreted as associational pathways rather than confirmed causal chains.

Student-level analysis with individual response times would permit testing whether confusion-inducing flaws impose uniform processing costs or disproportionately burden lower-ability examinees, a question the item-level design cannot address. Examining whether cue-giving flaws affect accuracy rather than speed would test the alternative interpretation that cueing operates on correctness without reducing response time. The taxonomy could also be applied to AI-generated items to determine whether large language models produce different flaw signatures than human item writers, a question of growing practical relevance as automated item generation scales. More broadly, the RT-based screening approach demonstrated here could be formalized into a practical quality assurance tool with decision thresholds that weight confusion flags above other flaw types, moving from the associational evidence presented in this study toward an actionable pre-deployment diagnostic.

7. CONCLUSION

Across 1,230 multiple-choice questions from three assessment domains, we show that weak or inconsistent relationships between item-writing flaws and item outcomes are often a measurement consequence of aggregation, not evidence that flaws are inconsequential. By introducing a cognitive-function taxonomy that separates cue-giving, confusion-inducing, and structural-complexity flaws, we reveal that different flaw types operate through distinct mechanisms, so collapsing them into a single flaw count can mask real effects. In particular, confusion-inducing flaws are the primary driver of processing burden, increasing mean response time by 14-29% across domains, an effect that held across

all three domains without significant moderation. When difficulty covaries with response time, the added time burden from confusion-inducing flaws is consistent with an indirect pathway to higher operational difficulty via response time. In the USMLE dataset, this indirect effect is 2.5× larger than the indirect effect based on the aggregate flaw count. For assessment practice, the implication is that not all IWFs deserve equal attention. Automated evaluation tools become more useful when they prioritize confusion-related issues, such as gratuitous information, that reliably add time due to wording rather than the target skill. As AI-generated assessment content scales, the taxonomy introduced here provides a cognitively grounded framework for quality assurance that moves beyond simply flagging flaws, to prioritizing the ones that impose measurable time costs on examinees and distort what assessments measure.

8. REFERENCES

[1] G. Abdelrahman, S. Abdelfattah, Q. Wang, and Y. Lin. Dbe-kt22: A knowledge tracing dataset based on online student evaluation. *arXiv preprint arXiv:2208.12651*, 2022.

[2] J. Abedi. Language issues in item development. In *Handbook of test development*, pages 391–412. Routledge, 2011.

[3] S. AlKhuzaey, F. Grasso, T. R. Payne, and V. Tamma. Text-based question difficulty prediction: A systematic review of automatic approaches. *International Journal of Artificial Intelligence in Education*, 34(3):862–914, 2024.

[4] S. Asthana, T. Arif, and K. C. Thompson. Field experiences and reflections on using llms to generate comprehensive lecture metadata. In *NeurIPS'23 Workshop on Generative AI for Education (GAIED)*, 2023.

[5] M. H. Balaha, M. T. El-Ibiary, A. A. El-Dorf, S. L. El-Shewaikh, and H. M. Balaha. Construction and writing flaws of the multiple-choice questions in the published test banks of obstetrics and gynecology: adoption, caution, or mitigation? *Avicenna Journal of Medicine*, 12(03):138–147, 2022.

[6] N. Balepur, B. Rajasekaran, H. J. Oh, M. Xie, A. Desai, V. Gupta, S. J. Moore, E. Choi, R. Rudinger, and J. L. Boyd-Graber. Benchmarker: An education-inspired toolkit for highlighting flaws in multiple-choice benchmarks. In *Association for Computational Linguistics*, 2026.

[7] L. Benedetto, A. Cappelli, R. Turrin, and P. Cremonesi. R2de: a nlp approach to estimating irt parameters of newly generated questions. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 412–421, 2020.

[8] E. Costello, J. Holland, and C. Kirwan. The future of online testing and assessment: question quality in moocs. *International Journal of Educational Technology in Higher Education*, 15(1):1–14, 2018.

[9] E. Costello, J. C. Holland, and C. Kirwan. Evaluation of mcqs from moocs for common item writing flaws. *BMC research notes*, 11(1):849, 2018.

[10] K. E. DeLeeuw and R. E. Mayer. A comparison of three measures of cognitive load: Evidence for separable measures of intrinsic, extraneous, and germane load. *Journal of educational psychology*, 100(1):223, 2008.

[11] D. DiBattista, J.-A. Sinnige-Egger, and G. Fortuna. The "none of the above" option in multiple-choice testing: An experimental study. *The Journal of Experimental Education*, 82(2):168–183, 2014.

[12] S. M. Downing. Construct-irrelevant variance and flawed test questions: Do multiple-choice item-writing principles make any difference? *Academic Medicine*, 77(10):S103–S104, 2002.

[13] S. M. Downing. The effects of violating standard item writing principles on tests and students: the consequences of using flawed test items on achievement examinations in medical education. *Advances in health sciences education*, 10(2):133–143, 2005.

[14] A. Dutulescu, S. Ruseti, M. Dascalu, and D. Mcnamara. How hard can this question be? an exploratory analysis of features assessing question difficulty using llms. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 802–808, 2024.

[15] X. Fan. Item response theory and classical test theory: An empirical comparison of their item/person statistics. *Educational and psychological measurement*, 58(3):357–381, 1998.

[16] T. Firoozi, L. Daniels, V. Daniels, and M. Gierl. An augmented intelligence system for automated quality control and feedback generation of multiple choice test items. In *International Conference on Artificial Intelligence in Education*, pages 86–93. Springer, 2025.

[17] F. Goldhammer, J. Naumann, A. Stelter, K. Tóth, H. Rölke, and E. Klieme. The time on task effect in reading and problem solving is moderated by task difficulty and skill: insights from a computer-based large-scale assessment. *Journal of Educational Psychology*, 106(3):608, 2014.

[18] V. Gupta, E. R. Williams, and R. Wadhwa. Multiple-choice tests: A–z in best writing practices. *Psychiatric Clinics*, 44(2):249–261, 2021.

[19] A. Gurung, K. Vanacore, A. A. McReynolds, K. S. Ostrow, E. Worden, A. C. Sales, and N. T. Heffernan. Multiple choice vs. fill-in problems: The trade-off between scalability and learning. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 507–517, 2024.

[20] T. M. Haladyna, S. M. Downing, and M. C. Rodriguez. A review of multiple-choice item-writing guidelines for classroom assessment. *Applied measurement in education*, 15(3):309–333, 2002.

[21] W. Haynes. Benjamini–hochberg method. In *Encyclopedia of systems biology*, pages 78–78. Springer, 2013.

[22] M. Jiang and J. E. McLaughlin. Item discrimination, difficulty, and response time for 4-option mcqs versus 3-option mcqs. *American Journal of Pharmaceutical Education*, page 101408, 2025.

[23] B. Johnson, J. Dittel, and R. V. Campenhout. Investigating student ratings with features of automatically generated questions: A large-scale analysis using data from natural learning contexts. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 194–202, 2024.

[24] H. A. Kiers. Weighted least squares fitting using ordinary least squares algorithms. *Psychometrika*, 62(2):251–266, 1997.

[25] D. A. Kodde and F. C. Palm. Wald criteria for jointly testing equality and inequality restrictions. *Econometrica: journal of the Econometric Society*, pages 1243–1248, 1986.

[26] P. C. Kyllonen and J. Zu. Use of response time for measuring cognitive ability. *Journal of Intelligence*, 4(4):14, 2016.

[27] J. Lee, D. Smith, S. Woodhead, and A. Lan. Math multiple choice question generation via human-large language model collaboration. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 941–946, 2024.

[28] U. Lee, Y. Kim, S. Lee, J. Park, J. Mun, E. Lee, H. Kim, C. Lim, and Y. J. Yoo. Can we use gpt-4 as a mathematics evaluator in education?: Exploring the efficacy and limitation of llm-based automatic assessment system for open-ended mathematics question. *International Journal of Artificial Intelligence in Education*, 35(3):1560–1596, 2025.

[29] Y.-H. Lee and Y. Jia. Using response time to investigate students' test-taking behaviors in a naep computer-based study. *Large-scale Assessments in Education*, 2(1):8, 2014.

[30] Z. Liu, Q. Liu, T. Guo, J. Chen, S. Huang, X. Zhao, J. Tang, W. Luo, and J. Weng. Xes3g5m: A knowledge tracing benchmark dataset with auxiliary information. *Advances in Neural Information Processing Systems*, 36:32958–32970, 2023.

[31] J. S. Long and L. H. Ervin. Using heteroscedasticity consistent standard errors in the linear regression model. *The American Statistician*, 54(3):217–224, 2000.

[32] P. Mair and R. Wilcox. Robust statistical methods in r using the wrs2 package. *Behavior research methods*, 52(2):464–488, 2020.

[33] W. Marinho, E. W. Clua, L. Martí, and K. Marinho. Predicting item response theory parameters using question statements texts. In *LAK23: 13th International learning analytics and knowledge conference*, pages 1–10, 2023.

[34] A. Masrur, A. Tayyab, Z. A. Khan, S. Jaffar, M. Mansoor, S. Mansoor, A. Rauf, and S. S. Shah. The impact of item-writing flaws on student test scores, pass rate and psychometric properties of the test in end of clerkship ophthalmology assessment. *Multicultural Education*, 8(4), 2022.

[35] S. Moore, E. Costello, H. A. Nguyen, and J. Stamper. An automatic question usability evaluation toolkit. In *International Conference on Artificial Intelligence in Education*, pages 31–46. Springer, 2024.

[36] R. Nedeau-Cayo, D. Laughlin, L. Rus, and J. Hall. Assessment of item-writing flaws in multiple-choice questions. *Journal for nurses in professional development*, 29(2):52–57, 2013.

[37] J. Norcini, I. Grabovsky, M. A. Barone, M. B. Anderson, R. S. Pandian, and A. J. Mechaber. The associations between united states medical licensing examination performance and outcomes of patient care. *Academic Medicine*, 99(3):325–330, 2024.

[38] Y. Ozyurt, S. Feuerriegel, and M. Sachan. Automated knowledge concept annotation and question representation learning for knowledge tracing. *arXiv preprint arXiv:2410.01727*, 2024.

[39] J. Pais, A. Silva, B. Guimarães, A. Povo, E. Coelho, F. Silva-Pereira, I. Lourinho, M. A. Ferreira, and M. Severo. Do item-writing flaws reduce examinations psychometric quality? *BMC Research Notes*, 9(1):399, 2016.

[40] Z. Papamitsiou, A. A. Economides, I. O. Pappas, and M. N. Giannakos. Explaining learning performance using response-time, self-regulation and satisfaction from content: An fsqca approach. In *Proceedings of the 8th international conference on learning analytics and knowledge*, pages 181–190, 2018.

[41] H. Pham, J. Besanko, and P. Devitt. Examining the impact of specific types of item-writing flaws on student performance and psychometric properties of the multiple choice question. *MedEdPublish*, 7:225, 2018.

[42] C. Pires, A. Cavaco, and M. Vigário. Towards the definition of linguistic metrics for evaluating text readability. *Journal of Quantitative Linguistics*, 24(4):319–349, 2017.

[43] K. A. Pituch, L. M. Stapleton, and J. Y. Kang. A comparison of single sample and bootstrap methods to assess mediation in cluster randomized trials. *Multivariate Behavioral Research*, 41(3):367–400, 2006.

[44] S. Puma, N. Matton, P.-V. Paubel, and A. Tricot. Cognitive load theory and time considerations: Using the time-based resource sharing model. *Educational Psychology Review*, 30(3):1199–1214, 2018.

[45] H. Reuss, A. Kiesel, and W. Kunde. Adjustments of response speed and accuracy to unconscious cues. *Cognition*, 134:57–62, 2015.

[46] B. R. Rush, D. C. Rankin, and B. J. White. The impact of item-writing flaws and item complexity on examination item difficulty and discrimination value. *BMC medical education*, 16(1):250, 2016.

[47] I. Rushkin, I. Chuang, and D. Tingley. Modelling and using response times in online courses. *Journal of Learning Analytics*, 6(3):76–89, 2019.

[48] A. Sahin and D. Anil. The effects of test length and sample size on item parameters in item response theory. *Educational Sciences: Theory and Practice*, 17(1n):321–335, 2017.

[49] R. Schmucker and S. Moore. The impact of item-writing flaws on difficulty and discrimination in item response theory. *arXiv preprint arXiv:2503.10533*, 2025.

[50] S. D. Schneid, C. Armour, Y. S. Park, R. Yudkowsky, and G. Bordage. Reducing the number of options on multiple-choice questions: response time, psychometrics and standard setting. *Medical Education*, 48(10):1020–1027, 2014.

[51] M. A. Shinnick and M. A. Woo. Validation of time to task performance assessment method in simulation: A comparative design study. *Nurse education today*, 64:108–114, 2018.

[52] A. Strugatski and G. Alexandron. Applying irt to distinguish between human and generative ai responses to multiple-choice assessments. In *Proceedings of the*

15th International Learning Analytics and Knowledge Conference, pages 817–823, 2025.

[53] J. Sweller, P. Ayres, and S. Kalyuga. Altering element interactivity and intrinsic cognitive load. In *Cognitive load theory*, pages 203–218. Springer, 2011.

[54] B. Tan and O. Bulut. The value of individual screen response time in predicting student test performance: Evidence from timss 2019 problem solving and inquiry tasks. *Journal of Intelligence*, 13(7):82, 2025.

[55] M. Tarrant, A. Knierim, S. K. Hayes, and J. Ware. The frequency of item writing flaws in multiple-choice questions used in high stakes nursing assessments. *Nurse Education Today*, 26(8):662–671, 2006.

[56] M. Tarrant and J. Ware. Impact of item-writing flaws in multiple-choice questions on student achievement in high-stakes nursing assessments. *Medical education*, 42(2):198–206, 2008.

[57] A. Telford, C. C. Taylor, H. M. Wood, and A. Gusnanto. Properties and approximate p-value calculation of the cramer test. *Journal of Statistical Computation and Simulation*, 90(11):1965–1981, 2020.

[58] D. R. Thomas, C. Borchers, S. Kakarla, J. Lin, S. Bhushan, B. Guo, E. Gatz, and K. R. Koedinger. Does multiple choice have a future in the age of generative ai? a posttest-only rct. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 494–504, 2025.

[59] M. H. Towns. Guide to developing high-quality, reliable, and valid multiple-choice assessments. *Journal of Chemical Education*, 91(9):1426–1431, 2014.

[60] A. Tripp and N. Tollefson. Are complex multiple-choice options more difficult and discriminating than conventional multiple-choice options?, 1985.

[61] S. Wang, S. Zhang, J. Douglas, and S. Culpepper. Using response times to assess learning progress: A joint model for responses and response times. *Measurement: Interdisciplinary Research and Perspectives*, 16(1):45–58, 2018.

[62] A. E. Wyse and S. Hao. An evaluation of item response theory classification accuracy and consistency indices. *Applied Psychological Measurement*, 36(7):602–624, 2012.

[63] V. Yaneva, K. North, P. Baldwin, L. A. Ha, S. Rezayi, Y. Zhou, S. R. Choudhury, P. Harik, and B. Clauser. Findings from the first shared task on automated prediction of difficulty and response time for multiple-choice questions. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 470–482, 2024.

The Semantic Gap in Behavioral Embeddings: Why Linear Methods Fail for Educational RAG in Mathematics

Ricky Gole
Morgan State University
rigol1@morgan.edu

Jamell Dacon
Morgan State University
jamell.dacon@morgan.edu

ABSTRACT

Educational AI systems often assume that behavioral failure patterns in student data reveal latent semantic structures useful for tutoring. We rigorously audit this assumption using 15.8 million student interactions from UK secondary mathematics. By correlating a 50-dimensional SVD-based behavioral manifold with neural semantic embeddings across 1,869 questions, we identify a critical **Semantic Gap**: a near-zero correlation ($r = 0.000$) demonstrating that behavioral patterns are functionally independent of semantic content. This gap causes systematic Topical Drift in naive RAG systems, where behavioral-only retrieval performs below random chance (12.3% pairwise win rate). To bridge this gap, we introduce the principle of **Semantic Primacy** and a hybrid **Manifold-Augmented Generation (MAG)** architecture. Our results show that MAG achieves superior diagnostic specificity (4.78/5) compared to zero-shot baselines by enforcing semantic filtering before behavioral personalization. Finally, we characterize the **Semantic Opacity** of behavioral clusters through expert audit, proving the necessity of architectural integration over manual curation. This work establishes a new baseline for understanding behavioral-semantic orthogonality in educational data mining.

Keywords

Educational Data Mining, Retrieval-Augmented Generation, Behavioral Patterns, Semantic Embeddings, Matrix Factorization, Educational AI

1. INTRODUCTION

Educational AI systems currently face a profound grounding problem. While foundation models generate pedagogically plausible responses, they often lack *diagnostic specificity*, leaving them unable to adaptively guide learners through different cognitive states such as confusion, partial understanding, and misconception [8]. This blindness leads to a reliance on linguistic patterns rather than pedagogical ones. Recent work has turned to behavioral manifolds: clustering questions by how students fail to reveal diagnostic structures invisible to semantic analysis [17]. The hypothesis is compelling: if students make identical errors across different problems, those problems likely share a behavioral attractor, a latent misconception that transcends the wording of the prompt in question.

Recent systematic evaluations [8] expose the depth of this diagnostic blindness and demonstrate that foundation models fail to perceive learner states accurately, to orchestrate appropriate instructional strategies, and to elicit reflection when students exhibit surface-level correctness while harboring fundamental misconceptions. A model might validate a student who correctly applies the quadratic formula yet cannot explain why the discriminant determines the number of solutions. This creates a pedagogical hazard: students receive affirmation for brittle knowledge that crumbles under transfer. The issue is not linguistic coherence but diagnostic precision. Without grounding in actual student failure patterns, even the most eloquent model remains pedagogically shallow.

However, this behavioral manifold hypothesis rests on an untested assumption of alignment. Do interaction patterns from student responses preserve the semantic coherence necessary for retrieval in agentic teaming? If behavioral clustering merely captures difficulty levels or generic test-taking patterns rather than conceptual relationships, systems using these patterns will retrieve topically irrelevant context. This Topical Drift undermines the agent's reliability, creating a mismatch where a student struggling with negative numbers receives hints about fractions simply because both show similar failure rates.

In this work, we rigorously audit the prevailing architectural standard of linear embeddings. While non-linear techniques are emerging, the vast majority of scalable educational platforms default to linear methods like SVD-based matrix factorization for real-time recommendation due to their computational frugality and interpretability. This focus on linear methods is not arbitrary. Singular Value Decomposition (SVD) based approaches dominate real-time educational platforms precisely because they provide the computational speed required for responsive tutoring at scale. Platforms serving millions of students require sub-second response times for hint generation and content sequencing. SVD decomposes sparse interaction matrices into low-dimensional representations that enable rapid

Ricky Gole, and Jamell Dacon. The Semantic Gap in Behavioral Embeddings: Why Linear Methods Fail for Educational RAG in Mathematics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 306–316. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040046

nearest-neighbor search without expensive recomputation.

Intelligent tutoring systems, adaptive learning platforms, and recommendation engines from Khan Academy to Carnegie Learning rely on variants of this approach. Given this
widespread deployment, auditing the semantic fidelity of linear behavioral embeddings directly impacts the pedagogical quality experienced by students daily. If these methods systematically fail to preserve topic coherence, millions of learners receive contextually inappropriate feedback through architectures designed for efficiency rather than semantic grounding. We treat these methods as the empirical baseline to investigate the following research questions (**RQs**):

- **RQ1:** Do linear projections of binary correctness preserve semantic similarity when measured against modern neural embeddings?
- **RQ2:** What are the practical consequences of semantic-behavioral orthogonality for retrieval-based tutoring systems?
- **RQ3:** Can architectural integration, such as Manifold-Augmented Generation (MAG), leverage orthogonal signals to outperform high compute zero-shot models?
- **RQ4:** Do behavioral patterns exhibit Semantic Opacity, rendering them unreadable to human experts and necessitating computational stabilization?

Using a massive dataset of 15.8 million student interactions from UK secondary mathematics, we construct a 50-dimensional behavioral manifold via SVD. This finding implies that Behavioral Sightedness and Semantic Understanding are orthogonal dimensions of the learning experience. We demonstrate that this gap leads to systematic failure in naive retrieval systems, where behavioral-only retrieval frequently retrieves topically irrelevant hints. However, we propose that this independence is a design opportunity rather than a defect.

By enforcing **Semantic Primacy**, a hybrid architecture that filters by topic before personalizing by behavior, we can leverage these orthogonal signals to catch invisible errors that linguistic models miss. Finally, we bridge the gap between data mining and the learning sciences through a qualitative *Expert Tagging* study to evaluate whether these manifolds represent human-readable pedagogical traps. This work advances what we term the Sighted-Frugal paradigm.

Rather than pursuing diagnostic precision through ever larger foundation models, we demonstrate that architectural constraint can achieve superior specificity at a fraction of the computational cost. The prevailing assumption in educational AI has been that pedagogical failures stem from insufficient model capacity. This fuels an arms race toward parameter scaling, where each generation of models grows more expensive to train and deploy. We challenge this assumption. Our results show that a modest model like GPT-4o-mini, when augmented with properly structured behavioral context, outperforms far larger zero-shot alternatives in diagnostic tasks. The key is not scale but grounding.

By respecting the orthogonality between behavioral and semantic dimensions, we can build systems that are simultaneously more accurate and more accessible. This matters for equity. Computationally intensive models concentrate educational innovation in well resourced institutions. Sighted-Frugal architectures democratize access to diagnostic AI.

Our primary contributions are as follows:

1. **Quantification of the Semantic Gap:** We provide the first large-scale empirical evidence that linear behavioral embeddings common in deployed systems are functionally independent of semantic content, providing a theoretical ceiling for naive RAG systems.
2. **The Principle of Semantic Primacy:** We establish a core design requirement for educational RAG: systems must strictly filter by semantic content before applying behavioral personalization to avoid pedagogical drift.
3. **Demonstration of Sighted-Frugal AI:** We validate a hybrid Manifold-Augmented Generation (MAG) architecture that outperforms high compute zero-shot models in diagnostic specificity, offering a path toward more equitable and sustainable educational AI.

2. RELATED WORK

2.1 Behavioral Data for Misconception Detection

Mathematics education research has long argued that effective instruction requires identifying systematic error patterns rather than mere correctness [9]. Cognitive evidence suggests that these behavioral signatures reflect underlying conceptual structures [2]. Recent large-scale efforts, such as the Eedi dataset [3] and diagnostic benchmarks from Wang et al. [17], have enabled the computational mining of these misconceptions.

The Gap: While these works provide the raw material for diagnostic AI, they often assume that behavioral clusters are inherently semantically meaningful. Our work explores and tests this assumption by auditing whether the clusters discovered in these datasets actually align with the conceptual categories recognized by expert educators and semantic models.

2.2 Knowledge Tracing and Interaction Modeling

Knowledge Tracing (KT) has evolved from Bayesian frameworks [4] to deep recurrent architectures such as RNNs [11] capable of predicting student mastery from high-dimensional interaction logs.

The Gap: These models are traditionally optimized for *predictive performance* (i.e., will a student get the next problem right?). However, being able to predict a failure does not necessarily mean the model understands the semantic

nature of that failure. Thus, we aim to move beyond prediction to investigate whether the latent spaces created by KT-like interaction modeling are sufficiently grounded in content to support *retrieval and generation* in tutoring applications, specifically for mathematics-based coursework.

2.3 Retrieval-Augmented Generation (RAG) in Education

RAG systems improve generation by retrieving external context based on semantic similarity [7, 13]. In education, this has roots in retrieving worked examples for Cognitive Tutors [14] and early automated hint generation from historical data [1].

The Gap: Standard educational RAG assumes that semantic similarity (how a problem is worded) is the best proxy for retrieval. Conversely, behavioral systems assume that failure patterns are the best proxy. To our knowledge, no prior study has measured the *alignment* between these two signals. We identify the Semantic Gap between these dimensions, necessitating the shift toward our proposed *Semantic Primacy* architecture.

2.4 Model Alignment and Diagnostic Precision

General-purpose foundation models are aligned for helpfulness and safety using techniques like RLHF [10] and DPO [12]. However, recent evaluations show that this alignment often lacks *diagnostic specificity*, causing models to provide polite but pedagogically vague feedback [8].

The Gap: Effective feedback requires precision [5], yet models often fall into a Black Box of Competence. We investigate whether incorporating behavioral manifolds as retrieval context can provide the Sightedness necessary to restore diagnostic precision without the heavy compute costs of model retraining.

2.5 Triangulated Evaluation Methodologies

The rise of LLM-as-a-Judge [18] has enabled scalable evaluation, but educational contexts require more rigorous, multi-dimensional validation to ensure replicability and improved metacognition [15].

The Pivot: We avoid reliance on a single metric by employing a triangulated methodology. We combine automated neural scoring with a qualitative Expert Tagging component to test for *Semantic Opacity*. This ensures that our findings are grounded not just in statistical correlation, but in pedagogical reality.

3. METHODOLOGY

3.1 Data Sources and Problem Formulation

We utilize the NeurIPS 2020 Education Challenge dataset [17], comprising 15.8 million interactions from the Eedi platform (UK secondary mathematics, ages 11-18). To ground our behavioral manifold in expert pedagogy, we cross-reference this with the 2024 Mining Misconceptions dataset [3], which provides expert-verified labels for common student errors.

From an initial pool of 27,613 questions, we isolated a core set of 1,869 questions with expert labels. For our comparative agentic experiments, we focused on 242 questions that met the threshold for high diagnostic utility: a minimum of 500 student interactions and a failure rate of at least 40%. This subset was divided into a development set ($n = 120$) and a held-out evaluation set ($n = 122$).

3.2 Auditing the Behavioral Manifold (SVD)

To represent the industry standard for scalable educational AI, we constructed a behavioral manifold using (SVD). We generated a sparse binary interaction matrix M, where $M_{i,j} = 1$ denotes a correct response from student i to question j.

We reduced this matrix to a 50-dimensional space, capturing the latent patterns of student success and failure. While modern deep learning techniques exist, we intentionally selected SVD to audit the specific linear assumptions prevalent in real-time production systems. To identify structural attractors within this space, we applied spectral clustering [16] with a nearest-neighbors affinity graph (n=10), identifying $k = 50$ clusters. This choice was based on domain considerations rather than optimization, though silhouette analysis yielded a score of -0.050, reflecting the pedagogically realistic property that trap clusters are not geometrically isolated from the broader manifold. Behavioral coherence was instead validated through downstream cluster statistics: identified trap clusters exhibited consistent failure rates and substantial interaction volumes, confirming that spectral clustering captured psychometric structure rather than geometric separation.

We prioritized pedagogical interpretability over statistical optimization: clusters with clean separation (high silhouette scores) but mixed difficulty profiles would be computationally elegant yet pedagogically meaningless. Our resulting clusters exhibited behavioral coherence, questions within each cluster shared similar failure rates and discrimination parameters, validating that spectral clustering captured psychometric structure rather than random noise. Critically, this behavioral coherence did not translate to semantic coherence, as subsequent correlation analysis would reveal.

While alternative cluster counts were not systematically explored, the resulting structure proved pedagogically interpretable: identified trap clusters exhibited consistent failure rates (40 to 51 percent) and substantial interaction volumes (1000 plus interactions per question), while mastery clusters demonstrated clear success patterns (success rates exceeding 70 percent). This behavioral coherence, despite the lack of formal hyperparameter tuning, suggests the clustering captured meaningful pedagogical structure rather than statistical noise.

3.3 Quantifying the Semantic Gap

To measure the alignment between behavior and content, we computed semantic embeddings for all questions using Sentence-BERT (S-BERT) [13]. We validated that these 384-dimensional vectors captured pedagogical relationships (e.g., questions with identical misconception labels showed significantly higher cosine similarity, $p < 0.001$).

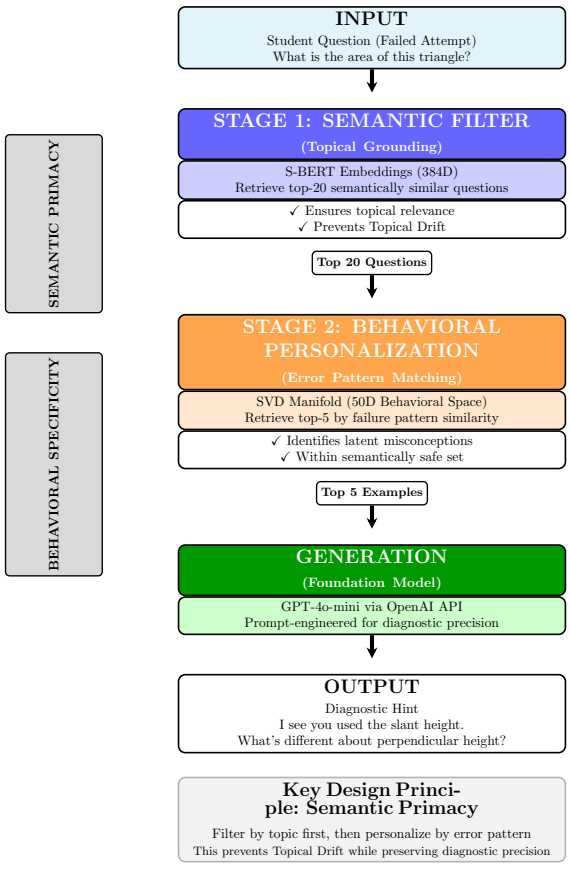


Figure 1: MAG architecture implementing Semantic Primacy. The pipeline enforces a two-stage constraint: semantic filtering narrows retrieval to topically relevant questions (top 20), then behavioral personalization identifies error patterns within this semantically bounded set (top 5). This sequential design prevents topical drift while preserving diagnostic specificity. Generation employs GPT-4o-mini via API with prompt engineering for pedagogical precision. Comparative evaluation (n=122) demonstrated that this architectural approach (4.75/5) substantially outperforms DPO-aligned Llama-3-8B (2.96/5, $p<0.001$), validating that architectural constraint on capable models dominates explicit alignment of smaller alternatives.

The Semantic Gap was measured by calculating the correlation between behavioral similarity (computed as $1/(1+d)$ where d is the Euclidean distance in the 50D SVD space) and semantic similarity (cosine similarity in the 384D Sentence-BERT space). We employed Pearson's r, Spearman's ρ, and permutation testing to ensure that the observed independence was not a result of difficulty-based confounds.

This operationalization requires explicit justification. Euclidean distance in the SVD-reduced space represents the standard metric for behavioral nearest neighbor search in production recommendation systems: it directly measures the dissimilarity used by retrieval algorithms when selecting contextually similar problems. Cosine similarity in neural embedding space serves as our semantic proxy because S-BERT explicitly optimizes for sentence-level semantic similarity, and we validated that it captures pedagogical relationships (questions sharing misconception labels show significantly higher similarity). The correlation between these distance measures tests whether behavioral neighborhoods preserve semantic structure: if they did, questions close in behavioral space would also be close semantically, yielding positive correlation. Our near-zero result indicates behavioral proximity provides no semantic information.

3.4 Agentic Architectures and Semantic Primacy

We compared four tutoring architectures to evaluate how different retrieval strategies impact the Sightedness of the agent. All systems utilized GPT-4o-mini as the generation backbone throughout, ensuring that differences in performance were attributable solely to the retrieval logic. GPT-4o was used exclusively as the evaluation judge for System D (see Section 3.5), not as a generator in any system.

- **System A (Zero-Shot):** A high compute baseline using GPT-4o-mini without external context, representing the ungrounded generation.
- **System B (Semantic RAG):** A standard RAG approach that retrieves the five most topically similar questions

based on S-BERT embeddings.

- **System C (Behavioral RAG):** A behavior-only approach that retrieves the nearest neighbor in the 50D SVD space, testing if failure patterns alone provide sufficient diagnostic context. This single-neighbor retrieval reflects how behavioral-only systems operate in practice, and tests whether even minimal behavioral context provides diagnostic value.
- **System D (MAG: Semantic Primacy):** Our proposed hybrid architecture. It operates via a two-stage **Filter-then-Personalize** process:
    1. **Semantic Filter:** The system identifies the top 20 most topically relevant questions across the dataset.
    2. **Behavioral Personalization:** Within this semantically safe subset, the system selects the top 5 questions nearest in distance within the behavioral manifold.

3.5 Evaluation and Pedagogical Constraint

Ensuring diagnostic precision requires pedagogical guardrails that prevent common failure modes such as false validation, shallow feedback, or premature solution revelation. We employed carefully engineered prompts that emphasize conceptual scaffolding and diagnostic specificity. All systems used identical prompt templates, ensuring that performance differences could be attributed solely to retrieval strategy rather than linguistic variation in generation.

Beyond prompt engineering, we investigated whether preference learning could offer complementary benefits. We fine-tuned Llama-3-8B-Instruct using Direct Preference Optimization (DPO) [12] with 600 contrastive pairs constructed from 120 anchor questions. Each anchor generated five preference pairs corresponding to distinct pedagogical boundaries: politeness bias (empty affirmation masking diagnostic gaps), shallow diagnosis (symptom labeling without mechanistic explanation), geometric imprecision (conflating structurally adjacent misconceptions), premature answer-giving (violating the Zone of Proximal Development), and technical opacity (exposing retrieval mechanics to learners). Training employed LoRA adapters [6] (r=16, α=32) over three epochs, explicitly teaching rejection boundaries through contrastive supervision. Rejected responses were systematically constructed to violate each pedagogical boundary: for politeness bias, rejected hints provided affirmation without diagnostic reasoning; for shallow diagnosis, they identified error symptoms without mechanistic explanation; for answer-giving, they revealed solutions rather than scaffolding discovery. Chosen responses demonstrated diagnostic precision aligned with each boundary constraint.

Comparative evaluation across all 122 test questions yielded a decisive result: the preference-aligned model achieved 2.96/5 quality, significantly trailing GPT-4o-mini within the MAG architecture (4.75/5[1], Δ = -1.79, $p < 0.001$). Across 122 pairwise comparisons, the DPO model never outperformed the baseline (0% win rate). This empirical outcome validates a central tenet of the Sighted-Frugal paradigm: architectural constraint applied to capable foundation models dominates explicit alignment of smaller alternatives. The performance gap suggests that model capacity and structured retrieval provide greater pedagogical leverage than parameter-efficient supervision signals, at least within the scale regime tested here.

Final evaluation employed triangulated measurement:

1. **Automated Scoring:** Hint quality was evaluated on a five-point diagnostic scale across 122 questions using GPT-4o as judge for System D and GPT-4o-mini as judge for baseline systems A, B, and C. Statistical robustness was ensured through 10,000 bootstrap resamples, yielding confidence intervals for all effect sizes.
2. **False Validation Detection:** A rule-based classifier flagged instances of inappropriate praise, where systems validated incorrect student reasoning. This provided an objective safety check independent of neural evaluation.
3. **Expert Audit:** Two mathematics educators with secondary teaching experience evaluated cluster interpretability (Semantic Opacity) and pedagogical utility of generated hints. Inter-rater reliability metrics (Cohen's κ, ICC) quantified alignment between computational and human assessments.

3.6 Reproducibility

This study is designed for full replication. We used publicly available datasets: the NeurIPS 2020 Education Challenge (15.8 million interactions) and Eedi Mining Misconceptions (expert-labeled questions). Our pipeline implements standard techniques with documented parameters: 50-component SVD for the behavioral manifold, spectral clustering with nearest-neighbors affinity (n=10, k=50), and Sentence-BERT embeddings (all-MiniLM-L6-v2, 384 dimensions). The MAG architecture performs two-stage retrieval (top-20 semantic, top-5 behavioral) with clearly specified similarity thresholds. We will release complete code, including SVD construction, clustering, RAG implementation, and evaluation protocols, via GitHub at `https://github.com/Rickygole/semantic-gap-educational-rag`.

4. RESULTS

4.1 The Semantic Gap: Quantifying Orthogonality

Our primary analysis across 1,869 expert-labeled questions confirms the existence of the **Semantic Gap**: a state of functional independence between how students fail and what a problem says. As shown in Figure 2, we observed a near-zero correlation between behavioral similarity and semantic similarity (Pearson $r \approx 0.000$, 95% CI $[-0.002, 0.001]$, $p = 0.861$; Spearman $\rho = -0.002$, $p = 0.004$).

[1]The MAG baseline score in this DPO comparison (4.75) is from a matched evaluation pass conducted alongside DPO scoring under identical judging conditions. The 4.78 reported in Table 1 is from the four-system tournament run. The 0.03 difference reflects run-to-run variance from generation at temperature 0.7 across separate API calls.

Figure 2: The Semantic Gap

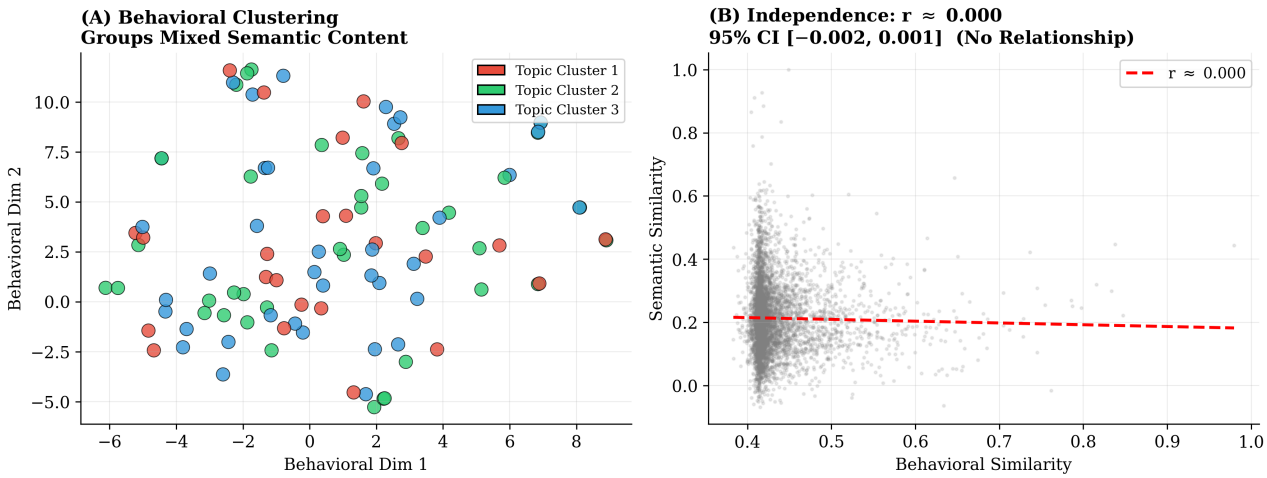


Figure 2: The Semantic Gap: Empirical independence between behavioral and semantic dimensions. Left: Behavioral clustering via t-SNE projection shows questions with mixed semantic content (colors represent different mathematical topics) scattered throughout behavioral space, demonstrating lack of semantic coherence within behavioral neighborhoods. Right: Near-zero correlation r ≈ 0.000, 95% CI [−0.002, 0.001] between behavioral similarity (Euclidean distance in 50D SVD space) and semantic similarity (Sentence-BERT cosine). The flat regression line demonstrates that linear projections of binary correctness are functionally independent to semantic structure required for content retrieval.

Permutation testing (1000 iterations) confirmed that this correlation falls entirely within the null distribution ($p = 0.872$). This near-zero correlation has direct retrieval consequences. Even weak positive correlation (r = 0.2 to 0.3) would allow behavioral signals to improve precision: most neighbors would fall within the correct semantic domain. Near-zero correlation means behavioral proximity provides no information about semantic relevance. A retrieved question is no more likely to be topically related than random selection. Worse, behavioral retrieval operates deterministically rather than randomly, systematically selecting off-topic problems with similar difficulty profiles. This transforms neutral noise into structured misdirection: students receive confident, detailed hints about the wrong mathematical domain.

We ruled out difficulty confounds through stratified analysis on 433 questions meeting high utility criteria (500+ interactions, 40%+ failure rate). Correlations remained near zero within difficulty tertiles: easy items (r = −0.02), moderate items (r = −0.01), and challenging items (r = 0.01). The Gap persists independent of psychometric difficulty. Stratified analysis confirmed that the Semantic Gap persists independently of psychometric item difficulty across all tertiles. Even within tightly constrained problem families such as triangle area calculations, behavioral clustering scattered questions across the manifold based on difficulty rather than geometric properties.

This suggests the Gap reflects a fundamental limitation of binary correctness signals rather than an artifact of aggregating heterogeneous content.

Interpretation: This visualization demonstrates that behavioral neighborhoods are semantically heterogeneous. A student's failure pattern on a Negative Numbers question may be behaviorally identical to their failure on a Fractions question. Knowing the *behavioral attractor* (how they failed) provides zero information regarding the *semantic content* (what they were failing at). This confirms our hypothesis that LLMs are behaviorally blind and SVD manifolds are semantically opaque.

4.2 Systemic Deficits and the Semantic Primacy Rescue

The practical consequences of this Gap are substantial for naive retrieval systems. As detailed in Table 1, System C (Behavioral Retrieval) achieved a pairwise win rate of only 12.3% falling below the 33.3% random baseline for a four-system pairwise tournament ($p < 0.001$). This failure is driven by *Topical Drift*: retrieving behaviorally similar but topically irrelevant context. Consider a diagnostic example from our evaluation set. When presented with a student struggling to subtract negative integers, System C retrieved a question about simplifying algebraic fractions. The only shared attribute was a 52 percent failure rate. The behavioral manifold correctly identified both as difficult items requiring procedural precision, yet this psychometric similarity provided zero pedagogical value.

The student received worked examples about fraction manipulation when they needed conceptual grounding in the number line. System D, by contrast, first filtered to arithmetic operations involving integers, then selected behaviorally similar problems within that semantic boundary. The resulting hint addressed sign confusion in subtraction, a structurally analogous trap within the correct mathematical domain. This contrast illustrates how Topical Drift trans-

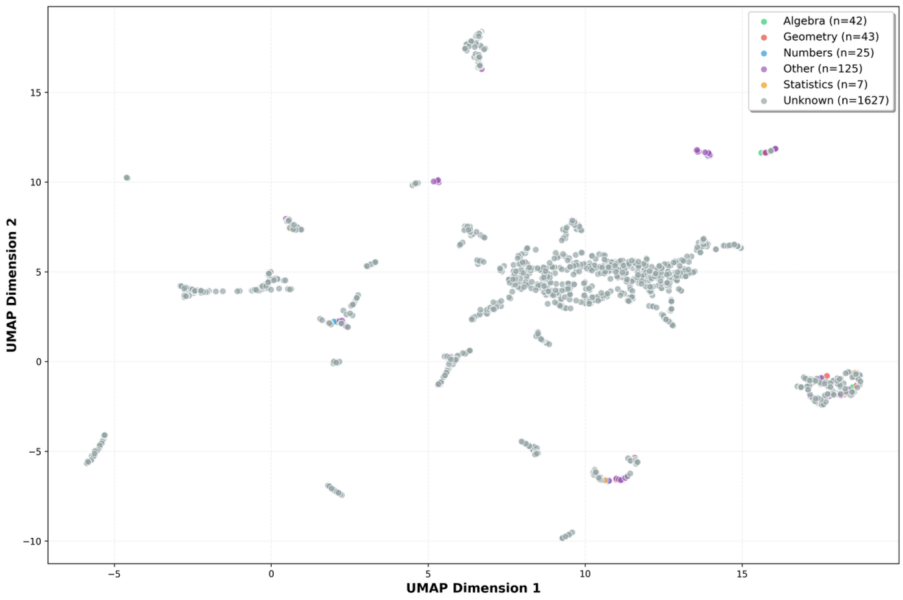


Figure 3: UMAP projection of behavioral space colored by semantic topic. Mathematical concepts (geometry, algebra, number operations, statistics) appear scattered across behavioral clusters rather than grouped, providing visual confirmation that behavioral neighborhoods lack semantic coherence. This dispersion pattern supports our quantitative finding of near-zero correlation between behavioral and semantic similarity.

Table 1: System Performance with Bootstrap Confidence Intervals (10,000 iterations)

Architecture	Quality (1-5)	Std. Dev.	Win Rate
System D: MAG (Hybrid)	**4.78 [4.69, 4.86]**	**0.49**	**49.2%**
System A: Zero-shot (GPT-4o-mini)	4.27 [4.06, 4.47]	1.18	32.5%
System B: Semantic RAG	3.94 [3.74, 4.15]	1.16	24.0%
System C: Behavioral RAG	3.36 [3.13, 3.60]	1.32	12.3%

[†]Win rate computed as pairwise tournament score: fraction of head-to-head comparisons each system wins against each of the three other systems (n=122 questions).

forms behavioral patterns from diagnostic signals into pedagogical noise.

However, our **Semantic Primacy** architecture (System D) successfully bridged this gap. By first filtering for topical relevance and then personalizing via the behavioral manifold, MAG achieved the highest quality (4.78/5) with the lowest variance ($\sigma = 0.49$). MAG demonstrates substantial improvements with effect sizes of d = 0.560 versus zero-shot ($p < 0.001$), d = 0.935 versus semantic RAG ($p < 0.001$), and d = 1.421 versus behavioral RAG ($p < 0.001$). These medium to large effects confirm that architectural integration of orthogonal signals yields measurable gains in diagnostic precision. Prompt sensitivity analysis across 30 test cases verified that behavioral RAG's deficit persists regardless of prompt engineering, isolating topical drift as the causal mechanism rather than suboptimal instruction design.

4.3 Semantic Opacity: The Human-in-the-Loop Barrier

Qualitative audit by two mathematics educators revealed that the behavioral manifold exhibits high **Semantic Opacity**. While the manifold identifies statistically robust clusters, human experts struggled to assign them consistent pedagogical labels (Cohen's $\kappa = 0.40$). Furthermore, inter-rater reliability on the quality of behaviorally-derived hints was poor ($ICC = -0.25$).

Consider Question 514 as a concrete illustration. Students solving a triangle area problem systematically substituted slant height for perpendicular height, a dimensional confusion sharing behavioral signatures with other problems involving oblique versus orthogonal measurements. Semantic RAG retrieved topically proximate geometry questions, yet failed to surface this specific error pattern. The behavioral manifold, however, identified structurally analogous traps across disparate topics. When constrained by semantic filtering, this behavioral signal elevated hint quality by 2.0 points on our diagnostic scale, revealing misconceptions that pure linguistic similarity missed entirely. The structural trap here is subtle. Students treating slant height as equivalent to perpendicular height are not making a calculation error but a dimensional category error. They fail to recognize that area formulas require measurements orthogonal to the base, not measurements along an oblique edge.

This confusion appears across multiple geometric contexts: using hypotenuse length in right triangle area calculations, applying diagonal measurements in rectangle problems, or substituting arc length for radius in circular sectors. Semantic RAG cannot detect this pattern because the questions use different vocabulary and geometric shapes. The

behavioral manifold, however, identifies the shared failure signature: students systematically choose the most visually salient measurement rather than the geometrically appropriate one. This is precisely the kind of latent misconception that benefits from computational detection paired with semantic constraint.

This finding suggests that student failure patterns contain latent traps that are mathematically detectable but pedagogically non-obvious. This necessitates the **computational stabilization** provided by our MAG architecture: rather than asking teachers to curate these clusters, we use semantic embeddings to ground the behavioral signal in human-readable content.

5. DISCUSSION AND LIMITATIONS

5.1 The Semantic Gap as a Structural Property

Our central finding identifies a fundamental structural reality: linear behavioral embeddings derived from binary correctness are functionally independent of semantic content. We argue that this **Semantic Gap** is not a temporary engineering hurdle that can be solved with more data or refined clustering; rather, it is an intrinsic property of what these representations capture.

Linear methods like SVD extract variance from correctness matrices that primarily reflect psychometric properties: item difficulty, student discrimination, and test-taking behaviors (e.g., fatigue or guessing). While these patterns are robust, they do not encode semantic relationships. A question on negative numbers and a question on fractions may share a behavioral attractor due to identical failure rates, but they possess zero conceptual overlap. This explains the substantial deficit of behavioral-only retrieval (12.3% pairwise win rate): the system lacks a mechanism to distinguish between a shared difficulty level and a shared concept. The 50-dimensional SVD space can be understood through the lens of classical test theory.

Each dimension likely corresponds to latent factors governing item response patterns: the first few dimensions capture global difficulty (analogous to the b-parameter in Item Response Theory), subsequent dimensions encode discrimination power (the a-parameter, measuring how sharply performance separates high and low ability students), and later dimensions reflect guessing patterns and test-taking strategies. However, these psychometric constructs are domain-agnostic. A dimension encoding requires multi-step reasoning applies equally to fraction division and quadratic factoring, collapsing conceptually distinct topics into a single behavioral signature. This is why behavioral distance preserves psychometric structure but destroys semantic boundaries.

The manifold encodes how hard questions are, not what they are about. This psychometric-semantic decoupling has profound implications for educational data mining. Traditional approaches to content recommendation implicitly assume that behavioral signals contain topic information that students who struggle similarly are struggling with similar concepts. Our findings falsify this assumption. The behavioral

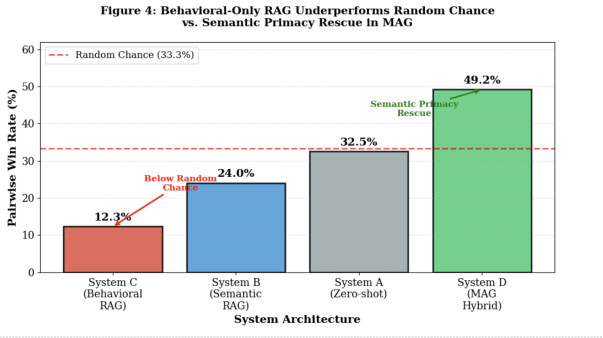


Figure 4: System performance across retrieval architectures. Behavioral-only retrieval (System C) achieves a 12.3 percent pairwise win rate, falling below the 33.3 percent random baseline (dashed line). MAG (System D) reaches 49.2 percent by enforcing semantic filtering before behavioral personalization, demonstrating that architectural constraint on orthogonal signals yields superior diagnostic quality. Win rate computed as pairwise tournament score across 122 evaluation questions.

manifold captures a parallel dimension of the learning experience: not conceptual understanding, but cognitive load, procedural complexity, and metacognitive demand. These are pedagogically valuable signals, but only when grounded within semantic boundaries.

5.2 Semantic Primacy: A Restorative Design Principle

The success of our hybrid MAG model (4.78/5) establishes **Semantic Primacy** as a core design principle for educational RAG. Our results prove that systems must strictly filter by semantic content before applying behavioral personalization. Attempting to learn semantics from interaction patterns alone is a category error that leads to *Topical Drift*.

This principle offers immediate practical value for Frugal AI deployment. It allows developers to leverage behavioral data where it is strongest identifying specific failure modes within a topic while relying on semantic embeddings for what they do best: grounding the conversation in the correct mathematical domain. The independence we discovered is thus not a bug, but a feature that can be exploited through deliberate architectural constraint. The computational efficiency of this approach has direct equity implications. GPT-4o-mini operates at roughly one-tenth the inference cost of GPT-4o while achieving superior diagnostic performance when augmented with behavioral context. This cost differential is not academic.

In under-resourced educational settings where per-student budgets constrain technology adoption, the difference between 10 cents and 1 cent per interaction determines whether AI tutoring is feasible at scale. Sighted-Frugal architectures that combine modest models with structured data enable equitable access to diagnostic AI. The alternative, pursuing diagnostic precision through ever-larger foundation models, concentrates educational innovation in wealthy institutions while leaving resource-constrained schools reliant on generic

chatbots. Our results demonstrate that architectural intelligence can substitute for parameter scale, democratizing access to pedagogically grounded AI.

5.3 Semantic Opacity and the Necessity of Computation

The observed **Semantic Opacity** ($ICC = -0.25$) indicates that the patterns captured by the behavioral manifold are human-unreadable. Because these patterns reflect psychometric variance rather than pedagogical categories, human educators cannot reliably recognize them through traditional pedagogical intuition.

This finding necessitates a shift in how we approach a Human-in-the-Loop design. Manual curation of behavioral clusters is likely infeasible. Instead, we must use computational stabilization like the MAG architecture to translate these opaque behavioral signals into semantically grounded, pedagogically useful hints. This bridges the gap between high-dimensional data mining and the practical needs of the classroom. This *Semantic Opacity* finding reframes the role of AI in educational systems. Rather than replacing teacher judgment, computational methods should function as translators between statistical patterns and pedagogical action.

Teachers cannot intuitively recognize that a triangle area problem shares behavioral signatures with cylinder volume calculations, yet the manifold detects this structural analogy through shared error patterns. The AI does not make pedagogical decisions but surfaces hidden relationships for human interpretation. This positions the system as an augmentation tool rather than an autonomous agent, a distinction critical for teacher trust and adoption. The design implication is clear: behavioral insights must always be presented within semantic containers that educators can evaluate and override.

5.4 Boundaries of the Linear Baseline

As a rigorous audit, this study is bounded by its focus on **Linear Methods** and **Binary Correctness**. While SVD is the industry standard for scalable systems, it represents the empirical baseline rather than the theoretical ceiling. It remains an open and vital question whether nonlinear techniques such as Variational Autoencoders (VAEs), Graph Neural Networks (GNNs), or Attention Mechanisms could bridge the Semantic Gap that linear methods cannot.

Furthermore, our use of binary correctness provides a "maximally impoverished" input. Future work should investigate if richer behavioral signals, such as response latencies, multi-attempt sequences, or option-choice analysis, might preserve semantic structure even under linear projection. Finally, while our dataset of 15.8 million interactions provides significant power, it is situated within the UK secondary mathematics curriculum. Cross-context replication is essential to determine if the Semantic Gap fluctuates across different cultures, age groups, or domains.

5.5 Methodological Reflection and Triangulation

We acknowledge the limitations inherent in using "LLM-as-a-Judge" to evaluate hint quality. System D was evaluated using GPT-4o while baseline systems were evaluated using GPT-4o-mini. We further acknowledge that the behavioral space (50D SVD) and semantic space (384D Sentence-BERT) use different underlying geometries, though both are transformed into comparable similarity measures before correlation; this choice does not affect our near-zero correlation finding, which holds across Pearson, Spearman, and permutation testing. To mitigate evaluation circularity, we employed a triangulated evaluation strategy, incorporating educator audits and rule-based classifiers to detect false validation. While these measures provide a robust signal for diagnostic specificity, definitive claims regarding student learning outcomes will require future longitudinal deployment studies in live classrooms.

By identifying the *Semantic Gap*, we provide a theoretical and practical framework for the next generation of educational AI. We have shown that the Sightedness required for effective tutoring does not come from larger models, but from architectures that respect the independence of behavior and semantics. By enforcing Semantic Primacy, we can move toward a Sighted-Frugal AI that is pedagogically precise, computationally efficient, and ethically grounded.

6. CONCLUSION

This paper identifies and quantifies the **Semantic Gap**: a state of empirical independence between student interaction patterns and semantic content. Through an audit of 15.8 million interactions, we demonstrated that while linear behavioral embeddings are functionally blind to topic, they capture robust psychometric properties that define how students struggle.

We proved that this Gap leads to systematic failure in naive RAG systems, which succumb to *Topical Drift* when retrieving based on behavioral signals alone. However, we have established a restorative design principle: **Semantic Primacy**. By enforcing a hybrid architecture that filters by topic before personalizing by behavior, our **Manifold-Augmented Generation (MAG)** system outperforms even high-compute zero-shot models in diagnostic specificity.

Our discovery of **Semantic Opacity** further suggests that behavioral data contains latent traps invisible to human pedagogical intuition, necessitating the computational stabilization provided by our MAG architecture. Ultimately, this work provides the first large-scale baseline for understanding the relationship between student behavior and content semantics. By respecting the independence structure of educational data, we can move toward a new generation of AI tutors that are not just linguistically fluent, but diagnostically sighted, computationally frugal, and pedagogically just.

7. FUTURE DIRECTIONS

While this study establishes a rigorous linear baseline, it opens several critical avenues for the evolution of Sighted-Frugal AI. We highlight four directions that we believe warrant systematic investigation by the research community.

Future research should investigate whether nonlinear archi-

tectures such as Variational Autoencoders (VAEs), Graph Neural Networks (GNNs), or Attention Mechanisms can bridge the Semantic Gap that linear methods cannot. Graph Neural Networks appear particularly promising given the relational structure of educational data. A message-passing architecture where nodes represent questions, edges encode co-occurrence in student trajectories, and node features combine behavioral statistics with semantic embeddings could learn joint representations. Specifically, a Graph Attention Network (GAT) with multi-head attention could weight connections by both behavioral similarity (questions students fail together) and semantic proximity (questions sharing misconception labels), potentially discovering hybrid distances that respect both dimensions. The question is whether such architectures can learn the Semantic Primacy constraint implicitly or whether architectural enforcement remains necessary.

We utilized binary correctness as a "maximally impoverished" input. Testing whether the inclusion of response latencies, multi-attempt sequences, and process traces can close the Gap even under linear projection remains a vital open question. Specific process signals merit investigation: response latency distributions (whether students pause before errors, suggesting confusion versus rapid incorrect attempts suggesting procedural errors), edit patterns (whether students erase and rewrite, indicating metacognitive awareness), attempt sequences (whether retry strategies differ across misconceptions), and navigation paths (whether students seek help resources differentially by trap type). These temporal and behavioral traces may encode semantic information absent from binary outcomes. For instance, students who pause extensively before answering geometry questions incorrectly may be struggling with visualization rather than calculation, a distinction invisible to correctness alone but potentially detectable through interaction timing.

Mapping the Semantic Gap across diverse domains, age groups, and cultural curricula will determine if this orthogonality is a universal property of educational data or a curriculum-specific phenomenon. Cross-context replication is essential to establish the generality of our findings beyond UK secondary mathematics.

Transitioning from automated quality metrics to longitudinal classroom studies is essential to measure how Semantic Primacy impacts actual skill acquisition and misconception resolution. A feasible deployment protocol would involve a randomized controlled trial in middle school algebra classrooms over one academic semester. Students in the treatment condition would receive MAG-generated hints when struggling with linear equations and systems of equations, while control students would receive standard computer-based practice with generic feedback. Pre and post assessments measuring both procedural fluency and conceptual understanding would quantify learning gains. Critically, the study would track misconception persistence: whether students who receive geometrically grounded hints demonstrate faster resolution of specific error patterns (e.g., sign errors in equation solving, confusion between solutions and equations) compared to controls. Implementation would require integration with existing learning management systems to ensure ecological validity and teacher buy-in.

Finally, larger educator cohorts from several distinct domains, particularly social sciences, humanities, and computer science education, would further probe *Semantic Opacity* and support the development of interfaces that help experts interpret and engage productively with behavioral manifolds.

8. REFERENCES

[1] T. Barnes and J. Stamper. Toward automatic hint generation for logic proof tutoring using historical student data. In *International Conference on Intelligent Tutoring Systems*, pages 373–382, 2008.

[2] S. Bugden and D. Ansari. Individual differences in children's mathematical competence are related to the intentional but not automatic processing of Arabic numerals. *Cognition*, 118(1):32–44, 2011.

[3] J. King, L. Burleigh, S. Woodhead, P. Kon, P. Baffour, S. Crossley, W. Reade, and M. Demkin. Eedi — Mining Misconceptions in Mathematics. Kaggle, 2024. `https://www.kaggle.com/competitions/eedi-mining-misconceptions-in-mathematics`.

[4] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[5] J. Hattie and H. Timperley. The power of feedback. *Review of Educational Research*, 77(1):81–112, 2007.

[6] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., Red Hook, NY, 2020.

[8] Y. Liu, C. Li, T. Zhang, M. Wang, Q. Zhu, J. Li, and H. Huang. Discerning minds or generic tutors? Evaluating instructional guidance capabilities in Socratic LLMs. *arXiv preprint arXiv:2508.06583*, 2025.

[9] P. Nesher. Towards an instructional theory: The role of student's misconceptions. *For the Learning of Mathematics*, 7(2):33–40, 1987.

[10] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., Red Hook, NY, 2022.

[11] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, volume 28, pages 505–513. Curran Associates, Inc., Red Hook, NY, 2015.

[12] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn. Direct preference optimization:

Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc., Red Hook, NY, 2023.

[13] N. Reimers and I. Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 3982–3992, 2019.

[14] S. Ritter, J. R. Anderson, K. R. Koedinger, and A. Corbett. Cognitive tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review*, 14(2):249–255, 2007.

[15] J. Stamper, K. Koedinger, R. S. Baker, A. Skogsholm, B. Leber, S. Demi, S. Yu, and D. Spencer. Managing the educational dataset lifecycle with DataShop. In *International Conference on Artificial Intelligence in Education*, pages 557–559, 2012.

[16] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[17] S. Wang, A. Ororbia, Z. Wu, K. Williams, I. Arroyo, B. Woolf, and N. Heffernan. Diagnostic questions: The NeurIPS 2020 education challenge. *arXiv preprint arXiv:2007.12061*, 2020.

[18] L. Zheng, W. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *arXiv preprint arXiv:2306.05685*, 2023.

Time as a geometric property of the latent space: An empirical study of Time-and-Order RoPE in Knowledge Tracing

William O. C. Silva
UFRPE
williamcostac@gmail.com

Filipe Dwan Pereira
UFRR
filipe.dwan@ufrr.br

Gabriel Alves
UFRPE
gabriel.alves@ufrpe.br

André Nascimento
UFRPE
andrecamara@gmail.com

Luiz Rodrigues
UTFPR
luizrodrigues@utfpr.edu.br

Mario A. P. Santos
Cogna Educação S.A.
mario.a.santos@cogna.com.br

Tyagi M. Lima
Cogna Educação S.A.
tyagi.lima@cogna.com.br

Rafael Ferreira Mello
UFRPE
CESAR School
rafael.mello@ufrpe.br

ABSTRACT

Knowledge Tracing (KT) models are essential for estimating student knowledge states, yet prominent attention-based architectures like SAINT and SAKT struggle with length extrapolation, often failing when sequence lengths exceed those seen during training. This limitation frequently arises from the use of absolute learnable positional embeddings. Furthermore, existing methods often treat interactions as uniformly spaced, failing to capture the continuous temporal dynamics of learning and forgetting. This paper investigates the incorporation of Time-and-Order rotation Positional Encoding (TO-RoPE), which treats both discrete interaction order and continuous real time as geometric properties of the latent space. By applying deterministic rotations to queries and keys, TO-RoPE allows for robust extrapolation to longer sequences while capturing heterogeneous temporal patterns. We conducted an empirical study across three datasets: ASSISTments2017, EdNet, and a Proprietary Dataset. The results demonstrate that replacing absolute embeddings with TO-RoPE significantly improves the AUC performance of SAINT and SAKT in extrapolated windows of up to 500 interactions. In particular, the SAINT + TOR configuration consistently outperformed the baselines, suggesting that explicit geometric modeling of continuous time is a viable approach to modeling long-range dependencies in digital education environments.

Keywords

Knowledge Tracing, Length Extrapolation, Time-aware Positional Encoding

Willian Costa, Filipe Dwan Pereira, Gabriel Alves, André Nascimento, Luiz Rodrigues, Mario A. P. Santos, Tyagi M. Lima, and Rafael Ferreira Mello. Time as a geometric property of the latent space: An empirical study of Time-and-Order RoPE in Knowledge Tracing. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 317–328. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039935

1. INTRODUCTION

The aim of personalized education at scale is fundamentally based on our ability to model learners' evolving cognitive states [30]. Knowledge Tracing (KT), the computational task of predicting student future performance based on historical interactions, serves as the analytical engine for personalized solutions, such as Intelligent Tutoring Systems (ITS) and Adaptive Systems [2, 16]. By accurately mapping a student's mastery over specific Knowledge Components (KCs), these systems can transition from static, one-size-fits-all curricula to adaptive environments that provide targeted feedback and optimized learning paths. Furthermore, educational platforms have begun to generate large volumes of data on student interactions, opening opportunities for analytical techniques to enhance the learning experience [23, 11, 3].

In this context, recent breakthroughs in Deep Learning, particularly with Transformer-based architectures such as SAINT and SAKT, have established new benchmarks in KT by using self-attention to capture complex dependencies across student histories [8, 9, 19]. However, these models share a structural dependency on absolute positional embeddings, which treat student interactions as a rigid, equidistant sequence. This approach fundamentally ignores the temporal reality of human learning, in which the time between interactions carries essential information about memory consolidation, interference from external activities, and the inevitable decay of knowledge [28]. Although previous efforts like AKT and LefoKT introduced temporal biases or handcrafted decay functions to mitigate this, they largely treat time as a discrete proxy rather than a continuous intrinsic property of the learning process [14, 5].

On the other hand, the reliance on absolute positions further restricts the models' capacity for length extrapolation. In typical research settings, KT models are trained and validated in truncated interaction windows [2]. However, a truly effective KT system is expected to maintain predictive stability throughout a student's entire academic tenure [13]. As a result, rather than the technical challenge of process-

ing longer sequences, length extrapolation is required for reliable long-term performance predictions [17]. That is, if the predictive accuracy of a model degrades simply because a student has exceeded the interaction count seen during the training phase, the system becomes an unreliable guide for high-stakes longitudinal outcomes, such as identifying early risks of course failure or certifying long-term retention [30].

To bridge this gap, this study investigates a new approach to positional and temporal encoding by integrating Time-and-Order rotation Positional Encoding (TO-RoPE) [27]. By encoding positions as relative rotations in a complex-valued latent space, TO-RoPE provides a theoretically grounded mechanism for both length extrapolation and continuous-time modeling. Unlike additive embeddings, which anchor the model to a fixed coordinate system, rotation encodings allow the attention mechanism to perceive the distance between interactions, both in terms of sequential order and in terms of actual time elapsed, as a relative geometric property. This potentially enables models trained on short interaction windows to generalize seamlessly to the much longer sequences characteristic of real-world educational data.

Therefore, this work evaluates whether substituting absolute embeddings with a dimension-separated TO-RoPE strategy can enhance the robustness of attention-based KT models. To this end, we developed new versions of the SAINT and SAKT architectures to isolate the effects of this geometric encoding and subject them to a rigorous train-short, test-long protocol. Thus, our main contribution lies in demonstrating how this approach affects the extrapolation of student mastery over extended periods and across varying temporal scales. By validating our findings across three diverse datasets, ranging from controlled mathematics tutoring (ASSISTments2017 [4]) to large-scale TOEIC preparation (EdNet [10]) and a proprietary higher-education dataset, we provide evidence for a more scalable and temporally aware framework for KT.

2. BACKGROUND

The KT problem is formally defined as a sequence-processing task aimed at modeling dependencies between past interactions of a student to predict future performance [30, 16]. Given an interaction history:

$$H_{t-1} = \{(q_1, a_1, \tau_1), \ldots, (q_{t-1}, a_{t-1}, \tau_{t-1})\}, \quad (1)$$

where q_i represents the question, $a_i \in \{0, 1\}$ the correctness, and τ_i the timestamp, KT models estimate the probability of correctness for a future question q_t:

$$P(a_t = 1 \mid q_t, H_{t-1}). \quad (2)$$

These estimates are fundamental for adaptive educational systems [2].

2.1 Attention-based Knowledge Tracing

The introduction of the self-attention mechanism [25] revolutionized KT by enabling modeling of long-range dependencies without the vanishing gradient limitations of recurrent neural networks (RNNs), such as deep knowledge tracking (DKT) [21].

Pandey et al. [19] pioneered this approach with Self-Attentive Knowledge Tracing (SAKT), an encoder-only architecture in which the current exercise serves as a query to retrieve relevant past interactions. Subsequently, Choi et al. [9] proposed SAINT, a Transformer-based encoder-decoder model that processes exercises and responses separately.

However, both SAKT and SAINT rely on learnable absolute positional embeddings to encode interaction order. In SAINT, for instance, position is handled by adding a learned vector $\mathbf{P}_i$ to the input:

$$\mathbf{E}_i = \mathbf{kc}_i + \mathbf{q}_i + \mathbf{p}_i. \quad (3)$$

A critical limitation of this approach is the lack of length extrapolation. Since $\mathbf{P}_i$ is learned for specific indices $i \in [0, L_{train}]$, these models fail to generalize when a student's history exceeds the duration of the training window L_{train}, which is a frequent scenario on lifelong learning platforms. Consequently, when $t > L_{train}$, the model lacks a defined positional representation, leading to performance degradation.

2.2 Time-aware Knowledge Tracing

To address the irregularity of learning sessions, several works have incorporated temporal information into attention mechanisms. Ghosh et al. [14] proposed Attentive Knowledge Tracing (AKT), which introduces a monotonic attention mechanism with exponential decay. This method modulates attention weights based on the temporal distance between interactions, reflecting the cognitive effect of forgetting.

More recently, Bai et al. [5] introduced LefoKT, which employs a *relative forgetting attention* mechanism. LefoKT adds a bias term $B_{i,j}$ to the attention logits, penalizing older interactions:

$$\text{Score}(i, j) = \frac{Q_i K_j^\top}{\sqrt{D_k}} + B_{i,j}. \quad (4)$$

Although AKT and LefoKT successfully incorporate temporal biases, they treat time primarily as a scalar penalty (a decay factor or additive bias) applied to the attention scores. This approach imposes an explicit monotonic decay, restricting their capacity to capture complex temporal dynamics, such as non-monotonic forgetting patterns and study periodicity. Furthermore, these models often still anchor their notion of time in discrete ordinal indices.

2.3 Geometric Positional Encodings

To overcome the limitations of absolute embeddings and scalar biases, recent advances in Large Language Models (LLMs) have explored encoding position directly into the geometry of the latent space. Rotating positional *embedding* (RoPE) [22] proposes that the inner product between a query and a key should depend only on their relative distance. Instead of adding a positional vector, RoPE rotates the affine-transformed word embeddings by an angle proportional to their position.

This rotation ensures that the attention score $QK^\top$ naturally captures relative distances, allowing the model to generalize to sequence lengths significantly larger than those seen during training (length extrapolation).

Wei et al. [27] extended this concept to irregular time series with *Time-and-Order RoPE* (TO-RoPE). Recognizing

that discrete order is insufficient for processes with irregular intervals, TO-RoPE treats both discrete order (i) and continuous time (τ) as sources of rotation. By interpolating these two dimensions, the model encodes the *distance* between interactions as a composite geometric property in a complex vector space.

To the best of our knowledge, the integration of Time-and-Order RoPE into the Knowledge Tracing domain remains unexplored. Although existing models like AKT and LefoKT address temporal dynamics through decay mechanisms, they do not leverage the geometric properties of the latent space to model continuous time. This work bridges this gap by replacing absolute positional embeddings in SAKT and SAINT with TO-RoPE, hypothesizing that treating both order and time as rotational components will unlock superior length extrapolation capabilities and more robust temporal modeling.

2.4 Research Questions

As detailed in Section 2, current Knowledge Tracing architectures face a fundamental structural dichotomy: they either rely on absolute positional embeddings that bind the model to fixed sequence lengths (as in SAINT and SAKT) or attempt to patch temporal dynamics through decay mechanisms or additive scalar biases (as in AKT and LefoKT). However, these methods treat time predominantly as an auxiliary bias rather than an intrinsic, continuous dimension of the latent space. In this work, we propose shifting the modeling paradigm from additive embeddings to *Time-and-Order rotation Positional Encoding* (TO-ROPE), postulating that encoding both sequence order and continuous elapsed time as geometric rotations offers a superior inductive bias. To validate this hypothesis, we investigate two critical dimensions of model performance: the capacity for length generalization and the interpretability of learned temporal patterns.

A primary failure of absolute positional embeddings, as described in Section 2.1, is the use of fixed training windows. Because position vectors are learned for specific indices ($i \in [0, L_{train}]$), standard models lack a defined representation of student histories that exceed the window size observed during training—a scenario inherent to lifelong learning platforms. We investigate whether replacing this fixed coordinate system with relative rotation encodings allows the model to decouple predictive capability from absolute positions, thereby maintaining stability over indefinitely long sequences. As such, the first research question is:

> **Research Question 1 (RQ1):**
> *To what extent does replacing absolute positional embeddings with Time-and-Order RoPE enable attention-based Knowledge Tracing models to extrapolate predictive performance to sequence lengths significantly exceeding those observed during training?*

Furthermore, although previous work explicitly enforces temporal forgetting via monotonic decay biases (Section 2.2), it remains unclear whether a model can naturally learn these dynamics solely from geometric constraints. By treating continuous time as a rotational dimension in the complex plane, the attention mechanism can dynamically discern the temporal distance between interactions. We aim to determine whether this geometric encoding leads to the spontaneous emergence of coherent attention distributions that reflect realistic cognitive phenomena—such as knowledge decay—without relying on handcrafted penalty functions.

> **Research Question 2 (RQ2):**
> *How does the incorporation of continuous time as a geometric property of the latent space influence the model's attention mechanisms, and does it yield interpretable temporal decay patterns distinct from those derived solely from discrete ordinal indices?*

3. METHOD

To address the proposed research questions and isolate the effects of positional encoding from other architectural variables, we utilize the established state-of-the-art models SAINT and SAKT as our baseline frameworks. These models were specifically selected for their benchmark performance in the literature [2] and their reliance on absolute positional embeddings. Unlike models such as AKT or LefoKT [14, 5], SAINT and SAKT do not incorporate explicit temporal decay mechanisms. By replacing their original positional mechanisms with TO-RoPE, we can directly observe the impact of geometric temporal encoding without interference from auxiliary decay components. The following section describes the model architecture.

3.1 Positional Encoding Adjustment

Standard Transformer-based KT models are based on additive absolute positional embeddings [2]. We modify the input formulation to remove these explicit position vectors (P_i), forcing the model to rely solely on the rotation mechanism for sequence information.

SAINT Reformulation. In the original SAINT architecture, the position is added as an absolute learnable vector in the response and exercise embeddings. We simplify the input representation for the encoder ($\mathbf{E}$) and the decoder ($\mathbf{R}$) as follows:

$$\mathbf{E}_i = \mathbf{kc}_i + \mathbf{q}_i \tag{5}$$

$$\mathbf{R}_i = \mathbf{a}_i \tag{6}$$

Where $\mathbf{kc}_i$, $\mathbf{q}_i$, $\mathbf{a}_i \in R^d$, represent, respectively, the embedding for the Knowledge Component, Question ID, and answer correctness at step i and d the dimension of the latent space. Note that the positional term $\mathbf{P}_i$ is explicitly removed.

SAKT Reformulation. Similarly for SAKT, we remove the positional embedding from the interaction matrix formulation. The activity matrix ($\mathbf{E}$) remains unchanged, while the interaction matrix ($\hat{\mathbf{M}}$) is defined as:

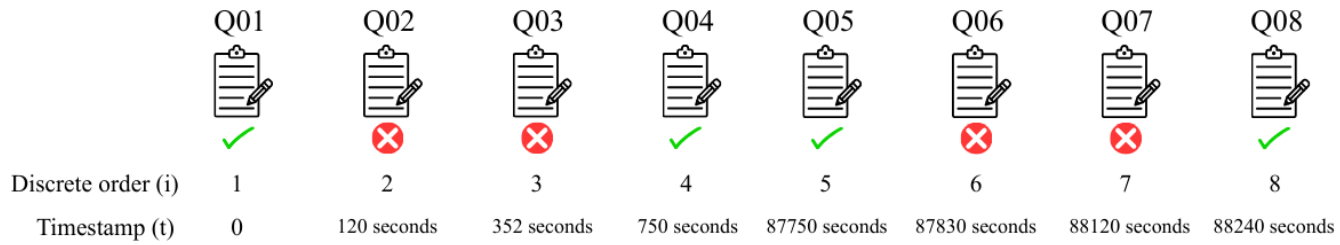


Figure 1: Illustration of the dual-coordinate system used in TO-RoPE. While the discrete order (i) treats interactions as uniformly spaced steps ($i \in \{1, \ldots, 8\}$), the continuous timestamp (t) captures the real-world irregularity of learning sessions, such as the large gap observed between Q04 and Q05 ($\Delta t \approx 24h$). TO-RoPE utilizes both coordinates to rotate the latent representations.

$$\hat{\mathbf{M}} = \begin{bmatrix} \mathbf{M}_{s_1} \\ \mathbf{M}_{s_2} \\ \vdots \\ \mathbf{M}_{s_i} \end{bmatrix}, \tag{7}$$

Where $\mathbf{M}_{s_i} \in R^{2E \times d}$ corresponds to the embedding projection of the student interaction tuple $(\mathbf{q}_i, \mathbf{a}_i)$.

With these modifications to the input embeddings, positional information is no longer additive; instead, it is injected directly into the attention mechanism via geometric rotations. We apply Time-and-Order RoPE (TO-RoPE) to rotate the query ($\mathbf{q}$) and key ($\mathbf{k}$) vectors conditioned on both the discrete sequence index (i) and the continuous real-time timestamp (t):

$$Q_{\text{rot}} = \text{TORoPE}(Q, i, t) \tag{8}$$

$$K_{\text{rot}} = \text{TORoPE}(K, i, t) \tag{9}$$

The attention score is subsequently computed using these rotated projections:

$$\text{Score}(i, j) = \frac{Q_{\text{rot}} K_{\text{rot}}^{\top}}{\sqrt{D_k}}, \tag{10}$$

By defining position as a rotation in the latent space, the inner product naturally encodes the relative distance between interactions in terms of both discrete order (Δi) and continuous elapsed time (Δt). In our implementation, we define the temporal coordinate t as the absolute cumulative time (in seconds) relative to the student's first interaction ($t_0 = 0$). Thus, while the model receives absolute timestamps as input, the rotation mechanism ensures that attention scores represent the relative time intervals between interactions. Figure 1 illustrates the alignment between these discrete order indices and the cumulative timestamps.

3.2 Separation Strategy

To effectively encode both discrete sequential order (i) and continuous elapsed time (t), we adopt the **Split-by-Dimension** strategy proposed by Wei et al. [27]. Wei et al. [27] suggested that integrating both signals into a single rotation plane can lead to destructive interference, in which conflicting phase shifts of time and order cancel each other out, destabilizing the attention mechanism. The Split-by-Dimension approach mitigates this by isolating the signals into orthogonal subspaces.

Formally, we partition the latent dimension d of each attention head into two disjoint sets of rotation planes: P (Order) and T (Time). We introduce a hyperparameter $\lambda \in [0, 1]$ to control the time capacity ratio. It defines the proportion of dimensions allocated to temporal encoding.

Let $\Psi_k \in \{0, 1\}$ be a gate function for the k-th rotation plane, determined by λ:

$$\Psi_k = \begin{cases} 1 & \text{if } k < (1-\lambda) \cdot \frac{d}{2} \quad \text{(Order Subspace)} \\ 0 & \text{if } k \geq (1-\lambda) \cdot \frac{d}{2} \quad \text{(Time Subspace)} \end{cases} \tag{11}$$

The rotation angle θ_k for a given interaction state (i, t_i) is then defined as:

$$\theta_k = \Psi_k \cdot (i \cdot \omega_k^p) + (1 - \Psi_k) \cdot (t_i \cdot \omega_k^t) \tag{12}$$

Here ω_k^p and ω_k^t are the angular frequencies derived from the geometric progression for order and time, respectively. $\lambda = 0.0$ implies a standard positional encoding (pure order), while $\lambda = 1.0$ results in a model driven purely by continuous time intervals. This mechanism allows the model to explicitly balance the inductive bias between sequence and time. Figure 2 illustrates this allocation.

3.3 Frequency Adjustment

To enable the model to capture both short-term immediate dependencies and long-term retention patterns, the angular frequencies of the rotation embeddings must be carefully calibrated. Following the methodology proposed by Wei et al. [27], we adopt a logarithmic frequency scale to ensure adequate resolution across different temporal scales.

The angular frequency ω_k for the k-th rotation plane within a specific subspace is defined as:

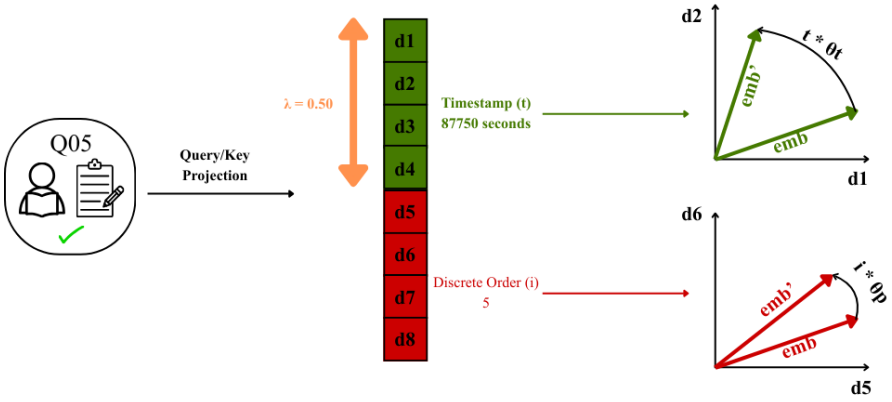


Figure 2: Illustration of the Split-by-Dimension strategy applied to the interaction $Q05$ with a capacity ratio $\lambda = 0.50$. The latent vector is partitioned into Time ($d1 - d4$) and Order ($d5 - d8$) subspaces. The diagram on the right shows how the first pair of dimensions in the time subspace ($d1, d2$) rotates under the timestamp ($t = 87550$), while the corresponding pair in the order subspace ($d5, d6$) rotates with the discrete index ($i = 5$). Note that this rotation process is repeated for all subsequent 2D sub-planes within each subspace (e.g., $d3, d4$ and $d7, d8$) using their respective frequencies from the geometric progression.

Table 1: Summary of the datasets used in our experiments, detailing the number of students with sequence lengths exceeding the extrapolation windows.

Dataset	Year(s)	Interactions	Students	Questions	KCs	Student Count with length $> L$				
						100	200	300	400	500
Proprietary	2023–2025	6,628,695	38,094	2,413	1,435	22,828	11,884	7,472	3,261	1,647
EdNet	2017–2019	95,293,926	784,309	13,169	188	112,558	76,718	59,108	48,087	40,260
ASSIST2017	2004–2006	942,816	1,709	3,162	102	1,647	1,390	1,139	930	750

$$\omega_k = \omega_{max} \left(\frac{\omega_{min}}{\omega_{max}} \right)^{\frac{k-1}{N-1}} \tag{13}$$

Where N represents the number of rotation planes (2D subspaces) allocated to the specific feature (time or order) and $k \in \{1, \ldots, N\}$ is the index of the plane. The boundary frequencies, ω_{max} and ω_{min}, are derived from the target wavelengths (periods) β. Where β_{min} and β_{max} represent the lower and upper bounds of the resolution (wavelengths), respectively.

$$\omega_{\max} = \frac{2\pi}{\beta_{\min}}, \qquad \omega_{\min} = \frac{2\pi}{\beta_{\max}} \tag{14}$$

4. EXPERIMENTS

4.1 Datasets

Previous works, such as Torralba and Efros [24] and Ben-David et al. [6], demonstrate that the intrinsic characteristics of each dataset, such as data distribution and collection biases, can significantly affect model performance. To mitigate performance gains arising from domain-specific overfitting, we selected three datasets with distinct characteristics for our evaluation.

The first is a **Proprietary Dataset**, a medium-sized corpus collected between 2023 and 2025 from response logs of students enrolled in various engineering courses within a Brazilian higher education institution. The second dataset is **EdNet** [10], a large-scale public benchmark containing interactions from Korean students collected between 2017 and 2019 for the TOEIC (Test of English for International Communication) exam. Finally, the third dataset is **ASSISTments 2017** [4], a small dataset collected between 2004 and 2006 that comprises data from middle school students interacting with an Intelligent Tutoring System (ITS).

The combined use of these datasets allows us to evaluate model behavior across diverse scenarios, ranging from controlled educational environments to large-scale, real-world contexts. Table 1 summarizes the descriptive statistics for each dataset.

4.2 Data Partitioning and data processing

Yadav and Shukla [29] demonstrate that k-fold cross-validation typically offers the most robust evaluation metric. However, the authors also highlight the need to balance performance generalization with computational cost, which scales linearly with k. Given the massive scale of the EdNet dataset and the requirement to perform grid searches for the TO-RoPE parameter λ, strictly adhering to k-fold cross-validation would render the experimental timeline infeasible. For instance, training a single variation of the SAINT architecture on EdNet requires approximately 13 hours on an NVIDIA A100 GPU.

Due to these computational constraints, we adopted a student-level holdout strategy for the proprietary and EdNet datasets, using an 80/10/10 split (Train / Validation / Test). The

partitioning was based on unique student identifiers, ensuring that students in the training set did not appear in the validation or test sets.

Regarding data processing, we addressed three critical scenarios:

- **Sequence Splitting:** To fully utilize extensive interaction histories, students with sequence lengths $L > L_{window}$ were partitioned into $N = L/L_{window}$ non-overlapping sub-sequences. This ensures that the model learns from the entire student's trajectory, rather than just the initial interactions.
- **Cold Start:** Following the findings of Bhattacharjee and Wayllace [7], students with fewer than 5 interactions were removed from the datasets, as models lack sufficient context to make reliable predictions for such sparse histories.
- **Padding:** For sequences (or sub-sequences) with lengths $L < L_{window}$, we appended padding tokens. These tokens are masked during training to prevent them from influencing the loss calculation.

In contrast, for the **ASSISTments 2017** dataset, due to its smaller sample size, we adopted a nested evaluation strategy to ensure statistical robustness. First, we randomly selected 20% of the students as a test set. The remaining 80% of the data was used for **5-fold cross-validation** to train the model and tune hyperparameters. This separation ensures that the final performance metrics are evaluated on a pristine set of unseen students.

4.3 RQ1: Length Extrapolation Analysis

To answer RQ1, we evaluated the capability of TO-RoPE to generalize to sequence lengths significantly longer than those observed during training by adopting a *train-short, test-long* protocol. The training and validation sets were divided into fixed windows of 100 interactions per student, following the configuration utilized by Choi et al. [9]. Interactions were classified chronologically by timestamp. During the testing phase, extrapolation capability was evaluated on windows of increasing lengths: 100, 200, 300, 400, and 500 interactions.

It is crucial to emphasize that the choice of a restricted training window ($L = 100$) serves a dual purpose: it aligns with previous baselines and, more importantly, acts as a rigorous **stress test** for length generalization. Training on longer sequences (e.g., 500) and testing on similar lengths would fundamentally assess the model's *interpolation* capabilities. By strictly limiting training to short sequences and evaluating on extended horizons, we effectively isolate and quantify the model's capacity for real *length extrapolation* based on the learned geometric properties of the latent space.

Based on previous work [2, 20, 8], the Area Under the ROC Curve (AUC) was used as the primary performance metric. To ensure robustness, each model was executed with 5 different random seeds, and the results are reported as mean and standard deviation ($\mu \pm \sigma$). Due to a limited sample size ($N = 5$) imposed by computational constraints, statistical significance was assessed using the Wilcoxon signed-rank test. We compared the baseline models (SAKT, SAINT) against their TO-RoPE variants. This non-parametric test was selected because it does not assume normality and is suitable for comparing classifiers across multiple runs [12]. Since our research hypothesis investigates the superiority of TO-RoPE over baselines, a one-sided test was adopted.

Additionally, to quantify stability, we report the slope coefficient (α) from a linear regression of AUC values across different test window sizes. This metric indicates the trend of performance gain or degradation as the sequence length increases. For readability, the reported coefficients were scaled by a factor of 10000.

4.4 Analysis of Attention Dynamics

To assess RQ2, we investigated how the model interprets the temporal dimension, and to validate whether TO-RoPE effectively encodes continuous time as a geometric property, we conducted an empirical analysis of the attention weights. Our objective was to determine whether the incorporation of time ($\lambda > 0$) induces distinguishable attention patterns compared to purely ordinal baselines ($\lambda = 0$), specifically in terms of the modeling of knowledge decay.

We extracted the attention matrices $A \in \mathbb{R}^{L \times L}$ from the inferences of the test set. To identify global patterns, we computed the mean attention scores averaged across all heads and layers for distinct configurations of λ. These scores were then aggregated based on the time distance (Δt) between interactions i and j. This aggregation strategy allows us to observe attention decay across varying temporal scales—ranging from immediate responses (seconds) to long-term retention intervals (weeks or months)—and to verify whether the model spontaneously learns to penalize older information based on elapsed time.

4.5 Hyperparameter Configuration

To ensure fair comparisons and reproducibility, hyperparameter tuning was tailored to each dataset's scale. For large-scale datasets (*EdNet* and *Proprietary*), we adopted established configurations from the literature to manage computational costs. In contrast, for the smaller dataset *ASSISTments2017*, we performed a grid search to identify optimal settings.

Across all model architectures, batch sizes were determined by dataset scale to optimize throughput on the NVIDIA A100 GPU: we employed a batch size of **512** for the massive *EdNet* and **128** for both the *Proprietary* and *ASSISTments2017* datasets.

For the **SAINT** architecture (and its TO-RoPE variants), we utilized the configuration proposed by Choi et al. [9] for EdNet and Proprietary datasets: latent dimension $d = 512$, $h = 8$ attention heads, $L = 4$ layers, and dropout $p = 0.1$. For *ASSISTments2017*, we searched over $d \in \{64, 256, 512\}$, $h \in \{4, 8\}$, $L \in \{1, 2, 4\}$, and $p \in \{0.05, 0.10, 0.20\}$.

For **SAKT**, we followed the recommendations of Pandey et al. [19] for large datasets, setting $d = 200$ and $p = 0.10$. For *ASSISTments2017*, we tuned the latent dimension over $d \in \{50, 100, 200\}$.

For the AKT and LefoKT-AKT-NF baselines (with ALiBi), we strictly adhered to the hyperparameter settings de-

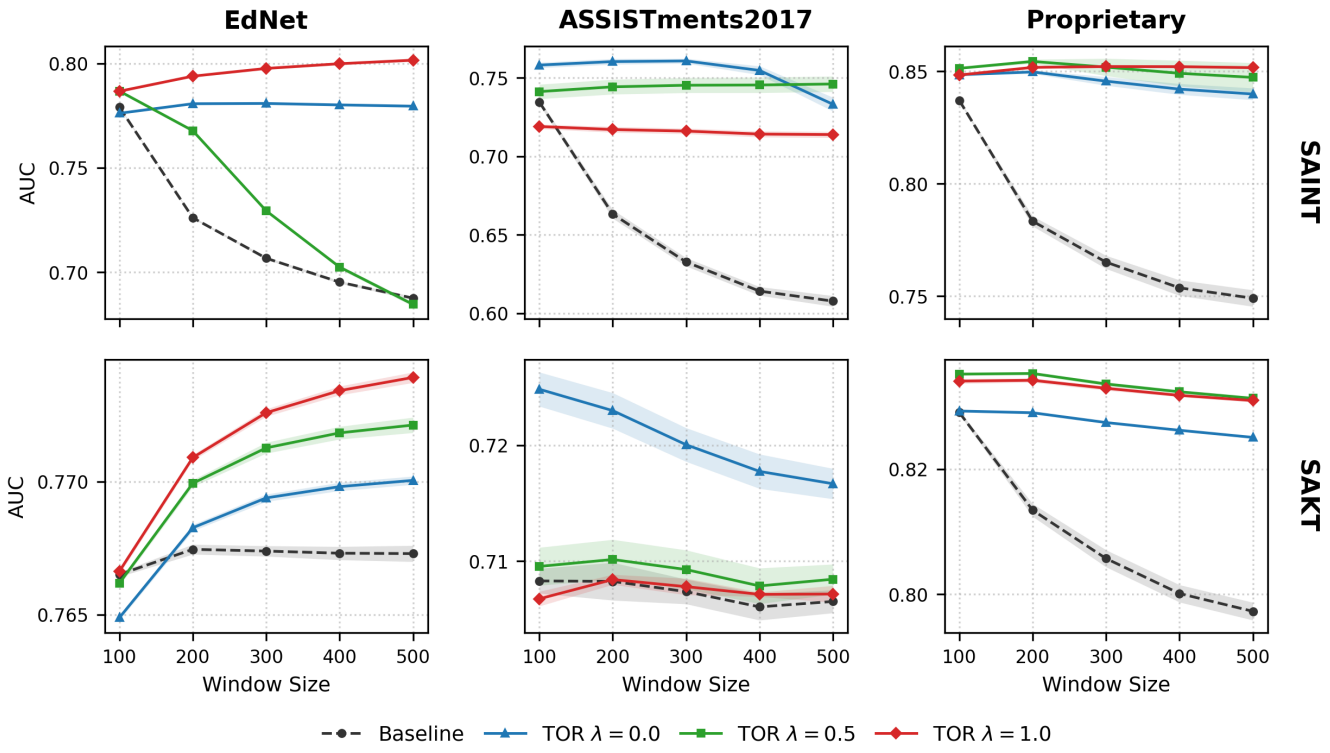


Figure 3: Average AUC per extrapolation window size over 5 independent runs

scribed in their respective original papers [14, 5] for large-scale datasets: $h = 8$ heads, latent dimension $d = 512$, and dropout $p = 0.10$. However, for *ASSISTments2017*, we performed a specific grid search over the latent dimension $d \in \{64, 256, 512\}$, number of heads $h \in \{4, 8\}$, and dropout rates $p \in \{0.05, 0.10, 0.20\}$.

All models were optimized using the Adam optimizer. We searched for the learning rate $l_r \in \{1e^{-5}, 5e^{-5}, 1e^{-4}, 5e^{-4}, 1e^{-3}\}$, finding that $l_r = 1e^{-4}$ is optimal for the majority configurations. Weights were initialized using the Xavier initialization scheme [15].

To avoid overfitting, we employed an *early stopping* mechanism with patience of 10 epochs, monitoring the AUC metric on the validation set with a minimum improvement delta of 10^{-4}. The maximum training duration was set at 50 epochs for SAINT-based models and 100 epochs for SAKT, AKT, and LefoKT.

Finally, for the **Time Subspace**, we set $\beta_{min} = 1$ second across all datasets to capture fine-grained interactions. The upper bound β_{max} is set to the nearest power of 10 exceeding the maximum timestamp observed in each dataset, ensuring that the rotation mechanism covers the entire temporal span without aliasing. Table 2 details these dataset-specific bounds.

For the **Order Subspace**, we fix $\beta_{min} = 1$ and $\beta_{max} = 500$. The latter corresponds to the maximum sequence length evaluated during our length extrapolation experiments, ensuring distinct positional signatures throughout the extended window.

Table 2: Frequency resolution bounds (β_{max}) for the temporal subspace across datasets.

Dataset	Max Time Observed (s)	$\beta_{\max}$ (s)
Proprietary	$\approx 7.80 \times 10^7$	10^8
EdNet	$\approx 8.17 \times 10^7$	10^8
ASSIST2017	$\approx 5.02 \times 10^7$	10^8

5. RESULTS

5.1 RQ1: Length Extrapolation

The initial results present the length extrapolation capabilities of the TO-RoPE models compared to baselines, with performance trajectories visualized in Figure 3 and quantitative metrics detailed in Table 3.

In the **EdNet** dataset, the SAINT + TO-RoPE configuration ($\lambda = 1.00$) achieved the highest AUC values across all test windows, distinguishing itself by a growing performance trend as the sequence extended ($10000 \cdot \alpha = +0.35792$). In contrast, both the SAINT baseline and the variation with $\lambda = 0.50$ exhibited a performance collapse, with sharp AUC drops starting from the first extrapolation window. Regarding SAKT, the time-focused variation ($\lambda = 1.00$) led across all window lengths; however, the gains were significantly less expressive than those observed in SAINT. Variations with $\lambda = 0.50$ and $\lambda = 0.00$ demonstrated intermediate performance, while the baseline remained stable but with AUC scores inferior to all rotation-based variations from 200 in-

Table 3: Length extrapolation performance (AUC) and curve slope (α) results

EdNet						
Models	100	200	300	400	500	$10000 \cdot \alpha$
AKT	0.7706 ± 0.0003	0.7761 ± 0.0004	0.7784 ± 0.0006	0.7794 ± 0.0005	0.7802 ± 0.0006	+0.22500
LefoKT-AKT-NF + ALiBi	0.7701 ± 0.0004	0.7740 ± 0.0005	0.7752 ± 0.0005	0.7757 ± 0.0006	0.7760 ± 0.0006	+ 0.13499
SAKT (baseline)	0.7665 ± 0.0001	0.7675 ± 0.0002	0.7674 ± 0.0002	0.7673 ± 0.0003	0.7673 ± 0.0003	+0.01424
SAKT + TOR ($\lambda = 0.00$)	0.7649 ± 0.0001	0.7683 ± 0.0001*	0.7694 ± 0.0001*	0.7698 ± 0.0002*	0.7700 ± 0.0002*	+0.11869
SAKT + TOR ($\lambda = 0.50$)	0.7662 ± 0.0001	0.7699 ± 0.0001*	0.7713 ± 0.0002*	0.7718 ± 0.0002*	0.7721 ± 0.0003*	+0.13786
SAKT + TOR ($\lambda = 1.00$)	0.7666 ± 0.0001	0.7709 ± 0.0001*	0.7726 ± 0.0001*	0.7734 ± 0.0002*	0.7739 ± 0.0002*	+0.17094
SAINT (baseline)	0.7790 ± 0.0004	0.7261 ± 0.0004	0.7067 ± 0.0006	0.6952 ± 0.0006	0.6875 ± 0.0008	-2.13790
SAINT + TOR ($\lambda = 0.00$)	0.7761 ± 0.0004	0.7806 ± 0.0005*	0.7808 ± 0.0006*	0.7801 ± 0.0006*	0.7795 ± 0.0007*	+0.06258
SAINT + TOR ($\lambda = 0.50$)	0.7863 ± 0.0004*	0.7677 ± 0.0005*	0.7294 ± 0.0007*	0.7025 ± 0.0009*	0.6846 ± 0.0012	-2.69629
SAINT + TOR ($\lambda = 1.00$)	**0.7866 ± 0.0004***	**0.7939 ± 0.0004***	**0.7976 ± 0.0006***	**0.7999 ± 0.0006***	**0.8016 ± 0.0008***	**+0.35792**
ASSISTments2017						
Models	100	200	300	400	500	$10000 \cdot \alpha$
AKT	0.7382 ± 0.0017	0.7388 ± 0.0021	0.7359 ± 0.0023	0.7329 ± 0.0024	0.7312 ± 0.0027	-0.19915
LefoKT-AKT-NF + ALiBi	0.7466 ± 0.0011	0.7492 ± 0.0009	0.7502 ± 0.0009	0.7508 ± 0.0009	**0.7512 ± 0.0010**	+0.10822
SAKT (baseline)	0.7083 ± 0.0011	0.7083 ± 0.0016	0.7074 ± 0.0011	0.7061 ± 0.0011	0.7065 ± 0.0009	-0.05739
SAKT + TOR ($\lambda = 0.00$)	0.7248 ± 0.0015*	0.7230 ± 0.0015*	0.7200 ± 0.0015*	0.7177 ± 0.0015*	0.7167 ± 0.0013*	-0.21541
SAKT + TOR ($\lambda = 0.50$)	0.7095 ± 0.0016*	0.7101 ± 0.0017*	0.7093 ± 0.0016*	0.7079 ± 0.0015*	0.7084 ± 0.0013*	-0.04493
SAKT + TOR ($\lambda = 1.00$)	0.7068 ± 0.0006	0.7084 ± 0.0004*	0.7078 ± 0.0007*	0.7071 ± 0.0002*	0.7072 ± 0.0007*	-0.00451
SAINT (baseline)	0.7345 ± 0.0015	0.6632 ± 0.0027	0.6326 ± 0.0026	0.6141 ± 0.0027	0.6077 ± 0.0034	-3.02679
SAINT + TOR ($\lambda = 0.00$)	**0.7580 ± 0.0017***	**0.7603 ± 0.0018***	**0.7607 ± 0.0016***	**0.7546 ± 0.0023***	0.7331 ± 0.0043*	-0.55579
SAINT + TOR ($\lambda = 0.50$)	0.7412 ± 0.0046	0.7442 ± 0.0047*	0.7452 ± 0.0048*	0.7454 ± 0.0049*	0.7459 ± 0.0049*	**+0.10864**
SAINT + TOR ($\lambda = 1.00$)	0.7189 ± 0.0014	0.7169 ± 0.0019*	0.7160 ± 0.0017*	0.7141 ± 0.0019*	0.7137 ± 0.0019*	-0.13214
Proprietary Dataset						
Models	100	200	300	400	500	$10000 \cdot \alpha$
AKT	0.8353 ± 0.0006	0.8386 ± 0.0007	0.8383 ± 0.0008	0.8382 ± 0.0009	0.8374 ± 0.0009	+0.03897
LefoKT-AKT-NF + ALiBi	0.8359 ± 0.0002	0.8392 ± 0.0002	0.8399 ± 0.0002	0.8406 ± 0.0002	0.8407 ± 0.0002	**+0.10907**
SAKT (baseline)	0.82891 ± 0.0003	0.8134 ± 0.0010	0.8057 ± 0.0014	0.8001 ± 0.0014	0.7973 ± 0.0014	-0.76963
SAKT + TOR ($\lambda = 0.00$)	0.8293 ± 0.0001*	0.8290 ± 0.0002*	0.8275 ± 0.0002*	0.8262 ± 0.0001*	0.8250 ± 0.0002*	-0.11241
SAKT + TOR ($\lambda = 0.50$)	0.8352 ± 0.0002*	0.8353 ± 0.0002*	0.8336 ± 0.0003*	0.8323 ± 0.0003*	0.8313 ± 0.0003*	-0.10678
SAKT + TOR ($\lambda = 1.00$)	0.8341 ± 0.0004*	0.8342 ± 0.0003*	0.8329 ± 0.0003*	0.8318 ± 0.0002*	0.8310 ± 0.0002*	-0.08638
SAINT (baseline)	0.8371 ± 0.0005	0.7834 ± 0.0022	0.7651 ± 0.0029	0.7537 ± 0.0034	0.7491 ± 0.0037	-2.05643
SAINT + TOR ($\lambda = 0.00$)	0.8485 ± 0.0003*	0.8497 ± 0.0008*	0.8456 ± 0.0019*	0.8421 ± 0.0024*	0.8399 ± 0.0026*	-0.24775
SAINT + TOR ($\lambda = 0.50$)	**0.8513 ± 0.0004***	**0.8543 ± 0.0009***	0.8518 ± 0.0038*	0.8491 ± 0.0057*	0.8472 ± 0.0065*	-0.13354
SAINT + TOR ($\lambda = 1.00$)	0.8484 ± 0.0005*	0.8518 ± 0.0005*	**0.8522 ± 0.0006***	**0.8521 ± 0.0006***	**0.8516 ± 0.0006***	+0.06858

Note: Values represent mean ± std. dev. over 5 independent runs. * denotes statistical significance ($p < 0.05$) compared to the baseline (one-sided Wilcoxon signed-rank test). **Bold** values indicate best performance per window. α: slope of linear regression (scaled by 10000 to ease legibility) over window sizes (stability metric).

teractions onward. Models such as AKT and LefoKT maintained positive slopes and outperformed SAKT variants, but did not reach the performance level of SAINT + TO-RoPE ($\lambda = 1.00$).

For **ASSISTments2017**, the SAINT + TO-RoPE variation with $\lambda = 0.00$ (purely order-based) presented the best performance in windows from 100 to 400 interactions. At the limit of 500 interactions, the LefoKT-AKT-NF + ALiBi model obtained the superior result. Contrary to trends in EdNet, the temporal variation ($\lambda = 1.00$) in SAINT underperformed, yielding consistently lower results than the order-based variants. The SAINT baseline showed an acute performance decline in the extended windows. In the SAKT model, the variation $\lambda = 0.00$ was the most effective, leading both in absolute AUC values and stability metrics, whereas temporal variation led to performance degradation. The AKT outperformed all SAKT variants in this dataset, though it remained inferior to SAINT.

Results on the **Proprietary** dataset exhibited mixed behavior. The SAINT + TO-RoPE configuration with $\lambda = 0.50$ led in the initial windows of 100 and 200 interactions, while the purely time-based version ($\lambda = 1.00$) obtained the best results in the 300, 400, and 500 windows, also registering the highest stability (α) among SAINT-based models. In the case of SAKT, the hybrid model ($\lambda = 0.50$) was superior to the other variations in all the extensions tested. The baselines for both models (SAINT and SAKT) suffered significant performance drops. Finally, the AKT and LefoKT-AKT-NF + ALiBi models demonstrated competitiveness, maintaining positive stability slopes, with LefoKT reaching the highest overall stability recorded for this dataset ($10000 \cdot \alpha = +0.10907$).

Additionally, we computed the average ordinal distance ($\overline{\Delta O}$) within each bucket to understand the density of sessions. These statistics are summarized in Table 4.

5.2 RQ2: Temporal Attention Dynamics

Figure 4 illustrates the attention decay patterns as a function of the temporal distance between interactions. A prominent feature is observed in the EdNet dataset with the SAINT architecture. The temporal model ($\lambda = 1.00$) concentrates significantly higher attention weights on short-term interactions ($1s \leq \Delta t \leq 1h$) compared to the ordinal baseline. As the time interval extends beyond 1 hour, both curves con-

Table 4: Temporal distribution and average ordinal distance per time bucket across datasets.

Dataset	First int.	Hour		Day		Week		Month		> 1 Month	
	%	%	$\overline{\Delta O}$	%	$\overline{\Delta O}$	%	$\overline{\Delta O}$	%	$\overline{\Delta O}$	%	$\overline{\Delta O}$
Proprietary	33.84%	28.44%	6.36	3.64%	12.91	9.84%	15.88	13.64%	22.42	10.60%	89.17
EdNet	16.44%	36.02%	7.71	14.25%	49.35	21.37%	106.07	7.92%	244.89	4.00%	379.29
ASSIST2017	7.55%	82.19%	3.24	0.21%	68.05	0.49%	64.19	3.87%	60.58	5.68%	158.86

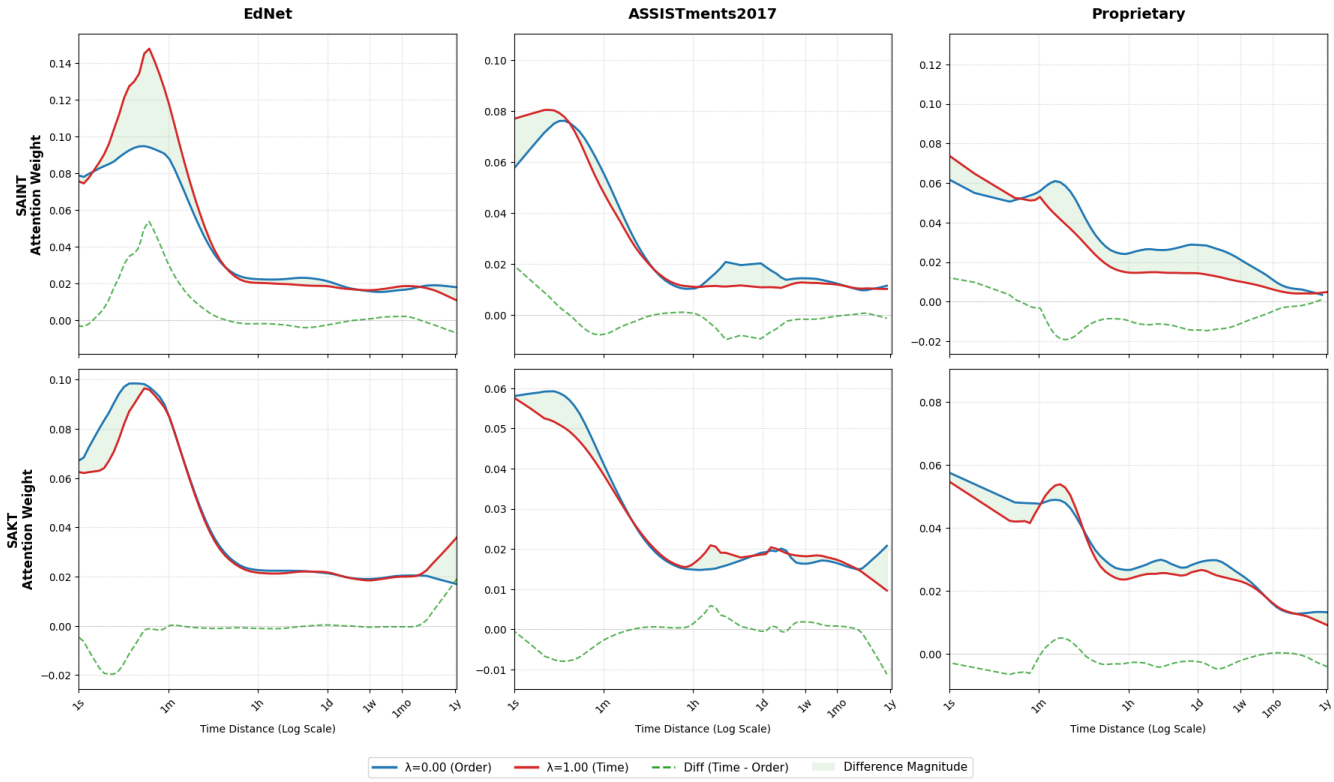


Figure 4: Average attention score per logarithmic time distance between interactions

verge to similar low-magnitude values. On the other hand, for the SAKT architecture on EdNet, a divergent pattern emerges: the ordinal model ($\lambda = 0.00$) assigns higher attention to immediate interactions ($1s \leq \Delta t \leq 1min$) than the temporal variant. For longer intervals ($1mo \leq \Delta t \leq 1y$), the difference between the two configurations stabilizes near zero.

Regarding ASSISTments2017, while there are differences between the ordinal and temporal attention distributions for both SAINT and SAKT in the short-to medium term, the magnitude of these deviations is less pronounced than those observed in the EdNet/SAINT scenario.

Finally, in the Proprietary dataset, the SAINT model with temporal encoding ($\lambda = 1.00$) exhibits a dual behavior: it assigns higher weights to very short-term interactions while systematically suppressing attention to distant past interactions more aggressively than the ordinal model. A similar trend is observed in SAKT, except that in the short term, the ordinal model predominates.

Despite these architecture-specific variations, a consistent global pattern is evident across all models and datasets: attention scores exhibit a natural decay as the temporal interval increases. This decay is notably sharper for EdNet and ASSISTments2017, whereas the Proprietary dataset displays a smoother, more gradual decline.

In an order-based model, such as SAINT + TO-RoPe ($\lambda = 0.00$), an item 20 steps ago is treated as a distant context. However, TO-RoPE ($\lambda = 1.00$) correctly identifies that these 20 items occurred on the same day, indicating a learning burst. This explains the attention patterns observed in Figure 5:

- **Order-based ($\lambda = 0.00$):** Attention is strictly diagonal, suffering from recency bias.
- **Time-based ($\lambda > 0$):** The attention mechanism diffuses off the diagonal, successfully attending to temporally clustered interactions that are ordinally distant.

This ability to compress bursts (treating them as simulta-

neous) while respecting long-term gaps allows the model to better capture complex temporal patterns, driving the superior extrapolation results seen in EdNet.

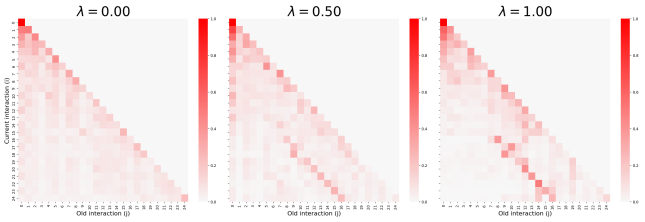


Figure 5: Average attention scores (per head) for distinct λ configurations on the EdNet dataset. Higher λ values allow the model to attend to ordinally distant but temporally close interactions (bursts).

The interplay between time and order is most nuanced in the Proprietary dataset, where the hybrid configuration ($\lambda = 0.50$) often yields superior stability. Unlike the self-paced nature of EdNet, higher education engineering courses follow a structured curriculum where sequential prerequisites are strictly enforced (favoring Order). However, students frequently engage in "cramming" sessions before exams, generating high-frequency bursts (favoring Time). A balanced λ allows the model to simultaneously respect the curricular sequence while geometrically compressing these intensive study sessions. This suggests that TO-RoPE is not a one-size-fits-all solution, but a flexible framework adaptable to the specific pedagogical design of the learning environment.

6. DISCUSSION

The inclusion of TO-RoPE significantly enhances length extrapolation capabilities, but the magnitude of these gains is highly sensitive to the dataset's characteristics. To deconstruct these performance variations, we analyzed the temporal distribution of the interactions. We aggregated interactions into time buckets based on the elapsed time since the student last engaged with the same KC.

Beyond specific dataset characteristics, the fundamental success of TO-RoPE in length extrapolation (RQ1) stems from the geometric property of translation invariance. Standard models fail because the absolute embeddings (P_i) learned for positions $i \in [0, L_{train}]$ are undefined or semantically meaningless for $i > L_{train}$. In contrast, TO-RoPE encodes the relative distance between interactions. For the attention mechanism, a temporal gap of $\Delta t = 1$ hours possesses the same geometric representation, whether it occurs at the beginning of a course or after thousands of interactions. This ensures that predictive stability is maintained indefinitely, effectively decoupling the model's context capacity from the fixed window sizes used during training.

The poor performance of the time-only variant ($\lambda = 1.00$) in ASSISTments2017 can be attributed to a low temporal signal-to-noise ratio. As shown in Table 4, nearly 90% of all interactions in this dataset occur within the first hour of contact with a KC. In such a highly dense short-term environment, the continuous timestamp provides negligible additional information beyond the discrete sequence order. Consequently, dedicating latent dimensions to time rotation ($\lambda > 0$) likely displaces valuable ordinal capacity without providing a discriminative signal, effectively acting as noise that degrades performance.

In contrast, EdNet exhibits a heavy-tailed distribution with a significant interaction volume spread across Days (14.25%), Weeks (21.37%), and even Months (7.92%). The table reveals a discrepancy between ordinal and temporal proximity: interactions within the same "Day" or "Week" bucket often have large ordinal distances ($\overline{\Delta O} \approx 49$ and 106, respectively).

Based on the findings regarding RQ2, incorporating continuous time as a geometric property alters the model's attention dynamics by decoupling ordinal distance from temporal distance. The empirical analysis, particularly on the EdNet dataset, reveals that purely order-based models exhibit a strict diagonal focus driven by recency bias—treating a prerequisite 20 steps ago as a distant context regardless of how quickly those interactions occurred. In contrast, the TO-ROPE mechanism effectively compresses rapid-fire bursts of activity. By geometrically recognizing that high-frequency sequences occur within a negligible time window despite their ordinal length, the model spontaneously diffuses attention across these clusters. This confirms that geometric encoding enables the attention mechanism to filter out noise from high-volume interaction bursts and to perceive the cognitive density of a learning session, rather than simply counting steps.

Furthermore, the study demonstrates that interpretable knowledge decay patterns emerge naturally from these geometric rotations, without the need for the handcrafted, monotonic penalty functions used in prior architectures such as AKT or LefoKT. While the attention weights exhibit a spontaneous forgetting curve as the temporal gap increases, the rotation mechanism preserves the norm of the latent vectors rather than imposing bias. This structural property enables the model to attenuate outdated information while still retaining the capacity to revisit and address distant but semantically critical prerequisites when vector alignment is strong. This finding suggests that geometric encoding offers a superior inductive bias, balancing the natural tendency to forget with the ability to recall long-term dependencies when the specific learning context demands it.

This research advances KT research by establishing that student learning trajectories are better modeled as continuous geometric paths in a latent space than as rigid, discrete sequences. The study shows that critical cognitive phenomena, such as memory decay and the spacing effect, can emerge from the relative rotation of the query and key vectors without the need for handcrafted penalty functions or additive biases. This shift challenges the prevailing reliance on absolute positional embeddings by proposing a unified framework that treats discrete order and continuous time as orthogonal yet complementary geometric dimensions.

The practical implications of this study extend to both the computational scalability of lifelong learning systems and the quality of personalized educational interventions. From a systems perspective, the findings demonstrate that models can be trained on efficient, truncated interaction windows (e.g., 100 steps) while maintaining predictive stability over substantially longer student histories (up to 500 steps). This decoupling of training context from inference capac-

ity prevents the performance collapse typical of standard Transformer architectures, allowing platforms to deploy robust models without incurring prohibitive computational costs. Consequently, this framework enables more stable performance predictions for learners with extensive interaction histories, which is essential for reliably tracking student progress across an entire course or even years of study. Furthermore, by accurately capturing natural memory decay, TO-RoPE empowers adaptive systems to deliver smarter, timely interventions, such as scheduling targeted reviews or spaced repetitions based on a student's probability to forget a concept.

7. LIMITATIONS AND FUTURE WORKS

Despite not changing the time complexity of the main attention mechanism, incorporating TO-RoPE adds a computational overhead when searching for the optimal time-capacity ratio hyperparameter (λ). Additionally, the model's effectiveness depends on the dataset's temporal signal-to-noise ratio. In environments with low temporal variance, such as ASSISTments2017, allocating dimensions to time can degrade performance by displacing valuable ordinal information. To address these challenges, future research could use interpretability techniques, such as Attention Rollout, to analyze how individual heads specialize in capturing either short-term bursts or long-term decay.

In terms of baselines, we also recommend expanding the comparative analysis to other time-aware models, such as DKT-Forget [18], Deep Graph Memory Network (DGMN) [1], Long Sequence Temporal Knowledge Tracing (LSTKT) [13], and HawkesKT [26], to evaluate how well TO-RoPE performs compared to these distinct approaches. Furthermore, this framework could be expanded to diverse educational domains, such as Law and Humanities, to investigate how different knowledge structures interact with geometric temporal properties.

8. ACKNOWLEDGMENTS

This research was partially supported by the Brazilian National Council for Scientific and Technological Development - CNPq (301415/2025-0).

9. REFERENCES

[1] G. Abdelrahman and Q. Wang. Deep graph memory networks for forgetting-robust knowledge tracing. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7844–7855, 2022.

[2] G. Abdelrahman, Q. Wang, and B. Nunes. Knowledge tracing: A survey. *ACM Comput. Surv.*, 55(11), Feb. 2023.

[3] I. Ahmad, S. Sharma, R. Singh, A. Gehlot, N. Priyadarshi, and B. Twala. Mooc 5.0: A roadmap to the future of learning. *Sustainability*, 14(18), 2022.

[4] ASSISTments. Assistments data mining competition 2017 dataset. https://sites.google.com/view/assistmentsdatamining/dataset, 2017. Accessed: [20/01/2026].

[5] Y. Bai, X. Li, Z. Liu, Y. Huang, M. Tian, and W. Luo. Rethinking and improving student learning and forgetting processes for attention based knowledge tracing models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(27):27822–27830, Apr. 2025.

[6] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[7] I. Bhattacharjee and C. Wayllace. Cold start problem: An experimental study of knowledge tracing models with new students. In *International Conference on Artificial Intelligence in Education*, pages 425–432. Springer, 2025.

[8] M. Cao, P. I. P. Jr., W. Chu, and L. Zhang. Integrating attentional factors and spacing in logistic knowledge tracing models to explore the impact of train-ing sequences on category learning. pages 526–532, July 2024.

[9] Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, and J. Heo. Towards an appropriate query, key, and value computation for knowledge tracing. In *Proceedings of the Seventh ACM Conference on Learning @ Scale*, L@S '20, page 341–344, New York, NY, USA, 2020. Association for Computing Machinery.

[10] Y. Choi, Y. Lee, D. Shin, J. Cho, S. Park, S. Lee, J. Baek, C. Bae, B. Kim, and J. Heo. Ednet: A large-scale hierarchical dataset in education. In *International conference on artificial intelligence in education*, pages 69–73. Springer, 2020.

[11] Y. Cui, Z. Ma, L. Wang, A. Yang, Q. Liu, S. Kong, and H. Wang. A survey on big data-enabled innovative online education systems during the covid-19 pandemic. *Journal of Innovation Knowledge*, 8(1):100295, 2023.

[12] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006.

[13] A. Gao and Z. Liu. Long sequence temporal knowledge tracing for student performance prediction via integrating lstm and informer. *PLoS One*, 20(9):e0330433, 2025.

[14] A. Ghosh, N. Heffernan, and A. S. Lan. Context-aware attentive knowledge tracing, 2020.

[15] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[16] K. Hong, K. Karbasi, and G. Pottie. Extracting causal relations in deep knowledge tracing. pages 586–591, July 2025.

[17] X. Li, Y. Bai, T. Guo, Z. Liu, Y. Huang, X. Zhao, F. Xia, W. Luo, and J. Weng. Enhancing length generalization for attention based knowledge tracing models with linear biases. In *Proceedings of the thirty-third international joint conference on artificial intelligence (IJCAI-24)*, pages 5918–5926, 2024.

[18] K. Nagatani, Q. Zhang, M. Sato, Y.-Y. Chen, F. Chen, and T. Ohkuma. Augmenting knowledge tracing by considering forgetting behavior. In *The world wide web conference*, pages 3101–3107, 2019.

[19] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing, 2019.

[20] M. Pankiewicz, Y. Shi, and R. S. Baker. srcml-dkt: Enhancing deep knowledge tracing with robust code representations from srcml. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and

L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 541–548. International Educational Data Mining Society, July 2025.

[21] C. Piech, J. Spencer, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing, 2015.

[22] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[23] A. C. Topuz, E. Saka, Ö. F. Fatsa, et al. Emerging trends of online assessment systems in the emergency remote teaching period. *Smart Learning Environments*, 9(1):17, 2022.

[24] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[26] C. Wang, W. Ma, M. Zhang, C. Lv, F. Wan, H. Lin, T. Tang, Y. Liu, and S. Ma. Temporal cross-effects in knowledge tracing. In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 517–525, 2021.

[27] X. Wei, J. Wu, D. Yi, R. Shirkavand, and M. Gong. Rotate both ways: Time-and-order rope for generative recommendation, 2025.

[28] Y. Wu, M. Arevalillo-Herráez, and D. Arnau. A study of the influence of the attention mechanism on knowledge tracing. In *International Conference on Artificial Intelligence in Education*, pages 462–469. Springer, 2025.

[29] S. Yadav and S. Shukla. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In *2016 IEEE 6th International conference on advanced computing (IACC)*, pages 78–83. IEEE, 2016.

[30] B. Yamkovenko, C. Hogg, M. Miller-Vedam, P. Grimaldi, and W. Wells. Practical evaluation of deep knowledge tracing models for use in learning platforms. pages 673–679, July 2025.

To What Extent Do Keystroke Dynamics Differentiate Correct from Incorrect Problem Solving

Oscar Blessed Deho
Centre for Change and Complexity in Learning (C3L)
Adelaide University
Australia
oscar.deho@adelaide.edu.au

Ryan S. Baker
Centre for Change and Complexity in Learning (C3L)
Adelaide University
Australia
ryan.baker@adelaide.edu.au

Digory Smith
Eedi
United Kingdom
digory.smith@eedi.com

Simon Woodhead
Eedi
United Kingdom
simon.woodhead@eedi.com

ABSTRACT

Understanding what behavioral patterns distinguish successful from unsuccessful problem solving could inform the design of more responsive learning environments. We investigated whether keystroke dynamics during typed mathematics responses reflect differences between correct and incorrect problem solving. Because problem format may influence student engagement and solution strategies, we also examined whether these keystroke patterns vary when problems include images versus text alone. Analyzing 21,938 responses from 570 students, we found that correct responses were characterized by slower, more variable typing with greater use of revision, while incorrect responses often showed patterns suggesting minimal engagement. Responses completed in under five seconds were almost never correct, suggesting a threshold below which meaningful problem solving appears not to occur. Machine learning models achieved modest but stable performance in distinguishing correct from incorrect responses based solely on typing patterns. Counterfactual analysis indicated that these models learned theoretically sensible relationships: the behavioral changes most frequently associated with shifting predictions from incorrect to correct aligned with increased deliberation and engagement. Notably, keystroke patterns remained largely stable across different problem formats, suggesting these behavioral signals may generalize across varied task presentations. These findings contribute to understanding how typing behavior during problem solving relates to correctness.

Keywords

Keystroke dynamics,Problem solving,Counterfactual analysis,Mathematics education

Oscar Deho, Ryan Baker, Digory Smith, and Simon Woodhead. To What Extent Do Keystroke Dynamics Differentiate Correct from Incorrect Problem Solving. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 329–339. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039808

1. INTRODUCTION

Mathematical problem solving is widely understood as a sequential, multi-stage cognitive process [27, 32]. One of the earliest and most influential articulations of this view dates back to Pólya's four phases—understand the problem, devise a plan, carry out the plan, and look back (evaluating the result) [27]. Each of these stages corresponds to distinct cognitive activities, ranging from initial sense-making to strategic planning, execution, and reflection. Subsequent research has further elaborated these staged models [32, 33, 34], emphasizing that effective problem solving depends not only on domain knowledge but also on strategic control over these phases. Consequently, examining how learners progress through these stages can shed light on why some responses are correct while others are not—for example, errors may arise from early misinterpretation or flawed planning, whereas success may reflect effective planning or careful review.

Because the problem-solving process is latent and multifaceted, researchers have devised various methods to make it observable. Think-aloud protocols have been a popular approach for a few decades: students are asked to verbalize their thoughts while solving problems, providing a running transcript of their reasoning [14]. Such protocols have yielded rich insights into strategies and misconceptions. Schoenfeld [32, 33] used think-aloud data to reveal, for example, that unsuccessful solvers often fail to plan or monitor their progress, whereas expert problem solvers exhibit effective self-regulation in real time. However, think-aloud methods have well-known limitations. They are intrusive—the act of continuously verbalizing can interfere with the natural flow of thinking—and thus may alter the very process under study [30]. Ericsson [13] notes that prompting students to articulate thoughts can slow cognitive processing or cause them to oversimplify decisions. Retrospective interviews (asking students to describe their solution after the task) lack these limitations but rely on memory and can thus be inaccurate [40].

Another window into problem-solving cognition is eye-tracking, which records where and for how long a solver looks at different parts of a problem [25, 24]. Eye movements provide

a fine-grained trace of attention and can indicate which information a student is focusing on or neglecting. For example, eye-tracking studies of algebra and word problems have shown that attention patterns differ between successful and unsuccessful solvers. Hegarty et al. [17] found that students who solved word problems correctly tended to form a coherent mental model of the situation, as evidenced by their gaze: they spent more time inspecting relevant relationships in the text/diagram, whereas poorer solvers often fixated on superficial keywords or numbers in a repetitive way. Similarly, Sušac et al. [36] used eye-tracking to compare novices and experts solving equations; experts had more efficient eye scans (fewer, more focused fixations), suggesting they quickly homed in on important elements, whereas novices' eyes wandered more, indicating uncertainty. Furthermore, Schindler et al. [31] demonstrated that eye-tracking can reveal the temporal sequence of solution steps a student takes—effectively mapping out their step-by-step approach—with greater precision than think-aloud or video analysis. However, eye-tracking too has practical limitations. It requires specialized hardware and controlled settings to achieve full precision [18] and scaling it to thousands of students in real classrooms is costly and logistically complex. Moreover, while less obtrusive than think-aloud [28], the presence of equipment and the laboratory setup may still not reflect a student's natural study environment.

Model tracing offers another method for studying students' problem-solving processes, when conducted using cognitive architectures like ACT-R [2]. The ACT-R architecture models cognition as a sequence of production rules and memory operations, enabling fine-grained simulations of reasoning. Koedinger and MacLaren [19] developed the Early Algebra Problem Solver (EAPS), which uses ACT-R to account for both informal and formal strategies students use when solving algebra problems and to predict common patterns of error based on cognitive processes. In model-tracing, each step of a student's process is evaluated against an expert-authored model for alignment to both correct and incorrect (buggy) reasoning [2]. However, this approach requires interfaces that explicitly reify each step of problem solving—meaning that students must type out or select every intermediate action [2]. Because this design can slow down the problem-solving experience and restrict the flexibility of students' strategies, many intelligent tutoring systems opt not to use this level of fine-grained model tracing [1].

In today's computer-based learning platforms, every student interaction can be logged [26, 11, 37]. This can (but often does not) include keystroke data—the sequence of keys pressed, along with timing information—when students type out responses. Such keystroke dynamics are an increasingly attractive source of cognitive process data in education because they are captured naturally, at scale, with minimal intrusion [15, 37]. Whenever students type an answer, how they type (pauses, deletions, revisions, etc.) is being recorded behind the scenes, without requiring the student to do anything extra. This yields a high-resolution trace of the writing or problem-solving process, time-stamped to the millisecond. In contrast to one-on-one think-aloud sessions or lab eye-tracking, keystroke logging is cheap and massively scalable—it can unobtrusively collect data from an entire online class or platform with hundreds of thousands of submissions [39]. It is also ecologically valid: students solve problems in the interface as they normally would (typing their response), so the data reflect authentic behavior in situ rather than under experimental observation [21].

Research in writing and programming has demonstrated the promise of keystroke analytics for inferring cognitive processes. For example, in essay writing tasks, keystroke logs have been used to measure pausing patterns, revision frequency, and text production bursts, which correlate with writing quality and strategy use [3, 38]. Several studies have linked certain keystroke features to performance outcomes. The number of revisions (edits and deletions) and the timing of those revisions have been tied to better performance in some contexts—e.g. writers who pause to revise frequently often produce more coherent texts [35, 42]. Keystroke-derived features such as interval timings (pauses between keystrokes or between sentences) can reflect the writer's cognitive effort or the points of difficulty [16]. Overall, prior work suggests that keystroke patterns carry informative signals about underlying cognitive processes [15].

Despite substantial prior work, gaps still remain in establishing how keystroke dynamics map to students' problem-solving processes. Keystroke data are inherently low-level behavioral traces, and their cognitive interpretation is often ambiguous: a prolonged pause may reflect difficulty, strategic planning, or disengagement, while rapid keystroke bursts may indicate fluent execution or unproductive guessing. Prior research has attempted to disambiguate meaning by extracting extensive sets of keystroke features and examining their associations with performance and other indicators [16]. However, findings across studies show limited convergence, in part because of substantial variability in the features selected for analysis [21, 35, 11]. Moreover, Conijn et al. [11] show that relationships between keystroke features and performance outcomes are context-dependent. As a result, further empirical work is needed to examine whether keystroke dynamics reliably differentiate between processes and even between correct and incorrect problem-solving outcomes.

Conijn et al. [11] note that these relationships can differ between tasks, yet there has been insufficient examination of how the problem context or format influences keystroke patterns. Other keystroke-based studies in education have looked at relatively uniform task types within-study (e.g. essay writing, short-answer responses in text form). However, in a domain like math, tasks can vary widely. For instance, an "open-ended" response might be prompted by pure text or accompanied by a diagram or image. In such cases, the presence of a diagram or image could plausibly alter students' engagement and problem-solving behaviour. Koedinger and Nathan [20] show that the format of a problem may shape students' solution approaches and performance. Furthermore, Chu et al. [9] demonstrate that student strategy is impacted by the presence of diagrams. However, to our knowledge, no prior work has systematically compared keystroke dynamics between math problems with diagrams or images and those presented in text-only format.

Finally, while prior studies have generally focused on correlating specific keystroke features with performance, they

rarely address the question of what changes in a student's keystroke behavior might be associated with a higher chance of success. Identifying minimal plausible shifts in behavior—a standard step in contemporary explainable artificial intelligence research [23]—could be very valuable.

This study thus aims to advance our understanding of fine-grained keystroke dynamics in mathematical problem solving along three dimensions: association with correctness, influence of problem format, and actionable differences between success and failure. We analyze an extensive dataset of 21,938 open-ended responses collected from a widely-used online math learning platform. Each response is accompanied by detailed keystroke logs capturing the student's typing and editing process. Our investigation centers on engineered features that quantify temporal aspects, revision behaviors, and the entropy or unpredictability of the keystroke sequence. We first examine the statistical associations between these keystroke features and answer correctness. Second, we explicitly compare image-based versus text-only questions. Third, we perform a counterfactual analysis using explainable artificial intelligence techniques to identify minimal behavioral changes associated with shifting an outcome from incorrect to correct. Aggregating such analyses over thousands of cases, we highlight which minimal changes would most frequently turn failure into success (according to the underlying machine-learned models being studied).

2. METHODS

2.1 Data

2.1.1 Source and Context

This study utilized de-identified data from an online mathematics learning platform that uses diagnostic assessment and adaptive learning pathways. The platform provides diagnostic quizzes, instructional videos, and practice activities intended to help students resolve misconceptions and improve their understanding. The learning sequence for a topic on the platform begins with five multiple-choice diagnostic questions. If a student answers a question incorrectly twice in a row, they enter a Learn phase to review worked examples and videos. Following this, the student proceeds to the Practice phase, which consists of a set of worksheet questions. Our study focuses exclusively on this Practice phase, as students provide open-ended, typed answers. These responses generate detailed keystroke logs. All personally identifiable information (PII) was removed prior to data access. The final dataset was constructed by merging several source files, including the raw keystroke logs, the students' final answers, and the worksheet question metadata. This merging process resulted in a final dataset of 21,938 unique student–question transactions from 570 students, which was used for all subsequent analyses.

2.1.2 Correctness Labelling

The worksheet data did not include pre-existing correctness labels. The dataset provided each student's final submitted response alongside an expert-authored reference answer. To label the data at scale, we developed an automated labeler using GPT-4o. The model was prompted, via a few-shot approach, to act as a mathematical evaluator, comparing each student's response to the expert-authored reference. The prompt instructed the model to determine if the two answers were mathematically equivalent, providing explicit rules to accept common variations. These rules included, for example, handling commutative expressions (e.g., $3x+2$ vs. $2+3x$), numerical approximations, differences in formatting (e.g., £1,057.50 vs. 1057.50), and the presence or absence of units. To validate this labeler, we first established a ground truth by having two human coders independently annotate a 200-response random sample, with high inter-rater reliability ($\kappa = 0.95$). In the small number of cases where the humans disagreed, social moderation was used to determine a final consensus label. The GPT-4o labeler, when tested against these human labels, demonstrated excellent performance, also achieving a Cohen's κ of 0.95, as well as a precision of 0.97 and a recall of 0.95. Following this successful validation, the GPT-4o labeler was used to provide correctness labels for the full dataset of 21,938 responses.

Table 1: Engineered keystroke and contextual features used in the analysis.

Feature	Definition
Keystroke Edit Distance	Levenshtein distance between the full raw keystroke sequence and the final submitted answer.
Seconds per Keystroke	Total time spent on the response divided by total number of keystrokes.
Keystroke Entropy	Shannon entropy of the keystroke sequence.
Correction Density	Number of correction keys (Backspace, Delete) divided by log-transformed total time.
Correction Proportion	Number of correction keys divided by total keystrokes.
Keystroke Efficiency Ratio	Final answer length divided by total keystrokes.
Average Seconds Between Keys	Mean inter-keystroke time (seconds) for a question attempt.
Variation in Seconds Between Keys	Standard deviation of inter-keystroke times (seconds) for a question attempt.
Fast Response	Indicator variable equal to 1 if total response time $\leq$ 5 seconds.
Slow Response	Indicator variable equal to 1 if total response time $>$ 300 seconds (5 minutes).
Pasted	Indicator variable equal to 1 if total keystrokes exceed total seconds taken.
Worksheet Question Has Image	Indicator variable equal to 1 if the question has an image attached.

2.1.3 Keystroke Data and Feature Engineering

For each student response to a worksheet question, a detailed keystroke log was produced. These logs contained the sequence of key presses along with associated metadata, capturing information on timing, editing behaviors, and other aspects of the input. An illustrative example of the data captured for a single student–question response is shown in Table 2. From these raw data, we engineered a set of

Table 2: Selected columns of a row in the raw data. HasImg = WorksheetQuestionHasImage, ⟨BS⟩ = Backspace, ←= Left Arrow

Question	HasImg	Reference Answer	Final Input	Time(sec)	Input Keystroke
Calculate 2548 − 362	False	2186	2186	167	← 218y ⟨BS⟩ 6

features to quantify the temporal (timing-related), corrective (editing-related), and compositional (content-related) aspects of the student's typing activity. These were analyzed alongside contextual data for each problem, including a feature indicating whether the question contained an image (see Figure 1 for an example). A summary of all engineered features, including their definitions is provided in Table 1. The distribution of the target label (correctness) across the final dataset was 72% incorrect and 28% correct, while the contextual image feature was nearly balanced, with 49% of transactions pertaining to questions with images and 51% to those without.

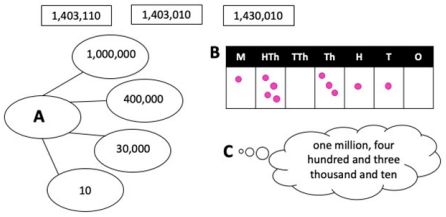


Figure 1: Example of an image-based worksheet question (QID 219: "Match each representation to its corresponding number in figures").

2.2 Analysis

As stated in Section 1, the goal of this study is to examine how fine-grained keystroke dynamics—such as typing latency, editing behaviors, and input variability—relate to answer correctness, and to identify how these process-level patterns, modulated by problem format (e.g., image vs. text), differentiate correct from incorrect problem-solving.

2.2.1 *Exploratory Analysis*

To investigate the relationships between student keystroke patterns, problem format, and answer correctness, we conducted a series of statistical analyses. First, to determine whether individual features were associated with the outcome, we tested each feature against the binary correctness label. We used the Mann–Whitney U test for continuous features (e.g., *variation in seconds between keys*) and Fisher's Exact Test for binary categorical features (e.g., *Worksheet Question Has Image*, *Pasted*), as appropriate for each data type. Second, we conducted a separate comparison to examine whether problem format itself influenced keystroke behavior. Specifically, we used the Mann–Whitney U test to compare continuous keystroke feature distributions based on problem format alone (image vs. text-only), to determine whether the presence of an image, in isolation, was associated with different keystroke patterns.

For all statistical tests, p-values from multiple comparisons were corrected using the Benjamini–Hochberg false discovery rate (FDR) procedure. For the Mann–Whitney U tests, effect sizes were computed using Cliff's Delta, which provides a non-parametric measure of group superiority. For the Fisher's Exact tests, we computed the odds ratio (OR) and its 95% confidence interval to quantify the strength of each association.

2.2.2 *Explanatory Analysis with Machine Learning Models*

Following the exploratory analysis, which established that individual features were significantly associated with correctness, the next step was to investigate their combined and interactive effects. We therefore turned to machine learning models, whose primary function is to learn complex, nonlinear, and interactive patterns from data—relationships that are not captured by the one-to-one statistical tests used in the exploratory phase. The goal of this stage was to construct and validate an explanatory model capable of capturing the underlying relationships between keystroke dynamics and correctness.

We examined several machine learning models, including Logistic Regression, Random Forest, CatBoost, and Support Vector Machine (SVM). Each model was tuned using hyperparameter optimization and evaluated using student-level five-fold cross-validation. This procedure ensured that all responses from a given student appeared exclusively in either the training or test set, thereby providing a robust estimate of generalization to unseen students.

The models were evaluated using a suite of metrics, including the area under the precision-recall curve (AUC-PR), the area under the receiver operating characteristic curve (AUC-ROC), macro-averaged F1-score, balanced accuracy (BACC) and the Matthews correlation coefficient (MCC). Within this explanatory framework, these metrics serve as goodness-of-fit measures, quantifying the extent to which the models successfully learned the underlying relationship between the features and the correctness label. We report the mean and standard deviation of each metric across the five folds to assess both average performance and model stability. Based on this evaluation, Random Forest emerged as the best-performing model and was selected for subsequent analysis.

2.2.3 *Counterfactual Analysis*

Using the validated Random Forest model, the final phase of the analysis focused on interpreting the specific behavioral patterns learned by the model. To achieve this, we employed Diverse Counterfactual Explanations (DiCE) [23]. DiCE is an optimization-based method that generates "what-if" scenarios by identifying the minimal and most plausible feature changes required to flip the model's prediction to a desired class (e.g., from incorrect to correct). Unlike feature-importance methods, which provide global rankings, DiCE yields concrete, instance-level explanations of the patterns

Table 3: Mann–Whitney U test results with Cliff's Delta effect sizes for continuous keystroke features by correctness.

Feature	$\text{Correct}_{\text{Mdn}}$	$\text{Incorrect}_{\text{Mdn}}$	Effect size (δ)	p-value
Average seconds between keys	0.85	0.46	0.35	< 0.001
Seconds per keystroke	1.25	2.33	0.31	< 0.001
Variation in seconds between keys	0.68	0.34	0.30	< 0.001
Keystroke entropy	2.00	2.00	0.13	< 0.001
Keystroke edit distance	1.00	1.00	0.11	< 0.001
Correction density	0.00	0.00	0.07	< 0.001
Correction proportion	0.00	0.00	0.06	< 0.001
Keystroke efficiency ratio	0.75	0.78	-0.06	< 0.001

the model associates with different outcomes. Importantly, it also generates diverse explanations, enabling examination of whether multiple behavioral profiles are associated with correctness.

A key methodological decision was to restrict analysis to verifiable ground-truth cases: true positives and true negatives. Because the model is imperfect, analyzing false positives or false negatives risks interpreting model-specific artifacts rather than stable relationships in the data. By focusing exclusively on cases that the model correctly classified, we ensure that the counterfactual analysis is grounded in the strongest and most reliable portion of the learned signal.

The counterfactual analysis was conducted within the five-fold cross-validation framework. For each fold, we first identified instances belonging to a verifiable group within the test set. For each instance, we generated three diverse counterfactuals, resulting in three distinct hypothetical scenarios per student. Each scenario represents a counterfactual setting in which the student's original feature values are minimally altered to flip the model's prediction. For example, for a true negative instance (incorrect and predicted incorrect), we generated three plausible counterfactuals in which the prediction changed from incorrect to correct.

To ensure plausibility, all counterfactuals were constrained to lie within an allowed range of feature values, defined as the 1st to 99th percentile of the training data for continuous features. We then aggregated the feature changes required to flip predictions. For continuous features, we computed the magnitude of change and analyzed directional proportions, defined as the percentage of changes corresponding to increases versus decreases. For categorical features, we analyzed the proportion of directional transitions (e.g., $0 \rightarrow 1$ or $1 \rightarrow 0$). Directional proportions were averaged across folds to produce a stable final set of results, identifying the feature changes most frequently associated with shifts in correctness.

3. RESULTS

3.1 Feature Association with Correctness

We first examined the relationship between the continuous keystroke features and correctness using the Mann–Whitney U test. The test established statistically significant differences across all eight features for correct versus incorrect responses, as shown in Table 3. Effect sizes (Cliff's Delta) ranged from 0.06 to 0.35, reflecting small to moderate differences across features. For nearly all latency, variability, and correction features, the positive effect sizes indicate that feature values tended to be higher for correct responses than for incorrect ones. The largest effects were observed for *average seconds between keys*, *seconds per keystroke*, and *variation in seconds between keys*. Specifically for these latency features, the effect size range corresponds to a 65% to 68% probability—using a standard derivation of Cliff's Delta—that a randomly selected correct response exhibits a higher feature value than an incorrect response.

Table 4: Fisher's exact test odds ratios (OR) for binary features vs. correctness.

Response:	OR [95% CI]	p-value
FastResponse	**0.03 [0.02, 0.05]**	< 0.001
Pasted	0.35 [0.31, 0.39]	< 0.001
SlowResponse	1.18 [0.96, 1.47]	0.163
WorksheetQuestionHasImage	0.71 [0.67, 0.76]	< 0.001

The Fisher's Exact Test results for the categorical features revealed several significant associations (see Table 4). The largest negative association was observed for *Fast Response* ($\text{OR} = 0.03$, $p < 0.001$). This categorical indicator flagged question transactions completed in five seconds or less and constituted 6% (1,209 transactions) of the total dataset. Questions answered between five seconds and five minutes served as a proxy for normal response time and were used as the comparison baseline. The probability of a fast response transaction being correct was only 1.41%, compared to the baseline correctness rate of 30%. This corresponds to a 95.3% decrease in the probability of a response being correct when completed within five seconds or less.

By contrast, *Slow Response*, defined as responses taking longer than five minutes, constituted 2% of the data (386 transactions) and did not differ statistically from the normal-time baseline ($\text{OR} = 1.18$, $p = 0.163$). Submitting a *Pasted* answer ($\text{OR} = 0.35$) and the presence of an image in the question (*Worksheet Question Has Image*, $\text{OR} = 0.71$) were also significantly associated with lower odds of correctness, though not as strongly as fast responses.

3.2 Impact of Problem Format on Keystroke Patterns

To examine whether problem format itself influenced keystroke patterns, we used the Mann–Whitney U test to compare continuous keystroke features across questions with images versus those without images (Table 5). The analysis revealed statistically significant, but negligible [22], differences in timing behaviors. Timing and variability features, including *average seconds between keys* ($\delta = -0.05$), *variation in seconds between keys* ($\delta = -0.04$), and *keystroke*

Table 5: Mann–Whitney U test comparing continuous keystroke features for questions with images versus those without image.

Feature	With $\text{Image}_{\text{Mdn}}$	No $\text{Image}_{\text{Mdn}}$	Effect Size (δ)	p-value
average seconds between keys	0.54	0.61	-0.05	< 0.001
variation in seconds between keys	0.39	0.44	-0.04	< 0.001
keystroke entropy	2.00	2.00	-0.03	< 0.001
seconds per keystroke	2.84	2.75	0.02	0.073
keystroke efficiency ratio	0.75	0.78	-0.02	0.022
keystroke edit distance	1.00	1.00	-0.01	0.430
correction density	0.00	0.00	-0.01	0.181
correction proportion	0.00	0.00	-0.01	0.261

Table 6: Model performance on predicting correctness from keystroke features. Values are mean ± standard deviation across student-level 5-fold cross-validation.

Model	F1	BACC	AUC	PR	MCC
Logistic Regression	0.44 ± 0.01	0.51 ± 0.00	0.65 ± 0.01	0.39 ± 0.02	0.05 ± 0.01
Random Forest	0.64 ± 0.01	0.69 ± 0.01	0.75 ± 0.00	0.48 ± 0.02	0.35 ± 0.01
CatBoost	0.64 ± 0.01	0.69 ± 0.00	0.75 ± 0.00	0.47 ± 0.02	0.34 ± 0.01
SVM	0.51 ± 0.01	0.53 ± 0.01	0.70 ± 0.01	0.42 ± 0.03	0.10 ± 0.03

entropy ($\delta = -0.03$), were all statistically significantly different between image-based and text-only questions ($p < 0.001$). Given the small effect sizes, these results indicate only slight shifts toward marginally faster and more regular typing on image-based questions—differences that are statistically detectable but practically inconsequential. In contrast, correction-related features ($\delta \approx -0.01$) were not statistically significant (e.g., *keystroke edit distance*, $p = 0.43$), suggesting that problem format did not meaningfully affect the frequency of observable typing errors.

3.3 Counterfactual Explanations of Keystroke Patterns and Correctness

The performance results for the machine-learned models, evaluated using student-level five-fold cross-validation, are presented in Table 6. Ensemble methods (Random Forest and CatBoost) substantially outperformed the simpler Logistic Regression and Support Vector Machine models across all evaluation metrics. The Random Forest classifier was selected for further analysis due to its *relatively* robust performance, achieving the highest or joint-highest score on every metric.

Given that the baseline for random guessing in this imbalanced dataset is the prevalence of the minority class (approximately 0.28 for the precision-recall AUC baseline), the Random Forest's mean precision-recall AUC of 0.48±0.02 indicates that the model learned a moderate, generalizable signal. The balanced accuracy of 0.69 further suggests reasonable discrimination between correct and incorrect responses, corroborated by a Matthews Correlation Coefficient of 0.35. This validated Random Forest model served as the basis for the subsequent counterfactual analysis.

We used DiCE to interpret the validated Random Forest model, identifying the keystroke pattern adjustments required to change verifiably incorrect answers (true negatives) into correct ones, as well as the adjustments required to change verifiably correct answers (true positives) into incorrect ones. For the true negatives, the analysis of continuous features (Table 7) consistently showed that the model required positive-direction adjustments (increase) across all features in order to flip an incorrect prediction to correct. This finding aligns with the correlational analysis, which showed that correct responses generally exhibited higher values for nearly all continuous keystroke features. The most frequently adjusted features were timing-related: *seconds per keystroke* (30.4% of all counterfactuals) and *average seconds between keys* (25.8%). The largest median adjustments included an increase of 34.1 units for seconds per keystroke and 8.3 units for average seconds between keys. Editing and variability features were also adjusted, with *keystroke entropy* modified in 9.31% of counterfactuals (median change = 1.67) and *keystroke edit distance* in 4.89% (median change = 26.0).

The analysis of categorical transitions (Table 8) revealed that the most frequent binary change required was transitioning *Fast Response* from 1 to 0, occurring in 5.69% of all counterfactuals. This directly corroborates the strong negative association observed in the exploratory analysis. Additionally, the question format feature *Worksheet Question Has Image* was required to change from present to absent in 2.37% of counterfactuals. Other categorical transitions occurred less frequently.

The analysis of true positives identified the minimal feature adjustments required to change a verifiably correct response into an incorrect one (Tables 9 and 10). These results largely mirror the true negative findings in reverse. For continuous features, the most frequent change was a decrease in *keystroke entropy* (20.20% of counterfactuals, median change = −1.74). For categorical features, the most frequent transition was changing *Fast Response* from 0 to 1, which occurred in 18.4% of counterfactuals. This indicates that shifting a response from non-fast (i.e., baseline) to fast was often sufficient for the model to flip a correct prediction to incorrect. Consistent with the true negative results, changing *Worksheet Question Has Image* from absent to present occurred in 1.86% of counterfactuals. Also, transitioning *Pasted* from 0 to 1 occurred in 2.67% of cases.

Table 7: Directional changes in continuous features required to turn answers that were incorrect into answers that become correct for true negatives (TN, 0→1 counterfactuals).

Feature	Proportion Increased(%) ↑	Proportion Decreased(%) ↓	Median ↑	Median ↓
average seconds between keys	**25.82**	0.15	8.31	-10.12
correction density	1.63	0.21	1.98	-1.46
correction proportion	1.11	0.64	0.20	-0.23
keystroke edit distance	4.89	0.20	26.00	-19.90
keystroke efficiency ratio	4.02	1.25	0.41	-0.34
keystroke entropy	9.31	0.62	1.67	-0.80
seconds per keystroke	**30.43**	0.14	34.05	-14.50
variation in seconds between keys	8.51	0.14	15.64	-22.45

Table 8: Categorical feature transitions required to turn answers that were incorrect into answers that become correct for true negatives (TN, 0→1 counterfactuals).

Feature	Transition	Proportion(%)
FastResponse	0→1	0.03
FastResponse	1→0	**5.69**
Pasted	0→1	0.54
Pasted	1→0	0.90
SlowResponse	0→1	0.97
SlowResponse	1→0	0.03
WorksheetQuestionHasImage	0→1	0.41
WorksheetQuestionHasImage	1→0	**2.37**

4. DISCUSSION

The central question motivating this study was whether keystroke dynamics captured unobtrusively during authentic mathematics problem solving can reliably differentiate between correct and incorrect responses. Our findings provide evidence that they can: across 21,938 open-ended responses, students who answered correctly exhibited systematically different typing behaviors than those who answered incorrectly. These differences emerged consistently across multiple analytical approaches (statistical comparisons, machine-learned models of the underlying relationships, and counterfactual interpretations of those learned patterns), suggesting that the temporal and behavioral traces left during response construction carry meaningful information about problem-solving success.

Correct responses were characterized by longer inter-keystroke intervals, greater variability in typing pace, and more frequent revision activity. Effect sizes ranged from small to moderate (Cliff's $\delta = 0.06$ to 0.35), with timing-based features showing the strongest associations. Students who typed more slowly and irregularly (pausing between keys, varying their pace, and editing their work) were more likely to submit correct answers than those who typed quickly and smoothly. What might account for these patterns? One interpretation is that slower, more variable typing reflects cognitive work happening between keystrokes: mental calculation, strategic deliberation, checking of intermediate results, or reconsideration of solution approaches. This interpretation aligns with theoretical accounts suggesting that keystroke-level information can shed light on cognitive and metacognitive processes involved in problem solving [15]. Long pauses may reflect planning or recall, while patterns of editing and revision may indicate monitoring and self-correction (behaviors associated with deliberate engagement). This interpretation is consistent with broader findings that deliberation and reflection are associated with better problem-solving outcomes. For instance, Chan et al. [8] found that middle-school students who paused longer before beginning to solve algebra problems in a learning game employed more efficient strategies, suggesting that taking time to think enhances performance. While their work examined pre-solving deliberation, our findings suggest that deliberation manifested during response construction (as slower, more variable typing with revision) similarly relates to success in open-ended mathematics problem solving.

Our results also align with work demonstrating relationships between keystroke patterns and performance outcomes across domains. In writing research, keystroke features such as revision frequency and timing have been linked to text quality [3, 38], with revision and pausing patterns differentiating more and less successful writers. In programming education, keystroke features including code editing patterns have proven informative for identifying at-risk students [7]. These findings extend to mathematics problem solving, demonstrating that typing dynamics capture aspects of cognitive engagement even in relatively brief, open-ended responses. The convergence of findings across writing, programming,and mathematics suggests that keystroke patterns may reflect domain-general characteristics of deliberate, monitored performance.

However, keystroke data present an inherent interpretive challenge. Unlike verbal protocols that provide direct access to students' articulated thoughts, or eye-tracking that reveals attentional focus, keystroke logs offer only indirect behavioral evidence. A pause between keystrokes might indicate careful checking, but it also could indicate retrieval difficulty, distraction, or indecision; the data alone cannot distinguish these possibilities. Moreover, both slower typing and correct answers could be independently caused by stronger mathematical knowledge or by better self-regulation skills. Students who understand the material well might type more deliberately because they recognize the importance of checking their work, while simultaneously being more likely to arrive at correct solutions due to superior domain knowledge.

While the continuous keystroke features showed moderate associations with correctness, we observed that one categorical threshold emerged with a particularly strong association: responses completed in five seconds or less were correct only 1.41% of the time, compared to a baseline rate of 30% for responses taking between five seconds and five minutes. This

Table 9: Directional changes in continuous features required to turn answers that were correct into answers that become incorrect for true positives (TP, 1→0 counterfactuals).

Feature	Proportion Increased(%) ↑	Proportion Decreased(%) ↓	Median ↑	Median ↓
average seconds between keys	5.63	1.98	9.46	-0.94
correction density	7.00	0.21	2.25	-0.64
correction proportion	7.20	0.26	0.40	-0.09
keystroke edit distance	5.99	0.45	26.43	-8.24
keystroke efficiency ratio	1.30	9.30	0.25	-0.56
keystroke entropy	1.57	**20.20**	1.02	-1.74
seconds per keystroke	4.51	1.93	37.50	-6.02
variation in seconds between keys	7.07	0.49	18.86	-2.84

Table 10: Categorical feature transitions required to turn answers that were correct into answers that become incorrect for true positives (TP, 1→0 counterfactuals).

Feature	Transition	Proportion (%)
FastResponse	0→1	**18.14**
Pasted	0→1	2.67
Pasted	1→0	0.07
SlowResponse	0→1	1.78
SlowResponse	1→0	0.03
WorksheetQuestionHasImage	0→1	1.86
WorksheetQuestionHasImage	1→0	0.37

represents a 95.3% reduction in the probability of correctness. This pattern is consistent with prior work demonstrating that extremely rapid responses in digital learning environments signal disengagement rather than genuine problem solving [6, 10, 41]. For instance, Beck [6] observed that extremely rapid responses in a reading tutor corresponded to chance-level performance, indicating students were guessing rather than genuinely processing the questions. Similarly, Cocea and Weibelzahl [10] found that rapid progression through learning materials without adequate time for comprehension reliably signaled disengagement across multiple e-learning platforms. Wright [41] reached comparable conclusions in non-timed formative assessments, demonstrating that responses completed in under five seconds often reflected guessing behavior that added error to ability estimates. Our results align with and extend these findings to open-ended mathematics problem solving: five seconds appears insufficient for the cognitive processes required to meaningfully engage with questions that demand calculation and written explanation. Students completing problems this quickly are almost certainly not progressing through the stages identified in classical models of mathematical problem solving (understanding, devising a plan, executing, and checking; [27, 32]). Rather, these extremely rapid responses likely represent giving up, guessing, entering placeholder text, or pattern-matching without comprehension. This stands in contrast to the deliberate engagement our data suggest characterizes successful problem solving, where students invest time in thinking through problems rather than rushing to submit responses.

The results from the counterfactual analysis further reinforced the centrality of this threshold in the model's learned patterns. When examining what minimal changes would flip the model's classification of an incorrect response to correct, moving students out of the fast response category occurred in 5.69% of counterfactuals. Conversely, for correct responses, transitioning into the fast response category was required in 18.4% of cases to flip the classification to incorrect, the single most common categorical change. This suggests that the five-second threshold captures a qualitatively different mode of engagement that the model learned to strongly associate with failure. Interestingly, the pattern was asymmetric for time outliers. While extremely fast responses were strongly associated with failure, extremely slow responses (greater than 5 minutes) showed no significant difference from baseline correctness. This asymmetry implies a minimum engagement threshold necessary for success (rushing through problems in seconds is almost universally associated with failure), but that additional time beyond approximately five minutes offers no systematic advantage. While deliberation is necessary, time alone does not ensure success.

Initially, we had hypothesized that problem format might influence keystroke behavior. Students responding to questions containing diagrams might type differently than those responding to text-only problems (perhaps pausing more frequently to inspect visual information or exhibiting altered pacing due to the cognitive demands of integrating visual and symbolic content). Prior research has shown that diagrams can substantially affect problem-solving strategies and performance in mathematics [9], and theories of problem representation emphasize that format fundamentally shapes solution approaches [20]. In this study, however, the keystroke patterns showed minimal variation across formats. While statistically significant differences emerged between image-based and text-only questions, effect sizes were negligible ($\delta = -0.01$ to -0.05). Students typed marginally faster and more regularly on image-based questions, but these differences were practically trivial. Correction behaviors showed no significant variation across formats.

Problem format was, however, associated with correctness. Image-based questions had lower odds of correctness (OR = 0.71, $p < 0.001$), and the counterfactual analysis supported this pattern. For incorrect responses, changing "Worksheet Question Has Image" from present to absent occurred in 2.37% of counterfactuals needed to flip the model's classification to correct. For correct responses, changing from absent to present occurred in 1.86% of counterfactuals needed to flip classifications to incorrect. The model learned that image presence was associated with lower correctness. This may be because these types of problems were less familiar to students, or because translating information across formats was difficult for these students. Overall, though, while images were associated with lower correctness, they do not appear to meaningfully change how students type their re-

sponses.

The machine learning models helped us better understand the full complexity of the relationships between keystroke features and correctness. The Random Forest model achieved reasonable goodness-of-fit (balanced accuracy = 0.69, MCC = 0.35, AUC-PR = 0.48), indicating that it successfully learned a meaningful signal from the data. These metrics provide evidence that the model identified stable, generalizable patterns in the relationship between typing behavior and problem-solving success. This level of performance is comparable to prior work using keystroke features for educational prediction. For instance, Casey [7] found that keystroke features improved models for identifying at-risk programming students, though with similarly modest improvements over baselines. Conijn et al. [11] reported that keystroke-based models for predicting essay quality achieved only modest gains over baseline models. Our results follow this pattern: keystroke dynamics provide meaningful but incomplete signals about outcomes, suggesting that typing behavior captures some (but not all) of the factors that differentiate success from failure.

To understand the structure of these learned relationships, we again employed counterfactual analysis. For incorrect responses (true negatives), the most frequently required adjustments involved increasing timing features, particularly seconds per keystroke (median increase: 34.1 units, modified in 30.4% of counterfactuals) and average seconds between keys (median increase: 8.3 units, modified in 25.8% of counterfactuals). For correct responses (true positives), the most common changes involved decreasing keystroke entropy (median decrease: 1.74, modified in 20.2% of counterfactuals). The model therefore found that deliberate, unhurried response construction is systematically associated with correctness, while rapid, streamlined typing is associated with failure. Interpreting these counterfactuals requires care. They reveal what the model learned about the relationship between keystroke patterns and correctness (which features it weighted most heavily and what magnitude of change it considers decision-relevant) but they do not establish causal pathways or prescribe interventions. The finding that increasing seconds per keystroke by 34 units would flip many model classifications does not imply that instructing students to "type slower" would improve their answers. Instead, this association may be mediated by unmeasured cognitive processes. Slower typing might be a consequence of deeper engagement rather than a cause of it. In general, the value of counterfactual analysis lies in revealing the structure of learned relationships and identifying behavioral thresholds that most strongly differentiate success from failure. The prominence of the fast response transition and timing-related features in counterfactual examination of the Random Forest model accords with the statistical analysis and provides additional confidence that these aspects of typing behavior mark meaningful distinctions in problem-solving processes.

4.1 Limitations

Several limitations should be noted. First, our outcome measure was binary correctness, which does not capture partial understanding or the quality of reasoning. A student might produce an incorrect answer due to a minor arithmetic error despite demonstrating sound problem-solving strategies, while another might guess correctly without genuine understanding [12]. Keystroke patterns may differentially relate to these scenarios, but our analysis treated all incorrect responses the same and all correct responses the same.

Second, our counterfactual analysis relied on a machine-learned model (Random Forest) as a representation of the relationship between keystroke patterns and correctness. While this model demonstrated reasonable goodness-of-fit, it likely did not capture all relevant dependencies or interactions. The counterfactuals generated by DiCE reflect what the model learned about the data, not necessarily the true underlying relationships. Our focus on verifiable ground-truth cases (true positives and true negatives) mitigates this concern by ensuring we interpret only the strongest part of the learned signal, but the possibility remains that some identified patterns are model-specific rather than reflecting genuine behavior–outcome relationships. Replication using different modeling approaches could potentially increase certainty about these findings.

Third, the explanatory models we developed capture associations between keystroke patterns and correctness but do not establish causation. We cannot determine from these analyses whether deliberate typing behaviors contribute to correct problem solving, whether both are products of underlying cognitive or motivational factors, or whether the relationship is bidirectional.

Finally, while this study found that problem format (image-based versus text-only) did not meaningfully affect keystroke patterns, this finding should be interpreted with caution. We examined only one dimension of format variation (presence or absence of a single static image), which does not capture the full range of visual complexity that affects problem solving. Images in our dataset varied widely in their content and informational value, yet our current analysis treated all image-based problems as equivalent. More broadly, the format invariance we observed may reflect the specific nature of our dataset rather than a general principle, and the boundary conditions under which keystroke patterns remain stable across task variations remain to be established through more systematic investigation of problem characteristics.

4.2 Future Directions

While our study is fundamentally explanatory rather than predictive or interventional, the patterns we identified may inform future work in several directions. The fast response threshold, in particular, emerges as a robust behavioral marker that could potentially serve as a trigger for lightweight prompts asking students to reflect before submitting [5]. The broader pattern—that slower, more variable, more effortful typing is associated with correctness—suggests that learning environments should encourage deliberation rather than rapid response entry. However, implementing such features would require careful validation, as we cannot assume that prompting students to engage in behaviors associated with success will actually improve their performance (e.g., Roll et al. [29]).

Future research could address several limitations and open questions. First, combining keystroke data with other pro-

cess measures—such as embedded probes asking students to explain their thinking, concurrent eye-tracking, or retrospective interviews—could help map particular keystroke patterns onto specific cognitive activities and disambiguate the interpretive ambiguity inherent in behavioral traces. Second, examining whether keystroke dynamics are associated with different types of errors (conceptual vs. procedural, early-stage vs. late-stage) or different qualities of correct responses (well-reasoned vs. lucky guesses, perhaps inferred through a contextual guessing model—e.g., Baker et al. [4]) could provide more nuanced understanding than our binary correctness measure allows.

Finally, cross-context replication would strengthen confidence in generalizability. Our dataset came from a single online mathematics platform serving middle school students; the extent to which these findings apply to other age groups, content areas, or learning environments remains uncertain. Different mathematical topics, problem structures, and student populations might produce different relationships between keystrokes and other constructs.

4.3 Conclusion

Keystroke dynamics reliably differentiate correct from incorrect responses in mathematics problem solving. Students who answered correctly typed more slowly, with greater variability and more frequent revision, compared to those who answered incorrectly. The strongest association involved responses completed in under five seconds, which were correct less than 2% of the time. While problem format (image versus text) was associated with correctness, it did not meaningfully affect keystroke patterns, suggesting that the behavioral indicators we identified may be reliable process signals even across different problem presentation formats. Machine learning models successfully learned the complex relationships between keystroke features and correctness (balanced accuracy = 0.69, MCC = 0.35), and counterfactual analysis revealed that timing-related features and the fast response threshold were the most decision-relevant factors in the model's representation. These findings demonstrate that keystroke logging can reveal meaningful signals about problem-solving processes. Our results extend prior work showing that keystroke patterns inform predictions of educational outcomes, demonstrating that typing dynamics also differentiate immediate problem-solving success in mathematics. While interpretive ambiguity remains regarding the specific cognitive processes underlying different typing patterns, and causal relationships cannot be established from these analyses, this work provides empirical evidence that fine-grained behavioral traces differentiate successful from unsuccessful mathematical problem solving. Future work combining multiple process-tracing methods, exploring intervention designs, and testing generalization across contexts can build on this foundation to deepen understanding of how students think through mathematics problems.

4.4 Acknowledgments

This work was supported by the Learning Engineering Virtual Institute (LEVI)

5. REFERENCES

[1] V. Aleven, B. M. McLaren, J. Sewall, and K. R. Koedinger. The cognitive tutor authoring tools (ctat): Preliminary evaluation of efficiency gains. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 61–70, Berlin, Heidelberg, 2006. Springer.

[2] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *Journal of the Learning Sciences*, 4(2):167–207, 1995.

[3] V. M. Baaijen, D. Galbraith, and K. de Glopper. Keystroke analysis: Reflections on procedures and measures. *Written Communication*, 29(3):246–277, 2012.

[4] R. S. D. Baker, A. T. Corbett, and V. Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian knowledge tracing. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 406–415, Berlin, Heidelberg, 2008. Springer.

[5] R. S. d. Baker, A. T. Corbett, K. R. Koedinger, S. Evenson, I. Roll, A. Z. Wagner, M. Naim, J. Raspat, D. J. Baker, and J. E. Beck. Adapting to when students game an intelligent tutoring system. In *International conference on intelligent tutoring systems*, pages 392–401. Springer, 2006.

[6] J. E. Beck. Engagement tracing: Using response times to model student disengagement. In C.-K. Looi, G. McCalla, B. Bredeweg, and J. Breuker, editors, *Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, pages 88–95, Amsterdam, 2005. IOS Press.

[7] K. Casey. Using keystroke analytics to improve pass–fail classifiers. *Journal of Learning Analytics*, 4(2):189–211, 2017.

[8] J. Y.-C. Chan, E. R. Ottmar, and J.-E. Lee. Slow down to speed up: Longer pause time before solving problems relates to higher strategy efficiency. *Learning and Individual Differences*, 93:102109, 2022.

[9] J. Chu, B. Rittle-Johnson, and E. R. Fyfe. Diagrams benefit symbolic problem-solving. *British Journal of Educational Psychology*, 87(2):273–287, 2017.

[10] M. Cocea and S. Weibelzahl. Log file analysis for disengagement detection in e-learning environments. *User Modeling and User-Adapted Interaction*, 19(4):341–385, 2009.

[11] R. Conijn, C. Cook, M. van Zaanen, and L. Van Waes. Early prediction of writing quality using keystroke logging. *International Journal of Artificial Intelligence in Education*, 32(4):835–866, 2022.

[12] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995.

[13] K. A. Ericsson. Protocol analysis. In *A Companion to Cognitive Science*, pages 425–432. 2017.

[14] L. Flower and J. R. Hayes. A cognitive process theory of writing. *College Composition and Communication*, 32(4):365–387, 1981.

[15] D. Galbraith and V. M. Baaijen. Aligning keystrokes with cognitive processes in writing. In E. Lindgren and K. Sullivan, editors, *Observing Writing*, pages 306–325. Brill, 2019.

[16] H. Guo, P. D. Deane, P. W. van Rijn, M. Zhang, and R. E. Bennett. Modeling basic writing processes from

keystroke logs. *Journal of Educational Measurement*, 55(2):194–216, 2018.

[17] M. Hegarty, R. E. Mayer, and C. A. Monk. Comprehension of arithmetic word problems: A comparison of successful and unsuccessful problem solvers. *Journal of Educational Psychology*, 87(1):18–32, 1995.

[18] S. Hutt, A. Wong, A. Papoutsaki, R. S. Baker, J. I. Gold, and C. Mills. Webcam-based eye tracking to detect mind wandering and comprehension errors. *Behavior Research Methods*, 56(1):1–17, 2024.

[19] K. R. Koedinger and B. A. MacLaren. Implicit strategies and errors in an improved model of early algebra problem solving. In *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*, pages 382–387. Lawrence Erlbaum Associates, 1997.

[20] K. R. Koedinger and M. J. Nathan. The real story behind story problems: Effects of representations on quantitative reasoning. *The Journal of the Learning Sciences*, 13(2):129–164, 2004.

[21] E. Lindgren and K. Sullivan, editors. *Observing Writing: Insights from Keystroke Logging and Handwriting*. Brill, Leiden, 2019.

[22] K. Meissel and E. S. Yao. Using cliff's delta as a non-parametric effect size measure: an accessible web app and r tutorial. *Practical Assessment, Research, and Evaluation*, 29(1), 2024.

[23] R. K. Mothilal, A. Sharma, and C. Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 607–617, 2020.

[24] M. Norqvist, B. Jonsson, J. Lithner, T. Qwillbard, and L. Holm. Investigating algorithmic and creative reasoning strategies by eye tracking. *The Journal of Mathematical Behavior*, 55:100701, 2019.

[25] A. Obersteiner and C. Tumpek. Measuring fraction comparison strategies with eye-tracking. *ZDM Mathematics Education*, 48(3):255–266, 2016.

[26] B. Plank. Keystroke dynamics as signal for shallow syntactic parsing. In *Proceedings of COLING 2016*, pages 609–619, 2016.

[27] G. Polya. *How to Solve It: A New Aspect of Mathematical Method*. Princeton University Press, Princeton, NJ, 1945.

[28] M. M. Porras, C. A. K. v. Campen, J. J. González-Rosa, F. L. Sánchez-Fernández, and J. I. N. Guzmán. Eye tracking study in children to assess mental calculation and eye movements. *Scientific Reports*, 14(1):18901, 2024.

[29] I. Roll, V. Aleven, B. M. McLaren, and K. R. Koedinger. Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, 21(2):267–280, 2011.

[30] J. E. Russo, E. J. Johnson, and D. L. Stephens. The validity of verbal protocols. *Memory & Cognition*, 17(6):759–769, 1989.

[31] M. Schindler, A. J. Lilienthal, R. Chadalavada, and M. Ögren. Creativity in the eye of the student. In *Proceedings of the 40th Conference of the International Group for the Psychology of Mathematics Education*, volume 4, pages 163–170, 2016.

[32] A. H. Schoenfeld. *Mathematical Problem Solving*. Elsevier, New York, 1985.

[33] A. H. Schoenfeld. Learning to think mathematically: Problem solving, metacognition, and sense making in mathematics. *Journal of Education*, 196(2):1–38, 2016.

[34] A. H. Schoenfeld. Why are learning and teaching mathematics so difficult? In *Handbook of Cognitive Mathematics*, pages 763–797. Springer, Cham, 2022.

[35] S. Sinharay, M. Zhang, and P. Deane. Prediction of essay scores from writing process and product features using data mining methods. *Applied Measurement in Education*, 32(2):116–137, 2019.

[36] A. N. Sušac, A. Bubić, J. Kaponja, M. Planinić, and M. Palmovic. Eye movements reveal students' strategies in simple equation solving. *International Journal of Science and Mathematics Education*, 12(3):555–577, 2014.

[37] Y. Tian, S. Crossley, and L. Van Waes. The klicke corpus: Keystroke logging in compositions for knowledge evaluation. *Journal of Writing Research*, pages 299–336, 2025.

[38] L. Van Waes and M. Leijten. Fluency in writing: A multidimensional perspective on writing fluency applied to l1 and l2. *Computers and Composition*, 38:79–95, 2015.

[39] L. Van Waes, M. Leijten, and D. Van Weijen. Keystroke logging in writing research: Observing writing processes with inputlog. *German as a Foreign Language*, (2):41–64, 2009.

[40] A. Wengelin, J. Frid, R. Johansson, and V. Johansson. Combining keystroke logging with other methods. In E. Lindgren and K. Sullivan, editors, *Observing Writing: Insights from Keystroke Logging and Handwriting*, pages 30–49. Brill, Leiden, 2019.

[41] D. B. Wright. Treating rapid responses as incorrect for non-timed formative tests. *Open Education Studies*, 1(1):56–72, 2019.

[42] C. Xu. Understanding online revisions in l2 writing: A computer keystroke-log perspective. *System*, 78:104–114, 2018.

Toward LLM-supported Automated Assessment of Critical Thinking Subskills

Marisa C. Peczuh*
University of Minnesota
peczu001@umn.edu

Nischal Ashok Kumar*
University of Massachusetts, Amherst
nashokkumar@umass.edu

Ryan S. Baker
Adelaide University
ryanshaunbaker@gmail.com

Blair Lehman
Brighter Research
blair.lehman@gmail.com

Danielle Eisenberg
Educational Testing Service
deisenberg@ets.org

Caitlin Mills
University of Minnesota
cmills@umn.edu

Payu Wittawatolarn
University of Massachusetts, Amherst
pwittawatola@umass.edu

Kushaan Naskar
University of Massachusetts, Amherst
knaskar@umass.edu

Keerthi Chebrolu
University of Massachusetts, Amherst
kchebrolu@umass.edu

Sudhip Nashi
University of Massachusetts, Amherst
snashi@umass.edu

Cadence Young
University of Massachusetts, Amherst
cadenceyoung@umass.edu

Brayden Liu
University of Massachusetts, Amherst
zliu@nutrition.umass.edu

Sherry Lachman
AugmentED, Advanced Education Research and Development Fund
slachman@aerdf.org

Andrew Lan
University of Massachusetts, Amherst
andrewlan@cs.umass.edu

*Joint first authors with equal contribution

ABSTRACT

As the world becomes increasingly saturated with AI-generated content, disinformation, and algorithmic persuasion, critical thinking – the capacity to evaluate evidence, detect unreliable claims, and exercise independent judgment – is becoming a defining human skill. Developing critical thinking skills through timely assessment and feedback is crucial; however, there has not been extensive work in educational data mining on defining, measuring, and supporting critical thinking. In this paper, we investigate the feasibility of measuring "subskills" that underlie critical thinking. We ground our work in an authentic task where students operationalize critical thinking by writing argumentative essays. We developed a coding rubric based on an established skills progression and completed human coding for a corpus of student essays. We then evaluated three distinct approaches to automated scoring: zero-shot prompting, few-shot prompting, and supervised fine-tuning, implemented across three large language models (GPT-5, Llama 3.1 8B, and ModernBERT). Fine-tuning Llama 3.1 8B achieved the best results and demonstrated particular strength on subskills with highly separable proficiency levels with balanced labels across levels, while lower performance was observed for subskills that required detection of subtle distinctions between proficiency levels or imbalanced labels. Our exploratory work represents an initial step toward scalable assessment of critical thinking skills across authentic educational contexts. Future research should continue to combine automated critical thinking assessment with human validation to more accurately detect and measure dynamic, higher-order thinking skills.

Keywords

Critical thinking, 21st century skills, large language models, argumentative essay

Marisa Peczuh, Nischal Ashok Kumar, Ryan Baker, Blair Lehman, Danielle Eisenberg, Caitlin Mills, Payu Wittawatolarn, Kushaan Naskar, Keerthi Chebrolu, Sudhip Nashi, Cadence Young, Brayden Liu, Sherry Lachman, and Andrew Lan. Toward LLM-supported Automated Assessment of Critical Thinking Subskills. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 340–352. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039869

1. INTRODUCTION

While the ability to think critically has always been an important skill for students to learn, it is increasingly important in today's ever-changing world [14]. Despite widespread recognition of its significance, the development of students' critical thinking has not been a major focus in the educational data mining (EDM) community to date. For example, a search of yearly EDM Conference Proceedings since 2020 revealed only one title and zero abstracts with the term 'critical thinking.'

There are likely several reasons why critical thinking has not been a central focus in educational data mining. Typically, in the EDM community, we must be able to create reasonably accurate ways to assess a skill or construct in order to either proactively or reactively respond to support the learner in a personalized way. Before a construct can be assessed, it must be clearly defined and operationalized, which has been a challenge for critical thinking skills to date. For example, a number of competing definitions and frameworks have articulated distinct visions of what critical thinking is, and how it manifests across contexts [33, 46, 60]. A limitation of many of these prior definitions and frameworks is that they have been vague and difficult to operationalize [4, 5]; others have been criticized for being narrow and over-simplified [57].

A recent definition of critical thinking that is both specific and feasible to operationalize comes from Pasquinelli and colleagues: critical thinking is "the capacity of assessing the epistemic quality of available information and—as a consequence of this assessment—of calibrating one's confidence in order to act upon such information" [46]. Within such a framework, we can think of critical thinking in terms of skills that can be demonstrated, processes needed to effectively use these skills in specific situations, and dispositions that motivate the use of these skills when needed [24]. Critical thinking is, of course, complex and multi-dimensional; successful critical thinking in context calls on skills such as understanding and analyzing information, evaluating evidence, making inferences, articulating one's own arguments, and monitoring one's own thinking process [17, 19, 24, 33, 47]. It's also worth noting that an open question is whether critical thinking skills are domain-general or domain-specific [15, 33, 41, 46]. For this particular submission, we operationalize a set of core "subskills" that are primarily involved in domains that support evaluating and synthesizing information to form new opinions as a foundational example for how to support such modeling.

A key step towards being able to support critical thinking more effectively is being able to assess it rapidly and in context to inform teachers and students [16] as well as adaptive learning platforms. Critical thinking assessments have been developed in multiple formats (e.g., multiple choice, open-ended questions), but these measurements have faced challenges with reliability and validity [33, 37]. Another important limitation to address is the scarcity of assessments that are easily implemented and actionable within educational settings [37]. Developing tools that can detect indicators of critical thinking skills in the moment and automatically, particularly in the context of authentic learning activities, could increase the feasibility for use in adaptive learning and contribute to formative assessment for teachers.

In the current paper, we provide insights into this problem by investigating the feasibility of measuring a set of critical thinking subskills that are core to critical thinking processes (e.g., evaluating evidence, drawing conclusions), rapidly and solely from an artifact produced by an authentic student learning activity, an argumentative essay. We developed a rubric for coding critical thinking subskills based on an existing skills progression, which human raters used to score a set of argumentative essays. We then explored the use of large language models (LLMs) in assessing critical thinking subskills from student-written argumentative essays. We investigated a range of methods, from prompting to fine-tuning, across three different LLMs, leveraging the same rubrics we provided the human coders.

Ultimately, we aim to develop methods that can be used to assess critical thinking within authentic learning activities, across educational activities and contexts. For this paper, we began with a single context (essays) and multiple subskills, providing a proof-of-concept methodology for EDM. We attempted to embrace the inherent complexity of critical thinking, in that it has multiple subskills that exist long a progression rather than being easily operationalized and modeled in a binary manner. Our goal was therefore to test whether we could develop effective models for the present dataset, and which subskills (if any) were amenable to such detection.

2. RELATED WORK

2.1 Critical Thinking Definitions and Frameworks

As mentioned above, there have been a range of definitions and frameworks for critical thinking [1, 46, 52, 60], but across definitions and frameworks, critical thinking generally involves having the appropriate background knowledge or acquiring information as needed [33], being motivated to engage in critical thinking processes [17, 24], possessing essential critical thinking competencies [19, 57], and being able to use the right competencies for a specific situation in the right fashion [7, 57]. While many frameworks for critical thinking skills exist (see reviews in [1, 33]), and there are many important differences between these frameworks, there are some subskills which the field has come to a consensus about. First, critical thinking involves evaluating the quality and credibility of information (e.g., source, evidence, argument) [17, 19, 30, 33, 47], analyzing the meaning of this information [17, 19, 24, 33, 47], and making inferences, such as how the information might apply in new scenarios [17, 19, 33]. Critical thinking also involves formulating one's own arguments [31], explaining one's own reasoning [17, 19, 33], and forming conclusions [47]. Finally, critical thinkers necessarily self-regulate their processes to continuously monitor and improve [17, 19, 24, 30, 33, 47]. Here we focus on these subskills rather than the overall construct of "critical thinking" in order to target the process involved in a more concrete and definable way.

2.2 Measuring Critical Thinking

Regardless of the ongoing debates on definitions and frameworks, there have been several attempts to measure critical thinking skills over the last three decades [8, 16, 33, 37]. The simplest approach to assessing critical thinking has been to give scenarios to students and then ask multiple choice questions [10, 62]. While these assessments can be time efficient, and can have good psychometric properties, there have been concerns about how much information they provide about critical thinking for real-world challenges and authentic activities [37], particularly in a way that can be used to support formative feedback.

Some assessments, such as the Halpern Critical Thinking Assessment [23], have attempted to use open response items and are more relevant to real world situations, where students need to respond authentically in the moment [37]. Still other instruments and rubrics have been used to assess critical thinking in artifacts [16, 20, 37]. For example, the Ennis Weir Critical Thinking Essay Test [18] was designed to assess critical thinking in a more realistic way by asking

participants to respond to a fictional argumentative letter. Another example, the Holistic Critical Thinking Scoring Rubric [20], can be used to assess critical thinking as demonstrated in an essay or presentation, whether authentic or designed for the purpose of assessment. However, instruments would ideally give students opportunities to develop skills in authentic learning contexts with quick and accurate AI-supported feedback.

2.3 Automated Essay Scoring and Qualitative Coding Approaches

One possible paradigm for how to address this may come from recent work in two areas: automated essay scoring (AES) and automated qualitative coding. AES has been studied for decades, with early approaches relying on surface features such as word counts, lexical diversity, and syntactic patterns to approximate human scoring [2, 48]. Then, neural and transformer-based models shifted the field toward more robust representations of text, enabling AES systems to capture deeper discourse and linguistic patterns [27, 51]. Most recently, large language models (LLMs) have introduced a new approach that does not require training: one can prompt LLMs with hand-crafted scoring rubrics and ask them to estimate the score of an essay (see review in [35]). Similar approaches using prompting of LLMs have been used to assess complex reasoning in shorter text as well [38, 58, 65], building on past work that used previous-generation natural language processing (NLP) techniques [28]. This progression has broadened the scope of automated assessment of text from holistic proficiency measures toward more fine-grained, rubric-aligned scoring tasks.

Recent studies have leveraged AES and automated qualitative coding methods for a number of purposes, and have investigated both their potential and limitations. For example, Xiao et al. [65] introduced a dual-process framework for human-AI collaborative essay scoring, leveraging LLMs to provide both initial predictions and justifications that can support teacher decision-making. Stahl et al. [56] demonstrated that integrating argument mining outputs into transformer-based models can enhance AES and provide a stronger basis for modeling reasoning skills. At the same time, Seßler et al. [53] compared LLM performance with teacher ratings in multidimensional essay scoring, showing that while models achieve promising agreement with human judgments, important gaps remain in transparency and trait sensitivity. Similarly, Liu and colleagues [38] investigated the strengths and weaknesses of different prompting techniques for three different textual data sets, and Simon and colleagues [54] investigated how multi-agent LLM systems could enhance the performance of automated coding for thematic analysis.

Early steps have already been taken to utilize this type of method for aspects of critical thinking. For example, early systems applied previous-generation NLP methods such as Latent Semantic Analysis (LSA) to open-ended responses to capture aspects of argumentation such as the relevance of an argument and the effective use of knowledge [49, 50], demonstrating potential but limited alignment with human judgments. More recently Yeginbergen et al. [66] introduced a framework for counter-argument generation that combines retrieved external evidence with LLM prompting and evaluates outputs using an LLM-as-a-Judge metric, advancing critical thinking assessment toward more evidence-grounded and generative application. These approaches show the possibility of using natural language processing to study and assess critical thinking in authentic or similar artifacts, but have not yet been successful at assessing critical thinking multi-dimensionally, in a fine-grained fashion, and across educational activities and contexts. Thus, while our task is not AES in the conventional sense of scoring overall writing quality, it is closely related methodologically: we similarly evaluate whether automated systems can assign human-aligned rubric scores to student essays, but we shift the scoring target from writing quality to critical thinking subskills.

3. METHOD

3.1 Dataset

We used a portion of the PERSUADE 2.0 corpus [11] that was made publicly available as part of the Feedback Prize - Predicting Effective Arguments competition [21] in our work. The PERSUADE 2.0 corpus includes more than 25,000 argumentative essays that were written by 6th to 12th grade students in the United States, for 15 prompts on two writing tasks: independent and source-based writing. The training data that was released as part of the competition included 4,115 essays from the original corpus. We did not have access to the essay prompt, writing task, or source materials (in the case of source-based writing) in the publicly available dataset, and therefore did not use this information in our human or LLM coding. We selected this dataset as an initial use case for the development of a critical thinking model since argumentative essays (and argumentation more generally) often include evidence of critical thinking processes (e.g., providing evidence for a claim, evaluating alternative viewpoints through counterarguments).

3.2 Critical Thinking Rubric

We developed a rubric for coding critical thinking based on an early version of the Critical Thinking Progression for High School Students developed by the Skills for the Future (SFF) initiative, a partnership between ETS and The Carnegie Foundation for the Advancement of Teaching [55].[1] The progression defined critical thinking as "the skill and disposition to actively seek and evaluate information and construct evidence-based arguments to reach well-founded conclusions or informed decisions, including recognizing and applying sound logical reasoning. It plays an important role in learning activities, such as evaluating multiple sources, identifying assumptions and reasoning flaws, and making sound arguments. Developing strong critical thinking skills enables students to tackle complex challenges more effectively and achieve greater learning outcomes" [55].

The skills progression included four skills that were further specified into subskills. The first skill, *Information Seeking*, was not included, as the focal dataset for the present work was completed essays. The remaining three skills were retained for the present work and are shown in Table 1. While these three skills were identified from the SFF initiative, we grounded the subskills in our literature review. For example, many definitions of critical thinking include the ability to determine the trustworthiness of information [17, 19, 30, 33, 47], otherwise known as *Evaluating Evidence Strength* in SFF.

[1] The current version of the Skills for the Future Progressions were made public in January 2026 (https://www.ets.org/newsroom/ets-carnegie-release-skills-progressions.html). They have four levels intended for understanding skill acquisition during secondary school. The project continues to iterate on ways to portray the ways in which skill development is not linear and is currently working to validate these progressions with educators and students.

Table 1. Critical thinking subskills

Skill	Subskill	Definition	Example
2: Information Analysis	2.1 Synthesizing Multiple Sources	Effectively synthesizes multiple pieces of information	Summarizes information from multiple sources (with citations) but does not integrate information
	2.2 Evaluating Evidence Strength	Evaluates the strength and relevance of evidence used to form a conclusion	Presents evidence and links evidence to a specific conclusion, but does not evaluate the relevance or strength of the evidence for the argument generated
3: Argument Generation	3.1 Using Counterarguments	Effectively addresses counterarguments	Acknowledges specific opposing viewpoint(s), counterargument(s), or qualifier(s)
	3.2 Using Facts and Opinions	Relies on data and/or facts over opinions	Uses facts and opinions about equally to support claim(s) and/or arguments
4: Logical Reasoning	4.1 Drawing Conclusions	Draws specific conclusions	Draws a specific conclusion to analyze simple and straight-forward relationships/argumentations
	4.2 Using Logical Fallacies	Recognizes and avoids logical fallacies	Uses logical fallacies and evidence about equally when generating arguments

Three proficiency levels were included in the original progression: 'Emerging' (representing lowest level of proficiency, where students are "beginning to identify and understand key aspects of critical thinking"), 'Expanding' ("developing [critical thinking] skills"), and 'Exemplifying' ("demonstrating proficiency" and mastery of critical thinking skills). After initially exploring the feasibility of using the progression for our desired purposes quantitatively and qualitatively, we decided to add two proficiency levels to the rubric that demonstrated lower proficiencies than the 'Emerging' level: 'Below Emerging' (where essays did not yet meet the threshold for 'Emerging' criteria) and 'Not Applicable' (the lowest proficiency level, where "certain critical thinking elements are absent from student contribution," which inhibits a proficiency rating").

We also adjusted the criteria across subskills for achieving various proficiency levels. First, we adjusted the progression to increase clarity in coding and reduce cognitive load of coders, such as collapsing across two sections for each subskill, using parallel wording across proficiency levels, and making relative rather than numerical statements (e.g., "Uses more opinions than facts," "Uses more logical fallacies than evidence"). Second, rather than modifying the progression directly, we provided additional context, such as by broadening the criteria for citing sources in *Synthesizing Multiple Sources* (e.g., direct quotes, reference to author, title of a text) and re-defining evidence in *Evaluating Evidence Strength* (e.g., anything used to support a claim, which can include both facts and opinions/anecdotes).

Third, we made distinctions between certain subskills (i.e., to avoid essays being scored at lower proficiency levels in multiple subskills for the same mistake) while also connecting subskills to align greater critical thinking with better quality arguments. For example, it was necessary for writers to have a clear conclusion (*Drawing Conclusions*) to connect evidence to the conclusion (*Evaluating Evidence Strength*) and evaluate the presence and quality of counterarguments (*Using Counterarguments*). Ultimately, the final rubric included five proficiency levels (from 'Not Applicable' to 'Exemplifying'), three skills and six subskills, and various definitions and resources to support the coding process.

3.3 Human Coding

3.3.1 Coders and Codebook

The original coders were four graduate students at a large university in the U.S. (Coder Group 1). All coders had Master's degrees and were advanced students (fourth through sixth years) in an educational psychology PhD program. Their engagement included three meetings to achieve sufficient reliability and independent coding of 50 essays (200 essays total). The codebook development primarily focused on adapting the original Critical Thinking Progression into a rubric that could be applied to coding a specific student-generated artifact (e.g., essay), which is detailed previously in Section 3.2. However, the need for supplementary information was identified during the codebook development and initial training meetings with coders. For example, clarification was needed for *Synthesizing Multiple Sources* as to what qualified as a citation since the essays were written by grade 6–12 students and how to discriminate between summarizing and integrating multiple sources.

A second group of coders (Coder Group 2) was recruited to expand our sample of human-coded essays. Coder Group 2 included three graduate students, two of which were from another large university in the U.S. and one from a large university in Australia. All coders had Master's degrees, and one was in the final year of another Master's program. All coders had extensive prior experience with qualitative coding. Their engagement included three meetings to achieve sufficient reliability and independent coding of 100 essays (300 essays total). The final codebook that was used for Coder Group 1 was used for Coder Group 2 with no additional modifications.

3.3.2 Reliability

Given the complexity of critical thinking as a construct and our overarching goal for this work as usable in practice (i.e., build a critical thinking model that can be used for similar purposes to iteratively evaluate AI tools), we decided on a level of reliability consistent with our preliminary work. Therefore, we did not use the typical thresholds for evaluation criteria for achieving interrater reliability for Coder Group 1 (e.g., Krippendorff's $\alpha \geq .8$) [29, 40, 63]. Instead, we set our threshold to a Krippendorff's Alpha greater than or equal to .6 as our criteria for sufficient interrater reliability. We adopted Krippendorff's Alpha as we had four coders who during the process of achieving reliability provided ordinal codes for

all subskills in all essays [25]. The coders scored each subskill holistically by providing a single rating for each subskill for each essay. We used the K-Alpha Calculator [40] to compute Krippendorff's Alpha for each subskill in each attempt to achieve sufficient interrater reliability, described next.

Two rounds of coding were required to achieve sufficient interrater reliability for Coder Group 1, and each round adopted the same approach. Coders were asked to code 10 essays for all six subskills (see Table 1). Prior to the first round of coding, coders engaged in a 90-minute training session. During this session, they also coded an essay collaboratively as a group. The coders engaged in a 60-minute disagreement resolution session that reviewed disagreements in round one codes as well as identified instances in which rubric revisions were necessary (see Section 3.2). The coders then completed a second round of coding with new essays that were coded for all six subskills. The second round of coding resulted in sufficient reliability for all subskills (Table 2), which exceeded our pre-defined threshold for sufficient reliability ($\alpha \geq .60$) and generally achieved a satisfactory level of agreement [40]. The one exception was *Using Facts and Opinions*, which did not reach our pre-defined threshold for sufficient reliability. We chose to proceed with coding of *Using Facts and Opinions* as the discussion of coder disagreements revealed that the disagreements generally stemmed from the coders not having access to the source material provided to students when writing the essays to determine whether statements were fact or opinion. Since our goal was to explore whether it was possible to engage in the process of defining, coding, and predicting critical thinking indicators (rather than develop the most "effective" model), we concluded the interrater reliability had reached a sufficient level to proceed.

Table 2. Reliability for human coder Group 1 and Group 2

Skill	Coder Group 1 Krippendorf's Alpha	Coder Group 2 Weighted Kappa *M (Min-Max)*
Synthesizing Multiple Sources	1.00	0.87 (0.79-0.91)
Evaluating Evidence Strength	0.68	0.74 (0.66-0.78)
Using Counterarguments	0.70	0.72 (0.65-0.78)
Using Facts and Opinions	0.57	0.82 (0.80-0.83)
Drawing Conclusions	0.75	0.69 (0.55-0.80)
Using Logical Fallacies	0.79	0.79 (0.69-0.84)

Although reliability was achieved after the second round, coders met for another 60-minute session, where additional revisions to the rubric were made based on coder disagreement and feedback (see Section 3.2). Then, each coder was assigned 50 essays to independently code for each of the six subskills, which resulted in 200 human-coded essays (5% of total essay sample).

Based on initial model performance, we determined that additional human-coded essays were necessary. Coder Group 2 engaged in the same 90-minute training session as Coder Group 1 and then proceeded to achieve reliability. Given that the goal was to increase the sample of human-coded essays, Coder Group 2 did not seek to achieve sufficient reliability within their group but rather to achieve sufficient reliability with Coder Group 1. This meant that essays coded by Coder Group 1 were leveraged for Coder Group 2 reliability along with the final version of the rubric and that reliability was computed between each coder from Coder Group 2 with Coder Group 1 codes. Weighted Kappas (Quadratic) were used for evaluating interrater reliability with the same threshold used for Coder Group 1 ($\alpha \geq 0.6$). Two rounds of coding (with a 60-minute meeting after each round) were required to achieve sufficient interrater reliability for each Coder Group 2 coder for each subskill (Table 2), except for one coder slightly below our threshold for *Drawing Conclusions* (*Kappa* = .55). Coders were then each assigned 100 essays to independently code for each subskill, which resulted in an additional 300 essays (7% of total essay sample).

The total human-coded essay sample included 500 essays (12% of total essay sample) that were used to evaluate the performance of the LLM critical thinking model. While we recognize this is a modest number of human-coded essays given the nuanced and multifaceted nature of critical thinking and its operationalization, the number is consistent with the overall exploratory goal of this work. Additionally, the sample size is in the range commonly used in other research involving evaluations of whether LLMs agree with human codes [6, 9, 38, 43, 45].

3.4 LLM-based Assessment

We evaluated two complementary approaches for LLM-based critical thinking assessment: (1) prompting proprietary LLMs, specifically OpenAI's GPT-5, and (2) fine-tuning two open-source models for classification purposes, Llama 3.1 8B [22] and ModernBERT [61], on our student essay dataset.

3.4.1 Prompting and Fine-tuning

We prompted GPT-5, a state-of-the-art LLM at the time of this work, with two settings: *zero-shot* [34] and *few-shot* [65]. In both settings, we instructed the LLM to assume the role of a calibrated educational scorer, tasked with classifying essays on fine-grained critical thinking subskills according to our structured rubric. The prompt included: (a) the general definitions of the five proficiency levels (0 = 'Not Applicable' through 4 = 'Exemplifying'), and (b) the subskill definition and subskill-specific rubric descriptors. In the **zero-shot condition,** each essay was paired with a single target subskill. We asked the model to output both a proficiency label (0–4) and a concise justification aligned with the rubric. All 500 essays in the dataset were evaluated under this setting. In the **few-shot condition,** the prompt included the same instructions as above, with in-context examples for each proficiency level for the target subskill. We experimented with one, three, and five randomly selected in-context examples per proficiency level for each target subskill. To avoid data leakage, we excluded essays used as examples from the evaluation and tested the model on the remaining essays. For GPT-5, we used the default *reasoning effort* setting to balance performance and efficiency, and set the maximum output length to 3000 tokens to accommodate the model's reasoning traces and justification.

To complement our prompting experiments, we fine-tuned two open-source LLMs, Llama-3.1 8B Instruct, a decoder-only model with strong performance on several popular natural language processing benchmarks and ModernBERT-base, an encoder-only transformer, chosen for its lightweight design (149M parameters) and strong performance-size tradeoff on classification tasks. ModernBERT improves upon its predecessor BERT [13] with

training on much larger data and an 8k-token context window, making it well-suited for efficiently handling student essays.

For fine-tuning Llama, we constructed the input prompt by including the subskill name, definition and subskill-specific rubric descriptors corresponding to the five proficiency levels followed by the essay text to be classified. The output of the model is the proficiency level label text itself ('Not Applicable,' 'Below Emerging,' 'Emerging,' 'Expanding,' 'Exemplifying'). We fine-tuned the model using PEFT with LoRA adapters (rank=32, α=64, dropout=0.05), updating only adapter weights [26]. Training used a chat-style input-output format, optimized with AdamW for 3 epochs (batch size 4, gradient accumulation 4) with a 1,536-token sequence length. We selected the checkpoint with the best validation loss on a held-out validation set for inference.

For fine-tuning ModernBERT, like Llama, we construct the input prompt by including the subskill name, definition and subskill-specific rubric descriptors corresponding to the five proficiency levels. The output of the model is a single proficiency label (0–4). We added a lightweight classification head over the [CLS] representation and trained with AdamW and focal loss [36] to mitigate class imbalance for 6 epochs, using early stopping and a maximum input length of 1,536 tokens.

For Llama fine-tuning, in addition to outputting the label text, we experimented with a variant, where the model first outputs a short justification and then outputs the label text in order to elicit a chain-of-thought behavior [64] in the LLM to help in the reasoning process to generate the label [65]. Since our dataset does not contain gold justifications written by the human coders, we synthetically generate them by prompting OpenAI's GPT-4.1 to generate a sentence justifying the human label for a particular subskill for a given essay. We call this fine-tuning method "Llama (with justification)". In contrast, the default variant outputs only the proficiency label text (without generating a justification), which we refer to as "Llama (without justification)".

Additionally, to study the role of the subskill-specific rubric descriptors corresponding to the five proficiency levels in the input prompt, we consider two input prompt variants for both Llama and ModernBERT. In the with-rubric variant, the input prompt includes the subskill name, definition, and all rubric descriptors, along with the essay text. In the without-rubric variant, we remove the rubric descriptors and include only the subskill name and its definition, and the essay text. We refer to these variants as "Llama (with rubric)/ ModernBERT (with rubric)" and "Llama (without rubric)/ ModernBERT (without rubric)", respectively.

We evaluated two data-splitting regimes: 1) **Essay-based split**: essays divided into 70/10/20 percentages for the train/validation/test sets, respectively, with subskills constant, and results averaged across five random seeds. 2) **Subskill-based split**: one subskill held out entirely for testing; the other two used for training and validation with a 90/10 essay split. The first splitting setup is standard and evaluates LLMs' scoring ability when given on-task training data, i.e., human scored essays on each subskill. The second splitting setup tests generalization to new subskills, without using any human scores for previously unseen subskills.

3.4.2 Evaluation Metrics

To compare the performance of LLMs against human-annotated data, we evaluated performance across several metrics, each chosen to capture a different perspective on model behavior. **Accuracy** measures the proportion of exact matches between predicted and true labels. It offers an intuitive sense of overall correctness, but does not capture the severity of errors when predictions fall close to, but not exactly on, the target proficiency level. It also does not take class imbalance into consideration. Root Mean Squared Error (**RMSE**) treats rubric levels as points on an ordinal scale and penalizes deviations according to their distance. By squaring errors before averaging, larger deviations between the predicted score and the actual score are weighted more heavily, making RMSE suitable for ordinal classification tasks. **Macro F1** calculates the F1 (harmonic mean of precision and recall) score independently for each class and then averages across classes, giving equal importance to rare and frequent proficiency levels. This setup highlights whether a model can handle minority categories, such as 'Exemplifying,' as effectively as majority ones. **Weighted F1** also averages per-class F1 scores but weights them by class frequency, producing a summary that reflects the overall distribution of labels. This setup makes it more representative of performance in imbalanced datasets where some categories dominate. **Krippendorff's α** treats the model as a second annotator and quantifies agreement with human labels beyond chance. Since it is defined for ordinal scales, α accounts for the graded nature of rubric levels, providing a more nuanced measure of reliability than accuracy alone.

It is also important to note that there is not an expectation of near perfect performance in this case. There are no "correct" or "incorrect" ways to critically think, which is the case even for human coders. Therefore, it is not expected that an LLM would detect critical thinking subskills even better than humans. Rather, it is more likely that the LLM's would perform similarly to the detection of other complex cognitive constructs in the EDM field, such as frustration, engagement, and mind wandering [32, 42, 44]. To strengthen the trustworthiness of our results, we therefore evaluate our approach using multiple complementary metrics and, for the essay-based split, report averages across five random train/validation/test splits. This setup reduces reliance on any single metric or data split and supports interpreting the results as stable trends within this exploratory dataset.

4. RESULTS

4.1 Human Coding

The independent human coding resulted in 500 essays with proficiency level codes for each subskill. Table 3 shows the distribution of essays coded for each proficiency level across subskills.

For *Synthesizing Multiple Sources*, the most prominent proficiency level was 'Not Applicable,' which represented that the essays did not cite any sources, followed by 'Below Emerging,' which signifies that the essays only cited one source. Thus, for the majority of essays (91%), there was no opportunity to evaluate the degree to which students synthesized multiple sources, since multiple sources were not cited. For *Evaluating Evidence Strength*, 'Emerging' was the most prominent proficiency level (73%), which signifies that the essays presented evidence that was linked to specific conclusions, but did not evaluate the strength or relevance of the evidence.

For *Using Counterarguments* 'Not Applicable' was the most prominent proficiency level (46%) and suggests that the essays did not include any counterarguments. The next most prominent proficiency level was 'Below Emerging (23%) which included essays that only acknowledged vague counterarguments (e.g., others may disagree; 20%). 'Expanding' was the most prominent proficiency level for *Using Facts and Opinions* (34%), which applies to essays that used more facts than opinions to support the stated claims or arguments, followed by 'Below Emerging' (28%), which applies to essays that used more opinions than facts to support the stated

Table 3. Distribution of critical thinking proficiency levels by subskill

Skill	Not Applicable	Below Emerging	Emerging	Expanding	Exemplifying
Synthesizing Multiple Sources	287	168	19	18	8
Evaluating Evidence Strength	48	73	363	14	2
Using Counterarguments	231	113	62	77	17
Using Facts and Opinions	58	140	122	172	8
Drawing Conclusions	–	66	299	113	22
Using Logical Fallacies	15	121	132	230	2

claims or arguments. This finding may be explained by the variation in essay prompts and writing tasks as a review of the essays revealed that students were likely prompted in some to give their opinion, whereas in others they were prompted to leverage facts in provided source materials.

'Emerging' was the most prominent proficiency level for *Drawing Conclusions* (60%), followed by 'Expanding' (23%). This suggests that essays were typically able to draw a specific conclusion but were more likely to analyze simple, rather than complex, relationships. For *Using Logical Fallacies*, the most prominent proficiency level was 'Expanding' (46%) with 'Emerging' as the next most prominent (26%). Essays in the present sample tended to include evidence at an equal or greater amount than logical fallacies when generating arguments.

It is also noteworthy that across subskills the 'Exemplifying' proficiency level was extremely rare (<1% – 4%). This may be due to the essays being authored by students between grades 6 and 12 while the learning progression was developed specifically for high school students. The uneven distribution of proficiency levels, especially the rarity of 'Exemplifying' responses, also highlights a broader challenge for automated critical thinking assessment: models must be validated on additional datasets with different student populations, tasks, and proficiency distributions before claims can be generalized beyond the present setting.

4.2 LLM Scoring Results

4.2.1 Subskill Scoring

In the first experiment, we compared the performance of all three methods (zero-shot prompting, few-shot prompting, fine-tuning) and all three base LLMs (GPT-5, Llama, ModernBERT) on each subskill. We show the overall results in Table 4, averaged across test sets in five random seed splits of the dataset. Overall, fine-tuning Llama yields the strongest performance. The best variant is Llama (with rubric, without justification), which conditions on the subskill-specific rubric descriptors and predicts only the label. Importantly, Llama is the only model that crosses the $\alpha \geq .6$ threshold we originally adopted as sufficient reliability for human coders. This result indicates that fine-tuning a capable base LLM on our dataset is necessary to reach satisfactory performance, considering what is deemed acceptable for human interrater agreement.

Fine-tuning ModernBERT (with rubric) is not as effective as fine-tuning Llama: it underperforms Llama across all metrics with a notable gap of 4% and 6.5% in accuracy and Krippendorff's α respectively. For ModernBERT, rubric descriptors have limited impact: ModernBERT (without rubric) is only .8% lower in accuracy, and ModernBERT (with rubric) is only 1% lower in Krippendorff's α, suggesting the model relies primarily on the essay text during fine-tuning. We also found that ModernBERT is effective at capturing majority labels but struggles with less-frequent, minority labels, reflecting the challenges of fine-tuning on an imbalanced dataset even with the focal loss. These results highlight ModernBERT's limited model size (149M parameters compared to 8B parameters), making it less capable than Llama. Also, the decoder-only nature of Llama means that it generates the label text directly, instead of using a separate classification head for each subskill for ModernBERT. Therefore, it is easier for Llama to make the most use of textual rubric definitions for subskills.

Prompting GPT-5 with few-shot (with three in-context examples) yields better performance in comparison with zero-shot (~5.5%) in terms of Krippendorff's α with minimal differences across other metrics. This result shows that the in-context examples help in classifying the minority labels, improving the overall agreement with humans. Few-shot prompting with different numbers of in-context examples (one, three, and five) yields very similar results, with three in-context examples being slightly better (by 1% on accuracy and α) showing that increasing the number of in-context examples does not yield significant gains. The modest gains from few-shot prompting for GPT-5 also highlight its strong zero-shot ability: the model can adapt to the rubric with little additional context.

Comparing fine-tuning to prompting, we see that ModernBERT, despite being a smaller model, matches GPT-5 few-shot in terms of Krippendorff's α, while Llama fine-tuning outperforms it by 6%. Overall, fine-tuning smaller, open-source models outperforms prompting larger, proprietary start-of-the-art LLMs. These results highlight the importance of training with on-task data, since LLMs are not pre-trained on our specific task of critical thinking assessment. There are also additional benefits in cost and security: few-shot prompting has significantly higher cost (≈ $40 for scoring 500 essays with GPT-5 via the OpenAI API) and raises concerns about student data privacy.

Table 4. Scoring results on the essay-based split, averaged across test sets from five random seeds

Model	Method	Accuracy	RMSE	F1 (Macro)	F1 (Weighted)	Krippendorff's α
Llama 8B	Fine-Tune	.604 ± .012	.933 ± .043	.489 ± .022	.599 ± .016	.606 ± .032
ModernBERT	Fine-Tune	.563 ± .021	1.072 ± .066	.448 ± .019	.552 ± .025	.541 ± .044
GPT-5	Zero-Shot	.543 ± .013	1.094 ± .045	.443 ± .011	.541 ± .013	.487 ± .031
	Few-Shot	.532 ± .013	1.079 ± .031	.457 ± .023	.534 ± .015	.542 ± .016

Table 5. Results showing Llama fine-tuning variants using rubrics and justification

Rubric	Justification	Accuracy	RMSE	F1 (Macro)	F1 (Weighted)	Krippendorff's α
With Rubric	Without Justification	.604 ± .012	.933 ± .043	.489 ± .022	.599 ± .016	.606 ± .032
	With Justification	.573 ± .014	.974 ± .051	.455 ± .011	.562 ± .018	.588 ± .022
Without Rubric	Without Justification	.591 ± .015	.942 ± .036	.481 ± .025	.586 ± .019	.599 ± .034
	With Justification	.576 ± .004	.983 ± .035	.462 ± .011	.563 ± .006	.584 ± .011

4.2.2 Llama 3.1 8B Results

Table 5 shows results on different variants of Llama fine-tuning using rubrics and justifications. We see that including the subskill-specific rubric descriptors in the input prompt yields only slight improvement than not including it (by 1.3% in accuracy and .7% in Krippendorff's α for the Llama [without justification] variant). This result suggests that the model attends mostly to the essay text and the subskill name when making its classification, while the rubric information does not play a very significant role. This result can perhaps be explained by the data split, where scored examples of the same subskills are already seen during training, resulting in on-task training data. However, such rubrics may be useful when we generalize to unseen subskills, which we study in Section 4.2.3. We also see that not including the justification performs slightly better than including it in the output (by 3.1% in accuracy and 1.8% in Krippendorff's α for the Llama [with rubric] variant). This result suggests that the justifications might be noisy: they are not written by humans, which may add additional distracting information during the model's decision-making process.

Table 6 shows the results for our best-performing variant, Llama (with rubric, without justification) broken down across different subskills. We see that both the subskills in *Subskill 2 - Information Analysis* achieve the highest accuracy (around .78) compared to other subskills. We note that this result is likely due to the label distribution for both subskills being highly skewed, with the model mostly predicting the majority label. However, we do see a significant difference in the Krippendorff's α scores across subskills, with *Synthesizing Multiple Sources* having a Krippendorff's α of .74, showing the fine-tuned LLM as quite consistent with human judgement. On the contrary, *Evaluating Evidence Strength* has a low Krippendorff's α score of only .23. This difference is because the model always predicts the majority 'Emerging' label for *Evaluating Evidence Strength*, ignoring the 'Not Applicable' and 'Below Emerging' labels that are not seen much during training.

We also see that both subskills under *Subskill 3 - Argument Generation* achieve the lowest accuracy among the other subskills. This result can be explained by the balanced label distribution across the proficiency levels, where subtle differences in essays led to differences in labels, making it hard for LLMs to learn during training. Despite the lower accuracy, we see that *Using Facts and Opinions* achieves a Krippendorff's α of .55, which is close to the threshold we set for interrater agreement between human coders.

Finally, both subskills in *Subskill 4 - Logical Reasoning* have higher accuracy than *Subskill 3 - Argument Generation* but lower than those of *Subskill 2 - Information Analysis*. In particular, *Using Logical Fallacies* has an accuracy of only .50, compared to .61 for *Drawing Conclusions*. This result underscores the challenge in identifying different types of logical fallacies and differentiating them from the evidence, since the fallacies are often subtle and rarely labeled explicitly.

Taken together, these results suggest that fine-tuning Llama performs best in contexts where different proficiency levels are separable and well-represented. Performance significantly drops when different proficiency levels have minor differences or the labeled data is dominated by one level. Therefore, it is important to evaluate models with metrics beyond accuracy before considering potential classroom use.

4.2.3 Generalizing to Unseen Subskills

In this experiment, we evaluate whether a Llama model fine-tuned on labeled data on some subskills can generalize to other subskills that are previously unseen during training. Intuitively, since the definition of subskills and scoring rubrics defining each proficiency level are used as input, a trained model may acquire some knowledge on how to evaluate students' critical thinking as a whole, as well as gain knowledge on the operational definitions of the different proficiency levels. Therefore, the trained model may perform better in the cold-start setting, where human scores are not readily available for previously unseen subskills [64].

Table 7 shows results on the subskill-based data split where we see a clear performance drop compared to the essay-based split. Among the variants, Llama (with rubric, with justification) performs best in this setting, achieving Krippendorff's α around .42. However, the without-rubric variants, either with or without justification, achieve α values near zero. This result suggests that rubric descriptors are essential for cross-subskill generalization: Including the rubric provides the model with the context it needs to map an unseen subskill to proficiency level labels. This observation is in stark contrast to the essay-based split, where rubrics are not

Table 6. Subskill-wise results for the best Llama fine-tuning variant

Subskill	Accuracy	RMSE	F1 (Macro)	F1 (Weighted)	Krippendorff's α
Synthesizing Multiple Sources	.780 ± .037	.638 ± .108	.444 ± .061	.761 ± .040	.740 ± .063
Evaluating Evidence Strength	.768 ± .026	.707 ± .097	.303 ± .043	.711 ± .033	.234 ± .126
Using Counterarguments	.471 ± .042	1.408 ± .170	.236 ± .050	.416 ± .065	.315 ± .148
Using Facts and Opinions	.487 ± .024	.958 ± .045	.325 ± .044	.440 ± .044	.555 ± .032
Drawing Conclusions	.613 ± .033	.672 ± .033	.361 ± .057	.555 ± .053	.382 ± .091
Using Logical Fallacies	.503 ± .027	.967 ± .032	.348 ± .036	.450 ± .061	.411 ± .049

Table 7. Scoring results on the subskill-based split for Llama fine-tuning

Test Subskill	Accuracy	RMSE	F1 (Macro)	F1 (Weighted)	Krippendorff's α
Information Analysis	0.564	1.221	0.358	0.549	0.379
Argument Generation	0.389	1.357	0.293	0.356	0.418
Logical Reasoning	0.418	1.066	0.310	0.406	0.302
Overall	0.457	1.22	0.392	0.448	0.422

essential since the data split does not test generalizability to new subskills. For the rubric-based variants, adding justifications improves generalization: Llama (with rubric, with justification) outperforms Llama (with rubric, without justification) by 10% in accuracy and 6% in Krippendorff's α. This result differs from the essay-based split, where justifications provided little benefit. This observation suggests that generating justifications improves the model's reasoning by encouraging it to explain its label using rubric descriptors and evidence from the essay, which matters more when the target subskill is unseen during training.

Unfortunately, accuracy is low for some subskills (between 0.39 and 0.56), and Krippendorff's α falls below the reliability threshold we set for human annotators. These results suggest that models struggle to generalize when the target subskill is unseen during training. This observation underscores the importance of subskill-specific context for guiding classification. In particular, models rely on the rubric descriptors for each subskill; removing this information during training substantially degrades performance. These results suggest that effective scoring of new subskills will require subskill-specific labeled data.

Despite this difficulty in generalization, Information Analysis achieves the highest accuracy among the unseen subskills (.56). This result likely follows because the subskill has simpler classification boundaries and higher human interrater agreement, which makes the dominant label easier for the model to identify and predict consistently. Argument Generation achieves the highest Krippendorff's α (.41) but lower accuracy and higher RMSE. This result suggests that the model struggles to separate adjacent labels when the label distribution is more balanced, leading to confusion among intermediate levels. Logical Reasoning achieves the lowest Krippendorff's α (.30). This result likely reflects the complexity of the subskill, especially *Using Logical Fallacies*, which requires recognizing specific fallacy types and distinguishing them from evidence.

Taken together, these findings highlight the difficulty of generalizing across subskills without on-task training data. These findings also show some promise in generalization, with accuracy comparable to the essay-based split for a few subskills and fair Krippendorff's α scores across all subskills. Compared to GPT-5 zero-shot, which does not rely on training data for unseen subskills either since it is a prompting method, Llama fine-tuning performs slightly worse, by 8.6% overall in accuracy and 6.5% overall in Krippendorff's α. This drop in performance likely follows from subskill definitions and rubric criteria being sufficiently distinct, making extrapolation to other subskills difficult. Consequently, robust scoring of unseen subskills remains challenging and may require some subskill-specific supervision or LLM fine-tuning techniques that encode subskill-agnostic features.

Error Analysis

We conducted an error analysis of the best-performing method, fine-tuning Llama (with rubric, with justification) on *Using Logical Fallacies* (Subskill 4.2). The most common misclassifications are 'Expanding' classified as 'Emerging' (80), 'Emerging' classified as 'Below Emerging' (53), and 'Emerging' classified as 'Expanding' (46). We qualitatively examined three representative examples, one for each misclassification type.

The first example discusses electoral rules. The essay includes concrete evidence such as "The Electoral College consists of 538 electors… 270 electoral votes" and "Because of the winner-take-all system… the candidates don't spend much time." The model instead foregrounds fairness appeals such as "It's not fair Texas gets 38… New York only gets 3," and similar overgeneralizations as logical fallacies and minimizes evidence, leading it to predict 'Emerging' instead of 'Expanding'. The second example discusses classroom emotion-detection software. The essay includes technical evidence describing the Facial Action Coding System, such as "orbicularis makes crow's feet around your eyes" and "the mouth is stretched sideways using the zygomatic major and a different muscle called the risorius." The model instead emphasizes slippery-slope and speculative claims (e.g., "more and more ads… you might even get a virus," "watching you all day, everyday"), rather than "concrete evidence," leading it to predict 'Below Emerging' instead of 'Emerging'. The third example discusses Venus exploration. The essay includes research-like statements such as "far more extreme than anything humans encounter on Earth" and "97 percent carbon dioxide blankets Venus." The model instead gives disproportionate weight to the volume of citations while not recognizing contradictions in the essay's conclusion, leading it to predict 'Expanding' instead of 'Emerging'.

Taken together, these results indicate that most misclassifications occur between adjacent proficiency levels. These errors are most frequent around 'Emerging', which requires judging whether evidence and fallacies are roughly balanced. This judgment is difficult when fallacy detection depends on fine-grained distinctions. This result can likely be attributed to the inherent difficulty of identifying specific fallacy types (e.g., causation, false equivalence). These distinctions were provided to human coders but are not explicitly available to the model [39]. Our findings suggest that while LLMs can approximate human scoring, biased judgments of the balance between evidence and fallacies underscore the need for additional scaffolding or hybrid human-AI approaches for critical thinking assessment. Future work can test whether providing explicit fallacy definitions and labeled examples improves performance.

5. DISCUSSION AND CONCLUSION

In this paper, we developed and evaluated automated methods for assessing critical thinking subskills in student essays, attempting to address the pressing need for more reliable and robust measures of this dynamic and understudied form of higher-order reasoning. Our work used two complementary strategies: prompting proprietary LLMs under both zero-shot and few-shot conditions and fine-tuning open-source LLMs using human-labeled data. By doing so, we investigated the capability of these systems to detect critical

thinking subskills, such as constructing coherent arguments, systematically integrating evidence, and articulating meaningful counterarguments. Our aim is to provide the foundation for future EDM work on modeling critical thinking so that eventually such detectors can be used in educational environments.

Overall, these methods involve trade-offs between scoring quality and deployment constraints. Fine-tuning open-source models achieves the strongest performance, with Llama fine-tuning outperforming and ModernBERT matching, respectively, GPT-5 few-shot's performance in Krippendorff's α, despite their smaller size. This result highlights the value of on-task training data for critical thinking assessment. Prompting remains easier to deploy, but it can be more expensive at scale (≈ $40 for scoring 500 essays with GPT-5) and raises student data privacy concerns. At the same time, even for our best model, Llama 3.1 8B fine-tuning using rubrics and justifications, we noted inconsistent results across subskills. The model was especially challenged by subskills where there was a class imbalance and only subtle (rather than clearer) distinctions across proficiency levels. These challenges suggest that we need to incorporate data-level balancing (e.g., synthetic minority over-sampling) and cost-sensitive training into LLM fine-tuning for critical thinking assessment [59].

Our experiments with fine-tuning Llama show that adapting to unseen subskills remains difficult without on-task training data. Rubric descriptors are necessary for cross-subskill generalization, and generating justifications further improves performance. This result likely follows because justifications encourage the model to explain its label using rubric criteria and evidence from the essay, which helps it map an unseen subskill to the proficiency levels more consistently. However, reliability remains below our target threshold. Performance also drops relative to settings where the target subskill is seen during training. This finding suggests that robust scoring of new subskills will require some subskill-specific human supervision, such as a small set of labeled essays or curated in-context examples. Future work may also investigate meta-prompting methods to automatically generate rubrics given description of previously unseen skills. Deploying automated critical thinking assessment in practice will require either subskill-specific labeled data or mechanisms to encode transferable traits across related subskills.

We acknowledge that selecting 0.6 as a threshold for reliability (compared to typical 0.8) affects the interpretation and use of our findings. However, this decision was intentional given the exploratory nature of this work and our long-term goal of developing a model for critical thinking that can be used in AI tools. A limitation of this initial work is that we evaluated a small set of models and prompting settings, even as the space of LLMs and prompting strategies for AES and related tasks continues to expand. It is therefore possible that other proprietary models (o3, o4-mini-high, Opus 4.6), alternative prompting strategies, or different open-weight models (Qwen [3], or DeepSeek [12]) would yield stronger performance for critical thinking assessment. The bottom line is that our experiments provide evidence that open-weight LLMs, after some training, can be competitive in this setting. The rapid pace of LLM development remains a challenge for interpreting and comparing results over time. Similarly, a broader range of critical thinking skills and contexts for its application will need to be studied before we can confidently draw conclusions about the usefulness of LLMs for measuring critical thinking as a whole.

Nonetheless, these preliminary results demonstrate that there is potential for LLMs to assess a range of critical thinking skills efficiently, rapidly, and cost-effectively, in the context of authentic student work. LLMs can potentially become a tool to support expanded and more extensive research on critical thinking, the relationship between its components, the processes involving its development, and its relationships to other forms of complex student cognition. This work lays the groundwork for AI models that can give teachers meaningful insights into how students' critical thinking is developing and that can be embedded in adaptive learning systems to help cultivate the skills humans will need most as they navigate an era of AI-generated content, disinformation, and increasingly complex global challenges.

6. REFERENCES

[1] Alsaleh, N.J. 2020. Teaching critical thinking skills: Literature review. *The Turkish Online Journal of Educational Technology*. 19, 1 (2020), 21–39.

[2] Attali, Y. and Burstein, J. 2006. Automated essay scoring With e-rater® V.2. *The Journal of Technology, Learning, and Assessment*. 4, 3 (Feb. 2006).

[3] Bai, J. et al. 2023. Qwen technical report. arXiv.

[4] Bailin, S., Case, R., Coombs, J.R. and Daniels, L.B. 1999. Conceptualizing critical thinking. *Journal of Curriculum Studies*. 31, 3 (May 1999), 285–302. https://doi.org/10.1080/002202799183133.

[5] Bailin, S. and Siegel, H. 2003. Critical thinking. *The Blackwell guide to the philosophy of education*. N. Blake, P. Smeyers, R. Smith, and P. Standish, eds. Wiley. 181–193.

[6] Borse, N.S., Subramaniam, R.C. and Rebello, N.S. 2025. Investigation of the inter-rater reliability between large language models and human raters in qualitative analysis. arXiv.

[7] Butler, H.A. 2012. Halpern critical thinking assessment predicts real-world outcomes of critical thinking. *Applied Cognitive Psychology*. 26, 5 (Sept. 2012), 721–729. https://doi.org/10.1002/acp.2851.

[8] Butler, H.A. 2024. Predicting everyday critical thinking: A review of critical thinking assessments. *Journal of Intelligence*. 12, 2 (Feb. 2024), 16. https://doi.org/10.3390/jintelligence12020016.

[9] Chen, J., Lotsos, A., Zhao, L., Wang, G., Wilensky, U., Sherin, B. and Horn, M. 2024. Prompts matter: Comparing ML/GAI approaches for generating inductive qualitative coding results. arXiv.

[10] Cornell Critical Thinking Test Level X: *https://www.criticalthinking.com/cornell-critical-thinking-test-level-x.html?srsltid=Afm-BOooDV8JaM31Gi2bROu9XXsqhWxR_ti7Rkha8H_Rw1z4rEPuzGWlC*.

[11] Crossley, S.A., Tian, Y., Baffour, P., Franklin, A., Benner, M. and Boser, U. 2024. A large-scale corpus for assessing written argumentation: PERSUADE 2.0. *Assessing Writing*. 61, (July 2024), 100865. https://doi.org/10.1016/j.asw.2024.100865.

[12] DeepSeek-AI et al. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv.

[13] Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (Minneapolis, Minnesota, 2019), 4171–4186.

[14] Dumitru, D. and Halpern, D.F. 2023. Critical thinking: Creating job-proof skills for the future of work. *Journal of Intelligence*. 11, 10 (Oct. 2023), 194. https://doi.org/10.3390/jintelligence11100194.

[15] Ennis, R.H. 1989. Critical thinking and subject specificity: Clarification and needed research. *Educational Researcher*. 18, 3 (Apr. 1989), 4–10. https://doi.org/10.3102/0013189X018003004.

[16] Ennis, R.H. 1993. Critical thinking assessment. *Theory Into Practice*. 32, 3 (June 1993), 179–186. https://doi.org/10.1080/00405849309543594.

[17] Ennis, R.H. 2011. *The nature of critical thinking: An outline of critical thinking dispositions and abilities*. University of Illinois.

[18] Ennis, R.H. and Weir, E. 1985. *The Ennis-Weir Critical Thinking Essay Test*. Midwest Publications.

[19] Facione, P.A. 1990. *Critical thinking: A statement of expert consensus for purposes of educational assessment and instruction*. American Philosophical Association.

[20] Facione, P.A. and Facione, N.C. 1994. Holistic Critical Thinking Scoring Rubric (HCTSR). California Academic Press / Insight Assessment.

[21] Feedback Prize – Predicting Effective Arguments: *https://kaggle.com/competitions/feedback-prize-effectiveness*.

[22] Grattafiori, A. et al. 2024. The Llama 3 herd of models. arXiv.

[23] Halpern Critical Thinking Assessment: *http://www.schuhfried.com/ vienna-test-system-vts/all-tests-from-a-z/test/hcta-halpern-critical-thinkingassessment-1/*.

[24] Halpern, D.F. 1998. Teaching critical thinking for transfer across domains: Disposition, skills, structure training, and metacognitive monitoring. *American Psychologist*. 53, 4 (1998), 449–455. https://doi.org/10.1037/0003-066X.53.4.449.

[25] Hayes, A.F. and Krippendorff, K. 2007. Answering the call for a standard reliability measure for coding data. *Communication Methods and Measures*. 1, 1 (Apr. 2007), 77–89. https://doi.org/10.1080/19312450709336664.

[26] Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S. and Chen, W. 2022. LoRA: Low-rank adaptation of large language models. arXiv.

[27] Ke, Z. and Ng, V. 2019. Automated essay scoring: A survey of the state of the art. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence* (Macao, China, Aug. 2019), 6300–6308.

[28] Kovanović, V., Joksimović, S., Waters, Z., Gašević, D., Kitto, K., Hatala, M. and Siemens, G. 2016. Towards automated content analysis of discussion transcripts: a cognitive presence case. *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge - LAK '16* (Edinburgh, United Kingdom, 2016), 15–24.

[29] Krippendorff, K. 2004. *Content analysis: An introduction to its methodology*. Sage.

[30] Kuhn, D. 1999. A developmental model of critical thinking. *Educational Researcher*. 28, 2 (Mar. 1999), 16–46. https://doi.org/10.3102/0013189X028002016.

[31] Kuhn, D. 2018. A role for reasoning in a dialogic approach to critical thinking. *Topoi*. 37, 1 (Mar. 2018), 121–128. https://doi.org/10.1007/s11245-016-9373-4.

[32] Kuvar, V., Kam, J.W.Y., Hutt, S. and Mills, C. 2023. Detecting when the mind wanders off task in real-time: An overview and systematic review. *International Conference on Multimodal Interaction* (Paris France, Oct. 2023), 163–173.

[33] Lai, E.R. 2011. *Critical thinking: A literature review*. Pearson.

[34] Lee, S., Cai, Y., Meng, D., Wang, Z. and Wu, Y. 2024. Unleashing large language models' proficiency in zero-shot essay scoring. *Findings of the Association for Computational Linguistics: EMNLP 2024* (Miami, Florida, USA, 2024), 181–198.

[35] Li, S. and Ng, V. 2024. Automated essay scoring: A reflection on the state of the art. *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing* (Miami, Florida, USA, 2024), 17876–17888.

[36] Lin, T.-Y., Goyal, P., Girshick, R., He, K. and Dollar, P. 2017. Focal loss for dense object detection. *2017 IEEE International Conference on Computer Vision (ICCV)* (Venice, Italy, Oct. 2017), 2999–3007.

[37] Liu, O.L., Frankel, L. and Roohr, K.C. 2014. Assessing critical thinking in higher education: Current state and directions for next-generation assessment. *ETS Research Report Series*. 2014, 1 (June 2014), 1–23. https://doi.org/10.1002/ets2.12009.

[38] Liu, X., Zambrano, A.F., Baker, R.S., Barany, A., Ocumpaugh, J., Zhang, J., Pankiewicz, M., Nasiar, N. and Wei, Z. 2025. Qualitative coding with GPT-4: Where it works better. *Journal of Learning Analytics*. 12, 1 (Mar. 2025), 169–185. https://doi.org/10.18608/jla.2025.8575.

[39] Logical fallacies: *https://owl.purdue.edu/owl/general_writing/academic_writing/logic_in_argumentative_writing/fallacies.html*.

[40] Marzi, G., Balzano, M. and Marchiori, D. 2024. K-Alpha Calculator–Krippendorff's Alpha Calculator: A user-friendly tool for computing Krippendorff's Alpha inter-rater reliability coefficient. *MethodsX*. 12, (June 2024), 102545. https://doi.org/10.1016/j.mex.2023.102545.

[41] McPeck, J.E. 1990. Critical thinking and subject specificity: A reply to Ennis. *Educational Researcher*. 19, 4 (May 1990), 10–12. https://doi.org/10.3102/0013189X019004010.

[42] Mills, C., Bosch, N., Graesser, A. and D'Mello, S. 2014. To quit or not to quit: Predicting future behavioral disengagement from reading patterns. *Intelligent Tutoring Systems*. S. Trausan-Matu, K.E. Boyer, M. Crosby, and K. Panourgia, eds. Springer International Publishing. 19–28.

[43] Misiejuk, K., Kaliisa, R. and Scianna, J. 2024. Augmenting assessment with AI coding of online student discourse: A question of reliability. *Computers and Education: Artificial Intelligence*. 6, (June 2024), 100216. https://doi.org/10.1016/j.caeai.2024.100216.

[44] Ocumpaugh, J., Baker, R., Gowda, S., Heffernan, N. and Heffernan, C. 2014. Population validity for educational data mining models: A case study in affect detection. *British Journal of Educational Technology*. 45, 3 (May 2014), 487–501. https://doi.org/10.1111/bjet.12156.

[45] Pankiewicz, M., Zambrano, A.F., Barany, A. and Baker, R.S. 2024. How we code code: Leveraging GPT and ordered networks for understanding introductory programming education. *Advances in Quantitative Ethnography*. Y.J. Kim and Z. Swiecki, eds. Springer Nature Switzerland. 225–240.

[46] Pasquinelli, E., Farina, M., Bedel, A. and Casati, R. 2021. Naturalizing critical thinking: Consequences for education, blueprint for future research in cognitive science. *Mind, Brain, and Education*. 15, 2 (May 2021), 168–176. https://doi.org/10.1111/mbe.12286.

[47] Paul, R. and Elder, L. 2006. *The miniature guide to critical thinking: Concepts and tools*.

[48] Phandi, P., Chai, K.M.A. and Ng, H.T. 2015. Flexible domain adaptation for automated essay scoring using correlated linear regression. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (Lisbon, Portugal, 2015), 431–439.

[49] Poce, A., Amenduni, F., De Medio, C. and Norgini, A. 2020. Assessing critical thinking in open-ended answers: An automatic approach. *European Distance and E-Learning Network (EDEN) Proceedings*. 1 (June 2020), 109–116. https://doi.org/10.38069/edenconf-2020-ac0008.

[50] Poce, A., De Medio, C., Amenduni, F. and Re, M.R. 2019. Critical Thinking assessment: A first approach to the automatic evaluation. *2019 18th International Conference on Information Technology Based Higher Education and Training (ITHET)* (Magdeburg, Germany, Sept. 2019), 1–8.

[51] Riordan, B., Horbach, A., Cahill, A., Zesch, T. and Chong Min, L. 2017. Investigating neural architectures for short answer scoring. *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications* (Copenhagen, Denmark, 2017), 159–168.

[52] Santos Meneses, L.F. 2020. Critical thinking perspectives across contexts and curricula: Dominant, neglected, and complementing dimensions. *Thinking Skills and Creativity*. 35, (Mar. 2020), 100610. https://doi.org/10.1016/j.tsc.2019.100610.

[53] Seßler, K., Fürstenberg, M., Bühler, B. and Kasneci, E. 2025. Can AI grade your essays? A comparative analysis of large language models and teacher ratings in multidimensional essay scoring. *Proceedings of the 15th International Learning Analytics and Knowledge Conference* (Dublin, Ireland, Mar. 2025), 462–472.

[54] Simon, S., Sankaranarayanan, S., Tajik, E., Borchers, C., Shahrokhian, B., Balzan, F., Strauß, S., Viswanathan, S.A., Ataş, A.H., Čarapina, M., Liang, L. and Celik, B. 2025. Comparing a human's and a multi-agent system's thematic analysis: Assessing qualitative coding consistency. Cham: Springer Nature Switzerland.

[55] Skills for the Future: A new way to measure what matters: *https://www.ets.org/skills-for-future.html*.

[56] Stahl, M., Michel, N., Kilsbach, S., Schmidtke, J., Rezat, S. and Wachsmuth, H. 2024. A school student essay corpus for analyzing interactions of argumentative structure and quality. *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (Mexico City, Mexico, 2024), 2661–2674.

[57] Sternberg, R.J. 1986. *Critical thinking: Its nature, measurement, and improvement*. National Institute of Education.

[58] Tai, R.H., Bentley, L.R., Xia, X., Sitt, J.M., Fankhauser, S.C., Chicas-Mosier, A.M. and Monteith, B.G. 2024. An examination of the use of large language models to aid analysis of textual data. *International Journal of Qualitative Methods*. 23, (Jan. 2024), 1–14. https://doi.org/10.1177/16094069241231168.

[59] Tan, J.S., Tan, I.K.T., Lay Ki Soon and Ong, H.F. 2022. Improved automated essay scoring using Gaussian multi-class SMOTE for dataset sampling. *Proceedings of the 15th International Conference on Educational Data Mining* (Durham, United Kingdom, July 2022).

[60] Thornhill-Miller, B., Camarda, A., Mercier, M., Burkhardt, J.-M., Morisseau, T., Bourgeois-Bougrine, S., Vinchon, F., El Hayek, S., Augereau-Landais, M., Mourey, F., Feybesse, C., Sundquist, D. and Lubart, T. 2023. Creativity, critical thinking, communication, and collaboration: Assessment, certification, and promotion of 21st century skills for the future of work and education. *Journal of Intelligence*. 11, 3 (Mar. 2023), 54. https://doi.org/10.3390/jintelligence11030054.

[61] Warner, B., Chaffin, A., Clavié, B., Weller, O., Hallström, O., Taghadouini, S., Gallagher, A., Biswas, R., Ladhak, F., Aarsen, T., Adams, G.T., Howard, J. and Poli, I. 2025. Smarter, better, faster, longer: A Modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vienna, Austria, 2025), 2526–2547.

[62] Watson-Glaser Critical™ Thinking Appraisal: *https://www.talentlens.com/watson-glaser.html*.

[63] Watts, F.M. and Finkenstaedt-Quinn, S.A. 2021. The current state of methods for establishing reliability in qualitative chemistry education research articles. *Chemistry Education Research and Practice*. 22, 3 (2021), 565–578. https://doi.org/10.1039/D1RP00007A.

[64] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E.H., Le, Q.V. and Zhou, D. 2022. Chain-of-thought prompting elicits reasoning in large language models. *36th Conference on Neural Information Processing Systems (NeurIPS 2022)* (New Orleans, 2022).

[65] Xiao, C., Ma, W., Song, Q., Xu, S.X., Zhang, K., Wang, Y. and Fu, Q. 2025. Human-AI collaborative essay scoring: A dual-process framework with LLMs. *Proceedings of the 15th International Learning Analytics and Knowledge Conference* (Dublin, Ireland, Mar. 2025), 293–305.

[66] Yeginbergen, A., Oronoz, M. and Agerri, R. 2025. Dynamic knowledge integration for evidence-driven counter-argument generation with large language models. *Findings of the Association for Computational Linguistics: ACL 2025* (Vienna, Austria, 2025), 22568–22584.

7. ACKNOWLEDGMENTS

The research reported here was supported by AugmentED, which is part of the of the Advanced Education Research and Development Fund (AERDF). The opinions expressed are those of the authors and do not necessarily represent views of the AERDF.

UNVaMP: Neural Knowledge Tracing with Variational Regularization of Latent Knowledge Dynamics

Carson J. Cook
Amplify Education, Inc.
ccook@amplify.com

Ahmed J. Zerouali
Amplify Education, Inc.
azerouali@amplify.com

Anthony Schmidt
Amplify Education, Inc.
aschmidt@amplify.com

Reginald Ziedzor
Amplify Education, Inc.
rziedzor@amplify.com

Paul Lin
Amplify Education, Inc.
plin@amplify.com

Luke G. Eglington
Amplify Education, Inc.
leglington@amplify.com

ABSTRACT

We introduce the Unified Neural Variational Measurement of Proficiency (UNVaMP) architecture, a knowledge tracing method that integrates observed student-item interactions with internal memory to produce evolving latent representations of student knowledge. These representations support accurate predictions of future responses while enabling explicit control over the smoothness of estimated learning trajectories. UNVaMP can be configured as either a purely neural model or a hybrid model that predicts responses through an interpretable measurement function over the latent space. We show that a pure neural configuration (UNVaMP-MLP) achieves the strongest predictive performance among compared models on three out of four datasets. Meanwhile, a hybrid configuration (UNVaMP-MIRT, using a 1PL MIRT measurement function) lags only slightly behind UNVaMP-MLP, indicating that the predictive cost of interpretability is modest.

Beyond predictive accuracy, UNVaMP provides the following: a principled mechanism for controlling volatility when estimating student latent variables, quantification of uncertainty over student knowledge state estimates, and flexible input specification that supports heterogeneous student-item interaction features. In addition, the hybrid UNVaMP-MIRT configuration generates interpretable moment-in-time student knowledge state estimates. Using an experimental dataset, we show that auxiliary inputs induce structured changes in the predictive behavior of UNVaMP-MIRT, consistent with sensitivity to underlying structure beyond response correctness. Furthermore, through a simulation study, we show that UNVaMP yields well-behaved knowledge state estimates under controlled measurement conditions. In total, these results indicate that UNVaMP is both practically useful for real-world education systems and capable of recovering meaningful underlying structure from student-item interactions.

Carson Cook, Ahmed Zerouali, Anthony Schmidt, Reginald Ziedzor, Paul Lin, and Luke Eglington. UNVaMP: Neural Knowledge Tracing with Variational Regularization of Latent Knowledge Dynamics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 353–364. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040074

Keywords

Knowledge Tracing, Deep Learning, Latent Variables, Interpretability, Uncertainty Quantification

1. INTRODUCTION

Building models of student learning often involves balancing competing needs for rigorous student knowledge estimation and flexible model fitting over large, sparse, and noisy datasets, while recognizing that student knowledge evolves with each interaction. In contrast, classical measurement models are typically designed for tightly controlled test environments (e.g., standardized tests) and impose strict assumptions about student latent knowledge structures. Knowledge tracing (KT) models seek to incorporate sequential (or temporal) information about student-item interactions and fall into two broad categories: those rooted in classical statistical frameworks [13, 33, 36], and those built upon modern deep learning methods [10, 27, 37]. Classical approaches often preserve much of the framing of traditional measurement models but relax certain constraints to capture learning dynamics and may support less controlled settings. Modern deep learning approaches to KT typically relax both statistical and latent knowledge assumptions in pursuit of flexibility and predictive power, although hybrid and theory-informed methods are increasingly prevalent in the literature.

In the remainder of this section, we briefly summarize the measurement and KT paradigms, followed by a discussion of existing KT methods, along with their strengths and limitations, then distill from this analysis a set of architectural desiderata outlining desirable properties for modern KT architecture designs. The UNVaMP architecture is then described in detail in Section 2. We then describe our datasets and experiments in Section 3, followed by a presentation of our results in Section 4. Finally, we summarize our findings and discuss limitations and future work in Section 5.

1.1 Classical Measurement Models

In classical measurement, models are built around an underlying construct of interest, typically represented by one or more latent variables that are assumed to be static within the measurement window [4]. In such models, student and item parameters are estimated directly, often via maximum likelihood, the expectation-maximization (EM) algorithm, or related variational approaches, rather than through amortized inference [20]. Although there are limitations with

these models in terms of predictive performance and flexibility, they are popular due to their interpretability, well-understood statistical properties, and mature implementations. Two very popular classical frameworks are Item Response Theory (IRT) and Cognitive Diagnostic Modeling (CDM). We briefly discuss IRT below, but point readers towards other materials discussing the properties and usage of CDMs [41, 44].

1.1.1 Item Response Theory

IRT is a classical measurement framework in which observed student-item interactions are modeled as a probabilistic function of latent student proficiency and item characteristics [20]. Multidimensional IRT (MIRT) extends this framework by defining latent proficiency as a vector, where each dimension represents a latent skill [39]. For instance, the two-parameter logistic (2PL) MIRT model is formulated as

$$P\left[Y_{i,j} = 1 \mid \theta_i, a_j, d_j\right] = \frac{1}{1 + \exp\{-(a_j^\top \theta_i + d_j)\}},$$

where $Y_{i,j} \in \{0, 1\}$ denotes an incorrect or correct student response to item j, θ_i is the latent proficiency vector for student i, a_j is the vector of item *discrimination* (or *loading*) parameters, and d_j is a scalar item intercept related to that item's *difficulty*. The vectors θ_i and a_j have one dimension per latent skill. The one-parameter logistic (1PL) MIRT model, also known as the multidimensional Rasch model, is a restricted case of 2PL MIRT where the dimensions of a_j have fixed values (typically 0 or 1).

1.2 Knowledge Tracing

Knowledge tracing is a broad category of methods that, unlike classical measurement models, treat student knowledge as evolving throughout a sequence of student-item interactions [42]. We group KT approaches into two broad categories: those grounded in classical measurement philosophies or statistical frameworks, and those built on modern deep learning methods. In our discussion of deep learning knowledge tracing (DLKT), we examine approaches that focus primarily on prediction, as well as hybrid methods that incorporate elements of classical measurement.

1.2.1 Classical and Statistical KT Methods

Classical and statistical KT methods typically model student knowledge directly and may utilize a longitudinal or dynamic adaptation of a classical measurement model. Popular families include Bayesian Knowledge Tracing (BKT) [13, 2], Elo [19, 36], and various dynamic IRT and CDM models [45, 53, 49, 24, 22, 7, 52]. Many variants exist, but all primarily consist of estimating student and item parameters given a sequence of student-item interactions.

Within classical KT, another well-studied and typically high-performing family of models is Logistic Knowledge Tracing (LKT) [32]. LKT is a framework that attempts to unify the many previous KT models that use logistic regression as their core predictive model. These prior models typically use hand-engineered features that could in principle be used together but were not (e.g., AFM [5], PFA [31], variants of PFA [18], R-PFA [14], spacing models like PPE [46, 47], and other features inspired by cognitive psychology [30]). However, selecting among these candidate features is challenging, so feature selection using lasso regression was included in the framework. The lasso regression approach was recently found to outperform several popular deep learning approaches on four benchmark datasets [35].

Despite their conceptual clarity, classical methods often suffer from several major shortcomings. First, longitudinal variants of IRT and CDM often require re-fitting student (and possibly item) parameters to obtain updated estimates given new observations [49]. Other variants of these IRT and CDM methods, along with models like Elo [36], support using updating rules for real-time parameter re-estimation, but these require analytical derivations or heuristics that limit model expressiveness, and these methods may still struggle to scale with larger datasets [25]. In addition, when many skills are present (e.g., dozens or hundreds), especially in large datasets, CDM and IRT methods may fail to converge without imposing significant assumptions or structural simplifications. Finally, classical approaches typically do not flexibly support auxiliary inputs beyond minimal predictors (e.g., sequential binary responses). Those that do, such as explanatory IRT or LKT models, utilize manually engineered features and require strong assumptions about functional relationships between inputs and outputs [56]. This manual approach can become cumbersome as new data sources become readily available (e.g., open-ended text responses), requiring significant resources to develop new features or adapt existing ones. This limitation extends to features intended to act as latent variables representing student knowledge, such as those intended to behave similarly to Elo parameters [34], since their interpretation will be confounded by the inclusion of new auxiliary inputs.

1.2.2 DLKT Methods

In response to the limitations of classical methods, numerous DLKT architectures have emerged over the past decade, ranging from those focused primarily on obtaining maximum predictive accuracy, to hybrid approaches that incorporate interpretable representations of student knowledge, item properties, or learning dynamics.

Prediction-focused DLKT architectures generally forgo interpretable student or item representations, limiting their scope to next-response prediction in sequences of student-item interactions. The first major example was DKT [37], a recurrent architecture focused exclusively on prediction. Following its introduction, numerous DLKT models have been proposed, many achieving state-of-the-art performance at the time of publication. Prominent examples include SAKT [27], AKT [16], and SAINT [10]. Within the prediction-focused DLKT paradigm, various extensions have been explored, such as incorporating uncertainty estimation [8, 12, 40]. DLKT models can exhibit volatility and unintuitive behavior in student learning dynamics [23, 59, 54, 62, 60]. Proposed remedies have included penalization of output-space inconsistency [59], attention mechanisms [54], and contrastive training procedures [60]. Despite these advances, such models remain largely uninterpretable and focused on prediction.

To address these interpretability limitations, several hybrid architectures incorporating elements of classical measurement, such as IRT- or CDM-based outputs, have been pro-

posed [51, 55, 57, 43]. Many early hybrid methods assumed static data generation and did not model learning dynamics. More recently, dynamic hybrid architectures suitable for KT have emerged, incorporating interpretability constraints such as IRT-style response functions [58, 6, 63], CDM-inspired interaction functions [50], or structured dynamics [28, 48]. However, to our knowledge, methods that introduce interpretability into the prediction mechanism do not tend to incorporate accompanying dynamics regularization [58, 6], while methods that impose structure on dynamics do not produce interpretable knowledge states [28, 48], and uncertainty representation is often left unaddressed.

1.3 Architecture Desiderata

Although knowledge tracing models are often judged primarily by predictive accuracy, prior work has argued that additional attributes are critical for practical usability [38]. We propose a set of architectural desiderata to guide DLKT architecture design and address common challenges with existing approaches. While many models satisfy subsets of these desiderata, we are unaware of any that explicitly address each of them.

1.3.1 Regularization of Dynamics

The volatility issues that have been observed in the DLKT paradigm may hamper their usability in many real-world contexts. In deployed systems, models with poorly constrained dynamics can yield unstable proficiency estimates that undermine reliability and interpretability. Accordingly, DLKT methods should incorporate explicit regularization mechanisms that promote smooth, stable trajectories of the inferred student knowledge states.

1.3.2 Interpretable Knowledge States

Although many DLKT architectures exist, few output interpretable knowledge states. Full transparency into all internal mechanisms (e.g., dynamics and response prediction) is ideal but impractical in DLKT without imposing constraints that undermine the benefits of deep learning. Instead, we suggest that moment-in-time summaries of student knowledge are sufficient for many practical use cases, provided the underlying dynamics are well-regularized, even if those dynamics are not themselves interpretable. To date, many hybrid methods either provide only partial knowledge state interpretability, or are too constrained to fully leverage DLKT's expressiveness. Thus, a model that allows users to specify an interpretable latent knowledge representation while retaining flexibility in modeling student learning dynamics would be highly useful.

1.3.3 Uncertainty Quantification

A common challenge for DLKT models is the absence of distributional and structural assumptions that underpin classical uncertainty quantification, making it difficult to define appropriate standards for uncertainty updating and calibration in practice. Nevertheless, a DLKT model should provide an internally consistent representation of uncertainty that reflects changes in its confidence as evidence accumulates. Even without statistical guarantees, such uncertainty estimates have practical utility, such as revealing diminishing returns as data accrue, informing when inference outputs should be surfaced, and enabling diagnostic analyses related to external signals such as proportion correct or recency.

1.3.4 Flexibility of Inputs

A major benefit of deep learning is that it relaxes two onerous requirements of classical modeling: pre-specifying functional relationships between inputs and outputs, and hand-engineering features to increase model capacity. Accordingly, a DLKT model should be capable of incorporating arbitrary sets of inputs, provided they do not introduce information leakage with respect to the response. While classical methods can, in principle, be expanded to accommodate additional inputs (e.g., via additional coefficients), this flexibility is largely nominal in practice. Such approaches scale poorly to large feature spaces, and still require pre-specified functional forms (e.g., linear effects or interaction terms).

2. THE UNVAMP ARCHITECTURE

We introduce the Unified Neural Variational Measurement of Proficiency (UNVaMP) architecture. Inspired by dynamic deep learning architectures that encode sequential inputs as latent distributions [12, 17], UNVaMP combines a memory model and an inference model (encoder) to produce moment-in-time latent knowledge state distributions, which are passed through a response model to predict student-item interaction outcomes. The response model thus serves as a modular decoder that determines how latent distributions map to predictions, and whether those latent distributions carry interpretable meaning. When configured with a neural decoder (UNVaMP-MLP), the latent space has no prescribed interpretation; when configured with a 1PL MIRT measurement function (UNVaMP-MIRT), the latent dimensions are directly interpretable as per-skill proficiency estimates, and item embeddings correspond to skill loadings and intercepts.

In UNVaMP, latent dynamics are regularized via a Kullback-Leibler divergence penalty between temporally adjacent distributions. When the response model is an interpretable measurement function, UNVaMP yields knowledge state estimates with prescribed meaning. Because inputs and memory are encoded as latent distributions, uncertainty representations are available through distributional parameters. All configurations support arbitrary auxiliary inputs. Use of the term "variational" refers to regularization of latent dynamics via a distributional divergence penalty, not variational Bayesian inference. Fig. 1 provides a full diagram of the architecture.

2.1 Inference Model and Memory Model

For a given student, the *inference model* (or encoder) f_z uses extracted features and hidden states—representing the model's internal memory—to output the mean and variance of a Gaussian distribution over the student's latent proficiency at a given time point. The hidden states are the outputs of the *memory model* f_h, a recurrent neural network (RNN) that uses extracted features to build an internal hidden memory representation of the student; see Fig. 1(a)–(b).

For student i with a sequence of student-item interactions at time points $t = 1, 2, ..., T_i$, the following inputs are available

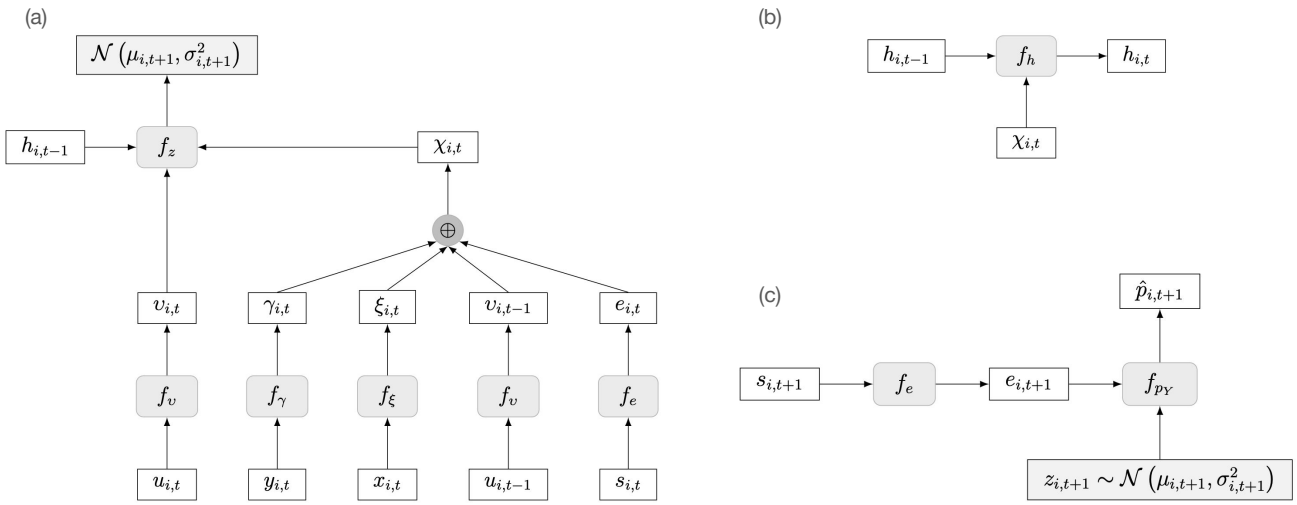


Figure 1: Overview of the UNVaMP architecture. Panel **(a)** shows the *inference model*, which combines internal memory about student i with extracted features from their most recent interaction to obtain an estimated latent knowledge distribution. Panel **(b)** shows the *memory model*, which combines the previous internal memory about student i with extracted features from their most recent interaction to obtain an updated internal memory. Panel **(c)** shows the *response model*, which receives latent variable samples from the knowledge state distribution for student i and generates predicted responses to items.

for each t:

$$(y_{i,t}, s_{i,t}, x_{i,t}, u_{i,t}),$$

where $y_{i,t}$ denotes the observed response data (e.g., correct/incorrect) and $s_{i,t}$ is an identifier (typically an integer index) for the item the student interacted with. The variable $x_{i,t}$ reflects any auxiliary data pertaining to the student-item interaction (e.g., item or skill metadata), and $u_{i,t}$ represents pre-response data that are measured before the next response $y_{i,t+1}$ is observed, such as the time-gap between t and $t+1$. The $x_{i,t}$ and $u_{i,t}$ inputs are optional, and can be flexibly specified, so long as they do not leak information about the response at $t+1$. In this study, predictions target only the binary correct/incorrect score at $t+1$.

During the forward pass, the inputs are sent through *feature extractors* f_γ, f_ξ, f_υ, and an *item embedder* f_e to obtain:

$$\gamma_{i,t} = f_\gamma(y_{i,t}), \quad \xi_{i,t} = f_\xi(x_{i,t})$$
$$\upsilon_{i,t} = f_\upsilon(u_{i,t}), \quad e_{i,t} = f_e(s_{i,t}),$$

Where $\gamma_{i,t},\ \xi_{i,t},\ \upsilon_{i,t},\ e_{i,t} \in \mathbb{R}^d$ and d is a specified (or tuned) embedding/encoding dimensionality. The representations $\gamma_{i,t}$, $\xi_{i,t}$, and $e_{i,t}$, along with $\upsilon_{i,t-1}$ (the extracted pre-response features from $t-1$), are then concatenated to obtain:

$$\chi_{i,t} = [\gamma_{i,t}, \xi_{i,t}, \upsilon_{i,t-1}, e_{i,t}].$$

The representation $\chi_{i,t}$ describes the full set of data available immediately after observing $y_{i,t}$. The hidden state $h_{i,t-1}$ is obtained from the memory model f_h using $\chi_{i,t-1}$ and $h_{i,t-2}$:

$$h_{i,t-1} = f_h(\chi_{i,t-1}, h_{i,t-2}),$$

where $h_{i,t-1}$ reflects UNVaMP's internal memory about student i, prior to observation t.

Then, $\chi_{i,t}$, $\upsilon_{i,t}$, and $h_{i,t-1}$ are passed to the inference model f_z, which outputs a Gaussian distribution over student latent variable $z_{i,t+1}$:

$$\mathcal{N}(\mu_{i,t+1}, \sigma^2_{i,t+1}) = f_z\left(\chi_{i,t}, \upsilon_{i,t}, h_{i,t-1}\right),$$

where $\mu_{i,t+1}$ represents the expected value of $z_{i,t+1}$, and $\sigma^2_{i,t+1}$, a vector representing the diagonal of a covariance matrix, reflects the model's uncertainty about $z_{i,t+1}$. Here, we have that $\mu_{i,t+1},\ \sigma^2_{i,t+1},\ z_{i,t+1} \in \mathbb{R}^k$, where k is the latent dimensionality. The $t+1$ subscripts denote that these values represent the model's belief about what the student's knowledge state will be at the time of response $t+1$. During training, $z_{i,t+1}$ is sampled from $\mathcal{N}(\mu_{i,t+1}, \sigma^2_{i,t+1})$ in the forward pass, while during inference, it is given by point estimate $z_{i,t+1} = \mu_{i,t+1}$.

We conclude our discussion of the inference model f_z and memory model f_h by noting several design choices. First, features $\chi_{i,t}$ are passed directly to f_z, rather than via f_h. This restricts the role of f_h to that of a memory mechanism only, preventing it from being the conduit of new information at time point t. Second, the latent variable $z_{i,t+1}$ is never ingested by f_h, imposing a separation between the internal representations learned by f_h and the point-in-time summaries given by f_z. In doing so, the internal expressiveness of the memory and inference models is left unconstrained by the latent distributions they produce. This separation has the added computational benefit of allowing the memory model to operate independently of the inference model's outputs during the forward pass, permitting the use of optimized sequence processing without the bottleneck of sequentially ingesting sampled latent variables.

2.2 Response Model

The *response model* f_Y, shown in Fig. 1(c), maps student latent knowledge states, along with item embeddings, to predicted response probabilities.

Once the student latent variable $z_{i,t+1}$ is obtained from f_z

and the next-item embedding $e_{i,t+1}$ is obtained from the embedder f_e via $e_{i,t+1} = f_e(s_{i,t+1})$, these values are passed into f_Y to obtain:

$$\hat{p}_{i,t+1} = f_Y(z_{i,t+1}, e_{i,t+1}),$$

where $\hat{p}_{i,t+1}$ is the scalar predicted probability that student i will respond correctly to item $s_{i,t+1}$ at time point $t+1$, conditioned on the latent variable $z_{i,t+1}$ and the item embedding $e_{i,t+1}$. The items $s_{i,t+1}$ are known during training, and can be chosen arbitrarily during inference.

When f_Y is a known measurement function, it determines the interpretation of the latent distribution. For the 1PL MIRT configuration used in this study, the item embedder f_e outputs interpretable embeddings

$$e_{i,t+1} = \left(a_{s_{i,t+1}}, d_{s_{i,t+1}}\right),$$

so that:

$$\hat{p}_{i,t+1} = \frac{1}{1 + \exp\{-(a_{s_{i,t+1}}^{\top} z_{i,t+1} + d_{s_{i,t+1}})\}}.$$

Here, the "embedding" is a tuple containing $a_{s_{i,t+1}}$ and $d_{s_{i,t+1}}$, which represent the loadings and intercept for item $s_{i,t+1}$, where $d_{s_{i,t+1}}$ can either be learned during fitting or supplied as data. In this study, we utilized fixed binary skill tags as loadings, giving $a_{s_{i,t+1}} \in \{0,1\}^k$, and learned $d_{s_{i,t+1}}$ during fitting. A key constraint of our implementation is that f_Y takes only student latent variables and item embeddings as input, following the standard practice in most measurement functions, allowing UNVaMP to support interpretable response model configurations.

2.3 Training Objective

The UNVaMP training objective takes the form:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \beta D_{\text{KL}},$$

where $\mathcal{L}_{\text{pred}}$ denotes the prediction loss (binary cross-entropy in our case), D_{KL} denotes the Kullback-Leibler (KL) divergence [3], and β is a regularization coefficient. Because UNVaMP represents student knowledge as latent probability distributions, we require a mechanism to regularize how these distributions evolve over time. As such, we compute D_{KL} over temporally adjacent latent distributions:

$$D_{\text{KL}}\left(\mathcal{N}(\mu_{i,t+1}, \sigma_{i,t+1}^2) \,\|\, \mathcal{N}(\mu_{i,t}, \sigma_{i,t}^2)\right),$$

constraining how much the new knowledge state distribution $\mathcal{N}(\mu_{i,t+1}, \sigma_{i,t+1}^2)$ diverges from the previous distribution $\mathcal{N}(\mu_{i,t}, \sigma_{i,t}^2)$. This results in smoother estimated student learning dynamics and tempers the local variation of f_z and f_h (indirectly), with β regulating the level of imposed smoothness. Note that $\mathcal{L}_{\text{pred}}$ and D_{KL} are scalar aggregated values. To prevent latent dimensionality from inflating or deflating the KL divergence, we perform mean-reduction rather than sum-reduction of these quantities. This is especially important in the case where the latent space is tied to an interpretable measurement function, since the dimensionality of the latent space should not artificially inflate or deflate the KL divergence penalty.

While UNVaMP is inspired by the dynamic VAE literature [12, 17], it does not perform variational Bayesian inference. Unlike standard VAE formulations, where the KL divergence is computed between an approximate posterior and a prior, UNVaMP uses the KL divergence solely as a regularization mechanism to impose smoothness on the latent knowledge distribution trajectories.

2.4 Alignment with Architecture Desiderata

UNVaMP was designed to address the DLKT architecture desiderata outlined in Section 1.3. Specifically, the β term in UNVaMP's training objective provides a mechanism for regularizing the estimated latent dynamics, allowing users to control the trade-off between smoothness and predictive performance. UNVaMP also supports using known measurement functions (e.g., MIRT) as the response model, yielding interpretable moment-in-time knowledge state estimates and item parameters (intercepts and loadings) when so configured. In addition, UNVaMP generates latent distributions representing student knowledge states, enabling representations of internal uncertainty. Although these estimates do not provide formal coverage guarantees, they capture relative model confidence in the latent estimates. Finally, UNVaMP accommodates arbitrary response, auxiliary, and pre-response inputs. In this study, the full set of desiderata is realized under the UNVaMP-MIRT configuration with $\beta > 0$, while UNVaMP-MLP retains the flexibility, uncertainty, and regularization properties without imposing interpretability constraints on the latent space.

3. METHODS

This section describes the models, datasets, and experimental procedures employed in this study. First, we introduce the UNVaMP configurations and baseline model architectures utilized in our experiments. Next, we discuss the public, internal, and simulated datasets on which our experiments were conducted. We then conclude with a description of the experimental setup and evaluation protocols.

3.1 Models

UNVaMP was benchmarked against a representative set of classical, hybrid, and deep learning KT models. These baselines were chosen to provide well-known comparison points with different approaches to dynamics, interpretability, and model capacity. Below, we describe the models and configurations used in our experiments.

3.1.1 UNVaMP

For benchmarking, we evaluated two configurations of UNVaMP that differ only in the specification of the response model. The first, UNVaMP-MLP, uses a neural network decoder as the response model. The second, UNVaMP-MIRT, is a hybrid configuration in which the response model is a 1PL MIRT measurement function. Here, the item embedder consumes pre-specified binary skill tags for each item, which are used as fixed loading parameters, while item intercepts are learned as one-dimensional embeddings. For the memory model, we used a Gated Recurrent Unit (GRU) RNN [9].

3.1.2 BKT

Bayesian Knowledge Tracing (BKT) is a Hidden Markov Model-based knowledge tracing method that uses response history to estimate future proficiency and correctness. The model uses four key parameters: initial knowledge probability, learning rate, guess rate (correct answer despite not

Table 1: Dataset summaries.

Dataset	Obs.	Students	Items	Skills
ASSISTments	172,200	1,561	548	78
EdNet	2,085,026	18,326	9,198	187
Cloze	45,115	478	144	36
Amplify	7,583,010	107,169	2,346	43

knowing), and slip rate (incorrect answer despite knowing). We used the BKT implementation in the pyBKT package [1].

3.1.3 Deep-IRT

Deep-IRT is an extension of the Dynamic Key-Value Memory Network (DKVMN) architecture that predicts the probability of a correct response using a Rasch-style measurement function [58]. Like DKVMN, Deep-IRT employs a static key memory to compute attention weights over knowledge components, and a dynamic value memory that is updated using embeddings of student-item interactions [61]. Deep-IRT then estimates latent student traits and item difficulties that are combined using the Rasch measurement function to predict probability of correctness. For our experiments, we used the Deep-IRT implementation provided by the pyKT Toolkit package [26].

3.1.4 SAINT

SAINT is a DLKT method based on the influential transformer architecture [10]. In SAINT, a multi-head attention mechanism is used within an encoder-decoder architecture, where the encoder models the exercise sequence and the decoder attends to prior responses and the encoded exercises to predict response correctness. For our experiments, we used the SAINT implementation provided by the pyKT toolkit, with minor implementation-level modifications that do not alter the model architecture. Specifically, we used PyTorch's built-in transformer encoder and decoder modules (rather than from-scratch implementations) and employed a shared positional embedding layer in the encoder and decoder blocks.

3.1.5 LKT

We used the lasso regression implementation in the LKT package in R [33] to create features and train models, in which a penalty term is applied to the objective function and may reduce coefficients to zero. This approach was implemented to avoid challenges with feature selection [35].

3.2 Data

The data used in this study fall into three categories: public, internal, and simulated. Public and internal datasets were used for evaluating UNVaMP against existing knowledge tracing models on known benchmarks and in a real-world context, while the simulated data were used to assess UNVaMP's latent structure recovery capacity and inference behavior under controlled conditions.

3.2.1 Public Datasets

Three public datasets were selected for this study: ASSISTments, EdNet, and Cloze. Dataset-specific preprocessing is described below. In addition, iterative threshold filtering was applied to retain only students with responses to at least ten unique items and items with responses from at least one hundred unique students. For multi-skill items, observed per-item skill combinations were used as skill tags for all models except UNVaMP-MIRT, where latent dimensions corresponded to atomic skills. Table 1 summarizes each dataset. Each dataset was divided into a 60/20/20 train/validation/test split unless otherwise noted.

ASSISTments. The ASSISTments dataset contains student-item interactions from the ASSISTments online math tutoring platform, which provides immediate feedback and optional scaffolds and hints [21]. In this study, we used the ASSISTments 2017 dataset released for the 2017 ASSISTments Longitudinal Data Mining Competition [29], a longitudinal dataset with nearly one million student actions from over 1,700 students spanning secondary through tertiary education. To better reflect non-intervention assessment moments, we retained only single-skill items (dropping untagged and multi-skill items) and excluded responses involving scaffolds or hints, as well as responses on student-item pairs where the initial response used assistance.

EdNet. The EdNet dataset contains student-item interaction data from Santa, an intelligent tutoring system used by South Korean students to prepare for the Test of English for International Communication (TOEIC) [11]. EdNet contains data from over 750,000 students and is organized into four datasets with varying levels of behavioral detail. We randomly sampled 20,000 students from the KT1 subset prior to threshold filtering. We then removed duplicate interactions within sessions and interactions missing responses, imposed a cap of 500 interactions per student (approximately three standard deviations above the in-dataset mean) to reduce computational demands during model fitting.

Cloze. The Statistics Cloze dataset (referred to as "Cloze" in this study) was obtained from Memphis Datashop via the LKT R package [33]. The data consist of responses from Amazon Mechanical Turk participants who studied statistical concepts through short readings and completed cloze-style exercises, with each sentence associated with one of 36 skills. Sentence presentation frequency and temporal spacing (narrow, medium, wide) were experimentally manipulated, with post-practice tests administered after delays of 2 minutes, 1 day, or 3 days. We removed observations without a correctness score. Due to the small dataset size, we used a 50/25/25 train/validation/test split with stratified sampling to maintain balance across experimental conditions.

3.2.2 Internal Dataset

Amplify Dataset. The Amplify dataset consists of student-item interactions from a K-8 mathematics curriculum delivered via an online platform, including summative pre-unit, mid-unit, and end-of-unit assessments, as well as formative in-lesson assessments. Though responses were primarily

auto-scored as correct or incorrect, when available, teacher re-scores were used in place of auto-scores. In this dataset, questions are tagged to a hierarchical proficiency map of math skills, and for this study, skill tags were drawn from the highest-level nodes. Only non-empty responses were retained and untagged items were excluded. The dataset was de-identified, with all personally identifiable information removed prior to use.

3.2.3 Simulated Dataset

We simulated student response data from a 1PL MIRT model. First, a global skill-skill covariance matrix Σ was constructed with unit variances on the diagonal and off-diagonal entries drawn uniformly from the range $[0.2, 0.6]$, with Σ constrained to be positive semi-definite. Next, a scalar overall proficiency x_i was drawn from a standard normal distribution for each student. Per-student θ_i vectors were then sampled from $\mathcal{N}(x_i \cdot \mathbf{1}, \Sigma)$. Item difficulties b_j were sampled from a standard normal distribution, with equal numbers of single-skill items per skill. Responses were generated by computing item intercepts $d_j = -b_j$ and applying the MIRT measurement function, then sampling Bernoulli outcomes. All students interacted with all items, and within-student item interaction order was randomized such that more difficult items tended to be seen later. Each simulated dataset contained 5 skills, 50 items per skill, and 10,000 students.

3.3 Experiments

3.3.1 Model Benchmarking

For the model benchmarking experiments, we fit all models to the public and internal datasets, then characterized predictive performance and smoothness.

Prior to benchmarking, hyperparameters for all deep learning models were tuned using Bayesian optimization. Across models, tuning included the learning rate, embedding dimensionality, and dropout probability. For UNVaMP, additional hyperparameters included the hidden state dimensionality and the number of hidden layers in the RNN and feature extractors; for UNVaMP-MLP, the number of hidden decoder layers was also tuned. For Deep-IRT, tuning included the batch size, the number of rows in the key and value memory matrices (constrained to be equal), and the maximum input sequence length. The latter was introduced to circumvent VRAM limitations, such that longer input sequences were split into subsequences and treated as additional samples during training. For SAINT, tuning included the batch size, the number of encoder and decoder layers (constrained to be equal), the number of attention heads, and the maximum input sequence length (used similarly to Deep-IRT). All DLKT models were trained using Adam-based optimizers with default settings, with UNVaMP trained using AdamW. For LKT, we included the same features as described in the original LKT-lasso paper, with the addition of power-law decay and PPE [35]. Hyperparameters for PPE were estimated via maximum likelihood on the training set. Other features with hyper-parameters were generated on a 0-1 grid (step=0.1).

Because LKT and UNVaMP allow flexible specification of input features, we evaluated two versions of each model. The minimal versions were trained using the same inputs as the other benchmark models: item ID, skill ID, sequence position, and response correctness (and features derived from these inputs). The (aux) versions included all minimal inputs plus available auxiliary inputs, which varied by dataset (e.g., problem type, bundle ID). All datasets provided submission timestamps, from which recency features (e.g., time since previous submission) were computed.

Benchmark fits of UNVaMP-MLP and UNVaMP-MIRT were conducted using $\beta \in \{0, 10^{-3}\}$ to reflect peak predictive performance. To examine the trade-off between predictive performance and regularization, we compared UNVaMP-MIRT (aux) fits across $\beta \in \{0, 10^{-3}, 10^{-2}, 10^{-1}\}$ for each dataset. To quantify the smoothing effect of β on the latent dynamics, we report what we call the *average latent total variation* (ALTV). ALTV can be viewed as a normalized, fixed-sample analogue of *total variation*, a concept in real analysis that describes the oscillatory behavior of functions. This metric was also inspired by the w_1 smoothness metric described in [59], though ALTV is computed in the latent space and gives equal weight to each student. ALTV is formulated as

$$\text{ALTV} = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{k(T_i - 1)} \sum_{t=1}^{T_i - 1} \| z_{i,t+1} - z_{i,t} \|_1 \right),$$

where $z_{i,t} \in \mathbb{R}^k$ is the latent proficiency vector for student $i \in \{1, \cdots, N\}$ at time point t. Under the 1PL MIRT measurement function, each latent dimension enters the logit with a binary coefficient. Consequently, holding all else fixed, a one unit perturbation to $z_{i,t}$ in any latent dimension has the same effect on the logit. This means latent dimensions share units and the use of the L^1 norm over the differences $(z_{i,t+1} - z_{i,t})$ is justified.

3.3.2 Simulation Study

In the simulation study, we evaluated UNVaMP-MIRT under five levels of Gaussian measurement noise, perturbing each student response using

$$P\left[Y_{i,j} = 1 \mid \theta_i, a_j, d_j\right] = \frac{1}{1 + \exp\{-(a_j^\top \theta_i + d_j + \epsilon)\}},$$

where $\epsilon \sim \mathcal{N}(0, \nu)$ and $\nu \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$. For each noise level, 5 replications were conducted (25 total runs). The skill-skill covariance matrix was fixed across runs, with student proficiencies, item difficulties, and responses re-sampled within each run. UNVaMP-MIRT and baseline MIRT models were fit separately within each run, with MIRT estimates obtained via maximum a posteriori (MAP) using a standard normal prior. MAP was chosen over maximum likelihood estimation (MLE) or weighted maximum likelihood estimation (WLE) for numerical stability and reliable convergence. Both models were fit with ground-truth item parameters a_j and d_j fixed, aligning latent scales such that student latent variables were the only estimated quantities. We examined two primary behaviors of UNVaMP-MIRT: (1) latent variable recovery accuracy, and (2) the behavior of uncertainty estimates as a function of the number of observations. Each replication comprised 10,000 students with 250 responses per student (2.5M observations per run). Except where otherwise noted, all simulation analyses use UNVaMP-MIRT with $\beta = 10^{-2}$.

4. RESULTS

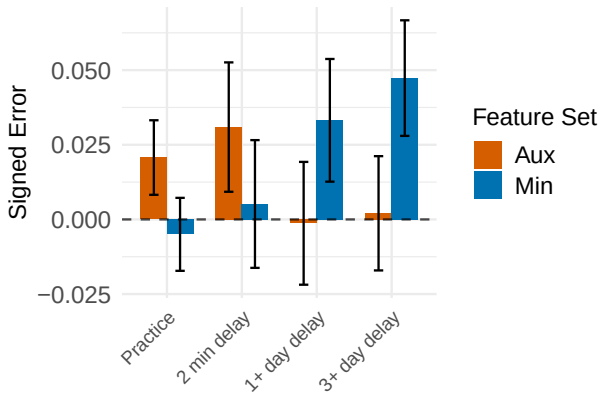


Figure 2: UNVaMP-MIRT signed error (mean$(\hat{p}-y)$) on the Cloze dataset with and without auxiliary inputs, under $\beta = 10^{-3}$. With auxiliary features included, UNVaMP-MIRT demonstrates reduced error in delayed post-tests. Error bars represent 95% confidence intervals. Standard errors were computed as the SD of prediction residuals divided by the square root of the subset sample size (practice session, 2 minute delay, 1+ day delay, or 3+ day delay).

4.1 Model Benchmarking

UNVaMP-MLP achieved the strongest overall performance during model benchmarking, outperforming or matching all other models on three of four datasets in both AUC and accuracy, with LKT performing best on the Cloze dataset (Table 2). UNVaMP-MIRT performed competitively, with only modest losses in predictive performance relative to UNVaMP-MLP, indicating that the cost of an interpretable response model is small in practice. Specifically, UNVaMP-MIRT outperformed SAINT on ASSISTments, EdNet, and Cloze, and performed slightly worse on the Amplify dataset. Compared to LKT, UNVaMP-MIRT performed comparably on ASSISTments and EdNet, and better on the Amplify dataset, though worse on Cloze. Both UNVaMP variants showed strong performance across datasets, despite varying dataset sizes, item and skill counts, and other structural attributes, indicating robustness to dataset-specific characteristics. By contrast, SAINT's performance appeared to be highly sensitive to dataset size, consistent with its transformer-based architecture. Differences between our baseline model results and those reported in the original works are primarily attributable to our data preprocessing and subsampling choices.

LKT's strong performance on Cloze is unsurprising, given that the data were collected under constrained, experimental conditions, with relatively few students and observations, limiting the performance of higher-capacity DLKT methods. LKT also benefited from hand-designed, research-based features, which can outperform deep learning methods in low- and medium-data settings [15]. Furthermore, Table 2 shows that the impact of auxiliary inputs on UNVaMP's aggregate predictive performance was minimal, with their inclusion producing negligible performance differences for UNVaMP-MLP and small, mixed effects for UNVaMP-MIRT.

Although auxiliary features had a limited impact on aggregate predictive performance for UNVaMP, results for the Cloze dataset—with its spacing, repetition, and retention interval manipulations—indicate that these features meaningfully influenced model behavior. Stratifying UNVaMP-MIRT performance by post-test session interval revealed that longer retention intervals were associated with smaller signed prediction errors for the auxiliary configuration (Fig. 2). This suggests that auxiliary features (e.g., elapsed time) can improve sensitivity to meaningful underlying structure, despite modest losses in aggregate accuracy. We suspect this sensitivity would improve, and perhaps come at reduced predictive cost (if any), given a larger dataset.

To examine the effect of β on predictive performance, we evaluated UNVaMP-MIRT (aux) under the conditions $\beta \in \{0, 10^{-3}, 10^{-2}, 10^{-1}\}$, revealing a clear pattern (Table 3). Increasing β from 0 to 10^{-3} produced no meaningful change in prediction accuracy, whereas the change from 10^{-3} to 10^{-2} led to a small but consistent decrease across datasets. A further increase to 10^{-1} resulted in a substantial accuracy drop in most cases. These results indicate that small values of β have little practical impact on predictive performance. Across all datasets, ALTV decreased monotonically with increasing β, confirming that β provides control over the smoothness of latent trajectories. With the exception of Cloze, the increase in β from 0 to 10^{-3} resulted in the largest decrease in ALTV by far. Subsequent β increases showed diminishing returns, along with increased performance penalties. The smoothing effects of β are further examined qualitatively in the simulation study results below.

4.2 Simulation Study

In the simulation study, we evaluated UNVaMP-MIRT's ability to recover student latent variables relative to a baseline MIRT model. Fig. 3(a) shows that, across all noise conditions, UNVaMP-MIRT performed comparably to the MIRT baseline once it had observed the full response sequence. Given the large number of responses per student, MAP shrinkage in the MIRT baseline model is unlikely to have materially affected accuracy. These results do not indicate that UNVaMP-MIRT will outperform MIRT under idealized conditions, but rather that it is capable of recovering latent variables with comparable accuracy given sufficient data. UNVaMP-MIRT's competitive recovery accuracy in this experiment may partly reflect its ability to implicitly exploit the cross-skill correlations that were built into the simulation process, whereas the baseline MIRT models were fit under a simple per-skill standard normal prior.

Fig. 3(b) shows that UNVaMP-MIRT's uncertainty estimates decreased alongside recovery RMSE across noise conditions, indicating that model uncertainty tracked recovery accuracy. Uncertainty declined more rapidly than RMSE early in the sequence, suggesting some possible early overconfidence, but both quantities exhibited a shared monotonic decrease. Fig. 4 illustrates the effect of β on latent variable estimation, with larger values producing less volatile estimation behavior. We emphasize that this simulation study considered static ground-truth latent knowledge states; further analysis is needed to evaluate model behavior under dynamically evolving states.

Table 2: Model performance across four real-world datasets. Best values per dataset, per metric are in bold.

	ASSISTments		**EdNet**		**Cloze**		**Amplify**	
Model	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.
BKT	0.635	0.595	0.668	0.659	0.724	0.674	0.654	0.614
Deep-IRT	0.687	0.633	0.697	0.675	0.753	0.689	0.768	0.697
SAINT	0.698	0.639	0.748	0.704	0.673	0.625	0.868	**0.780**
Lasso LKT	0.732	0.665	0.761	0.713	0.846	0.766	0.810	0.734
Lasso LKT (aux)	0.736	0.670	**0.767**	0.716	**0.858**	**0.775**	0.822	0.745
UNVaMP-MLP ($\beta = 0$)	0.743	0.677	**0.767**	**0.717**	0.833	0.750	0.867	0.779
UNVaMP-MLP ($\beta = 10^{-3}$)	0.742	0.675	0.766	**0.717**	0.827	0.740	**0.869**	**0.780**
UNVaMP-MLP (aux, $\beta = 0$)	**0.745**	0.677	0.765	0.716	0.837	0.756	0.865	0.777
UNVaMP-MLP (aux, $\beta = 10^{-3}$)	**0.745**	**0.678**	0.765	0.716	0.831	0.751	0.865	0.776
UNVaMP-MIRT ($\beta = 0$)	0.730	0.666	0.762	0.714	0.821	0.742	0.844	0.759
UNVaMP-MIRT ($\beta = 10^{-3}$)	0.729	0.662	0.755	0.709	0.810	0.737	0.845	0.760
UNVaMP-MIRT (aux, $\beta = 0$)	0.736	0.673	0.755	0.709	0.815	0.738	0.849	0.764
UNVaMP-MIRT (aux, $\beta = 10^{-3}$)	0.738	0.670	0.751	0.706	0.803	0.734	0.850	0.764

Table 3: Impact of increasing β on AUC and ALTV. Highest AUC and lowest ALTV values per dataset are in bold.

	ASSISTments		**EdNet**		**Cloze**		**Amplify**	
Model	AUC	ALTV	AUC	ALTV	AUC	ALTV	AUC	ALTV
UNVaMP-MIRT (aux, $\beta = 0$)	0.736	0.418	**0.755**	0.185	**0.815**	0.322	0.849	0.765
UNVaMP-MIRT (aux, $\beta = 10^{-3}$)	**0.738**	0.039	0.751	0.017	0.803	0.209	**0.850**	0.192
UNVaMP-MIRT (aux, $\beta = 10^{-2}$)	0.722	0.015	0.747	0.012	0.733	0.039	0.844	0.072
UNVaMP-MIRT (aux, $\beta = 10^{-1}$)	0.703	**0.007**	0.744	**0.009**	0.713	**0.017**	0.802	**0.023**

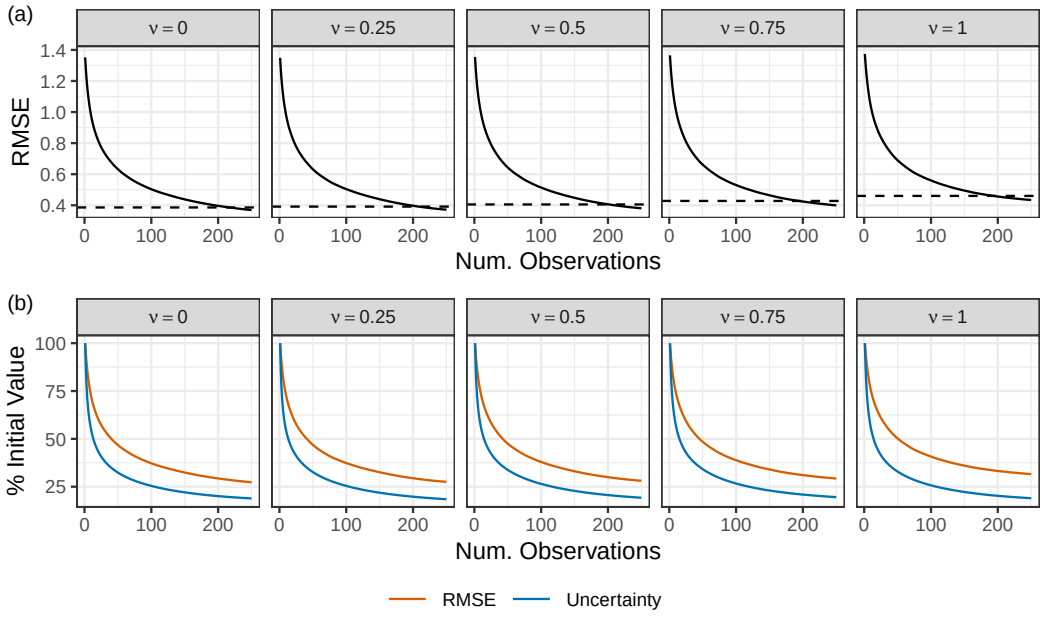


Figure 3: Relationship between the fit metrics of interest and the number of observations (student-item interactions). Panel **(a)** shows RMSE vs. observation count, averaged across replications. Cross-replication SD is omitted from the figure due to small magnitude (averaging below 0.01 for all noise conditions). The dashed line indicates baseline MIRT final RMSE, computed after observing all data. Panel **(b)** shows the percent of the initial value (RMSE and uncertainty) vs. observation count, averaged across replications. Again, cross-replication SD is omitted from the figure due to small magnitudes (averaging below 1% for RMSE and below 3% for uncertainty across all noise conditions).

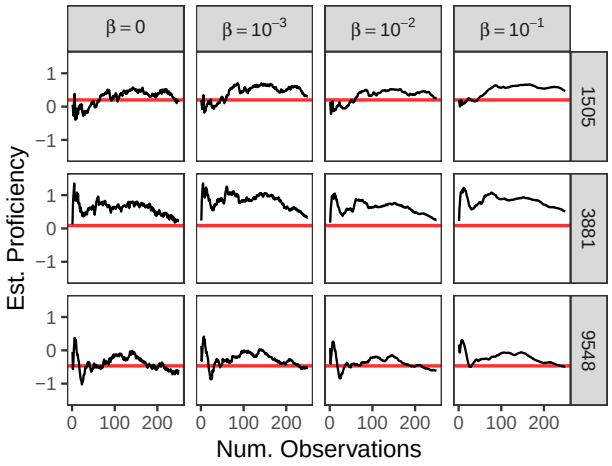


Figure 4: Effect of β on latent proficiency estimation. Columns correspond to $\beta \in \{0, 10^{-3}, 10^{-2}, 10^{-1}\}$; rows correspond to individual simulated students. The black line shows the UNVaMP-MIRT estimate for one latent skill plotted against the cumulative number of responses across all skills. The red horizontal line denotes the true value of the selected skill.

5. DISCUSSION

In total, our findings show that UNVaMP offers practical mechanisms for regularizing latent dynamics and modeling latent structure within student-item interaction data, all while maintaining competitive predictive performance. Below, we clarify the scope and limitations of the current work and place these results in a broader context.

5.1 Limitations and Future Work

Although ALTV quantifies the effect of β on latent trajectory smoothness, connecting such metrics to practical criteria for β selection is not straightforward. For instance, student knowledge estimates are expected to exhibit high early volatility, due to both estimation noise and genuinely rapid early learning, followed by more moderated trajectories. It remains unclear how to weigh smoothness against predictive accuracy, or whether such measures would have an intuitive interpretation. These issues arise because, although a hybrid UNVaMP configuration offers interpretable snapshots of estimated student knowledge, the underlying dynamics are still the product of a deep learning model and are not governed by interpretable updating rules.

Likewise, while UNVaMP provides quantitative uncertainty estimates, those estimates are not calibrated according to any interpretable statistical framework and should not be used to provide coverage guarantees; rather, they are best used to identify changes in relative confidence within the model as evidence accumulates. Because our current study examined the relationship between UNVaMP's latent uncertainty estimates and recovery accuracy only under simulated conditions, the relationship between uncertainty and external signals (e.g., predictive accuracy) should be investigated in a real-world setting; however, such investigations would need to account for the sparsity, noise, and heterogeneity of real-world interaction data.

UNVaMP can utilize arbitrary engineered features as auxiliary inputs, such as those used in LKT models. Future work could examine the impact of incorporating these features on both predictive performance and interpretability. In addition, subsequent studies could investigate UNVaMP's ability to implicitly recover experimentally demonstrated learning phenomena from data in which such structure is known to be present. While experimentally generated datasets such as Cloze are valuable for this purpose, their limited size constrains the direct application of DLKT methods. Accordingly, future work could include large experimentally informed simulation studies to examine how auxiliary inputs influence UNVaMP's sensitivity to known learning phenomena.

5.2 Conclusions

In this paper, we introduced UNVaMP, a neural network knowledge tracing architecture that estimates latent distributions over student knowledge as it evolves over time. We found that a pure neural configuration (UNVaMP-MLP) achieved the strongest predictive performance among compared models on three out of four datasets, while a hybrid configuration (UNVaMP-MIRT) lagged only slightly behind, highlighting the relatively small penalty incurred for improving interpretability. In addition, UNVaMP-MIRT with auxiliary inputs exhibited sensitivity to underlying temporal structure in the experimental Cloze dataset. Under simulated conditions, the UNVaMP-MIRT configuration further exhibited robust latent variable recovery and internally consistent uncertainty behavior. Taken together, these results indicate that UNVaMP, particularly in its hybrid configuration, is a practical solution for applications that require not only strong predictive performance, but also regularized latent knowledge trajectories, interpretable knowledge state estimates, internally consistent uncertainty quantification, and flexible input modeling.

6. REFERENCES

[1] A. Badrinath, F. Wang, and Z. Pardos. pybkt: An accessible python library of bayesian knowledge tracing models, 2021. arXiv:2105.00385.

[2] R. S. J. d. Baker, A. T. Corbett, and V. Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian knowledge tracing. In B. P. Woolf, E. Aïmeur, R. Nkambou, and S. Lajoie, editors, *Intelligent Tutoring Systems*, pages 406–415, Berlin, Heidelberg, 2008. Springer.

[3] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[4] D. Borsboom. Latent variable theory. *Measurement*, 6:25–53, 2008.

[5] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*, pages 164–175. Springer, 2006.

[6] J. Chen, Z. Liu, S. Huang, Q. Liu, and W. Luo. Improving interpretability of deep sequential knowledge tracing models with question-centric cognitive representations, 2023. arXiv:2302.06885.

[7] Y. Chen, S. A. Culpepper, S. Wang, and J. Douglas.

A hidden markov model for learning trajectories in cognitive diagnosis with application to spatial rotation skills. *Applied Psychological Measurement*, 42(1):5–23, Jan. 2018.

[8] W. Cheng, H. Du, C. Li, E. Ni, L. Tan, T. Xu, and Y. Ni. Uncertainty-aware knowledge tracing, 2025. arXiv:2501.05415.

[9] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches, 2014. arXiv:1409.1259.

[10] Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, and J. Heo. Towards an appropriate query, key, and value computation for knowledge tracing. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, pages 490–496, Apr. 2020.

[11] Y. Choi, Y. Lee, D. Shin, J. Cho, S. Park, S. Lee, J. Baek, C. Bae, B. Kim, and J. Heo. Ednet: A large-scale hierarchical dataset in education. In *International conference on artificial intelligence in education*, pages 69–73. Springer, 2020.

[12] S. T. Christie, C. Cook, and A. N. Rafferty. Uncertainty-preserving deep knowledge tracing with state-space models, 2024. arXiv:2407.17427.

[13] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modelling and User-Adapted Interaction*, 4(4):253–278, 1995.

[14] A. Galyardt and I. Goldin. Move your lamp post: Recent data reflects learner knowledge better than older data. *Journal of Educational Data Mining*, 7(2):83–108, 2015.

[15] T. Gervet, K. Koedinger, J. Schneider, T. Mitchell, et al. When is deep learning the best approach to knowledge tracing? *Journal of Educational Data Mining*, 12(3):31–54, 2020.

[16] A. Ghosh, N. Heffernan, and A. S. Lan. Context-aware attentive knowledge tracing, 2020. arXiv:2007.12324.

[17] L. Girin, S. Leglaive, X. Bie, J. Diard, T. Hueber, and X. Alameda-Pineda. Dynamical variational autoencoders: A comprehensive review. *Foundations and Trends in Machine Learning*, 15(1–2):1–175, 2021.

[18] Y. Gong, J. E. Beck, and N. T. Heffernan. Comparing knowledge tracing and performance factor analysis by using multiple model fitting procedures. In *International conference on intelligent tutoring systems*, pages 35–44. Springer, 2010.

[19] J. González-Brenes, Y. Huang, and P. Brusilovsky. General features in knowledge tracing: Applications to multiple subskills, temporal item response theory, and expert knowledge. In *Proceedings of the 7th International Conference on Educational Data Mining*, 2014.

[20] R. K. Hambleton, H. Swaminathan, and H. J. Rogers. *Fundamentals of item response theory*, volume 2. Sage, 1991.

[21] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[22] Y. Kaya and W. L. Leite. Assessing change in latent skills across time with longitudinal cognitive diagnosis modeling: An evaluation of model performance. *Educational and Psychological Measurement*, 77(3):369–388, June 2016.

[23] M. Khajah, R. V. Lindsey, and M. C. Mozer. How deep is knowledge tracing?, 2016. arXiv:1604.02416.

[24] H. Lee, J. B. Cho, D. S. Matteson, and B. W. Domingue. Dynamic bayesian item response model with decomposition (D-BIRD): Modeling cohort and individual learning over time, 2025. arXiv:2506.21723.

[25] R. V. Lindsey, J. D. Shroyer, H. Pashler, and M. C. Mozer. Improving students' long-term knowledge retention through personalized review. *Psychological science*, 25(3):639–647, 2014.

[26] Z. Liu, Q. Liu, J. Chen, S. Huang, J. Tang, and W. Luo. pyKT: A python library to benchmark deep learning based knowledge tracing models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[27] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing, 2019. arXiv:1907.06837.

[28] S. Pandey and J. Srivastava. RKT: Relation-aware self-attention for knowledge tracing. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 1205–1214. ACM, Oct. 2020.

[29] T. Patikorn, R. S. Baker, N. T. Heffernan, et al. Assistments longitudinal data mining competition special issue: a preface. *Journal of Educational Data Mining*, 12(2):i–xi, 2020.

[30] P. I. Pavlik and J. R. Anderson. Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied*, 14(2):101–117, 2008.

[31] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis – a new alternative to knowledge tracing. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, page 531–538, NLD, 2009. IOS Press.

[32] P. I. Pavlik, L. G. Eglington, and L. M. Harrell-Williams. Logistic knowledge tracing: A constrained framework for learner modeling. *IEEE Transactions on Learning Technologies*, 14(5):624–639, 2021.

[33] P. I. Pavlik Jr. and L. G. Eglington. *LKT: Logistic Knowledge Tracing*, 2024. R package version 1.7.0.

[34] P. I. Pavlik Jr and L. G. Eglington. Evolutionary features for mitigating cold starts in logistic knowledge tracing. *International Educational Data Mining Society*, 2025.

[35] P. I. Pavlik Jr, L. G. Eglington, et al. Automated search improves logistic knowledge tracing, surpassing deep learning in accuracy and explainability. *Journal of Educational Data Mining*, 15(3):58–86, 2023.

[36] R. Pelánek. Applications of the Elo rating system in adaptive educational systems. *Computers & Education*, 98:169–179, July 2016.

[37] C. Piech, J. Spencer, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing, 2015. arXiv:1506.05908.

[38] N. Rachatasumrit, P. Carvalho, and K. Koedinger. Beyond accuracy: Embracing meaningful parameters in educational data mining. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 203–210, 2024.

[39] M. D. Reckase. Multidimensional item response theory. *Handbook of statistics*, 26:607–642, 2006.

[40] S. Ruan, W. Wei, and J. Landay. Variational deep knowledge tracing for language learning. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, LAK21, pages 323–332, New York, NY, USA, Apr. 2021. Association for Computing Machinery.

[41] A. A. Rupp, J. Templin, and R. A. Henson. *Diagnostic measurement: Theory, methods, and applications*. Diagnostic measurement: Theory, methods, and applications. The Guilford Press, New York, NY, US, 2010.

[42] S. Shen, Q. Liu, Z. Huang, Y. Zheng, M. Yin, M. Wang, and E. Chen. A survey of knowledge tracing: Models, variants, and applications. *IEEE Transactions on Learning Technologies*, 17:1858–1879, 2021.

[43] L. Song, M. He, X. Shang, C. Yang, J. Liu, M. Yu, and Y. Lu. A deep cross-modal neural cognitive diagnosis framework for modeling student performance. *Expert Syst. Appl.*, 230(C), Nov. 2023.

[44] J. Templin and L. Bradshaw. Measuring the reliability of diagnostic classification model examinee estimates. *Journal of Classification*, 30(2):251 – 275, 2013.

[45] N. D. Verhelst and C. A. W. Glas. A dynamic generalization of the rasch model. *Psychometrika*, 58(3):395–415, Sept. 1993.

[46] M. M. Walsh, K. A. Gluck, G. Gunzelmann, T. Jastrzembski, and M. Krusmark. Evaluating the theoretic adequacy and applied potential of computational models of the spacing effect. *Cognitive science*, 42:644–691, 2018.

[47] M. M. Walsh, K. A. Gluck, G. Gunzelmann, T. Jastrzembski, M. Krusmark, J. I. Myung, M. A. Pitt, and R. Zhou. Mechanisms underlying the spacing effect in learning: A comparison of three computational models. *Journal of Experimental Psychology: General*, 147(9):1325, 2018.

[48] C. Wang, W. Ma, M. Zhang, C. Lv, F. Wan, H. Lin, T. Tang, Y. Liu, and S. Ma. Temporal cross-effects in knowledge tracing. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, page 517–525, New York, NY, USA, 2021. Association for Computing Machinery.

[49] C. Wang and S. W. Nydick. On longitudinal item response theory models: A didactic. *Journal of Educational and Behavioral Statistics*, 45(3):339–368, June 2019. Publisher: American Educational Research Association.

[50] F. Wang, Z. Huang, Q. Liu, E. Chen, Y. Yin, J. Ma, and S. Wang. Dynamic cognitive diagnosis: An educational priors-enhanced deep knowledge tracing perspective. *IEEE Trans. Learn. Technol.*, 16(3):306–323, June 2023.

[51] F. Wang, Q. Liu, E. Chen, Z. Huang, Y. Chen, Y. Yin, Z. Huang, and S. Wang. Neural cognitive diagnosis for intelligent education systems, 2019. arXiv:1908.08733.

[52] S. Wang, Y. Yang, S. A. Culpepper, and J. A. Douglas. Tracking skill acquisition with cognitive diagnosis models: A higher-order, hidden markov model with covariates. *Journal of Educational and Behavioral Statistics*, 43(1):57–87, Feb. 2018. Publisher: American Educational Research Association.

[53] X. Wang, J. O. Berger, and D. S. Burdick. Bayesian analysis of dynamic item response models in educational testing. *The Annals of Applied Statistics*, 7(1), Mar. 2013.

[54] X. Wang, Z. Zheng, J. Zhu, and W. Yu. What is wrong with deep knowledge tracing? attention-based knowledge tracing. *Applied Intelligence*, 53(3):2850–2861, 2022.

[55] Z. Wang, Y. Gu, A. Lan, and R. Baraniuk. VarFA: A variational factor analysis framework for efficient bayesian learning analytics, 2020. arXiv:2005.13107.

[56] M. Wilson and P. De Boeck. *Descriptive and explanatory item response models*, pages 43–74. Springer New York, New York, NY, 2004.

[57] M. Wu, R. L. Davis, B. W. Domingue, C. Piech, and N. Goodman. Variational item response theory: Fast, accurate, and expressive, 2020. arXiv:2002.00276.

[58] C.-K. Yeung. Deep-IRT: Make deep learning based knowledge tracing explainable using item response theory, 2019. arXiv:1904.11738.

[59] C.-K. Yeung and D.-Y. Yeung. Addressing two problems in deep knowledge tracing via prediction-consistent regularization, 2018. arXiv:1806.02180.

[60] Y. Yin, L. Dai, Z. Huang, S. Shen, F. Wang, Q. Liu, E. Chen, and X. Li. Tracing knowledge instead of patterns: Stable knowledge tracing with diagnostic transformer. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 855–864, New York, NY, USA, 2023. Association for Computing Machinery.

[61] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing, 2016. arXiv:1611.08108.

[62] Q. Zhang, Z. Chen, N. Lalwani, and C. MacLellan. Modifying deep knowledge tracing for multi-step problems. In *Proceedings of the 15th International Conference on Educational Data Mining*, 2022.

[63] Y. Zhou, W. Han, and J. Chen. DKT2: Revisiting applicable and comprehensive knowledge tracing in large-scale data, 2025. arXiv:2501.14256.

Utility-Preserving De-Identification for Math Tutoring: Investigating Numeric Ambiguity in the MathEd-PII Benchmark Dataset

Zhuqian Zhou
Cornell University
zz968@cornell.edu

Kirk Vanacore
Cornell University
kpv27@cornell.edu

Bakhtawar Ahtisham
Cornell University
ba453@cornell.edu

Jinsook Lee
Cornell University
jl3369@cornell.edu

Doug Pietrzak
Fresh Cognate
doug@freshcognate.com

Daryl Hedley
Fresh Cognate
daryl@freshcognate.com

Jorge Dias
Fresh Cognate
jorge@freshcognate.com

Chris Shaw
UPchieve
chris.shaw@upchieve.org

Ruth Schäfer
Saga Education
DrRuthSchafer@gmail.com

René F. Kizilcec
Cornell University
kizilcec@cornell.edu

ABSTRACT

Large-scale sharing of dialogue data is key to advancing the science of teaching and learning, yet rigorous de-identification remains a major barrier. In mathematics tutoring transcripts, numeric expressions frequently resemble structured identifiers (e.g., dates or IDs), leading generic Personally Identifiable Information (PII) detection systems to over-redact core instructional content and reduce data utility. This work asks how to detect PII while preserving educational utility, focusing on this "numeric ambiguity" problem. We introduce MathEd-PII, the first benchmark dataset for PII detection in math tutoring dialogues, built with human-in-the-loop LLM annotation. Using density-based segmentation, we show that false PII redactions cluster in math-dense regions, confirming numeric ambiguity as a key failure mode. We then compare four detection strategies: a Presidio baseline and three LLM-based approaches with basic, math-aware, and segment-aware prompting. Domain-aware prompting, including both math-aware (F1: 0.802) and segment-aware versions (F1: 0.821), substantially outperforms the baseline (F1: 0.379) while reducing numeric false positives, demonstrating that de-identification must incorporate domain context to preserve analytic utility. This work provides a new benchmark and evidence that utility-preserving de-identification for tutoring data requires domain-aware modeling.

Keywords

De-identification, PII Detection, Math Tutoring, Large Language Models, Privacy, Segmentation

Zhuqian Zhou, Kirk Vanacore, Bakhtawar Ahtisham, Doug Pietrzak, Daryl Hedley, Jorge Dias, Jinsook Lee, Rene Kizilcec, Chris Shaw, and Ruth Schäfer. Utility-Preserving De-Identification for Math Tutoring: Investigating Numeric Ambiguity in the MathEd-PII Benchmark Dataset. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 365–375. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039941

1. INTRODUCTION

The Educational Data Mining (EDM) community has long relied on large-scale open source data sets from digital learning platforms [14, 18]. Many of these data sources can be easily de-identified because they are composed of action logs that, for the most part, do not typically contain personally identifiable information (PII). As more and more educational technologies provide dialogue-based tutoring at scale, either with humans or with artificial intelligence (AI) chatbots [3, 8, 22, 26], access to large corpora of tutoring conversations has become essential. However, sharing such data requires rigorous de-identification to remove direct identifiers (e.g., names and email addresses) and quasi-identifiers (e.g., gender, ages, locations, and school names) that could enable re-identification of specific individuals.

Recent work has begun to develop tools and workflows for de-identifying educational text [9, 10, 24, 25, 31, 32]. Yet specialized domains like mathematics tutoring introduce a distinct and underexplored challenge. In math dialogue, numbers are the primary medium of exchange. A teacher might ask, "Are 4/12 and 2/6 the same?" (fractions), or a student might calculate "500 - 70 - 2000" (subtraction). Standard PII detectors frequently misidentify these mathematical expressions as `DATE` or `US_SSN` due to *numeric ambiguity*, the overlap between the surface forms of mathematical content and sensitive identifiers. Over-identifying these math-related spans results in low precision, stripping the data of its core educational content and rendering it useless for downstream tasks.

A second barrier is the lack of domain-specific benchmarks. To date, no publicly available dataset exists for evaluating PII detection in math tutoring contexts, limiting the ability to rigorously study and improve de-identification methods for this increasingly important data modality.

To overcome these barriers, this work addresses the following research question (RQ): How can we accurately detect PII in math tutoring transcripts while preserving the educational utility of the data? We operationalize this question

through three sub-questions that structure the remainder of the paper:
RQ1 (Failure analysis): To what extent does numeric ambiguity in mathematical discourse lead to false positive PII detection?
RQ2 (Dataset construction): How can a reliable benchmark for evaluating PII detection be created when original unredacted data cannot be shared?
RQ3 (Method evaluation): Do math-aware prompting strategies improve utility-preserving PII detection compared to domain-agnostic baselines?

To answer these questions, we begin with a large corpus of math tutoring transcripts that had already been redacted for PII by the data provider. The original unredacted data are unavailable because the data provider's internal data safety regulation requires responsible removal of PII before any data sharing can happen. Therefore, we developed a human-in-the-loop large language model (LLM) workflow to audit and repair PII redactions done by the data provider before they shared the data with us and to generate privacy-preserving surrogates. This process yields **MathEd-PII**[1], a benchmark dataset of 1,000 tutoring sessions (115,620 messages; 769,628 tokens) for evaluating PII detection in math tutoring dialogue.

We find that numeric expressions are a major source of false PII detections in math tutoring dialogue. Using a segmentation analysis to examine how detection errors differ between math-dense and conversational regions of transcripts, we observe 55.5% of false redactions to occur in math-dense segments that cover only 36.6% of the corpus, whereas true PII appears substantially less often in these regions (22.2%). We evaluate four PII detection strategies: (i) a Presidio-based baseline, (ii) LLMs with basic prompting, (iii) LLMs with math-aware prompting, and (iv) LLMs with segment-aware prompting. Math-aware prompting substantially improves performance over the baseline (F1: 0.821 vs. 0.379), primarily by reducing numeric false positives. These findings demonstrate that de-identification for tutoring dialogue cannot be treated as a domain-agnostic preprocessing step; preserving analytic utility requires incorporating mathematical discourse context.

This paper makes three contributions aligned with the RQs above. We identify numeric ambiguity as a central failure mode in de-identifying math tutoring dialogue (RQ1). We then introduce MathEd-PII, the first benchmark dataset for PII detection in math tutoring transcripts (RQ2). Finally, we show that math-aware prompting substantially improves utility-preserving de-identification (RQ3).

2. RELATED WORK

2.1 Policy Context

De-identification is a prerequisite for sharing educational interaction data at scale, but what counts as "sufficient" de-identification depends on the legal regime and the assumed adversary model. In the United States, the Children's Online Privacy Protection Act (COPPA) and the Family Educational Rights and Privacy Act (FERPA) govern education records and only permits disclosure of de-identified information when there is no reasonable basis to believe the remaining information can be used to identify an individual [4, 29].

In parallel, the Health Insurance Portability and Accountability Act (HIPAA) provides two common pathways for de-identification of health information–*Safe Harbor* (removal of 18 enumerated identifiers, e.g., names, geographic subdivisions smaller than a state, dates directly related to an individual, and unique identification numbers) and *Expert Determination* (a documented, risk-based assessment) [30]. Although specific to health data, the 18 identifiers are widely used as a practical reference beyond that domain because they offer "the promise of a straightforward application of rules, a repeatable process, and a known result: a dataset that is legally de-identified" [6, p. 24]. Despite this apparent clarity, the National Institute of Standards and Technology (NIST) emphasizes that de-identification is inherently dynamic, framing it along an "identifiability spectrum" and cataloging transformation families such as suppression, generalization, perturbation, masking, and synthesis. Stronger privacy protections, it notes, typically come at the expense of analytic fidelity [6, 7].

2.2 PII Detection in Unstructured Text

Most practical pipelines for unstructured text de-identification begin with *entity detection* (identifying spans that correspond to PII) and then apply an *anonymization action* (e.g., redaction, masking, or surrogate replacement). Historically, detection approaches fall into three broad categories–rule-based, model-based, and LLM-based methods.

Rule-based PII detection systems rely on hand-engineered patterns and resources, including regular expressions for structured identifiers (e.g., emails, phone numbers, and dates), curated lexicons for names and locations, and deterministic context heuristics [21]. These methods can be highly precise for well-formed identifiers but brittle under conversational noise (misspellings, fragments, idiosyncratic formatting), and they require continual maintenance as domains shift [19].

A complementary line of work frames PII detection as a named entity recognition (NER) task, in which statistical or neural models predict entity spans over tokens or characters based on contextual cues. Contemporary NLP toolkits, most notably spaCy, provide pretrained and fine-tunable NER models that are commonly used as baseline PII detectors or adapted to domain-specific corpora [12]. In practice, many production systems combine *rule-based* and *model-based* paradigms to exploit their complementary strengths. Microsoft Presidio exemplifies this hybrid design by integrating rule-based recognizers for high-precision patterns with neural-network-based entity detectors within a unified anonymization framework [16].

More recently, LLMs have been applied to de-identification both via prompting and fine-tuning [24, 25, 31]. In prompting-based setups, the model is typically instructed (often with a schema) to tag or extract PII spans and entity types from raw text–either by returning structured outputs (e.g., JSON with character offsets and labels) or by

[1]https://huggingface.co/datasets/NationalTutoringObservatory/MathEd-PII

emitting an annotated/redacted version of the transcript—whereas fine-tuning-based setups train the model on labeled examples to produce consistent token-level tags or span predictions under a fixed output format [24].

2.3 Augmented and Synthetic data for PII Detection and Evaluation

A practical challenge in PII detection research is the scarcity of richly annotated, shareable corpora, as releasing PII labels alongside text can increase re-identification risk. To address this, two complementary strategies are increasingly common: data augmentation and synthetic data generation.

Data augmentation for PII detection starts from de-identified or manually redacted text and inserts synthetic PII spans, often through surrogate replacement schemes such as "hiding in plain sight," which preserve discourse coherence while reducing disclosure risk [2, 27]. In parallel, fully synthetic datasets have been constructed using template- or rule-driven generators and, more recently, LLMs conditioned on schema constraints (e.g., entity types, formats, and conversational style) [17, 23, 28]. These approaches offer controllability and improved coverage of rare entity types, but their effectiveness depends on the realism of the generated text: synthetic data that fails to capture domain-specific language, error patterns, or contextual ambiguity may yield overly optimistic performance estimates and limited transfer to real-world settings [23, 28].

2.4 De-Identification of Educational Data

Early work on de-identification in educational text focused on providing practical tooling and datasets to support privacy-preserving analysis of student writing and interaction data. Holmes et al. [9, 10] introduced PIILO, an open-source system for labeling and obfuscating PII in educational corpora, and examined de-identification challenges in student-authored texts across digitally mediated learning environments. Building on this foundation, Singhal et al. [25] conducted one of the first systematic evaluations of LLMs for educational data de-identification, using online course discussion forums as a testbed and demonstrating that LLMs substantially improve recall but often over-redact, motivating the need for utility-aware evaluation. Subsequent work by Zambrano et al. [31] expanded this analysis with larger-scale experiments and error analyses, further documenting precision–utility trade-offs in LLM-based anonymization of student forum posts. More recently, Shen et al. [24] proposed enhanced LLM-based pipelines that combine prompting and verification strategies to improve robustness across educational datasets, while Zent et al. [32] introduced an anonymization framework tailored to question-anchored tutoring dialogues, emphasizing deployability and domain adaptation. Together, this line of work reflects a rapid shift from rule-based and toolkit-oriented approaches toward LLM-centric methods, alongside growing recognition that educational domains pose distinct challenges for balancing privacy protection and analytic utility as it is conversational and tightly interleaves task content with incidental personal references.

Math tutoring transcripts present an additional challenge: beyond conversational noise, they contain dense, non-sensitive numerals and operators whose surface forms overlap with common PII categories such as `AGE`, `DATE`, and `ID`. Over-redacting this math-related content would severely reduce the downstream educational and research-related utility of the dataset. However, numerical false positives in PII detection within mathematical contexts have received little attention. A major reason is the scarcity of labeled data: there are only two publicly available educational datasets with PII annotations, one consisting of student-authored articles submitted to a massive open online course (MOOC) on critical thinking through design[2] [11] and the other containing teacher–student chatroom messages for English language learning[3] [1]. To our knowledge, no public datasets exist for PII detection in math education contexts.

3. CURRENT RESEARCH OVERVIEW

To answer the research questions outlined above, we follow a three-phase research workflow moving from dataset construction to analytical validation and finally to evaluation of math-aware de-identification strategies.

Phase 1: Dataset Preparation involves constructing MathEd-PII, a benchmark dataset for PII detection in math tutoring context. The construction includes identifying original over-redaction and establishing a clean ground truth through human-in-the-loop LLM-powered PII surrogation.

Phase 2: Math Segmentation focuses on developing a density-based module to isolate math-heavy regions, serving as an analytical lens to validate the hypothesis of numeric ambiguity that false PII redactions are structurally concentrated in mathematical content while true redactions are not.

Phase 3: PII Detection and Evaluation implements and compares four PII detection strategies—a Presidio-based baseline, LLMs with a basic prompt, a general math-domain-aware prompt, and a specific math-segment-aware prompt—evaluating Precision (P), Recall (R), and F1 overall, as well as stratified by PII types and by math versus non-math segments.

4. PHASE 1: DATASET PREPARATION

Reference datasets for PII detection in math education are currently lacking. To enable rigorous evaluation, we constructed a benchmark dataset, **MathEd-PII**, from a PII-redacted large corpus.

4.1 Source Corpus

Our source corpus comprises 1,000 math tutoring sessions (115,620 messages; 769,628 tokens) from a U.S.-based tutoring platform. Among these, 4,648 (4%) messages spread in 786 sessions feature 5,265 PII redactions applied by the platform's in-house PII detection system–Microsoft Presidio, which, as introduced in 2.2, is an industry-standard open-source software development kit (SDK) for PII detection by integrating rule-based recognizers for high-precision patterns with neural-network-based entity detectors within a unified anonymization framework [16] with customizable

[2]Available at https://www.kaggle.com/datasets/langdonholmes/cleaned-repository-of-annotated-pii/data

[3]Available via request at forms.gle/pKc48WMhnySC8zDk9

rules. The provider redacts 12 types of PII, such as PERSON, LOCATION, US_DRIVER_LICENSE, etc. The full list of PII types can be found in Table 6.

Initial inspection suggests that several redaction types are not consistently reliable indicators of true PII in math tutoring dialogue. In particular, some types that resemble structured numeric identifiers (e.g., COURSE_NUMBER and US_DRIVER_LICENSE) are frequently triggered by math content rather than identity disclosure (see an example of incorrect redactions and inferred original text[4] in Table 1). Other types (notably LOCATION, NRP[5], and URL) appear mixed, containing both correct redaction and substantial over-redaction.

Table 1: Incorrect PII Redactions in the Source Corpus and Inferred Original Text

Redacted Message	Inferred Original Message
9351 / 7 = 1335 <US_DRIVER_LICENSE>	9351 / 7 = 1335.8571
go from radians to degrees it would <COURSE_NUMBER>/pi	go from radians to degrees it would 180/pi
so it would be 1.15x= 1.15x=<PHONE_NUMBER>368000	so it would be 1.15x=

4.2 Method: Human-In-The-Loop PII Redaction Evaluation and Surrogate Generation

To evaluate the extent of incorrect redactions and construct a reliable benchmark for PII detection from the current dataset, we employed a human-in-the-loop LLM workflow to assess the quality of existing PII labels and generate plausible surrogates, thereby restoring transcript completeness for downstream investigation. This design choice is motivated by three practical constraints. First, the original un-redacted transcripts are unavailable due to privacy and data-protection requirements, precluding direct comparison against ground-truth PII. Second, manually reviewing and reconstructing every redaction at scale would be prohibitively costly and time-consuming. Third, LLMs are well-suited for this task because they can leverage rich contextual information to infer the most plausible lexical realizations consistent with surrounding discourse, enabling systematic approximation of redacted content without reintroducing real personal identifiers. Human oversight is incorporated throughout the workflow to validate model outputs, refine prompts, and ensure that generated surrogates remain contextually coherent while preserving privacy. This workflow consists of the following three steps and is illustrated in Figure 1.

First, **Initial LLM-Based Assessment**: We first drafted an initial prompt that instructed the LLM to evaluate messages from a randomly sampled set of 100 transcripts, using a context window of at least three utterances preceding and following each target message. For each message, the LLM was prompted to identify any redacted PII or PII missed by the data provider's internal redaction process, determine its PII type, and classify the redaction as "PII", "Not PII", or "Uncertain" based on contextual evidence. When PII was identified, the LLM was further instructed to generate context-appropriate surrogate text that preserved semantic coherence while remaining substantially different from any original PII, when available.

Second, **Human-In-The-Loop Validation and Refinement**: Three researchers—the first three authors of this paper—then manually reviewed the LLM outputs. Each researcher was assigned one-third of the outputs and down-voted instances in which any component of the output was deemed incorrect, including the identified PII type, quality classification, or the generated surrogate. Based on these annotations, the prompt was revised iteratively to address observed failure modes. This process continued until at least 95% of previously down-voted issues were successfully resolved by the updated prompt.

Third, **Pipeline Execution at Scale**: The final, validated prompt[6] for redaction evaluation and surrogate generation was subsequently applied to the full 1,000-transcript dataset to produce the benchmark used in downstream analyses.

All three steps were conducted using an open-sourced annotation web application, Sandpiper[7], optimized for stable, human-in-the-loop annotation iteration. The system provides a robust, interactive user interface that allows researchers to directly review, up-vote, or down-vote AI-generated annotations, enabling efficient human verification and correction at scale. Its modular architecture and worker-based execution model ensure stable handling of large annotation workloads and repeated LLM inference, supporting consistent evaluation across iterations.

4.3 Source Corpus Evaluation Results

Applying the procedures described above, we discovered both recall and precision issues of the data provider's internal PII redaction presented in the source corpus as described below.

The Recall Issue: The human-in-the-loop LLM workflow discovered 201 messages containing PII that the data provider's system failed to redact, spanning PERSON (159), GRADE_LEVEL (23), AGE (8), DATE (4), COURSE_NUMBER (3), NRP[8] (3), SCHOOL (2), and URL (1). Notably, DATE and AGE were newly introduced PII types under the Safe Harbor framework (see Section 2.1) and were not included among the original dataset's 12 PII categories. Still, this demonstrates a substantial recall issue: a widely used approach to de-identification, Microsoft Presidio, failed to identify 159 names in tutoring transcripts. Table 2 presents simulated examples of missed names. These examples were generated from failure patterns observed in the original dataset and do not refer to any real

[4]Original un-redacted text is not available due to the data provider's privacy compliance. The original text here is inferred by an LLM based on the context. Please see Section 4.2 for more details.

[5]A person's Nationality, Religious or Political group.

[6]see 1 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

[7]https://sandpiperresearch.org/

[8]see foot note 5

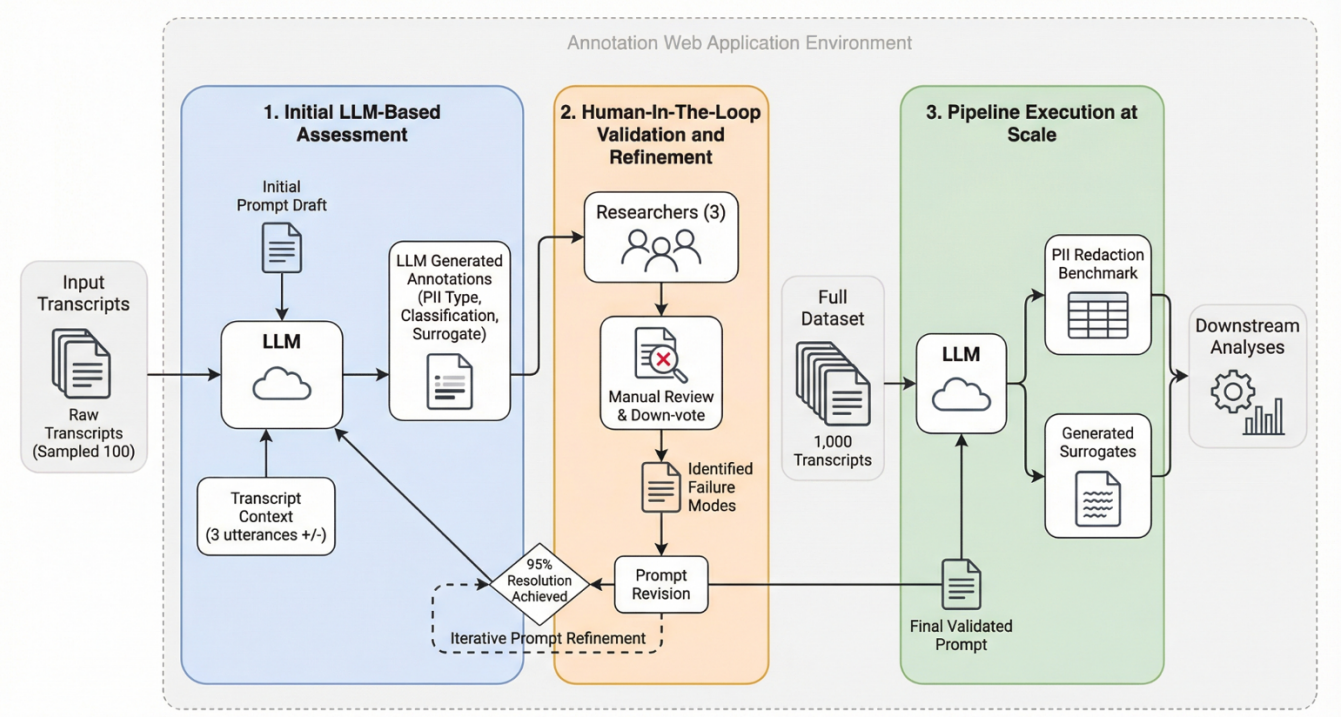


Figure 1: The Three-Step Human-In-The-Loop MathEd-PII Benchmark Dataset Preparation Workflow.

people. Readers can replicate the same failure using Presidio's default Analyzer Engine with customized rules.

Table 2: Simulated Examples of `PERSON` that Presidio Failed to Detect and Similar Cases that Presidio Successfully Detected

Unsuccessful Detection	Successful Detection
hey olivia	hey Olivia
Hi Alex	Hi Alex,
I am Lam	i am Lam

The Precison Issue: Of the 4,648 originally redacted messages, 3,015 (64.87%) contained redactions evaluated as "Not PII", indicating low precision in their internal system. Several PII types contained predominantly false positives, with the following four types receiving more than 90% "Not PII" flags–`US_DRIVER_LICENSE` (99.36%), `COURSE_NUMBER` (96.65%), `PHONE_NUMBER` (93.33%), and `NRP` (90.64%), the first three of which are all numeric identifiers further corroborating our initial findings. Table 1 presents some examples of such over-redaction.

4.4 The MathEd-PII Surrogate Benchmark

By replacing all of the originally redacted PII and newly identified PII with LLM-generated, context-aware surrogates and removing PII annotations for those that were labeled as "Not PII", we obtained the MathEd-PII surrogate benchmark dataset. A comparison of basic statistics for the source corpus and MathEd-PII is presented in Table 3.

Table 3: PII Statistics Comparison between Source Corpus and MathEd-PII (ordered by the most common PII to the least in MathEd-PII)

Category	Source Corpus	MathEd-PII
Transcripts	1,000	1,000
Messages	115,620	115,620
PII Labels (Total)	5,263	1,995
PERSON	1,915	1,424
URL	245	187
LOCATION	595	121
GRADE_LEVEL	87	107
SCHOOL	88	73
COURSE_NUMBER	1,103	40
NRP	235	25
AGE	0	8
DATE	0	4
US_DRIVER_LICENSE	941	2
PHONE_NUMBER	30	2
IP_ADDRESS	2	2
US_BANK_NUMBER	20	0
US_PASSPORT	2	0

5. PHASE 2: MATH SEGMENTATION AND NUMERIC AMBIGUITY

Since major "Not PII" redactions are numeric identifiers, a natural explanation is *numeric ambiguity*: mathematical content in tutoring transcripts often shares surface-level similarities with numeric PII, making the two difficult to distinguish. Assessing the extent to which numeric ambiguity degrades PII detection—and developing effective mitigation

strategies—therefore requires distinguishing math-dense regions from non-mathematical conversational text.

5.1 Method: Math Density, Semantic Similarity, and Segmentation Optimization

To distinguish math-focused sections in a transcript, we applied a segmentation strategy by first identifying anchor math messages with high density of math vocabulary and then expanding the segment based on semantic similarity [5, 13, 20].

The construction of the math vocabulary set was informed by a comprehensive list of 2,824 K-12 math learning components[9] curated by the Learning Commons initiative and the Achievement Network [15], which are granular representations of individual math skills and concepts encompassing the Common Core State Standards for Mathematics (CCSSM) and 18 U.S. state math standards (e.g., California, New York, Texas, and Washington). Some examples of the learning components are "Use conversions to solve multi-step real-world problems", "Determine the lateral surface area of three-dimensional cylinders in real-world problems", and "Represent proportional relationships with equations". The math vocabulary used for math density calculation includes a list of math terms derived from these learning components and regular expressions (RegEx) of common math expressions. The vocabulary can be found here[10].

The **Math Density** (D_{math}) of a message is a weighted sum of single-word math vocabulary terms (V_{math}), multi-word phrases (P_{phrase}), and RegEx patterns ($R_{pattern}$):

$$D_{math}(m) = \frac{1}{|W|}\left[\sum_{w\in W}\mathbb{I}(w \in V) + 1.5\sum_{p\in P}C(p,m) + 2.0\sum_{r\in R}C(r,m)\right] \quad (1)$$

where W is the set of tokens in the message, $\mathbb{I}(\cdot)$ is the indicator function, $C(\cdot, m)$ counts Occurrences in message m. The weights were empirically set to attribute higher significance to multi-term concepts and structural math patterns.

A message exceeding a threshold of Math Density, *Anchor Threshold* (T_{anchor}) at 0.05, becomes an anchor message and initializes a math segment. Adjacent messages are added to the math segment if their cosine similarity to the segment centroid (the aggregated semantic meaning of the math context accumulated in the segment computed via `all-MiniLM-L6-v2` embeddings) exceeds a *Similarity Threshold* (T_{sim}) at 0.3. The thresholds were determined via a grid search optimization process around the average math density of the dataset (0.07) from 0.05 to 0.10 in increments of 0.01 and a range of similarity threshold from 0 to 0.5 with a step size of 0.1 with the objective to maximize the capture of "Not PII" (false positives) and minimize the capture of "PII" (true positives) by the math segments. Notably, across all threshold configurations evaluated in the grid search, the proportion of false positives consistently exceeded the proportion of true positives captured by the math segments. This indicates that the finding elaborated below is robust to the choice of threshold. The optimization process and analysis can be seen here[11].

5.2 False Positive PII in Math Segments and Numeric Ambiguity

Figure 2 illustrates the distribution of PII categories across mathematical and non-mathematical segments, distinguishing between true positives (validated PII) and false positives (Not PII) arising from numeric ambiguity.

It reveals a stark contrast in detection accuracy between the two domains. In math segments, the data provider's in-house detection system produced 2,539 false positives compared to only 520 true PII instances—a false-to-true ratio of approximately 4.88:1. In contrast, non-math segments exhibited a much higher precision, with 929 false positives and 1,455 true PII instances (a ratio of 0.64:1).

Specific numeric categories were nearly entirely composed of false positives in mathematical contexts. For instance, as shown in the MATH Segment of Figure 2, categories such as `DATE`, `US_DRIVER_LICENSE`, and `LOCATION` contain substantial numbers of false positives. Interestingly, the `PERSON` category also presents a high number of false positives within math segments. Qualitative review indicates that these names are frequently tied to mathematical word problems (e.g., "If Alex has 5 apples...") rather than the actual identities of the student or tutor. While over-redacting these fictional entities is perhaps less detrimental to educational utility than over-redacting the mathematical operations themselves, it further illustrates the inability of domain-agnostic systems to distinguish between the *context* of a name (math problem vs. personal disclosure).

This empirical evidence confirms that numeric ambiguity is a structural failure mode in de-identifying math tutoring transcripts: the surface forms of mathematical expressions and problem-related entities systematically overlap with structured personal identifiers, leading generic detection systems to over-redact critical instructional content.

6. PHASE 3: PII DETECTION AND EVALUATION

6.1 PII Detection Methods

6.1.1 Baseline: Microsoft Presidio

As a baseline, we deployed Microsoft Presidio (v2.2) [16]. We utilized its default analyzer which orchestrates a set of predefined recognizers. For high-structure entities (e.g., `EMAIL_ADDRESS`, `URL`, `IP_ADDRESS`, `PHONE_NUMBER`, `US_SSN`, `US_DRIVER_LICENSE`), Presidio employs regular expressions (RegEx). For context-dependent entities (e.g., `PERSON`, `LOCATION`, `NRP`), it leverages a spaCy-based NER model (`en_core_web_lg` and `en_core_web_trf` used here). To address domain-specific requirements, we implemented custom RegEx-based recognizers for `SCHOOL`, `COURSE_NUMBER`, and

[9] https://docs.learningcommons.org/knowledge-graph/understanding-knowledge-graph/about-knowledge-graph

[10] see 2 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

[11] see 3 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

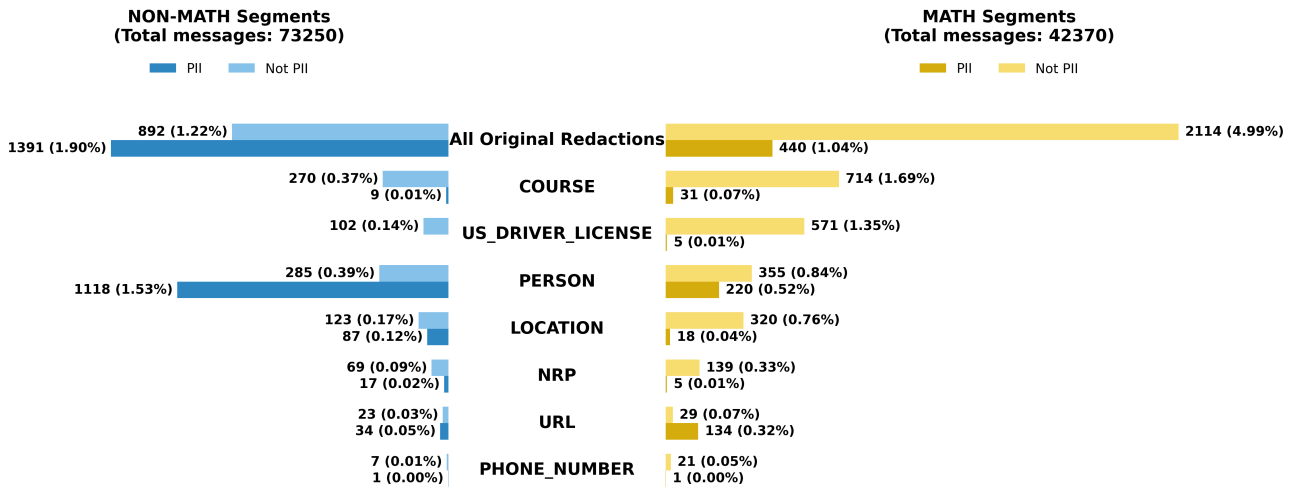


Figure 2: Distribution of Original PII Redactions from the Data Provider's In-House System (Presidio and Customized Rules) across Different PII Types and Math and Non-Math Segments in terms of true PII and false positives ("Not PII"). Types are ordered by the frequency of "Not PII" in Math Segments. The first seven types are presented.

`GRADE_LEVEL`. We mapped Presidio's output entities to the 17 target PII categories.

6.1.2 *LLMs using Basic, Math-Aware, and Segment-Aware Prompts*

We evaluated the performance of six LLMs—Claude 4.5 Haiku, Claude 4.5 Opus, GPT-4 Turbo, GPT-5.2, Gemini 2.5 Flash, and Gemini 3 Pro—on the task of PII detection to examine the robustness of LLM-based methods. These models were selected because they are widely used and span lightweight, general-purpose, and high-capability model regimes. To ensure data security and compliance, all API requests were routed through Cornell University's LLM Gateway, LiteLLM[12]. This centralized gateway provides a unified interface while enforcing strict privacy protocols; crucially, it operates under a zero-retention policy where payload data is neither logged by the gateway nor used for model training, aligning with institutional IRB requirements.

We first compared model performance under basic prompting against the baseline and selected the higher-performing model from each provider for subsequent evaluation with the math-aware and segment-aware prompting strategies, in order to assess whether these refinements further improve PII detection performance. The three prompt variants are described below.

Basic Prompt: A straightforward instruction acting as a PII specialist. It lists the 17 PII categories and requests the model to identify "ALL PII" in the message, returning a JSON list of text spans and types (exact text here[13]).

Math-Aware Prompt: A refined version of the basic prompt that explicitly addresses the domain challenge. This prompt aims to reduce false positives stemming from numeric ambiguity by leveraging the model's general knowledge about math tutoring (exact text here[14]).

Segment-Aware Prompt: A refined version of the basic prompt that explicitly points to the math segment label for each message. This prompt aims to reduce false positives stemming from numeric ambiguity through explicit contextual grounding (exact text in here[15]).

6.1.3 *Evaluation Metrics and Statistical Analysis*

We evaluated PII detection performance using standard metrics: Precision, Recall, and the F1-score. Precision measures the proportion of detected text spans that correctly match the ground truth, Recall indicates the proportion of actual PII spans successfully identified, and the F1-score represents their harmonic mean. To account for the inherent variability and potential clustering of errors within different tutoring sessions, we calculated 95% confidence intervals (CIs) using a bootstrap resampling method. We performed 1,000 bootstrap iterations by resampling tutoring transcripts (the primary unit of our corpus) with replacement. For each iteration, metrics were aggregated across the resampled sessions. The 95% CIs reported in our results correspond to the 2.5th and 97.5th percentiles of the resulting bootstrap distribution.

6.2 PII Detection Performance

6.2.1 *Overall Performance*

Table 4 summarizes the performance of Microsoft Presidio (baseline) and various LLMs using three types of prompts.

The baseline Presidio models achieved high recall (with the

[12]https://docs.litellm.ai/

[13]see 4 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

[14]see 5 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

[15]see 6 in https://github.com/ZhuqianZhou/MathEd-PII/blob/69012c0dbde3b3976f36533e2e44f5a5bfdfd718/EDM2026_paper_183_Appendix.pdf

Table 4: PII Detection Performance Metrics (with 95% Bootstrap Confidence Intervals)

Model	Precision	Recall	F1
Baseline			
Presidio (Large)	0.254 [0.225, 0.285]	0.747 [0.694, 0.794]	0.379 [0.344, 0.415]
Presidio (Transformer)	0.230 [0.208, 0.253]	**0.781** [0.731, 0.826]	0.355 [0.328, 0.385]
Basic Prompting			
Claude 4.5 Haiku	0.831 [0.802, 0.860]	0.620 [0.565, 0.675]	0.710 [0.670, 0.752]
Claude 4.5 Opus	0.911 [0.861, 0.946]	0.730 [0.677, 0.783]	0.810 [0.762, 0.853]
Gemini 2.5 Flash	0.848 [0.806, 0.887]	0.771 [0.723, 0.818]	0.807 [0.769, 0.844]
Gemini 3 Pro	0.887 [0.863, 0.910]	0.760 [0.706, 0.812]	0.818 [0.781, 0.851]
GPT-4 Turbo	0.732 [0.695, 0.769]	0.578 [0.530, 0.630]	0.646 [0.609, 0.685]
GPT-5.2	0.759 [0.723, 0.795]	0.707 [0.656, 0.759]	0.732 [0.695, 0.767]
Math-Aware Prompting			
Claude 4.5 Opus	0.905 [0.860, 0.938]	0.720 [0.661, 0.769]	0.802 [0.756, 0.842]
Gemini 3 Pro	0.869 [0.826, 0.906]	0.735 [0.685, 0.787]	0.796 [0.756, 0.835]
GPT-5.2	0.774 [0.740, 0.809]	0.721 [0.670, 0.764]	0.747 [0.711, 0.780]
Segment-Aware Prompting			
Claude 4.5 Opus	**0.934** [0.913, 0.951]	0.730 [0.682, 0.784]	0.820 [0.785, 0.855]
Gemini 3 Pro	0.888 [0.862, 0.913]	0.764 [0.713, 0.811]	**0.821** [0.788, 0.851]
GPT-5.2	0.800 [0.760, 0.838]	0.720 [0.670, 0.766]	0.758 [0.720, 0.793]

Transformer version reaching a point estimate of 78.1%), but suffered from exceptionally poor precision (spanning 0.23–0.25). This confirms that generic PII detection tools tend to over-redact math-rich content by failing to distinguish mathematical notation from sensitive identifiers. While the point estimates for Presidio's recall appear high, several top-performing LLMs achieved recall levels that are statistically comparable (with overlapping 95% bootstrap confidence intervals) while providing vastly superior precision and F1 scores. Based on these initial results, Claude 4.5 Opus (Anthropic), Gemini 3 Pro (Google), and GPT-5.2 (OpenAI) were selected as the primary focus for evaluating domain-specific prompt refinements.

Math-aware prompting generally improved the precision of the selected models compared to the basic prompt across most configurations. Segment-aware prompting, which explicitly grounds the detection process in the mathematical context of the message, yielded the highest overall performance for all three models. Notably, Claude 4.5 Opus achieved the highest precision (0.934, 95% CI [0.913, 0.951]) under this condition, while Gemini 3 Pro achieved the highest overall F1 score (0.821, 95% CI [0.788, 0.851]). Crucially, the segment-aware versions of these models maintain recall levels that are on par with the baseline, while reducing false positives by as much as 90%. These results demonstrate that incorporating mathematical discourse context is essential for minimizing over-redaction.

6.2.2 Performance by PII Categories

Given that both the baseline and the top-performing LLMs achieved comparable recall levels (as discussed in the previous section), we zoom in on category-level performance to investigate how domain-aware prompting reduces false positives to achieve superior precision. Table 5 provides a breakdown of the specific sources of over-redaction, focusing on False Positives (FP) and Precision across the target PII categories under two model configurations: the highest-performing baseline, Presidio LG (using the `en_core_web_lg` SpaCy NER model), and the highest-performing LLM configuration, Gemini 3 Pro with segment-aware prompting.

Table 5: PII Detection False Positives and Precision by Category (ordered by Precision from the highest to the lowest from Gemini 3 Pro with Segment Prompting)

PII Type	Total	Presidio LG		Gemini 3 Pro Seg.	
		FP	Prec	FP	Prec
US_DRIVER_LICENSE	2	344	0.000	0	1.000
PERSON	1422	330	0.775	63	0.949
LOCATION	121	153	0.349	5	0.896
NRP	25	45	0.286	2	0.857
URL	187	21	0.859	21	0.852
SCHOOL	73	17	0.638	10	0.831
GRADE_LEVEL	103	2	0.973	22	0.750
IP_ADDRESS	2	0	1.000	1	0.667
PHONE_NUMBER	2	6	0.250	1	0.667
AGE	8	3	0.000	6	0.538
COURSE_NUMBER	40	46	0.193	43	0.411
DATE	3	3386	0.001	16	0.059
US_BANK_NUMBER	0	5	0.000	-	-

The most striking improvement is observed in numeric categories where the baseline is particularly prone to over-redact mathematical entities. For instance, Presidio generated 3,386 false positives for `DATE` and 344 for `US_DRIVER_LICENSE`, confirming that rule-based systems frequently misinterpret math spans as structured identifiers. In contrast, Gemini 3 Pro with segment-aware prompting reduced `DATE` false positives by over 99.5% (down to 16) and eliminated false positives for `US_DRIVER_LICENSE` entirely. Similarly, for context-heavy categories like `PERSON` and `LOCATION`, the LLM significantly improved precision by reducing the number of false detections. These results underscore that the superior performance of domain-aware LLMs is primarily driven by their ability to selectively filter mathematical noise while maintaining a capture rate on par with standard tools.

6.2.3 Performance by Segment Types

To further investigate how well the LLM addressed numeric ambiguity, we evaluated model performance separately for messages labeled as NON-MATH versus MATH conversational segments. Table 6 reveals a consistent performance gap across the models, with precision being systematically lower in the MATH segment compared to the NON-MATH segment.

The stratified analysis in Table 6 provides strong empirical evidence for the numeric ambiguity hypothesis and demonstrates the effectiveness of segment-aware prompting in mitigating it. The baseline Presidio LG model exhibits a big drop in precision when moving from NON-MATH (0.346) to

Table 6: PII Detection Performance by Segment Type (NON-MATH vs. MATH)

Model	Prompt	NON-MATH Segments			MATH Segments		
		Prec	Rec	F1	Prec	Rec	F1
Presidio LG	-	0.346	0.756	0.475	0.142	0.720	0.237
Gemini 3 Pro	Basic	0.913	0.819	0.864	0.800	0.591	0.680
Gemini 3 Pro	Segment	0.909	0.815	0.860	0.819	0.618	0.705

MATH segments (0.142), confirming that generic rule-based systems frequently misinterpret mathematical notation as sensitive identifiers. This 59% decline in precision within math-dense regions highlights the risk of over-redaction in tutoring transcripts when domain context is ignored.

In contrast, the Gemini 3 Pro model maintains substantial precision in both contexts, though a performance gap between segments remains. Crucially, the introduction of segment-aware prompting specifically improves the model's ability to navigate numeric ambiguity within mathematical discourse. In MATH segments, segment-aware prompting increases both precision (0.800 to 0.819) and recall (0.591 to 0.618) compared to basic prompting. This simultaneous gain in both metrics is particularly noteworthy; it suggests that providing the model with an explicit context label allows it to adopt a more nuanced decision boundary—correctly identifying PII that it might otherwise miss (higher recall) while simultaneously avoiding false detections triggered by mathematical symbols (higher precision). The negligible difference in performance within NON-MATH segments further indicates that the benefits of this strategy are structurally concentrated in the regions where numeric ambiguity is most severe.

7. DISCUSSION AND CONCLUSION

This study investigated the challenge of utility-preserving de-identification in the context of math tutoring transcripts, focusing on the phenomenon of numeric ambiguity. By introducing MathEd-PII, the first benchmark dataset for this domain, we provided a rigorous foundation for evaluating PII detection methods that balance privacy protection with instructional fidelity.

Our findings provide strong empirical evidence that numeric ambiguity is a structural failure mode in standard PII detection systems. Generic tools like Microsoft Presidio often misinterpret mathematical expressions as sensitive identifiers, such as dates or driver's license numbers. Our density-based segmentation analysis revealed that over half of all false positives (55.5%) were concentrated in math-dense regions, which comprise only about a third of the corpus tokens. In these regions, the ratio of false positives to true PII reached nearly 5:1, highlighting a significant risk of removing core instructional content that is essential for educational research.

We demonstrated that LLMs, particularly when guided by domain-aware prompting strategies, significantly outperform traditional baselines. While Presidio achieved a respectable recall, its low precision (F1: 0.380) renders it problematic for creating high-utility datasets. In contrast, Gemini 3 Pro with segment-aware prompting achieved an F1 of 0.821, maintaining high recall while reducing false positives by over 90% in critical numeric categories. The success of segment-aware prompting suggests that providing models with explicit contextual grounding—identifying whether a message is likely to be mathematical—allows them to adopt more nuanced decision boundaries that better distinguish task-content from personal disclosure.

The creation of MathEd-PII through a human-in-the-loop LLM surrogation workflow also offers a methodological contribution. It demonstrates how high-quality benchmark data-sets can be generated even when original, unredacted data cannot be directly shared, by leveraging the generative capabilities of LLMs to create privacy-preserving surrogates that maintain discourse coherence.

However, several limitations and avenues for future work remain. First, while MathEd-PII is the first math-focused PII surrogate benchmark, it is based on 1,000 sessions from a single U.S.-based tutoring platform. As a result, the generalization behavior of the proposed methodology across broader educational domains remains an open question. Expanding this benchmark to include more diverse curricula, grade levels, instructional settings, and international contexts would improve the generalizability of our findings and enable more systematic evaluation across heterogeneous datasets. More broadly, our approach to ensuring applicability across different contexts is to emphasize domain-aware prompting principles that are adaptable to new educational settings, while future work will focus on validating these strategies on additional datasets and subject areas beyond mathematics tutoring.

Second, while this work focused primarily on addressing the precision issues caused by numeric ambiguity, further optimizing recall remains an important avenue for future exploration. Although our domain-aware prompting strategies yielded substantial improvements in F1 and maintained recall levels comparable to standard baselines, a portion of PII still remains undetected. While a detailed investigation into maximizing recall was beyond the scope of this paper, there are several promising directions to bridge this gap, such as: (1) developing multi-stage "cascade" architectures that combine high-recall rule-based systems with high-precision LLM verification; (2) exploring advanced prompting techniques like multi-agent debate to enhance sensitivity to subtle PII mentions; and (3) leveraging the MathEd-PII dataset to fine-tune specialized models that capture domain-specific patterns more effectively and robustly than prompt-only approaches. In addition, the current methodology relies entirely on prompting large language models, which may be brittle across model versions, prompting styles, or unseen distributions. Fine-tuning specialized domain-adapted models therefore represents an important future direction for improving robustness, consistency, and transferability across datasets and deployment settings. Finally, future work could also investigate making these solutions more efficient for low-cost, on-device deployment at educational institutions.

In conclusion, as dialogue-based interaction data becomes increasingly available to educational data mining community, the need for domain-aware de-identification becomes paramount. This work shows that by incorporating mathe-

matical context into the de-identification pipeline, we can move closer to the goal of "utility-preserving privacy"—protec-ting student and tutor identities without sacrificing the very data that makes educational research possible.

8. ACKNOWLEDGMENTS

This material is based upon initial work completed under National Science Foundation Grant No. 2321499, and support from the Gates Foundation and the Chan Zuckerberg Initiative. Any opinions expressed in this material are those of the authors and do not necessarily reflect funders' views.

9. DATA AVAILABILITY

The MathEd-PII dataset is publicly available at https://huggingface.co/datasets/NationalTutoringObservatory/MathEd-PII.

10. REFERENCES

[1] A. Caines, H. Yannakoudakis, H. Allen, P. Pérez-Paredes, B. Byrne, and P. Buttery. The teacher-student chatroom corpus version 2: more lessons, new annotation, automatic detection of sequence shifts. In D. Alfter, E. Volodina, T. François, P. Desmet, F. Cornillie, A. Jönsson, and E. Rennes, editors, *Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning*, pages 23–35, Louvain-la-Neuve, Belgium, Dec. 2022. LiU Electronic Press.

[2] D. S. Carrell, B. Malin, J. Aberdeen, S. Bayer, and C. Clark. Hiding in plain sight: Use of realistic surrogates to reduce exposure of protected health information in clinical text. *Journal of the American Medical Informatics Association*, pages 342–348, 2013.

[3] G. Deacon and G. Chojnacki. Impacts of upchieve on-demand tutoring on students' math knowledge and perceptions. middle years math grantee report series. *Mathematica*, 2023.

[4] Federal Trade Commission. Children's online privacy protection act (coppa) guidance. `https://www.ftc.gov/business-guidance/resources/complying-coppa-frequently-asked-questions`, 2013. Accessed: 2026-02-10.

[5] M. Galley, K. McKeown, J. Hirschberg, and E. Shriberg. Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 562–569. Association for Computational Linguistics, 2003.

[6] S. L. Garfinkel. De-identification of personal information. NIST Interagency Report 8053, National Institute of Standards and Technology, Oct. 2015.

[7] S. L. Garfinkel. De-identifying government data sets: Techniques and governance. NIST Special Publication 800-188, National Institute of Standards and Technology, Sept. 2023.

[8] O. Henkel, H. Horne-Robinson, N. Kozhakhmetova, and A. Lee. Effective and scalable math support: Evidence on the impact of an ai-tutor on math achievement in ghana. *arXiv preprint arXiv:2402.09809*, 2024.

[9] L. Holmes, S. Crossley, N. Hayes, D. Kuehl, A. Trumbore, and G. Gutu-Robu. De-identification of student writing in technologically mediated educational settings. In *Polyphonic Construction of Smart Learning Ecosystems: Improving Inclusive Digital Education*, pages 177–189, Singapore, 2023. Springer Nature Singapore.

[10] L. Holmes, S. Crossley, H. Sikka, and W. Morris. Piilo: an open-source system for personally identifiable information labeling and obfuscation. *Information and Learning Sciences*, 124(9-10):266–284, Oct. 2023.

[11] L. Holmes, S. Crossley, J. Wang, and W. Zhang. The cleaned repository of annotated personally identifiable information. In P. Benjamin and D. E. Carrie, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 790–796, Atlanta, Georgia, USA, July 2024.

[12] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spacy: Industrial-strength natural language processing in python. `https://doi.org/10.5281/zenodo.1212303`, 2020.

[13] Z. Huang, W. Xu, and K. Yu. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv: 1508.01991*, 2015.

[14] K. R. Koedinger, R. S. Baker, K. Cunningham, A. Skogsholm, B. Leber, and J. Stamper. A data repository for the edm community: The pslc datashop. *Handbook of educational data mining*, 43:43–56, 2010.

[15] Learning Commons Initiative. Learning components. `https://docs.learningcommons.org/knowledge-graph/entity-and-relationship-reference/learning-components`, 2024. Accessed: 2026-02-05.

[16] Microsoft. Presidio: Data protection and de-identification sdk. `https://github.com/microsoft/presidio`, 2020. GitHub repository. Accessed 2026-01-20.

[17] Microsoft. presidio-research: Research utilities for Presidio (including synthetic text generation). `https://github.com/microsoft/presidio-research`, 2021. GitHub repository. Accessed 2026-01-20.

[18] M. C. Mihaescu and P. S. Popescu. Review on publicly available datasets for educational data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 11(3):e1403, 2021.

[19] K. Mishra, H. Pagare, and K. Sharma. A hybrid rule-based NLP and machine learning approach for PII detection and anonymization in financial documents. *Scientific Reports*, 15(1):22729, 2025.

[20] H. Misra, F. Yvon, O. Cappé, and C. Jose. Text segmentation: A topic modeling perspective. *Information Processing & Management*, 47(4):528–544, 2011.

[21] I. Neamatullah, M. M. Douglass, L.-W. H. Lehman, A. Reisner, M. Villarroel, W. J. Long, P. Szolovits, G. B. Moody, R. G. Mark, and G. D. Clifford. Automated de-identification of free-text medical records. *BMC Medical Informatics and Decision Making*, 8(32), 2008.

[22] E. Prihar, A. Moore, and N. Heffernan. Identifying explanations within student-tutor chat logs. In *Proceedings of the 15th International Conference on Educational Data Mining*, page 773, 2022.

[23] M. Savkin, T. Ionov, and V. Konovalov. SPY: Enhancing privacy with synthetic PII detection

dataset. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 236–246. Association for Computational Linguistics, 2025.

[24] Y. Shen, Z. Ji, J. Lin, and K. R. Koedinger. Enhancing the de-identification of personally identifiable information in educational data. *arXiv preprint arXiv: 2501.09765*, 2025.

[25] S. Singhal, A. F. Zambrano, M. Pankiewicz, X. Liu, C. Porter, and R. S. Baker. De-identifying student personally identifying information with gpt-4. In *Proceedings of the 17th International Conference on Educational Data Mining (EDM)*, pages 559–565, 2024.

[26] N. Slijepcevic and A. Yaylali. Leveraging "khanmigo" generative ai-powered tool for personalized tutoring to learn scientific concepts. *Journal of Teaching and Learning*, 19(4):155–178, 2025.

[27] A. Stubbs and Ö. Uzuner. Automated systems for the de-identification of longitudinal clinical narratives: Overview of the 2014 i2b2/uthealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19, 2015.

[28] R. Tang, X. Han, X. Jiang, and X. Hu. Does synthetic data generation of LLMs help clinical text mining? *arXiv preprint arXiv: 2303.04360*, 2023.

[29] U.S. Department of Education, Privacy Technical Assistance Center (PTAC). Data de-identification: An overview of basic terms. `https://studentprivacy.ed.gov/sites/default/files/resource_document/file/data_deidentification_terms_0.pdf`, 2012. Updated May 2013. Accessed 2026-01-20.

[30] U.S. Department of Health and Human Services, Office for Civil Rights. Guidance regarding methods for de-identification of protected health information in accordance with the HIPAA privacy rule. `https://www.hhs.gov/sites/default/files/ocr/privacy/hipaa/understanding/coveredentities/De-identification/hhs_deid_guidance.pdf`, 2012. Accessed 2026-01-20.

[31] J. Zambrano, K. Singhal, L. Pankiewicz, R. Baker, L. Porter, and L. Liu. De-identifying student personally identifying information in discussion forum posts with large language models. *Information and Learning Sciences*, 126(5/6):401–424, 2025.

[32] M. Zent, D. Smith, and S. Woodhead. PIIvot: A lightweight NLP anonymization framework for question-anchored tutoring dialogues. *arXiv*, 2025.

What Changes When the Interlocutor Is an AI? Interactional Fluency and Linguistic Uptake in L2 Spoken Dialogue

Russell Scheinberg
Portland State University
rschein2@pdx.edu

Ameeta Agrawal
Portland State University
ameeta@pdx.edu

Tetyana Sydorenko
Portland State University
tsydorenko@pdx.edu

Kalab Kahsay
Portland State University
kalabk@pdx.edu

Nina Vyatkina
University of Kansas
vyatkina@ku.edu

Griet Boone
University of Antwerp
griet.boone@uantwerpen.be

ABSTRACT

Voice-based AI systems are increasingly used for L2 speaking practice, but evaluations rarely characterize the interactional processes they create. We analyze 78 university learners of German across four sites completing a counterbalanced spot-the-difference task with both a human peer and a real-time AI partner. From diarized ASR transcripts, we extract measures of interactional fluency, linguistic uptake, and learner experience. Human dialogue was faster and more balanced, with many short turns; AI dialogue resembled supported monologue, with fewer, longer turns, reduced learner floor share, and greater within-turn fluency. The AI's verbose, syntactically regular input was associated with greater short-term uptake and stronger syntactic priming after controlling for input volume. Attitudes toward AI improved after the task, and satisfaction was predicted by production fluency rather than uptake. The results show complementary affordances for AI and human dialogue in L2 practice.

Keywords

interactional fluency, L2 speech, voice-based AI, turn-taking, educational data mining

1. INTRODUCTION

Voice-based conversational AI is increasingly positioned as a scalable speaking partner for second language (L2) learners. Yet these systems do not simply provide more practice, they change the *nature* of the interaction [1, 2, 29, 11, 22]. AI-based interlocutors, especially using large language models (LLMs), produce longer turns, respond at atypical moments, and lack the backchannels and floor-negotiation strategies of human partners [22, 29, 11, 30, 5]. If interaction structure shapes learning opportunities, then characterizing *how* AI-mediated dialogue differs from human dialogue is essential for informed system design and deployment.

Russell Scheinberg, Ameeta Agrawal, Tetyana Sydorenko, Kalab Kahsay, Nina Vyatkina, and Griet Boone. What Changes When the Interlocutor Is an AI? Interactional Fluency and Linguistic Uptake in L2 Spoken Dialogue. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 376–387. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039907

Most evaluations of dialogue-based computer-assisted language learning (CALL) focus on outcome measures: pre/-post speaking scores, perceived confidence, or learner satisfaction. Meta-analytic evidence confirms positive effects overall but highlights substantial heterogeneity across systems, tasks, and evaluation criteria [2, 17, 10]. What remains underexplored is the *interactional process* itself—how learners and systems jointly construct spoken interaction during practice [19].

We address this gap through three complementary lenses. The first is *interactional fluency*: measurable properties of conversational coordination—response latency, turn-length distributions, floor sharing, and within-turn pausing—that reflect how speakers manage participation in real-time dialogue. While the distinction between monologic and dialogic fluency is well established in SLA research [38], less is known about how these dynamics shift when the interlocutor is a machine.

The second is *linguistic uptake*. Conversational alignment theory holds that interlocutors converge on lexical and syntactic forms during interaction [28], a process proposed as a mechanism supporting L2 learning [6, 23]. When partners differ fundamentally in how they generate language, it is an open question whether they afford similar uptake opportunities. We operationalize this as direction-sensitive lexical and morphosyntactic uptake: the extent to which a learner reuses words or constructions introduced by the interlocutor in the preceding context.

The third is *learner experience*. If AI and human dialogue create different interactional conditions, learners may perceive these differences, and the behavioral features of the interaction (e.g., production fluency, floor balance) may predict satisfaction independently of what learners linguistically adopt.

These three lenses motivate the following research questions:

RQ1: How do **interactional fluency** measures differ between human–human and human–AI dialogue?

RQ2: Do learners differ in **lexical and morphosyntactic uptake** from a human partner versus an AI-interlocutor, and do these differences vary across linguistic levels?

RQ3: How do learner **attitudes toward AI** change after direct experience, and what behavioral features of the interaction predict satisfaction?

To investigate these questions, we conducted a controlled, multi-site study with second- and third-year L2 German learners ($N = 78$) at four universities in the United States and Europe. Participants completed a spot-the-difference task under four within-speaker conditions: two monologic scene descriptions and two dialogues (one with a human peer, one with a real-time AI partner). All measures are extracted from diarized, timestamped ASR transcripts without manual coding. Results reveal two distinct interaction modes: human dialogue is rapid and balanced, whereas AI dialogue resembles a supported monologue, with fewer but longer turns, slower responses, reduced learner floor share, and greater within-turn fluency. AI verbosity and syntactic regularity afford more uptake opportunities, and learners show stronger short-term syntactic priming with the AI partner even after controlling for input volume. Attitudes toward AI improve after the task, with satisfaction tied to production fluency rather than linguistic uptake.

This paper makes three contributions: (1) we operationalize interactional fluency and direction-sensitive uptake for spoken AI practice using metrics extractable from diarized ASR output without manual linguistic annotation; (2) we leverage a within-speaker design across four instructional sites to isolate how the same learner's interaction changes with a human versus AI partner; and (3) we provide a reusable analysis pipeline for rapid, scalable evaluation of conversational AI in educational settings. The remainder of the paper describes the study design (§3), methods and results for interactional fluency (§4.1), uptake (§4.2), and survey analyses (§5), followed by discussion and design implications (§6).

2. RELATED WORK

Prior work establishes the promise of conversational practice for language learning, but also suggests that who (or what) the interlocutor is can reshape the interaction itself.

Conversational practice and transcript-based measurement. Dialogue-based computer-assisted language learning (CALL) has a long history motivated by the role of interaction in second language development. Meta-analytic evidence suggests that dialogue-based CALL systems can yield positive learning outcomes overall, while also reporting substantial heterogeneity across system types, tasks, and evaluation measures [2]. Recent reviews of conversational agents for language learning echo both pedagogical promise and persistent concerns about reliability, pedagogical alignment, and evaluation practices [17, 10]. As voice-based LLM systems become plausible speaking partners, a central question is not only whether learners improve, but what kinds of interaction these systems create and how those interactional properties shape practice opportunities [19].

A parallel development in learning analytics is the growing treatment of dialogue as analyzable behavioral trace data. This shift enables process-level evaluation of learning-support interactions, but it also raises a practical challenge for spoken settings: extracting reliable measures without extensive manual annotation. Recent work demonstrates that transcript-based pipelines can extract educationally meaningful structure from naturalistic collaborative speech, enabling scalable, fine-grained analysis [39]. This line of work motivates our use of diarized, timestamped transcripts to quantify interaction dynamics and linguistic behavior across interlocutor conditions and sites.

Interactional fluency and turn-taking with AI partners. Within second language acquisition (SLA) research, fluency is often decomposed into speed, breakdown, and repair, and dialogic fluency work argues that monologic temporal measures can miss key interactional phenomena in conversation [38]. Overlap, interruptions, and between-turn gaps can materially affect fluency assessments, motivating measurement at the level of turns and turn transitions. Pauses, repetitions, and stalling may likewise function as interactional resources rather than purely as disfluency [25]. These perspectives build on classic accounts of turn-taking norms that describe strong pressure to minimize gaps and avoid overlap, with response latency as a central coordination variable [33, 37, 20].

When one interlocutor is a machine, these norms can shift systematically. Studies of voice interfaces document altered turn design and recurring breakdown and repair patterns [29, 21], and users may perform additional interactional work to maintain progressivity when systems are slow or insufficiently contingent [11]. With LLM-based agents, system contributions become more content-rich, yet interface timing and modality still constrain turn-taking and repair; users often scaffold the interaction through reformulations and related strategies [30]. Emerging work in language learning contexts suggests that L2 learners' turn-taking practices may shift when interacting with conversational AI [5]. Interaction structure is also increasingly treated as consequential for learning processes, with dialog acts in AI-supported peer tutoring linked to learning rates [3]. Together, these strands motivate our analysis of interactional fluency as a way to characterize how human–AI dialogue differs from human–human dialogue in the interactional conditions it affords.

Lexical and morphosyntactic uptake as alignment in dialogue. Psycholinguistic theories of alignment propose that interlocutors tend to converge on lexical and syntactic representations through priming mechanisms, supporting efficient coordination [28]. Related work on lexical entrainment and conceptual pacts shows how partners establish shared referring expressions as part of building common ground [4]. In SLA, alignment has been proposed as a mechanism that can support learning, since repeated exposure to and reuse of forms in interaction may increase accessibility and entrenchment [6, 23], and empirical work suggests that alignment patterns vary with interactional context [6, 18]. In the present one-shot design, we therefore treat uptake as short-term mining or alignment, not as evidence of durable

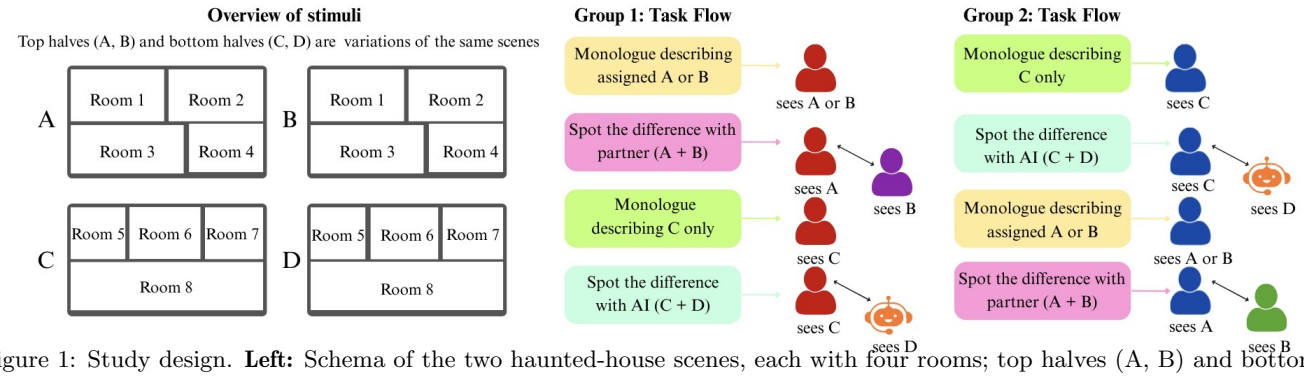


Figure 1: Study design. **Left:** Schema of the two haunted-house scenes, each with four rooms; top halves (A, B) and bottom halves (C, D) are slight variations of the same scenes. In the human condition each partner sees a different version (A or B), while in the AI condition the participant describes C and the AI is provided a description of D. **Right:** Task flow for the two counterbalanced groups. Each participant completes four tasks: two monologues and two dialogues (one with a human partner, one with the AI). Group 1 completes the human block first; Group 2 completes the AI block first.

learning; repeated alignment may provide affordances for learning over time, but our data measure immediate reuse opportunities rather than acquisition.

Measuring alignment at scale remains challenging. However, computational dialogue research offers measures ranging from global similarity to targeted accounts of repetition and construction reuse [32]. Work in human computer interaction further shows that people align their language with spoken dialogue systems, and that the strength and form of this alignment can vary with system framing and design [7, 35]. Educational natural language processing has also moved toward turn-linked measures that capture responsiveness beyond surface similarity [9]. Recent work similarly demonstrates the feasibility of automated, language-based measurement in educational settings, including LLM-based detection of learning-relevant constructs in verbal protocols [41]. At the same time, these approaches foreground familiar concerns about construct validity and fairness in model-based inference [36]. Building on these threads, we operationalize lexical and morphosyntactic uptake as a direction-sensitive form of alignment, measuring the extent to which learners reuse lexical items and morphosyntactic patterns introduced by the interlocutor in the immediately preceding context.

Learner affect and attitudes. CALL studies of chatbot and LLM-based speaking practice frequently include self-report measures—perceived usefulness, enjoyment, comfort, willingness to communicate—since these perceptions shape sustained engagement [17, 12]. Recent work shows that some conversational AI designs can increase willingness to communicate and reduce speaking anxiety [40], and that perceived enjoyment and satisfaction predict intentions to continue using AI tools for speaking practice [16]. However, these attitudinal outcomes are rarely connected to fine-grained interaction traces, leaving open which moment-by-moment behaviors drive learner perceptions—a gap we address by relating survey responses to transcript-derived interaction metrics.

3. DATA COLLECTION

3.1 Task and Stimuli

Participants completed a spot-the-difference task, a cooperative information-gap activity widely used in communicative language teaching [27]. A human artist created a "haunted house" scene containing eight rooms, each drawn in two matched versions with systematic differences in object placement (e.g., repositioned ghosts, added or removed objects). Participants communicated in the L2 language to identify these differences. The eight rooms were roughly matched for complexity and divided into two sets of four; one set was used for the human dialogue condition and one for the AI dialogue condition. Figure 1 provides an overview of the study design.

3.2 Conditions and Procedure

The study used a within-participant design with four speaking conditions along two dimensions: (i) task format (monologue vs. dialogue) and (ii) partner type (human peer vs. AI model).

Participants were assigned to a counterbalanced design, with one group completing the human–human condition first and the other completing the human–AI condition first. In the human-human condition, participants first completed a monologue describing either image A or image B, and then engaged in a dialogue about the same room set with a human partner who had previously described the complementary version. In the human-AI condition, all participants first described image C, followed by a dialogue with the AI system, which was provided a textual description of image D.

In human dialogues, participants were paired with a classmate and completed the task face-to-face or on Zoom; no constraints on turn-taking or strategy were imposed. AI dialogues were conducted through a web application with audio input. This modality asymmetry is representative of how each mode would typically be deployed, but co-presence in the human condition provides non-verbal coordination cues unavailable in the AI condition. Each session lasted approximately 90 minutes including a background survey, vocabulary assessment, the four speaking tasks, and a post-task survey. Data were collected between October and December 2025.

All procedures were approved by the institutional review boards at the participating sites. Before completing the background survey and speaking tasks, participants provided informed consent for audio recording and transcript-based analysis; recordings without valid consent were excluded.

3.3 AI Interlocutor System

We developed the AI system using OpenAI's `gpt-realtime` model[1], selected for its native speech-to-speech capability (eliminating separate ASR/TTS latency), support for a plurality of languages, and real-time turn-taking via server-side voice activity detection. Because direct image input produced excessive hallucination during piloting, we provided the AI with manually verified English-language text descriptions of each room, written by the research team and checked against the images for accuracy. This choice kept the AI grounded in the task materials but creates a stimulus-representation asymmetry: human partners viewed images directly, whereas the AI received textual descriptions of the rooms.

At session initialization, all room descriptions were included in the system prompt[2]; at each room transition, the current room's description was re-sent. The AI retained full conversation history within a session.

The system parameters were set as follows: voice `alloy`, temperature 0.8, max response tokens 150. Turn-taking used server-side VAD with an open microphone (prefix padding 300 ms, silence threshold 200 ms). After pilot testing, we found it useful to instruct the AI model to use informal register, take short turns (one to two sentences), allow time for learner responses, and avoid excessive praise. Despite these constraints, the AI consistently occupied a larger share of the conversational floor than human partners.

3.4 Participants

All participants were enrolled in second- or third-year university German courses at one of four sites: two Flemish universities in Belgium ($n = 24, 34$), a Midwestern US university ($n = 14$), and a Pacific Northwest US university ($n = 14$), totaling $N = 86$. After excluding non-consented, failed, and incomplete recordings, $N = 78$ participants remained, yielding 78 AI dialogues and 39 human-human conversations (one per dyad). Per-analysis sample sizes vary by minimum-turn thresholds and are reported with each analysis. Approximately 50% of participants listed Dutch as their sole L1, English was the second largest group (34.5%), and the remaining 15% comprised speakers of Romanian, Vietnamese, Spanish, Albanian, Hindi, and Ndebele, several of whom reported bilingual combinations with Dutch.

3.5 Transcription

We analyze diarized transcripts rather than raw audio, a deliberate choice favoring scalability: transcript-based measures avoid prosodic confounds across L1 backgrounds and recording environments, require no manual coding, and yield an easily replicable pipeline.

All sessions were transcribed using ElevenLabs `scribe_v2` with automatic speaker diarization (language: `deu`), producing word-level timestamps and speaker attribution. German-speaking authors reviewed all transcripts, verifying speaker labels, trimming non-task speech at recording boundaries, and confirming coherence with assigned room content. Word-level ASR errors were not systematically corrected. In three AI-dialogue transcripts, mid-stream speaker label swaps were detected via an automated disfluency-rate classifier (the AI produces zero filled pauses) and corrected manually.

4. METHODS

From the diarized, timestamped transcripts we extract two complementary sets of metrics. *Interactional fluency* (§4.1) captures how speakers coordinate in real time – how long they hold the floor, how quickly they respond, and how much they hesitate within turns. *Linguistic uptake* (§4.2) captures what speakers do with the language they hear – whether they adopt vocabulary or reproduce morphosyntactic patterns from the interlocutor's speech. The first captures *how* learners interact; the second captures *what language* they take away from the interaction.

4.1 Interactional Fluency

Dialogue involves continuous coordination over who speaks, for how long, and when transitions occur. These properties can differ systematically between human and AI partners, shaping the kind of practice an interaction affords. We capture this through seven features extracted from word-level timestamps, organized into two groups.

Conversation structure. Four features describe the overall shape of a dialogue. *Conversation length*: total session duration in seconds. *Turn count*: number of participant turns. *Mean turn duration*: average length of a participant turn in seconds. *Words per turn*: average words per participant turn. Together, these indicate whether the dialogue consists of many short exchanges or fewer, longer stretches of speech.

Turn-taking dynamics. Three features capture how speakers manage transitions and share the floor. *Response latency*: the gap between the partner's last word and the participant's first word; only gaps within $[0, 10]$ s are retained. *Floor ratio*: the proportion of session time the participant is speaking. *Within-turn pause ratio*: the proportion of a participant's turn spent in silence (≥ 0.3 s gaps between words), reflecting online planning difficulty. Figure 2 illustrates how within-turn pauses are distinguished from between-turn gaps.

Session-level metrics (conversation length, turn count) were compared using paired t-tests ($n = 78$). Per-turn metrics use sessions with ≥ 8 participant turns, yielding unbalanced samples; these were analyzed using linear mixed-effects models with condition as a fixed effect and participant as a random intercept.

[1]Specifically, `gpt-realtime-2025-08-28`

[2]An anonymized repository containing the full LLM prompt is available at `https://osf.io/9kxsu/overview?view_only=ff7b5f764f1647a788c8541bc0b3fc31`.

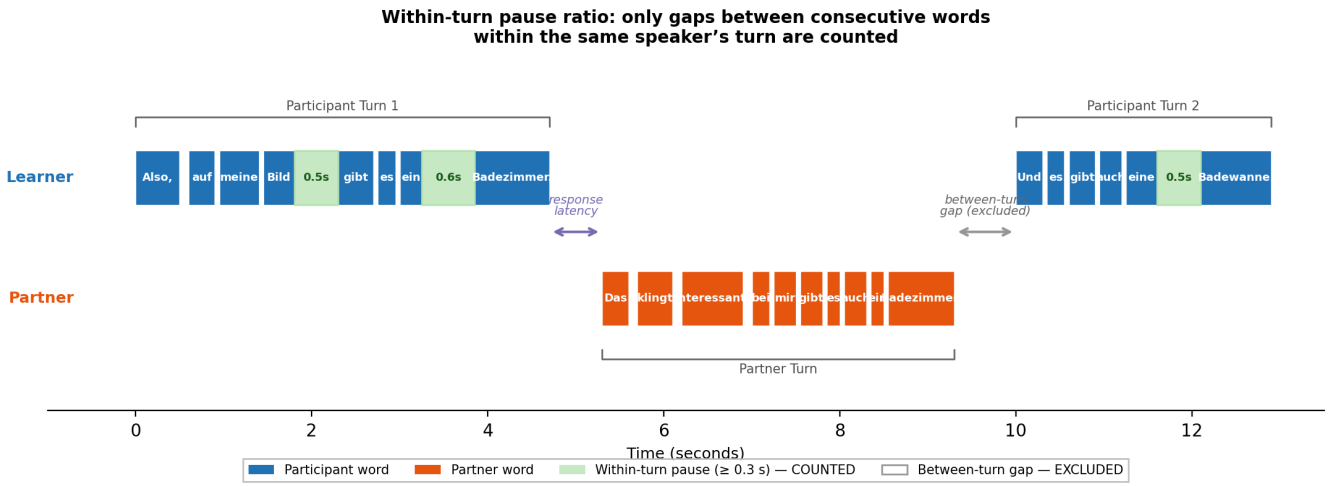


Figure 2: Illustration of temporal metrics extracted from word-level timestamps. Blue blocks are participant words; orange blocks are partner words. *Within-turn pause ratio* counts only gaps ≥ 0.3 s between consecutive words within the same speaker's turn (green). Gaps between speakers' turns (gray) are excluded from the pause ratio but are used to compute *response latency* (purple). Turn boundaries are reconstructed by grouping consecutive words from the same speaker.

4.2 Linguistic Uptake

Interactive alignment theory [28] predicts that interlocutors converge on shared linguistic representations during dialogue, from lexical choices to syntactic structures. We operationalize this as linguistic uptake: whether L2 learners adopt linguistic material from their interlocutor's speech. Figure 3 presents an illustration of how uptake is estimated. We measure uptake at two complementary levels:

Lexical uptake tracks whether learners adopt vocabulary items (lemmas) introduced by the partner, capturing breadth of vocabulary activation across the full session.
Syntactic uptake tracks whether learners reproduce morphosyntactic patterns (e.g., determiner–noun agreement, prepositional case government) from the partner's prior speech, capturing structural alignment within a dialogue.

Both measures impose a **first-use constraint**, such that each linguistic unit is scored only when it first appears in the learner's speech. This operationalization follows an input-mining paradigm in L2 research, where learners' later production is interpreted relative to prior independent production and intervening input [15]. In our design, the monologue preceding each dialogue provides a learner-specific baseline for lexical uptake, while syntactic uptake applies the same conservative first-use logic within dialogue by crediting templates only after prior partner use within a bounded recency window. We use uptake/mining to mean short-term reuse of available input, not acquisition; repeated task performance and the shared visual context may also contribute to some cases.

4.2.1 Lexical Uptake

The lexical uptake algorithm uses a **global timeline** approach, tracking each lemma across all four tasks in sequential order. The counterbalanced design creates two orderings:

- **Group 1** ($n = 46$): Monologue A/B → Human dialogue → Monologue C → AI dialogue
- **Group 2** ($n = 32$): Monologue C → AI dialogue → Monologue A/B → Human dialogue

The participant's first monologue establishes a **baseline vocabulary**. Words are tokenized, filtered (removing determiners, 49 pre-task vocabulary items, 153 English code-switching tokens, and words ≤ 2 characters), and lemmatized using spaCy's `de_core_news_sm` model. Words appearing on the Goethe-Institut A1 wordlist [26], approximately 714 high-frequency items, are also excluded as uninformative adoption candidates. A lemma counts as **adopted** if: (1) the interlocutor produced it before the participant, (2) it is not in the participant's baseline, and (3) it has not already been attributed to another source (**first-source-wins** rule). We use "adopted" rather than "learned" because we are measuring activation through interactive exposure, not acquisition.

Adoption is measured cumulatively without a recency window, consistent with accounts of lexical entrainment as a *historical* phenomenon in which interlocutors maintain partner-specific conceptual precedents beyond immediate adjacency [4], and with L2 alignment work that omits distance constraints under an interaction-as-learning framing [24]. Because the AI interlocutor produces substantially more speech than human partners, we report both raw adoption counts and adoption rates controlled for verbosity by dividing by the number of unique novel lemmas each interlocutor introduced beyond the participant's baseline (**pool size**). To test whether interaction type predicts adoption beyond input volume, we fit a mixed-effects regression with condition, pool size, and learner word count as fixed effects and participant as a random intercept.

4.2.2 Syntactic Uptake

Syntactic uptake measures whether learners reproduce morphosyntactic patterns from the interlocutor's speech **within**

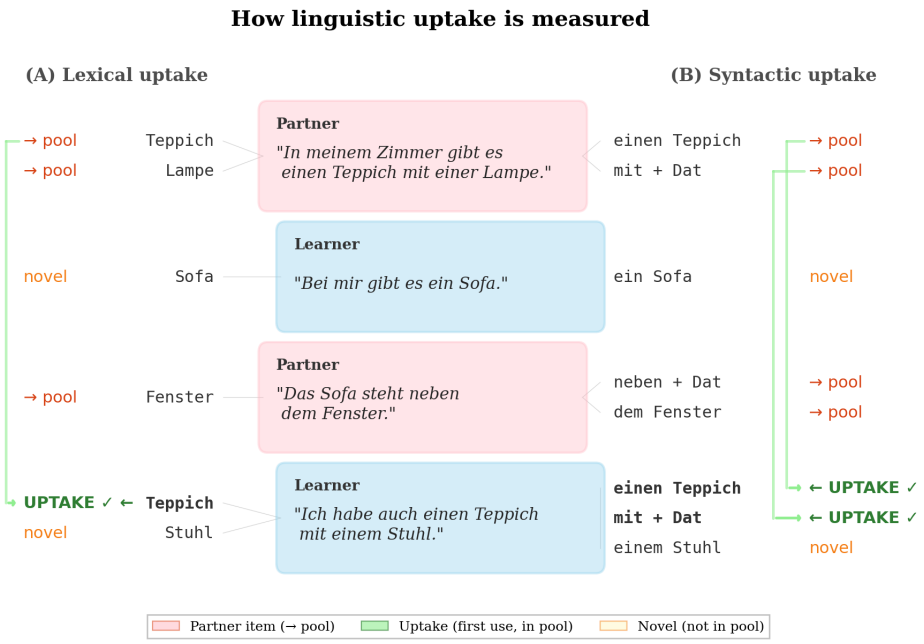


Figure 3: Illustration of linguistic uptake measurement at two levels, applied to the same example dialogue ($N_{\text{AI}} = 78$, $N_{\text{Human}} = 78$ in the full dataset). **(A)** Lexical uptake: content words used by the partner accumulate in a pool; when the learner first produces a matching word, it is counted as uptake. **(B)** Syntactic uptake: morphosyntactic templates—determiner–noun agreement (e.g., *einen Teppich*, masculine accusative) and preposition case government (e.g., *mit* + dative)—are tracked analogously. Green arrows trace each uptake event to its source in the partner's speech. Only the learner's first use of each item counts (first-use constraint); repeated uses are not double-counted. For syntactic templates, grammatical correctness is additionally validated against a noun-gender lexicon (e.g., *Teppich* → masculine).

Table 1: Target morphosyntactic patterns for syntactic uptake.

Pattern	Example	What it captures
DET-NOUN	*der Tisch* (Masc/Nom/Sg)	Det–noun gender/case/number agreement
ADP-CASE	*mit dem Hund* (mit → Dat)	Preposition case government
ADJ-NOUN	*großes Gemälde* (Neut/Nom/Sg)	Attributive adjective agreement
CONTRACTION	*im* (= *in* + *dem*, Dat)	Preposition–article fusion

a single dialogue. We extract four pattern types that constitute the core of German nominal morphology (see Table 1): determiner–noun agreement, prepositional case government, attributive adjective agreement, and preposition–article contractions. These are pervasive in the data ($M = 86$ learner-produced template instances per AI dialogue, 113 per human dialogue) and target well-documented areas of L2 German difficulty.

Extraction pipeline. Each turn is lightly preprocessed (removing filled pauses such as *äh* and *ähm*) and parsed with the Stanza German UD pipeline [31]. Each extracted template has an **anchored** representation that includes the noun lemma, surface form, and morphological features (e.g., DET_NOUN|*tisch*|*dem*|Masc|Dat|Sg), tying uptake to specific lexical items rather than abstract morphological categories.

Correctness checking. For DET-NOUN and ADJ-NOUN templates, we verify gender agreement against a lexicon of 91,270 German nouns. For ADP-CASE, correctness is checked for strict-case prepositions (e.g., *mit* → dative); two-way prepositions (e.g., *auf*, *in*), which can take dative or accusative case, are included in uptake matching but excluded from correctness analysis, as case selection depends on semantic context that cannot be determined from syntax alone. Contractions are inherently correct.

Uptake scoring. On the participant's first production of a template, we check whether it appeared in the source's prior speech. Unlike lexical adoption, syntactic uptake uses a recency window, reflecting the lag-sensitivity of structural priming documented in both L1 [14] and L2 task-based interaction [8], where prime–target contingency is typically coded within a bounded span [24]. Because AI turns are longer and syntactically denser than human turns, a fixed turn-count window captures unequal amounts of input across conditions. We therefore compare three window operationalizations: **turn-based** (last 3 source turns), **time-based** (last 15 seconds of source speech), and **template-count** (last 15 source

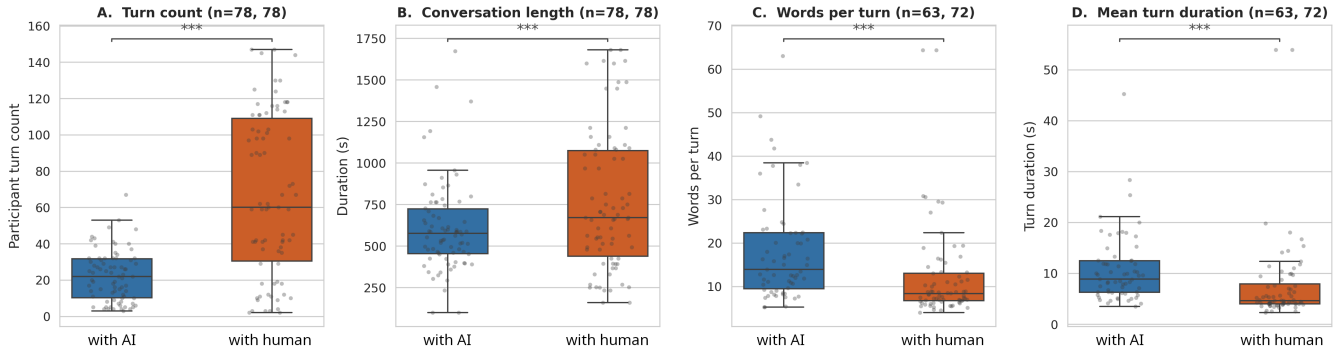


Figure 4: Conversation structure: AI vs. Human partner. (A) Participant turn count and (B) conversation length use all sessions ($n = 78$ per condition). (C) Words per turn and (D) mean turn duration use sessions with ≥ 8 participant turns ($n_{\text{AI}} = 63$, $n_{\text{Human}} = 72$). Box plots show median and interquartile range; individual sessions are overlaid as points. $^{***}p < .001$.

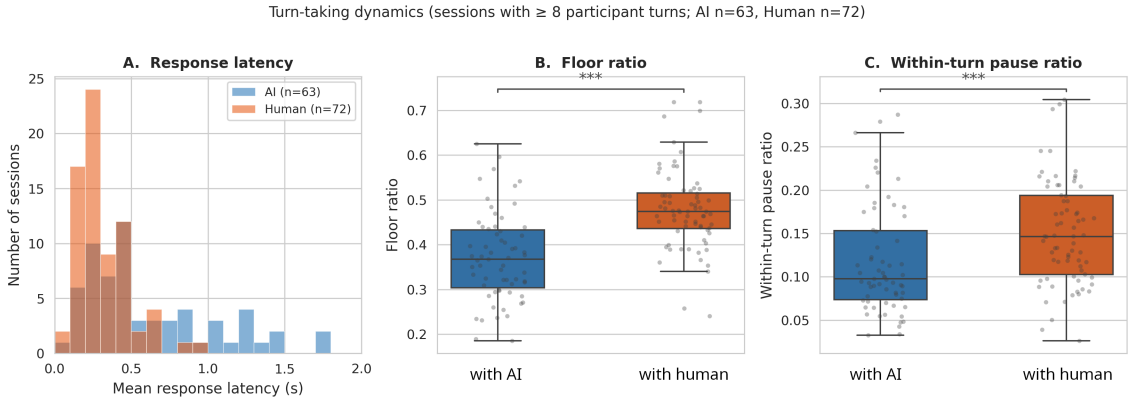


Figure 5: Turn-taking dynamics: AI vs. Human partner (sessions with ≥ 8 participant turns). (A) Overlapping histograms of mean response latency show clear separation between conditions. (B) Floor ratio: learners hold the floor for a smaller share of session time with the AI. The dashed line marks equal floor time (0.5). (C) Within-turn pause ratio: learners pause *less* within turns when speaking with the AI. $^{***}p < .001$.

templates, explicitly equalizing input volume). All results are tested via mixed-effects regression with condition, source pool size, and learner unique template count as fixed effects and participant as a random intercept.

The algorithm also detects **error–correction sequences**: the participant produces an incorrect form, the source subsequently models the correct form, and the participant later produces the corrected version.

5. RESULTS

We present results in three parts. Interactional dynamics (§4.1) characterize how dialogue structure and turn-taking differ between AI and human partners. Lexical and syntactic uptake (§4.2) examine whether and how learners adopt language from each interlocutor type. Survey results (§5.3) connect interaction behavior to learner attitudes and satisfaction.

5.1 Interactional Dynamics

We organize the findings into three domains: conversation structure, turn-taking dynamics, as well as consistency of results across four sites.

5.1.1 *Conversation Structure*

Human-human dialogues were significantly longer ($M = 764$ s vs. 608 s, $d = -0.44$, $p < .001$) and contained nearly three times as many turns ($M = 66.5$ vs. 22.2, $d = -1.35$, $p < .001$). Human partners engaged in rapid back-and-forth exchange, while AI dialogues were characterized by fewer, longer turns. With the AI, participants produced significantly longer turns ($M = 10.8$ s vs. 7.7 s, $d = 0.39$, $p < .001$) containing more words per turn ($M = 18.0$ vs. 12.1, $d = 0.51$, $p < .001$). Learners pack more content into each turn with the AI, but the threefold reduction in turn count means that total learner output is substantially lower than in human dialogue.

Figure 4 illustrates this trade-off: human dialogues produce many more turns (A), but learners produce more words per turn with the AI (C).

5.1.2 *Turn-Taking Dynamics*

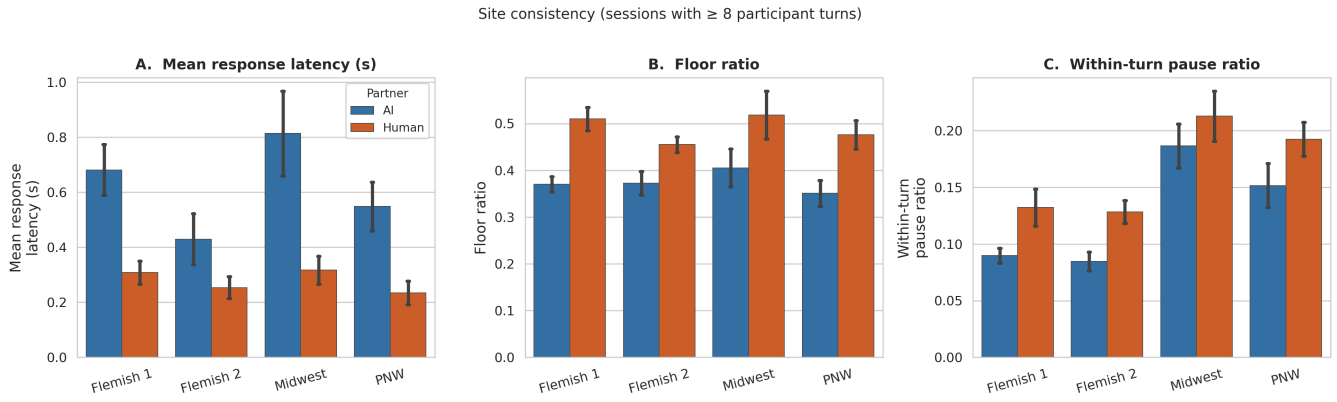


Figure 6: Site consistency: the three turn-taking metrics broken down by site and partner type (sessions with ≥ 8 turns). Error bars show ± 1 SE. At every site, response latency is higher with the AI, floor ratio is lower, and within-turn pause ratio is lower. No site reverses the direction of any effect.

Three interrelated measures reveal qualitatively different turn-taking patterns across conditions. Figure 5 visualizes the three turn-taking metrics, showing clear separation between conditions for response latency (A) and consistent differences in floor ratio (B) and within-turn pause ratio (C).

Participants took roughly twice as long to begin speaking after the AI ($M = 0.61$ s) as after a human partner ($M = 0.32$ s; $d = 0.90$, $p < .001$), indicating substantially longer response latencies in AI dialogue. Most human-partner latencies cluster below 0.3 s, while AI latencies spread between 0.2–1.0 s, likely reflecting the AI's longer, more information-dense turns.

Learners held the floor for a smaller proportion of session time when interacting with the AI ($M = 0.37$) than with a human partner ($M = 0.48$; $d = -1.14$, $p < .001$), representing one of the largest effects observed in the dataset. The AI's longer turns account for much of this difference, as they occupy a greater share of the conversational time.

Learners paused *less* within their own turns when speaking with the AI ($M = 0.12$ vs. 0.15; $d = -0.53$, $p < .001$), indicating greater within-turn fluency. When learners do hold the floor with the AI, they tend to speak more fluently within each turn.

Taken together, these measures show a consistent shift in interactional organization: AI dialogue features slower turn transitions and reduced learner floor time, but more fluent production when learners hold the floor.

5.1.3 *Site Consistency*

Figure 6 shows the contrasts were consistent in direction across all four sites. Response latency was higher with the AI at every site; floor ratio was lower; within-turn pause ratio was lower. The magnitudes varied—Midwest showed the largest response-latency difference, while Flemish 2 showed the smallest floor-ratio gap—but no site reversed the direction of any effect.

5.2 Linguistic Uptake

5.2.1 *Lexical Uptake*

The AI produced over twice as many novel lemmas as human partners (pool size: $M = 79.6$ vs. 36.4), yielding a higher descriptive adoption rate for human partners (16.2% vs. 9.3% of pool adopted). However, a mixed-effects regression controlling for pool size and learner word count found no significant condition effect ($\beta = -1.77$, $SE = 0.95$, $z = -1.87$, $p = .062$). Pool size ($\beta = 0.10$, $p < .001$) and learner word count ($\beta = 0.005$, $p < .001$) were the significant predictors: after accounting for input volume and learner output, condition does not reliably predict adoption.

To disentangle partner type from task order, we compared adoption at the same sequential position. Table 2 presents the results of this analysis. In position 2—where the depleted vocabulary pool provides a more sensitive test—AI participants adopted significantly more words ($M = 4.6$ vs. 2.6; $d = 0.72$, $p = .003$). In position 1, the direction is consistent but nonsignificant ($d = 0.42$, $p = .079$).

5.2.2 *Syntactic Uptake*

A total of 156 participant-level dialogue records were analyzed: 78 AI dialogue records, and 78 human-partner records derived from 39 human-human conversations. The results are summarized in Table 3. Learners' cumulative uptake rate was higher with the AI ($M = .16$, $SD = .11$) than with human partners ($M = .08$, $SD = .06$; $d = 0.93$), but as with lexical adoption, the AI also provided a substantially larger source pool ($M = 155$ vs. 90 unique templates; $d = 1.35$). After controlling for pool size and learner unique template count, cumulative uptake showed a positive but nonsignificant trend ($\beta = 1.55$, $SE = 0.92$, $z = 1.69$, $p = .090$). Recency-windowed uptake told a different story: the AI advantage was significant across all three operationalizations—turn-based ($\beta = 3.18$, $p < .001$), time-based ($\beta = 3.04$, $p < .001$), and template-count ($\beta = 2.84$, $p = .001$)—including the window that explicitly equalizes input volume. This confirms that the AI's short-term priming advantage reflects genuine structural priming, not merely a larger input pool.

Correctness rates were comparable: AI $M = .82$ ($SD = .13$)

Table 2: Lexical adoption by sequential position and partner type. Group 1 ($n = 46$) had the human partner first; Group 2 ($n = 32$) had the AI first. Position-matched comparisons use between-subjects Welch's t-tests.

Position	Partner type	n	M	SD	t	p	d
First dialogue	Human (Group 1)	46	8.6	5.5	1.79	.079	0.42
	AI (Group 2)	32	11.2	6.8			
Second dialogue	Human (Group 2)	32	2.6	2.8	3.12	.003**	0.72
	AI (Group 1)	46	4.6	2.8			

Table 3: Syntactic uptake by condition. Descriptive statistics (Welch's t-tests) and mixed-effects regression results (β = condition coefficient, AI = 1; models control for source pool size and learner unique templates with participant random intercept).

Measure	**AI** M (SD)	**Human** M (SD)	t	d	β	SE	z	p
Cumulative uptake (rate)	.16 (.11)	.08 (.06)	5.84	0.93	—	—	—	—
Cumulative uptake (count)	7.3 (5.6)	5.4 (4.6)	2.28	0.36	1.55	0.92	1.69	.090
Recency: 3 turns (count)	6.3 (5.2)	2.9 (2.9)	4.98	0.80	3.18	0.86	3.69	<.001***
Recency: 15 s (count)	5.7 (5.0)	2.7 (2.7)	4.65	0.74	3.04	0.84	3.61	<.001***
Recency: 15 templates (count)	5.2 (5.3)	3.2 (2.8)	3.00	0.48	2.84	0.87	3.27	.001**

Table 4: Key baseline attitudes: AI vs. Human peers (N=55). All items rated on a 5-point Likert scale (1 = Strongly Disagree, 5 = Strongly Agree).

Item	M	SD
Practicing speaking is valuable	4.56	0.69
AI effectiveness	3.22	0.76
AI enjoyment	2.73	0.89
Relaxation: AI	3.25	1.02
Relaxation: Proficient speakers	2.82	1.19
Access to proficient speakers	2.67	1.11

vs. Human $M = .80$ ($SD = .13$); $p = .316$, $d = 0.16$. Higher uptake with the AI does not come at the cost of morphological accuracy. Correction sequences were numerically more frequent with human partners (total: 38 vs. 27), but per-participant rates did not differ significantly ($M = 0.49$ vs. 0.35; $p = .215$).

5.3 Survey Analysis

Of the 78 participants, 55 completed matched pre- and post-surveys; 52 had complete data for the directional sentiment composite used in the paired t-test below. These data allow us to assess how learner attitudes toward AI shift after direct experience and whether behavioral features of the interaction predict satisfaction.

Baseline attitudes. Participants entered with predominantly skeptical attitudes toward AI: 60% described themselves as skeptical or unsure, and only 14.5% as enthusiastic adopters. AI-specific Likert items scored near or below neutral (perceived effectiveness $M = 3.22$, enjoyment $M = 2.73$), while general speaking practice was highly valued ($M = 4.56$). Notably, relaxation with AI ($M = 3.25$) already exceeded relaxation with proficient speakers ($M = 2.82$), and the lowest-scoring item, having enough opportunities to speak with proficient speakers ($M = 2.67$), points to a perceived access gap (see Table 4).

Post-experience shift. Attitudes improved significantly after the study: a paired t-test on directional sentiment scores showed a mean increase of +0.48 ($t(51) = -4.58$, $p < .0001$, $d = 0.64$), with 75% of participants becoming more positive. Post-study, 67.3% liked the AI interaction, 81.8% agreed AI can be useful for L2 speaking practice, and 34.5% reported learning new vocabulary.

Behavioral predictors of satisfaction. To connect attitudinal outcomes to interaction behavior, we fit OLS regressions predicting post-satisfaction and attitude change from behavioral features of the AI dialogue. Table 5 presents the results of survey regressions, fit using ordinary least squares (`statsmodels`; [34]) with site as a categorical covariate.

Pre-study attitude predicted post-satisfaction ($\beta = 0.36$, $p = .006$), but the strongest behavioral predictor was within-turn pause ratio ($\beta = -4.78$, $p = .008$): learners who paused more during AI turns, reflecting greater online planning difficulty, reported lower satisfaction and smaller attitude gains ($\beta = -5.91$, $p = .006$). Floor ratio was not significant, suggesting that *how fluently* learners spoke mattered more than *how much* they spoke. Neither lexical adoption rate nor syntactic uptake rate predicted satisfaction ($p > .45$), indicating that the subjective experience of AI interaction is shaped by production fluency rather than by receptive alignment.

6. DISCUSSION

The fluency, uptake, and survey results converge on one conclusion: AI and human dialogue create distinct but complementary conditions for L2 practice.

Monologue-in-dialogue. AI dialogue structurally approximates supported monologue. Learners produce sustained output with fewer, longer turns, slower responses, and less floor competition, while producing more fluent speech within each turn. The AI's patient, non-competitive style shifts learners from reactive turn management to more deliberate, content-focused production. Human dialogue, by contrast, is rapid and interactive: frequent turns, fast responses, balanced floor time, and more within-turn hesitation. The interactional work of peer conversation (collaborative reference-building, repair negotiation, and affiliative floor manage-

Table 5: OLS regressions predicting post-interaction satisfaction and attitude change from behavioral features of the AI dialogue ($n = 55$). Site is dummy-coded with Flemish 1 as the reference category. $^{*}p < .05$, $^{**}p < .01$.

	Satisfaction			Attitude change			Satisfaction (uptake)		
Predictor	β	SE	p	β	SE	p	β	SE	p
Intercept	4.01	0.54	<.001**	2.10	0.46	<.001**	3.34	0.51	<.001**
Pre-AI attitude	0.36	0.13	.006**				0.31	0.14	.030*
Within-turn pause ratio	−4.78	1.71	.008**	−5.91	2.07	.006**			
Floor ratio	−0.95	0.92	.307	−1.16	1.12	.306			
Lexical adoption rate							1.22	1.62	.454
Syntactic uptake rate							−0.39	1.08	.717
Site: Flemish 2	−0.53	0.26	.046*	−0.22	0.31	.482	−0.58	0.27	.038*
Site: Midwest	−0.44	0.35	.209	−0.18	0.42	.667	−0.98	0.32	.004**
Site: PNW	−0.34	0.26	.196	−0.04	0.31	.907	−0.60	0.28	.037*
R^2 (Adj. R^2)	.43 (.36)			.24 (.16)			.33 (.25)		
F	6.03**			3.10*			3.98**		

ment) is largely absent from AI dialogue. This is not a failure of the AI to "be human" but a description of what the interaction actually is.

What the AI scaffolds. The AI's verbose, syntactically regular speech, in this specific model-prompt-task configuration, creates favorable conditions for short-term structural alignment. Overall, AI dialogue exposes learners to more input and elicits more uptake in absolute terms. The deeper finding is that the AI's syntactic priming advantage survives all controls for input volume, pointing to a qualitative difference: complete, well-formed constructions without the fragments and restarts of peer speech yield cleaner targets for short-term reuse. At the lexical level, the AI provides greater exposure but not more efficient uptake per unit of input ($p = .062$ after controls). The AI's distinctive pedagogical contribution is thus specifically structural. This is particularly notable given that meaning-focused tasks like spot-the-difference typically direct learner attention toward vocabulary rather than grammar [13]; the syntactic priming observed here operates implicitly, suggesting a potential affordance of AI interaction worth investigating in future work.

What human partners scaffold. Human dialogue exercises a different set of capacities. The rapid turn exchange, floor competition, and negotiation sequences characteristic of peer interaction create conditions for developing interactional competence, skills not exercised in AI dialogue. The information-gap task produces its richest interactional outcomes with a partner who *cannot* resolve gaps efficiently. Whether human dialogue also promotes more durable lexical learning through interactional grounding remains a question our data suggest but cannot confirm.

Learner experience. Attitudes toward AI improved significantly after the task, but satisfaction was predicted by within-turn production fluency, not by uptake. Learners who paused more within their turns reported lower satisfaction, suggesting that the subjective quality of AI interaction hinges on whether the learner can produce fluent speech in the space the AI provides, not on how much language they absorb from the AI's input.

7. CONCLUSIONS

AI and human dialogue scaffold L2 production through complementary mechanisms: structural priming through regular input in one case, interactional competence through coordinative demand in the other. The design challenge is not to make AI dialogue more like human dialogue, but to exploit what each mode does well.

This points to concrete principles for integrating voice-based AI into language curricula: For curricula, this suggests four design principles. First, AI tasks are well suited to focused morphosyntactic practice, where regular input can support priming of constructions such as dative prepositions and gender agreement. Second, AI practice should complement rather than replace peer speaking, since human dialogue exercises turn competition, clarification requests, repair, and collaborative floor management. Third, AI verbosity should be calibrated: shorter, less complete AI turns may reduce floor dominance and create more space for learner-initiated language. Finally, uptake metrics can support adaptive feedback on vocabulary and constructions adopted, missed, or produced incorrectly from ASR transcripts.

Limitations. Several limitations apply. Uptake measures cannot fully separate priming from parallel activation driven by shared task context; causal claims require manipulating interlocutor language. NLP pipelines trained on Standard German may degrade on disfluent L2 speech; word-level ASR errors were not corrected, although German-speaking authors verified speaker labels, task boundaries, and broad coherence. First-use counting captures breadth rather than depth of uptake, and the lexical first-source-wins rule creates a task-order confound ($d = 1.71$), addressed with lower-powered position-matched comparisons. The study cannot determine whether short-term priming becomes durable learning; delayed post-tests are needed. Transcript-based analysis also excludes prosody, pronunciation, stress, intonation, and segmental accuracy. The comparison involved stimulus and modality asymmetries: human partners viewed images directly and interacted face-to-face or by Zoom, whereas the AI received English text descriptions and interacted through audio only. Some temporal effects may therefore partly reflect interface latency, VAD behavior, and visual coordination cues rather than interlocutor type alone. Finally, the AI condition reflects one system configuration (model, voice, prompt, VAD setting, and task interface) with university German learners, mostly Dutch- or English-dominant, performing one task type; future work should replicate across AI systems, L1 backgrounds, proficiency levels, target languages, and tasks.

Acknowledgments

We would like to thank all of the professors, teachers and students who worked with great dedication to plan for, collect and process all of the data for these experiments.

References

[1] S. Bibauw, T. François, and P. Desmet. Discussing with a computer to practice a foreign language: Research synthesis and conceptual framework of dialogue-based call. *Computer Assisted Language Learning*, 32(8):827–877, 2019.

[2] S. Bibauw, W. Van den Noortgate, T. François, and P. Desmet. Dialogue systems for language learning: A meta-analysis. *Language Learning & Technology*, 26(1):1–24, 2022.

[3] C. Borchers, K. Yang, J. Lin, N. Rummel, K. R. Koedinger, and V. Aleven. Combining dialog acts and skill modeling: What chat interactions enhance learning rates during ai-supported peer tutoring? In B. Paassen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 117–130, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[4] S. E. Brennan and H. H. Clark. Conceptual pacts and lexical choice in conversation. *Journal of experimental psychology: Learning, memory, and cognition*, 22(6):1482, 1996.

[5] S. Choi and S. Oh. Developing L2 turn-taking with ChatGPT: A longitudinal conversation analytic study. *System*, 138:103959, 2026.

[6] A. Costa, M. J. Pickering, and A. Sorace. Alignment in second language dialogue. *Language and Cognitive Processes*, 23(4):528–556, 2008.

[7] B. R. Cowan and H. Branigan. Does voice anthropomorphism affect lexical alignment in speech-based human-computer dialogue? In *Proc. Interspeech 2015*, pages 155–159, 2015.

[8] P. Dao, P. Trofimovich, and S. Kennedy. Structural alignment in l2 task-based interaction. *ITL - International Journal of Applied Linguistics*, 169(2):293–320, 2018.

[9] D. Demszky, J. Liu, Z. Mancenido, J. Cohen, H. Hill, D. Jurafsky, and T. B. Hashimoto. Measuring conversational uptake: A case study on student-teacher interactions. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 1638–1653, 2021.

[10] J. Du and B. K. Daniel. Transforming language education: A systematic review of ai-powered chatbots for english as a foreign language speaking practice. *Computers and Education: Artificial Intelligence*, 6:100230, 2024.

[11] J. E. Fischer, S. Reeves, M. Porcheron, and S. Sharples. Progressivity for voice interface design. In *Proceedings of the 1st International Conference on Conversational User Interfaces*, CUI '19. Association for Computing Machinery, 2019.

[12] L. Fryer and R. Carpenter. Bots as language learning tools. 2006.

[13] M. d. P. García Mayo and A. Azkarai. Efl task-based interaction: Does task modality impact on language-related episodes? In *Peer interaction and second language learning: Pedagogical potential and research agenda*, pages 241–266. John Benjamins Publishing Company, 2016.

[14] R. J. Hartsuiker, S. Bernolet, S. Schoonbaert, S. Speybroeck, and D. Vanderelst. Syntactic priming persists while the lexical boost decays: Evidence from written and spoken dialogue. *Journal of Memory and Language*, 58(2):214–238, 2008.

[15] H. Hoang and F. Boers. Re-telling a story in a second language: How well do adult learners mine an input text for multiword expressions? *Studies in Second Language Learning and Teaching*, 6(3):513–535, 2016.

[16] F. Huang and B. Zou. English speaking with artificial intelligence (ai): The roles of enjoyment, willingness to communicate with ai, and innovativeness. *Comput. Hum. Behav.*, 159(C), Oct. 2024.

[17] W. Huang, K. F. Hew, and L. K. Fryer. Chatbots for language learning—are they really useful? a systematic review of chatbot-supported language learning. *Journal of Computer Assisted Learning*, 38(1):237–257, 2022.

[18] Y. Kim and M. Michel. Linguistic alignment in second language acquisition: A methodological review. *System*, 115:103007, 2023.

[19] W. Y. W. Lai and J. S. Lee. A systematic review of conversational ai tools in elt: Publication trends, tools, research methods, learning outcomes, and antecedents. *Computers and Education: Artificial Intelligence*, 7:100291, 2024.

[20] S. C. Levinson and F. Torreira. Timing in turn-taking and its implications for processing models of language. *Frontiers in Psychology*, 6:731, 2015.

[21] E. Luger and A. Sellen. Like having a really bad PA: The gulf between user expectation and experience of conversational agents. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 5286–5297, San Jose, CA, USA, 2016. Association for Computing Machinery.

[22] E. Mayor, L. M. Bietti, and A. Bangerter. Can large language models simulate spoken human conversations? *Cognitive Science*, 49(9):e70106, 2025.

[23] K. McDonough. Interaction and syntactic priming: English l2 speakers' production of dative constructions. *Studies in Second Language Acquisition*, 28(2):179–207, 2006.

[24] M. Michel, C. Appel, and S. Cipitria. Lexical and syntactic alignment during english-spanish teletandem meetings. *System*, 111:102930, 2022.

[25] P. Peltonen. Temporal fluency and problem-solving in interaction: An exploratory study of fluency resources in L2 dialogue. *System*, 70:1–13, 2017.

[26] M. Perlmann-Balme. Goethe-zertifikat A1: Start Deutsch 1 — Wortliste. Technical report, Goethe-Institut, 2012.

[27] T. Pica, H.-S. Kang, and S. Sauro. Information gap tasks: Their multiple roles and contributions to interaction research methodology. *Studies in Second Language Acquisition*, 28(2):301–338, 2006.

[28] M. J. Pickering and S. Garrod. The interactive-alignment model: Developments and refinements. *Behavioral and Brain Sciences*, 27(2):212–225, 2004.

[29] M. Porcheron, J. E. Fischer, S. Reeves, and S. Sharples. Voice interfaces in everyday life. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 1–12, Montreal, QC, Canada, 2018. Association for Computing Machinery.

[30] O. Pütz and E. Esposito. Performance without understanding: How chatgpt relies on humans to repair conversational trouble. *Discourse & Communication*, 18(6):859–868, 2024.

[31] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning. Stanza: A python natural language processing toolkit for many human languages. In A. Celikyilmaz and T.-H. Wen, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online, July 2020. Association for Computational Linguistics.

[32] D. Reitter and J. D. Moore. Alignment and task success in spoken dialogue. *Journal of Memory and Language*, 76:29–46, 2014.

[33] H. Sacks, E. A. Schegloff, and G. Jefferson. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4):696–735, 1974.

[34] S. Seabold and J. Perktold. Statsmodels: Econometric and statistical modeling with python. *SciPy 2010*, 2010.

[35] L. Spillner and N. Wenig. Talk to me on my level – linguistic alignment for chatbots. In *Proceedings of the 23rd International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '21. Association for Computing Machinery, 2021.

[36] F. Stinar, H. Lee, C. Belitz, N. Nasiar, S. Fancsali, S. Ritter, H. Almoubayyed, R. Baker, J. Ocumpaugh, and N. Bosch. Fairness of bayesian knowledge tracing for math learners of different reading ability. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 170–181, Palermo, Italy, July 2025. International Educational Data Mining Society.

[37] T. Stivers, N. J. Enfield, P. Brown, C. Englert, M. Hayashi, T. Heinemann, G. Hoymann, F. Rossano, J. P. de Ruiter, K.-E. Yoon, and S. C. Levinson. Universals and cultural variation in turn-taking in conversation. *Proceedings of the National Academy of Sciences*, 106(26):10587–10592, 2009.

[38] P. Tavakoli. Fluency in monologic and dialogic task performance: Challenges in defining and measuring L2 fluency. *International Review of Applied Linguistics in Language Teaching*, 54(2):133–150, 2016.

[39] V. Venkatesha, A. Nath, I. Khebour, A. Chelle, M. Bradford, J. Tu, J. Pustejovsky, N. Blanchard, and N. Krishnaswamy. Propositional extraction from natural speech in small group collaborative tasks. In B. Paassen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 169–180, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[40] Y. Wang, Y. Wang, K. Crace, and Y. Zhang. Understanding attitudes and trust of generative ai chatbots for social anxiety support. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, page 1–21. ACM, Apr. 2025.

[41] J. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using large language models to detect self-regulated learning in think-aloud protocols. In B. Paassen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 157–168, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

What Do LIWC Features Capture in Multilingual Classroom Talk? A Systemic Functional Linguistic Analysis

Kaycie Barron
University of Wisconsin–Madison
kdbarron@wisc.edu

Ananya Ganesh
University of Wisconsin–Madison
aganesh27@wisc.edu

Shamya Karumbaiah
University of Wisconsin–Madison
shamya.karumbaiah@wisc.edu

Mariana Castro
University of Wisconsin–Madison
mcastro@wisc.edu

ABSTRACT

Classroom discourse among multilingual learners (MLLs) routinely integrates cognitive, interpersonal, and organizational work, yet scalable discourse analytics in educational data mining (EDM) rely on linguistic features whose relationship to these meanings is often unclear. This study examines how psycholinguistic features derived from Linguistic Inquiry and Word Count (LIWC) correspond to expert interpretations of meaning in multilingual small-group science discourse. Grounded in Systemic Functional Linguistics (SFL), we analyze associations between LIWC categories and human annotations of the ideational and interpersonal metafunctions across ten classroom transcripts. Using three complementary analyses, we evaluate empirical alignment patterns and theory-driven predictions. Results show that LIWC features align most consistently with interpersonal meaning, capturing stance and social positioning, whereas ideational meaning is reflected more selectively in measures of elaboration and in references to participants and actions. These findings clarify what dimensions of classroom meaning are accessible as LIWC-based representations and highlight important limitations of psycholinguistic features for modeling learning-relevant discourse in multilingual settings.

Keywords

Educational Data Mining, Collaborative Learning, Multilingual Discourse, Systemic Functional Linguistics, Interpretability

1. INTRODUCTION

Classroom discourse plays a central role in collaborative learning by shaping how multilingual learners (MLLs) construct ideas, negotiate understanding, and coordinate participation during instruction. In multilingual classrooms, these processes are often realized through translanguaging practices in which learners flexibly draw on multiple linguistic resources to foster joint sense-making and participation [2, 16]. Prior research shows that student talk in small-group STEM activities routinely serves multiple communicative functions simultaneously, such as combining problem solving, interpersonal positioning, and discourse organization within a single spoken utterance [20, 5, 16].

Educational data mining (EDM) has increasingly turned to discourse analysis to study classroom interaction and learning processes at scale [25, 6]. This approach offers powerful tools for characterizing multilingual meaning-making and collaborative processes across large datasets. However, many commonly used linguistic representations rely on surface-level features whose relationship to learning-relevant discourse functions remains underspecified, particularly in multilingual settings where it does not conform to monolingual or standardized norms [3, 12]. For instance, a student might say, "No sé... [I don't know] but I think it's because it evaporates faster," where the phrase may register pronoun use, negation, or cognitive markers in surface feature models, while simultaneously contributing to collaborative hypothesis building that may not be fully represented.

Linguistic Inquiry and Word Count (LIWC) provides one widely adopted and interpretable approach to discourse analysis by quantifying psycholinguistic dimensions such as cognitive processing, affective tone, and social orientation [27, 22]. LIWC features have been used extensively in educational research to examine student engagement, performance, and experience, including in STEM learning contexts [23, 17]. Despite its widespread use, LIWC is often applied without an explicit theory of discourse and assumes monolingualism, leaving open questions about what kinds of communicative work these features actually reflect in linguistically diverse classroom interactions. In addition, MLLs routinely draw on multiple languages for both conceptual sense-making and social interaction, even when English is the official medium of instruction [8, 21]. Approaches that rely on language-specific dictionaries risk under-representing how meaning is distributed across languages, with implications for how discourse is interpreted and used in downstream models of learning.

Kaycie Barron, Ananya Ganesh, Shamya Karumbaiah, and Mariana Castro. What Do LIWC Features Capture in Multilingual Classroom Talk? A Systemic Functional Linguistic Analysis. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 388–399. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040154

This study addresses this gap by examining how LIWC features align with human interpretations of discourse in K-12 multilingual science classrooms. Drawing on Systemic Functional Linguistics (SFL), which theorizes how meaning is realized through language across the ideational, interpersonal, and textual metafunctions [10, 19], we analyze the correspondence between select LIWC variables and expert-annotated metafunctions in small-group science discussions. Rather than treating linguistic features as predictors of downstream outcomes, we adopt a theory-informed analytic approach to estimate how LIWC's psycholinguistic variables represent different dimensions of meaning-making.

1.1 Multilingual Classroom Discourse

Research on classroom discourse has long emphasized that knowledge construction is achieved through interactional processes (i.e., explanation, questioning, critique, and uptake) instead of isolated individual contributions [20, 5]. In multilingual classrooms, these interactions are frequently characterized by translanguaging practices in which students coordinate meaning across and beyond languages and registers [7, 9]. Studies of multilingual STEM classrooms show that student contributions often integrate cognitive work, social alignment, and discourse organization within a single turn, reflecting the inherently multifunctional nature of collaborative sense-making [16].

While discourse analysis approaches have documented these patterns in detail, they typically rely on labor-intensive qualitative methods and small datasets, limiting their scalability for EDM applications [16]. This tension motivates the need for computational representations that remain sensitive to the functional complexity of classroom talk.

1.2 LIWC

LIWC is one of the most widely used dictionary-based tools for analyzing psychological, cognitive, and social dimensions of language. It operationalizes linguistic features as proportions of words associated with categories such as cognitive processes, affect, social relationships, and function words, as well as composite summary measures including Analytical Thinking, Clout, Authenticity, and Emotional Tone [27, 22].

In educational research, LIWC has been used to study how linguistic patterns relate to student outcomes and experience. For example, Lin, Yu, and Dowell [17] showed that cognitive and social LIWC features were differentially associated with performance and sense of belonging in online STEM courses. Other work by [23] showed through LIWC analysis that students' use of function words and analytic language patterns in their writing predicts academic performance.

Foundational LIWC research cautions that psycholinguistic categories do not map transparently onto communicative functions. The interpretation of any given LIWC feature depends on context, genre, and interactional setting, requiring theoretical grounding [27]. In multilingual classroom discourse, this limitation is compounded by LIWC's reliance on language-specific dictionaries. As MLLs fluidly integrate linguistic resources, analysis must either restrict data to one language or transform translanguaged discourse into monolingual representations. Both approaches risk distorting how meaning is distributed across languages and interaction, particularly in collaborative settings where ideational and interpersonal work are tightly integrated.

1.3 Systemic Functional Linguistics

SFL provides a principled framework for interpreting discourse by distinguishing among ideational and interpersonal metafunctions, which capture how speakers construe experience and negotiate social relations during interaction [19]. These dimensions have been widely used to analyze classroom talk and multilingual academic language practices, particularly in studies of collaborative learning and sense-making [26, 28].

- **Ideational**: construing experience, representing processes or phenomena, and elaborating content.
- **Interpersonal**: expressing stance, evaluation, uncertainty, and social alignment with others.

SFL also theorizes a *textual* metafunction concerned with cohesion, information flow, and discourse organization across extended stretches of discourse. However, textual meaning is primarily realized across turns and interactional sequences instead of isolated speaker turns. Because LIWC operates at the level of short utterances and relies on surface lexical and punctuation cues, it is poorly suited for capturing textual resources that develop thematically in dialogic classroom interaction. Accordingly, while textual meaning was annotated to characterize the multifunctional nature of classroom discourse, subsequent LIWC analyses in this study focus exclusively on ideational and interpersonal metafunctions.

2. METHODS

2.1 Dataset

The dataset consists of 204 utterances drawn from ten small-group science classroom transcripts collected across three Midwestern K–12 school districts. Transcripts were selected from longer classroom sessions as part of a broader study of multilingual interaction in science classrooms in Fall 2024 and Spring 2025. For the present analysis, we focus on pre-selected segments capturing focal episodes of collaborative sense-making and translanguaging.

Across these segments, 123 students participated. Classrooms varied in science content and linguistic demographics (four 6th-grade science classrooms, one 9th-grade biology classroom, and one 10th-grade chemistry classroom), and included MLLs with diverse language backgrounds (e.g., Spanish, Arabic) as well as monolingual English-speaking peers. The selected segments represent authentic small-group problem-solving interactions in which students collaboratively interpreted evidence, negotiated understanding, and coordinated action using their multilingual resources.

The data used in this study were collected as part of a broader research project approved by the Institutional Review Board (IRB) (ID: 2024-1029-CP004) at the University of Wisconsin-Madison. Informed consent was obtained from parents/guardians and assent from students in English and Spanish. All data were de-identified before analysis.

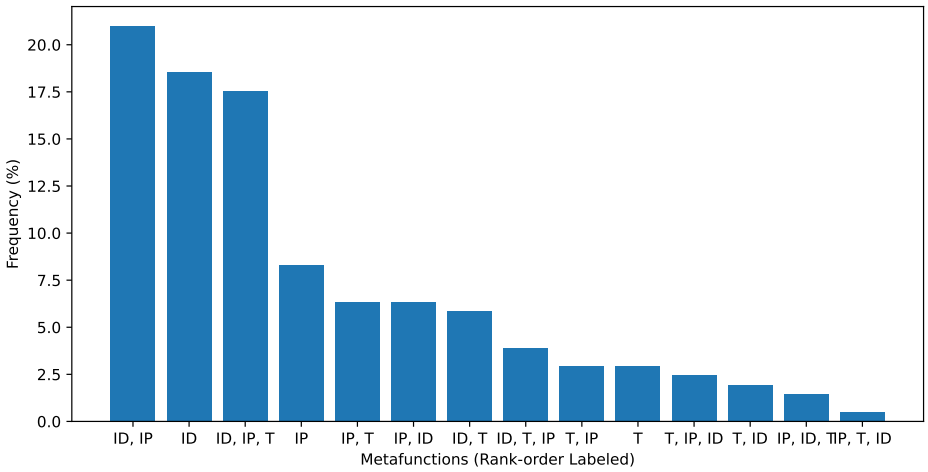


Figure 1: Distribution of adjudicated SFL metafunction labels. ID = ideational; IP = interpersonal; T = textual.

2.2 SFL Annotation

Two bilingual annotators fluent in English and Spanish independently annotated all utterances using a finalized annotation protocol grounded in SFL. Both annotators had expertise in qualitative analysis of multilingual classroom discourse. Annotations served as the reference standard for examining how psycholinguistic features align with functional meanings in multilingual student talk. Disagreements were resolved through structured discussion, with annotators jointly revisiting interpretations and reaching consensus iteratively [15].

Each utterance was labeled for the presence of one or more SFL metafunctions to reflect the multifunctional nature of classroom discourse. Annotators were permitted to assign multiple labels to a single utterance when meanings overlapped, consistent with prior discourse-analytic research on multilingual STEM classrooms. Table 1 provides an illustrative excerpt showing how ideational, interpersonal, and textual meanings can co-occur across short stretches of multilingual peer interaction.

Unit of analysis. The unit of analysis was a single speaker turn, defined as one sentence or short utterance. Utterances are understood as socially and interactionally situated units of meaning-making [1]. Multi-clause turns were annotated as a whole when they reflected a unified communicative intent.

Rank-order labeling. When multiple metafunctions were present, annotators assigned all relevant labels and ranked them by prominence, with the primary communicative function listed first. For example, the utterance *¿Qué pasó? ¿Qué vieron? [What happened? What did you see?]* was labeled as primarily ideational, reflecting the solicitation of experiential information, and secondarily interpersonal, reflecting engagement with peers and the initiation of joint sense-making. Annotators relied on expert judgment rather than fine-grained grammatical segmentation.

Annotations revealed substantial multifunctionality across the corpus. As shown in Figure 1, utterances expressing more than one metafunction were more common than single-function realizations, with ideational–interpersonal combinations occurring most frequently, followed by singular ideational turns and three-way metafunctional combinations. Purely interpersonal or purely textual utterances were comparatively rare.

2.3 Label Reliability

Inter-rater reliability was assessed using Cohen's kappa for each metafunction annotation, as seen in Table 2. Agreement was substantial for ideational meaning ($\kappa = 0.551$), high for interpersonal meaning ($\kappa = 0.693$), and moderate for textual meaning ($\kappa = 0.444$), following conventional interpretation thresholds [14]. Overall micro-averaged agreement was $\kappa = 0.595$, with an exact set-level agreement of 0.620. All disagreements were adjudicated by the primary author to produce a single gold-standard label set used in subsequent analyses.

2.4 LIWC Feature Extraction

Each utterance was analyzed using LIWC-22. LIWC produces a broad set of variables capturing summary dimensions, psychological processes, affective and social language, and function word use. Most variables are calculated as proportions of total words, while summary variables are standardized composite scores derived from underlying categories [27]. The psychometric foundations of LIWC-22 are documented in the LIWC-22 manual [4], and the English LIWC-22 and Spanish ES-LIWC2007 variables in this study are listed in the Appendix.

Because the unit of analysis is a short speaker turn, many LIWC variables were sparse or zero-inflated at the utterance level. To reduce noise associated with short texts and enable

Table 1: Excerpt of multilingual small-group science discourse with SFL metafunction labels.

Speaker	Utterance	Meta-functions
Yesenia	¿Qué pasó? ¿Qué vieron? *[What happened? What did you see?]*	Ideational, Interpersonal
Trevor	Las personas de … o los científicos … *[The people … or the scientists …]*	Ideational
Yesenia	Ajá, ¿qué están haciendo los científicos? *[Uh-huh, what are the scientists doing?]*	Ideational
Trevor	Matando a las … um … urchins *[Killing the … um … urchins]*	Ideational, Interpersonal
Yesenia	¿Erizos? *[Urchins?]*	Ideational, Interpersonal
Trevor	Erizos, yeah *[Urchins, yeah]*	Ideational
Yesenia	¿Están matando a los erizos porque qué pasó con los erizos? *[Are they killing the urchins because what happened with the urchins?]*	Ideational, Textual
Trevor	Se comen la kelpo *[They eat the kelp.]*	Ideational
Yesenia	Entonces, ¿hay muchos erizos o pocos erizos? *[So, are there many urchins or few urchins?]*	Ideational
Alonso	Muchos *[Many.]*	Ideational, Textual

Table 2: Human–human agreement for metafunction labels per utterance.

Metafunction	κ
Ideational	0.551
Interpersonal	0.693
Textual	0.444
Overall micro	0.595
Exact set-level agreement	0.620

interpretable comparison across features, LIWC variables appearing in fewer than 5% of utterances were excluded from analysis. The remaining variables were then carried forward to the correlation analyses described below, where 52 exceeded the median absolute correlation threshold for at least one metafunction and were retained for reporting.

2.5 Statistical Analysis

To examine how LIWC features correlate with functional discourse, we conducted three complementary analyses. Together, these analyses examine empirical patterns, theory-driven expectations, and robustness across language conditions.

Analysis 1: Empirical alignment between LIWC features and SFL metafunctions. First, we examined associations between LIWC variables and adjudicated SFL metafunction labels using correlation analysis. The ideational and interpersonal metafunctions were represented as binary indicators indicating whether they were present in a given utterance, which allowed both metafunctions to co-occur within the same turn.

Spearman rank-order correlations were computed between each LIWC variable and each metafunction indicator. Spearman correlation was selected to accommodate non-normally distributed variables and the ordinal nature of feature ranks. Given the exploratory goal of identifying interpretable alignment patterns, we focused subsequent analyses on LIWC variables whose correlations exceeded the median absolute correlation value for each metafunction. This thresholding strategy was used to distinguish features showing relatively stronger alignment with expert annotations from those exhibiting weak or inconsistent associations, without assuming statistical significance in a small corpus. Correlation results were visualized using heatmaps to facilitate comparison across metafunctions and LIWC feature groups.

Analysis 2: Theory-driven alignment with SFL lexicogrammatical resources. As a second line of analysis, we adopt a theory-driven approach to evaluate whether psycholinguistic features derived from LIWC align with lexicogrammatical resources theorized in SFL. Whereas Analysis 1 examined empirical associations between LIWC variables and annotated metafunctions, Analysis 2 tests whether LIWC variables hypothesized to index ideational or interpersonal meaning *a priori* exhibit differentiated alignment with these metafunctions in the annotated transcript.

Based on SFL discourse semantics and genre theory [18, 11], we specified LIWC variables expected to align with ideational and interpersonal meaning, respectively (see the Appendix). Variables were selected as surface linguistic indicators plausibly associated with different meaning orientations: ideational predictors index elaboration and references to processes or phenomena (e.g., *WC*, *WPS*, *Perception*, *space*, *differ*); interpersonal predictors index stance, social positioning, and participant orientation (e.g., *ppron*, *you*, *Social*, *Clout*). Spearman correlations were computed between each LIWC variable and metafunction.

Analysis 3: Language and Dictionary-Based Error Analysis. As a third analysis, we examined the robustness of LIWC–SFL alignment across language conditions while accounting for constraints of LIWC dictionary design. Currently, LIWC-22 supports analysis using only one language-specific dictionary at a time[1]. Accordingly, we conducted two parallel analyses across all three transcript conditions: (a) original translanguaged utterances, (b) English-only versions, and (c) Spanish-only versions using the LIWC-22 English dictionary and the internal ES-LIWC2007 Spanish dictionary.

Within each dictionary, LIWC features were computed sep-

[1]We confirmed this constraint with the creators of LIWC-22.

arately for all transcript conditions and related to the same expert-annotated SFL metafunction labels. Following prior LIWC cross-language methodology, analyses focused on within-dictionary patterns and not on direct comparisons across languages [24]. Although the English and Spanish LIWC dictionaries share category structures, subtle differences in linguistic realization and coverage exist (e.g., subject pronoun omission in Spanish), motivating a within-language analytic approach.

Together, the three analyses serve complementary purposes. Analysis 1 establishes empirical alignment patterns between LIWC features and expert interpretations of meaning. Analysis 2 evaluates whether these patterns align with theory-driven expectations about how meaning is realized lexicogrammatically. Analysis 3 probes the stability of these alignments under common preprocessing and language-restriction decisions in EDM. This design supports both exploratory and theory-driven analysis.

3. RESULTS

3.1 Analysis 1: LIWC Predictors and SFL Metafunctions

We examined associations between LIWC variables and expert-annotated SFL metafunctions using Spearman rank-order correlations. Ideational and interpersonal metafunctions were represented as binary indicators denoting whether each meaning was present in a given utterance. Fifty-two LIWC variables exceeded the median correlation threshold for at least one metafunction and were retained for analysis. For interpretability, variables were grouped post hoc into six LIWC-informed categories. Separate heatmaps were generated for each category to visualize alignment patterns with ideational and interpersonal meanings.

3.1.1 Summary Variables

Figure 4 presents correlations between SFL metafunctions and LIWC summary variables. Ideational meaning showed the strongest alignment within this category, with positive correlations for word count (WC; $\rho = 0.28$), words per sentence (WPS; $\rho = 0.28$), and BigWords ($\rho = 0.22$). Clout was also positively associated with ideational meaning ($\rho = 0.16$), while dictionary coverage (Dic) showed a negative association ($\rho = -0.18$).

Interpersonal meaning exhibited weaker and more selective alignment with summary variables. Clout showed a positive correlation ($\rho = 0.19$), while Analytic was negatively associated ($\rho = -0.23$). Other summary variables showed correlations close to zero.

3.1.2 Linguistic Dimensions

Figure 2 shows correlations for LIWC linguistic dimension variables. Interpersonal meaning exhibited the clearest alignment in this category, with the strongest associations observed for personal pronouns (ppron; $\rho = 0.28$), pronouns overall ($\rho = 0.21$), first-person singular *I* ($\rho = 0.20$), and second-person *you* ($\rho = 0.16$).

Ideational meaning showed more moderate and distributed associations across linguistic dimensions, including positive correlations for ipron ($\rho = 0.17$), determiners ($\rho = 0.17$), prepositions ($\rho = 0.17$), auxiliary verbs ($\rho = 0.15$), and conjunctions ($\rho = 0.14$). The aggregate Linguistic category was negatively associated with ideational meaning ($\rho = -0.23$).

3.1.3 Psychological Processes

Figure 3 presents correlations between metafunctions and LIWC psychological process variables. Ideational meaning showed its strongest association with *differ* ($\rho = 0.18$), followed by affiliation ($\rho = 0.11$) and cause-related language ($\rho = 0.08$). Other cognitive process categories showed weak or near-zero associations.

Interpersonal meaning exhibited modest positive correlations for discrepant language (*discrep*; $\rho = 0.12$), cognition ($\rho = 0.10$), and cognitive processing more broadly ($\rho = 0.09$), while *differ* showed a small negative association ($\rho = -0.06$).

3.1.4 Affect and Social Processes

Figure 5 shows correlations for LIWC affective and social process variables. Interpersonal meaning aligned most clearly with this category, with positive correlations for Social ($\rho = 0.22$), socrefs ($\rho = 0.18$), socbehav ($\rho = 0.12$), and assent ($\rho = 0.15$).

Ideational meaning showed weak associations with most affective and social variables, with the exception of a strong negative correlation with assent ($\rho = -0.35$), indicating that brief agreement tokens were unlikely to be annotated as ideational.

3.1.5 Perception and Time Orientation

Figure 6 presents correlations for LIWC perception and time orientation variables. Ideational meaning showed consistent positive alignment with perception ($\rho = 0.22$), spatial language ($\rho = 0.19$), motion ($\rho = 0.13$), and temporal focus, including past ($\rho = 0.12$) and future ($\rho = 0.12$) orientation.

Interpersonal meaning showed weak associations across this category, with correlations near zero for most variables and small positive correlations for present ($\rho = 0.08$) and future focus ($\rho = 0.05$).

3.1.6 Conversational and Punctuation Features

Figure 7 shows correlations for LIWC conversational and punctuation variables. Ideational meaning exhibited consistent negative correlations with conversational markers, including Conversation ($\rho = -0.29$), nonfluencies ($\rho = -0.18$), overall punctuation density (AllPunc; $\rho = -0.16$), and comma use ($\rho = -0.13$). In contrast, Apostro ($\rho = 0.12$) and OtherP ($\rho = 0.11$) showed small positive associations.

Interpersonal meaning showed weak and inconsistent alignment across conversational and punctuation features, with correlations ranging from $\rho = -0.03$ to 0.10.

3.2 Analysis 2: LIWC Alignment with SFL Lexicogrammatical Resources

We examined whether LIWC variables predicted *a priori* to align with ideational or interpersonal lexicogram-

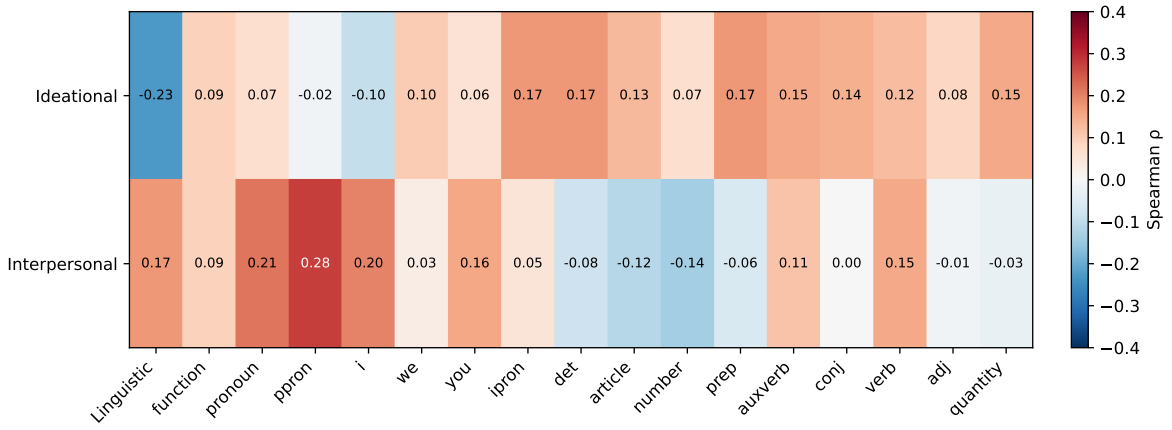


Figure 2: Spearman correlations between SFL metafunctions and LIWC Linguistic Dimensions.

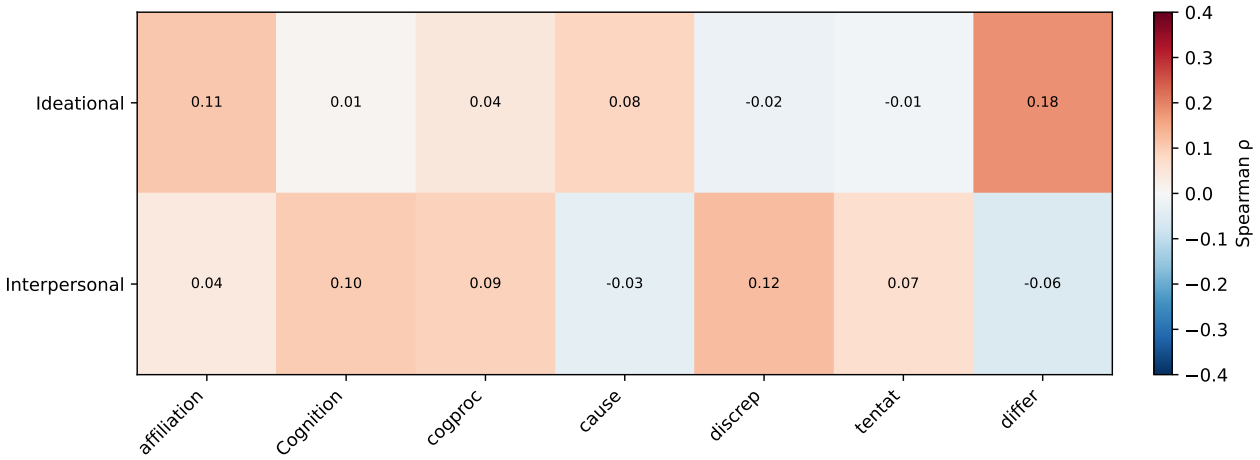


Figure 3: Spearman correlations between SFL metafunctions and LIWC Psychological Processes.

matical resources exhibit empirical alignment with expert-annotated metafunctions. For each theoretically-specified LIWC variable, Spearman correlations were computed with both ideational and interpersonal indicators to give direct comparison across metafunctions.

Ideational-predicted LIWC variables. Figure 8 shows Spearman correlations for LIWC variables hypothesized to index ideational meaning, displayed for both ideational and interpersonal metafunctions. Measures of elaboration showed the strongest positive correlations with ideational meaning, including word count (WC; $\rho = 0.28$) and words per sentence (WPS; $\rho = 0.28$). Perceptual and experiential variables also showed positive correlations with ideational meaning, including *Perception* ($\rho = 0.22$), *space* ($\rho = 0.19$), *motion* ($\rho = 0.13$), and *differ* ($\rho = 0.18$).

Across these variables, correlations with interpersonal meaning were weaker or near zero. LIWC cognitive process variables (e.g., *cogproc*, *cause*, *insight*, *discrep*) showed weak or inconsistent correlations with ideational meaning (all $\rho < 0.10$).

Interpersonal-predicted LIWC variables. Figure 9 presents correlations for LIWC variables hypothesized to reflect interpersonal meaning for ideational and interpersonal metafunctions. Pronoun-based features showed the strongest positive correlations with interpersonal meaning, including personal pronouns (*ppron*; $\rho = 0.28$), first-person singular (I; $\rho = 0.20$), and second-person address (*you*; $\rho = 0.16$). Social reference variables also showed positive correlations with interpersonal meaning, including *Social* ($\rho = 0.22$), *socrefs* ($\rho = 0.18$), *affiliation* ($\rho = 0.11$), and *Clout* ($\rho = 0.19$).

The same variables showed weaker correlations with ideational meaning, including *ppron* ($\rho = -0.02$), I ($\rho = -0.10$), *you* ($\rho = 0.06$), *Social* ($\rho = 0.13$), *socrefs* ($\rho = 0.13$), *affiliation* ($\rho = 0.11$), and *Clout* ($\rho = 0.16$).

Affective and modality-related variables showed comparatively weaker correlations with interpersonal meaning, in-

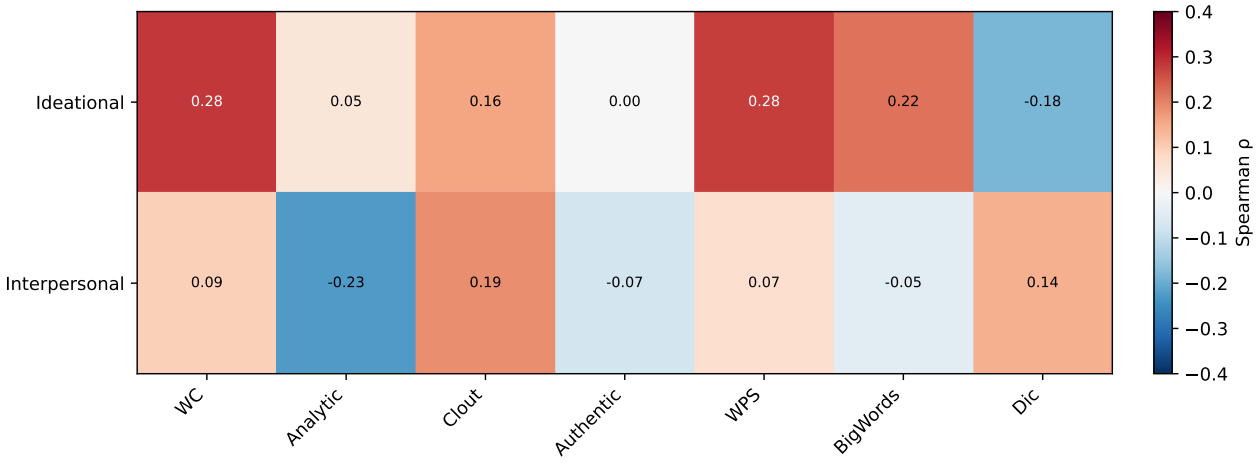


Figure 4: Spearman correlations between SFL metafunctions and LIWC Summary Variables.

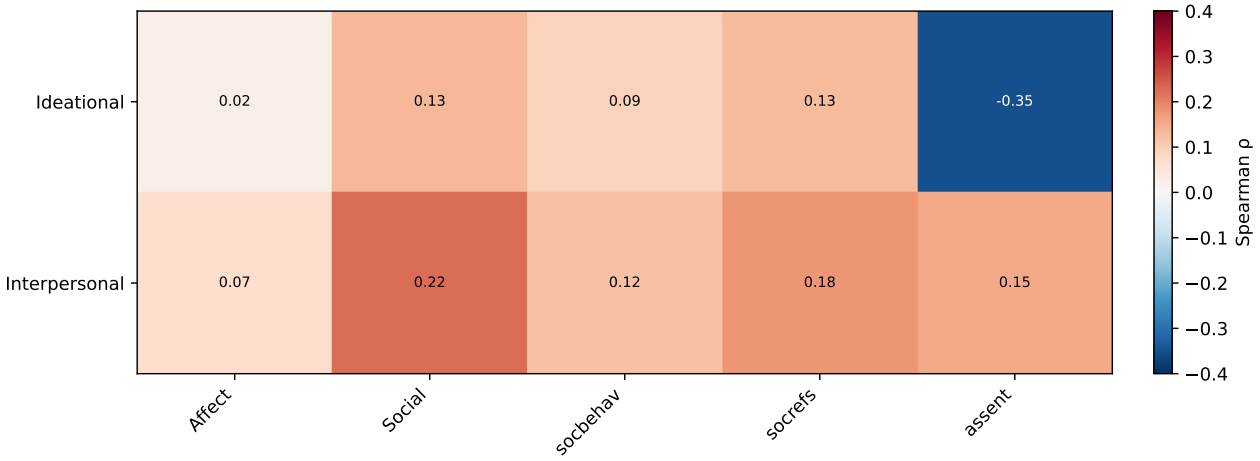


Figure 5: Spearman correlations between SFL metafunctions and LIWC Affective and Social Processes.

cluding *Affect* ($\rho = 0.07$), *tentat* ($\rho = 0.07$), and *negate* ($\rho = 0.06$).

The patterns observed in Analysis 2 support the construct validity of the theory-driven mapping. Variables predicted to align with ideational and interpersonal meaning showed differentiated correlations across metafunctions. This divergence would not be expected if LIWC features reflected only generic verbosity or affect, but instead indicates theoretical alignment with dimensions of meaning. At the same time, the weak and inconsistent performance of cognitive-process variables highlights limits in how these categories map onto interactional reasoning in multilingual classroom discourse.

3.3 Analysis 3: Language, Translation, and Error Analysis

LIWC outputs varied systematically across the original translanguaged, English-only, and Spanish-only transcript conditions, with distinct patterns depending on dictionary choice (Tables 3 and 4).

Using the LIWC-22 English dictionary (Table 3), English-only transcripts showed higher values than the original translanguaged condition for dictionary-based and function-heavy categories, including *Dic* (89.03 vs. 71.28), *Linguistic* (79.82 vs. 62.83), and *function* (55.85 vs. 42.60). The Spanish-only condition showed pronounced reductions in these same variables (e.g., *Dic* = 16.91; *function* = 8.00), while punctuation- and segmentation-related measures increased, including *Analytic* (63.81 vs. 30.43), *Period* (26.74 vs. 12.61), and *AllPunc* (57.59 vs. 42.12). Pronoun- and verb-based categories were near zero in the Spanish-only condition under the English dictionary.

Using the ES-LIWC2007 Spanish dictionary (Table 4), Spanish-only transcripts showed higher values than the original for dictionary match and grammatical categories, including *Dic* (80.70 vs. 28.28), *Funct* (51.59 vs. 19.02), *Verbos* (18.89 vs. 5.36), and *TotPron* (12.72 vs. 4.48). English-only transcripts showed consistently lower values across these same categories. Punctuation-related variables (*AllPunc*,

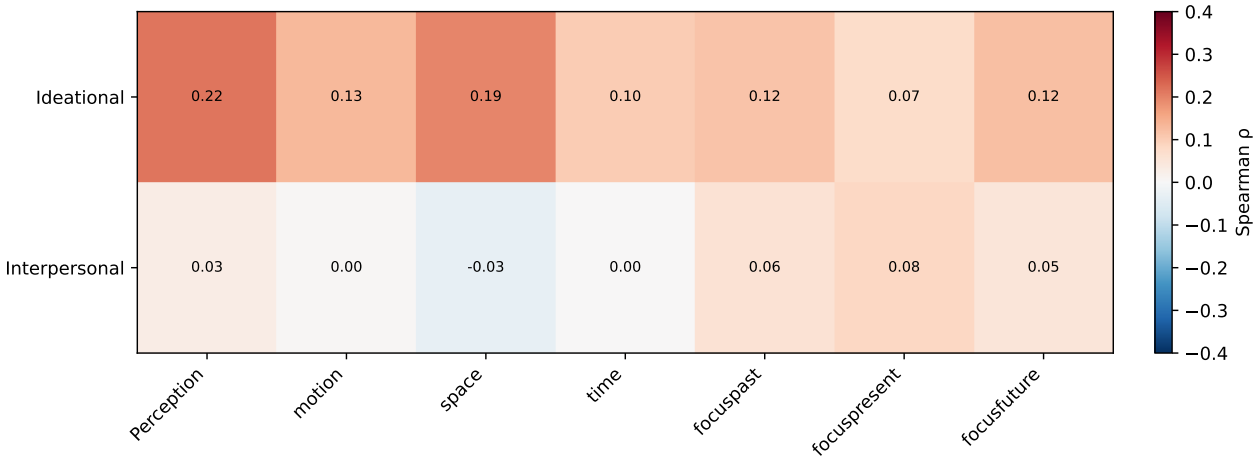


Figure 6: Spearman correlations between SFL metafunctions and LIWC Perception and Time Orientation variables.

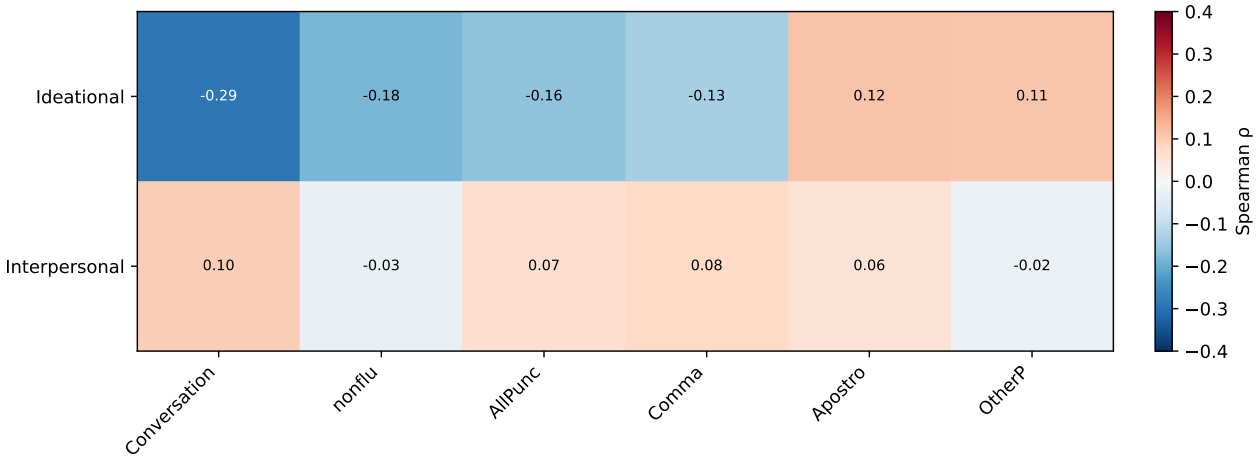


Figure 7: Spearman correlations between SFL metafunctions and LIWC Conversational and Punctuation variables.

Period, *QMark*) showed higher values in the Spanish-only condition across both dictionaries.

Across both dictionary settings, the original translanguaged condition generally fell between the English-only and Spanish-only conditions on most variables (Tables 3–4), indicating that language restriction and translation systematically reshaped LIWC feature distributions.

4. DISCUSSION

This study uses SFL to address a central challenge in EDM: how to improve interpretation of automated linguistic measures in ways that are theoretically grounded and reflect authentic language use, particularly in multilingual STEM classrooms. SFL provides a principled account of how language simultaneously construes experience, enacts social relations, and organizes discourse, making it well suited for analyzing the meaning-making practices of MLLs whose linguistic resources span languages and registers.

Against this backdrop, we examined how psycholinguistic features derived from LIWC align with expert annotations of SFL metafunctions in K-12 multilingual small-group science discourse. By grounding analysis in ideational and interpersonal meaning, the results clarify which dimensions of classroom talk are captured by commonly used psycholinguistic features and which aspects may remain less visible within current analytic approaches.

Across analyses, ideational meaning showed limited alignment with LIWC features. It was most reliably associated with surface measures of elaboration, particularly *WC* and *WPS*. Although often treated as coarse indicators, these measures indexed experiential and explanatory content in collaborative science talk, suggesting that longer and more elaborated turns were more likely to involve ideational meaning. Perceptual and spatial categories (e.g., *Perception*, *space*, *motion*) also showed modest alignment, reflecting the role of observation and spatial reasoning during scientific sense-making.

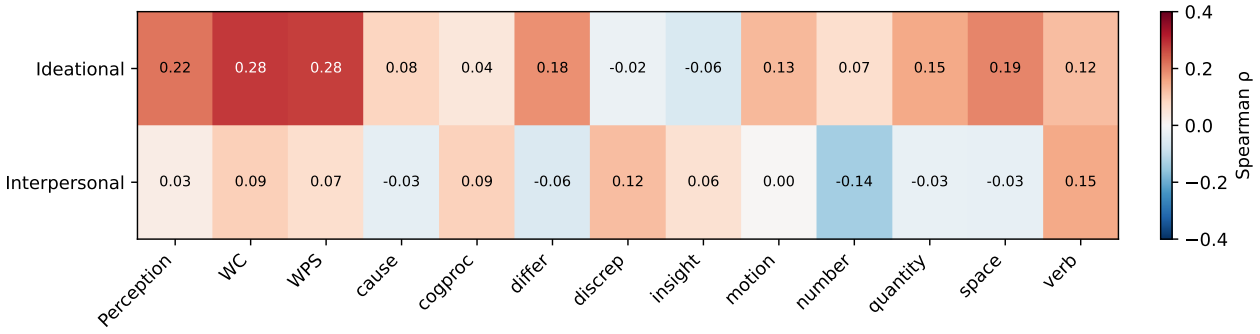


Figure 8: Spearman correlations between LIWC variables and the ideational metafunction.

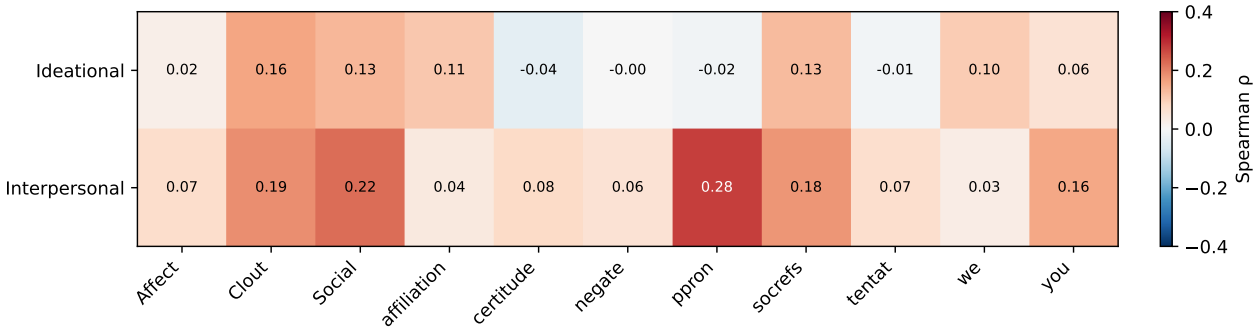


Figure 9: Spearman correlations between LIWC variables and the interpersonal metafunction.

Several LIWC variables commonly interpreted as indicators of conceptual processing showed weak or inverse relationships with ideational meaning. Higher values for *Dic* and aggregate linguistic measures (e.g., *Linguistic*, function word totals, and overall punctuation density) were negatively associated with ideational meaning. In multilingual classroom discourse, lower dictionary coverage may reflect translanguaging practices and domain-specific terminology that fall outside of monolingual dictionary inventories. As a result, utterances that are rich in disciplinary and collaborative content may register as having a lower dictionary match, despite contributing substantively to meaning-making.

In contrast, interpersonal meaning exhibited the strongest and most coherent alignment with LIWC features. Pronoun variables (e.g., *ppron*, *I*, *you*), social reference categories, and summary measures related to social positioning (e.g., *Clout*) showed the highest and most stable correlations across analyses. Agreement markers such as *assent* followed this pattern, showing a positive correlation with interpersonal meaning and a strong negative correlation with ideational meaning ($\rho = -0.35$). This pattern is most clearly reflected in Analysis 2, where variables predicted to align with ideational and interpersonal meaning showed clearly differentiated correlations across metafunctions. These patterns align with SFL accounts in which interpersonal meaning is enacted through stance-taking, evaluation, and participant positioning.

Relatedly, several LIWC categories often interpreted as indexing cognition (e.g., *cogproc*, *discrep*) aligned more strongly with interpersonal than ideational meaning. In dialogic classroom interaction, these features frequently function to signal epistemic stance, hesitation, or negotiation (e.g., "I think," "I don't know," "maybe") within interaction rather than to elaborate scientific explanations. This suggests that LIWC cognitive-process variables in collaborative settings are more closely associated with interactional positioning than with experiential or explanatory content.

Findings from the third analysis sharpen these interpretations by isolating the role of language-bound dictionary assumptions. When multilingual classroom talk was constrained to a single-language representation, the prominence of LIWC features shifted systematically, with greater emphasis on grammatical density, punctuation, and segmentation, and reduced sensitivity to social (e.g., *Social*, *socrefs*, *affiliation*) and experiential (e.g., *Perception*, *space*) categories. These shifts varied by dictionary, indicating that alignment patterns are shaped by how linguistic resources are partitioned across language-specific inventories. The translanguaged condition did not behave as an aggregation of English and Spanish features, but instead exhibited a distinct distribution of LIWC features, suggesting a redistribution of meaning-making resources that is not fully preserved under single-language dictionaries. These findings empirically challenge assumptions that academic reasoning occurs primarily in English while other languages serve only social or supportive functions, showing instead that MLLs use their full linguistic repertoires for both ideational and inter-

Table 3: Mean LIWC values across transcript conditions in the LIWC-22 English dictionary.

LIWC Variable (English dict)	Original	English-only	Spanish-only
Dic	71.28	89.03	16.91
Linguistic	62.83	79.82	12.45
function	42.60	55.85	8.00
Tone	41.83	54.45	20.60
Cognition	22.85	28.01	5.87
verb	16.39	21.61	0.00
pronoun	14.50	18.17	0.50
Clout	36.11	41.26	12.43
Social	5.92	8.53	1.13
Perception	7.22	9.29	1.24
Analytic	30.43	31.25	63.81
AllPunc	42.12	40.42	57.59
Period	12.61	13.33	26.74
QMark	12.16	8.03	17.91
BigWords	10.42	10.64	17.96

Table 4: Mean LIWC values across transcript conditions in the ES-LIWC2007 Spanish dictionary.

Spanish LIWC var	English counterpart	Original	English-only	Spanish-only
Dic	Dic	28.28	12.30	80.70
Funct	function	19.02	7.78	51.59
MecCog	Cognition	7.92	0.61	26.59
Verbos	verb	5.36	1.17	18.89
TotPron	pronoun	4.48	0.37	12.72
Present	focuspresent	4.59	1.17	16.21
Relativ	relativ	2.11	0.17	10.31
Conjunc	conj	1.23	0.00	7.23
AllPunc	AllPunc	42.12	40.42	57.59
Period	Period	12.61	13.33	26.74
QMark	QMark	12.16	8.03	17.91
Sixltr	BigWords	10.42	10.64	17.96

personal meaning-making.

Taken together, these outcomes have direct implications for how psycholinguistic features are interpreted and used in EDM models of learning in multilingual classroom contexts. In particular, LIWC categories conventionally treated as indicators of cognition or reasoning consistently aligned with interpersonal meaning in dialogic interaction. As a result, models that do not account for these learner variabilities may underestimate MLLs' knowledge and shape both assessment and downstream instructional decisions in ways that do not reflect their full meaning-making practices [13]. We therefore recommend that LIWC-based approaches more explicitly account for multilinguality, and that future work compare these features with methods that are better suited to multilingual analysis (e.g., parts-of-speech tagging).

Several limitations should be considered when interpreting these findings. The dataset consists of a relatively small number of annotated utterances drawn from focal segments of authentic classroom interactions, which enables fine-grained functional analysis but limits statistical power and generalizability to broader classroom populations. Multilingual classroom discourse highlights a significant limitation of dictionary-based approaches, as translanguaging practices and multilinguality are not fully captured by language-specific resources. In addition, SFL metafunction labels were applied at the utterance level and allowed to co-occur, reflecting the multifunctional nature of classroom talk. While theoretically grounded, this design attenuates feature–label correlations and complicates interpretation. Future work should explore representations that capture temporal context, turn-taking structure, and the development of meaning across turns, including textual and thematic organization, to better align computational features with interactional processes.

5. CONCLUSION

This study uses SFL theory to examine what LIWC features capture in multilingual classroom discourse. LIWC aligns most strongly with interpersonal meaning by capturing stance and social positioning. Ideational meaning is only partially reflected through elaboration and experiential language, while cognitive-process categories show limited alignment with explanatory content. Integrating SFL with computational analysis can support more precise interpretation of linguistic features and the development of methods that capture ideational and interpersonal meaning-making in multilingual learning contexts.

6. ACKNOWLEDGMENTS

The research reported in the article was made possible (in part) by a grant from the Spencer Foundation (#202500160). The views expressed are those of the authors and do not necessarily reflect the views of the Spencer Foundation. Support for this research was provided by the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin-Madison with funding from the Wisconsin Alumni Research Foundation. We thank the Multilingual Learning Research Center and Institute for Diversity Science at the University of Wisconsin-Madison for supporting our work. Ananya Ganesh was supported

by a postdoctoral fellowship from the Institute of Diversity Science. We also thank Dr. Diego Román, Dr. Ruslana Westerlund, Pragati Maheshwary, Alina Guha, and EDM reviewers for their valuable feedback.

7. REFERENCES

[1] M. M. Bakhtin. *Speech Genres and Other Late Essays*. University of Texas Press, Austin, 1986.

[2] K. Barron, N. Tseng, S. Karumbaiah, and C. T. Baeza. Evaluating language models using linguistic variations in multilingual learners' writing: A teacher study. In *Proceedings of the 19th International Conference of the Learning Sciences (ICLS 2025)*, pages 1389–1393. International Society of the Learning Sciences, 2025.

[3] S. L. Blodgett, S. Barocas, H. Daumé III, and H. Wallach. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[4] R. L. Boyd, A. Ashokkumar, S. Seraj, and J. W. Pennebaker. The development and psychometric properties of liwc-22. Technical report, University of Texas at Austin, Austin, TX, 2022.

[5] M. T. H. Chi. Active-constructive-interactive: A conceptual framework for differentiating learning activities. *Topics in Cognitive Science*, 1(1):73–105, 2009.

[6] D. Demszky, J. Liu, H. C. Hill, S. Sanghi, and A. Chung. Automated feedback improves teachers' questioning quality in brick-and-mortar classrooms: Opportunities for further enhancement. *Computers & Education*, 227:105183, 2025.

[7] O. García and A. M. Y. Lin. Translanguaging in bilingual education. In O. García, A. M. Y. Lin, and S. May, editors, *Bilingual and Multilingual Education*. Springer, 2 edition, 2017.

[8] O. García and L. Wei. *Translanguaging: Language, Bilingualism and Education*. Palgrave Macmillan, London, 2014.

[9] A. Guha, S. Karumbaiah, C. Baeza, M. Castro, K. White, and D. Roman. Teacher perceptions on ai support for bi/multilingual learners: Superpowers and constraints. In *International Conference on Artificial Intelligence in Education*, pages 85–91. Springer, 2025.

[10] M. A. K. Halliday. *An Introduction to Functional Grammar*. Edward Arnold, 1994.

[11] R. Harman. *Bilingual Learners and Social Equity: Critical Approaches to Systemic Functional Linguistics*. Springer, Cham, Switzerland, 2018.

[12] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[13] S. Karumbaiah, A. Ganesh, S. Yang, and N. Tseng. Prompt vs. supervise: A case of using language models to assess students' science explanations. In *European Conference on Technology Enhanced Learning*, pages 244–258. Springer, 2025.

[14] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977.

[15] A. Lane and M. L. Kent. Dialogic engagement. *The handbook of communication engagement*, pages 61–72, 2018.

[16] J. Langman and H. Hansen-Thomas. Discourse analysis in multilingual classrooms. *Language and Education*, 2017.

[17] Y. Lin, R. Yu, and N. Dowell. Liwcs the same, not the same: Gendered linguistic signals of performance and experience in online stem courses. In *Proceedings of AIED 2020*, volume 12163 of *Lecture Notes in Computer Science*. Springer, 2020.

[18] J. R. Martin. *Discourse Semantics*. Bloomsbury Academic, London, UK, 2014.

[19] J. R. Martin and D. Rose. *Working with Discourse: Meaning Beyond the Clause*. Continuum, 2 edition, 2007.

[20] N. Mercer. Sociocultural discourse analysis: Analysing classroom talk as a social mode of thinking. *Journal of Applied Linguistics*, 1(2):137–168, 2004.

[21] R. Otheguy, O. García, and W. Reid. Clarifying translanguaging and deconstructing named languages. *Applied Linguistics*, 36(3):281–307, 2015.

[22] J. W. Pennebaker. *The Secret Life of Pronouns*. Bloomsbury, 2011.

[23] J. W. Pennebaker, C. K. Chung, J. Frazee, G. M. Lavergne, and D. I. Beaver. When small words foretell academic success: The case of college admissions essays. *PLOS ONE*, 9(12):e115844, 2014.

[24] N. Ramírez-Esparza, C. K. Chung, E. Kacewicz, and J. W. Pennebaker. The psychology of word use in depression forums in english and in spanish: Testing two text analytic approaches. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 2, pages 102–108, 2021.

[25] C. P. Rosé, O. Ferschke, V. Dutt, Y. Jo, Y. Cui, R. Wang, R. Kumar, and E. Fancher. Analyzing collaborative learning processes automatically: Exploiting the advances of natural language processing. *International Journal of Artificial Intelligence in Education*, 29(3):305–346, 2019.

[26] M. J. Schleppegrell. Review of classroom discourse analysis: A functional perspective by f. christie. *Language in Society*, 33(3):443–446, 2004.

[27] Y. R. Tausczik and J. W. Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54, 2010.

[28] F. J. Troyan, J. D. Herazo, and M. Ryshina-Pankova. Sfl pedagogies in language education: Special issue introduction. *System*, 104:102694, 2022.

APPENDIX
A. LIWC VARIABLES

Table 5: LIWC variables used in analyses. ID = ideational; IP = interpersonal; (–) = no a priori prediction specified.

Category	Abbrev.	Description / Exemplars	Predicted
Summary Variables			
Word Count	WC	Total number of words.	ID
Analytical Thinking	Analytic	Formal, logical, hierarchical language.	ID
Clout	Clout	Confidence, leadership, social status.	IP
Authenticity	Authentic	Honest, personal, genuine language.	IP
Emotional Tone	Tone	Overall emotional positivity or negativity.	IP
Words per Sentence	WPS	Average sentence length.	ID
Big Words	BigWords	Words with six or more letters.	–
Dictionary Words	Dic	Words matched by LIWC dictionary.	–
Linguistic Processes			
Linguistic Processes	Linguistic	Aggregate function word usage.	–
Function Words	function	Articles, pronouns, prepositions, auxiliaries.	–
Pronouns (total)	pronoun / TotPron	All personal and impersonal pronouns.	–
Personal Pronouns	ppron	I, we, you, they.	IP
First-person singular	I	I, me, my.	IP
First-person plural	we	we, us, our.	IP
Second-person	you	you, your.	IP
Impersonal Pronouns	ipron	it, that, those.	–
Determiners	det	this, that, these.	–
Prepositions	prep	to, with, from, entre.	–
Auxiliary Verbs	auxverb	be, have, do.	–
Conjunctions	conj / Conjunc	and, but, or, porque.	–
Negations	negate	no, not, never.	IP
Verbs	verb / Verbos	Action and relational verbs.	ID
Quantifiers	quantity / Cuantif	many, few, much.	ID
Numbers	number	Numeric digits and number words.	ID
Adjectives	adj	Descriptive modifiers.	–
Cognitive Processes			
Cognitive Processes	cogproc / MecCog	Thinking, knowing, reasoning.	ID
Insight	insight	Realization, understanding.	ID
Causation	cause	Cause–effect relations.	ID
Discrepancy	discrep	Should, would, could.	ID
Tentativeness	tentat	Maybe, perhaps.	IP
Differentiation	differ	Contrast (but, however).	ID
Certitude	certitude	Certainty and confidence markers.	IP
Affective and Social Processes			
Affect	Affect	Emotional expression (happy, sad, angry).	IP
Social Processes	Social	Social interaction language.	IP
Social References	socrefs	References to others.	IP
Social Behavior	socbehav	Helping, talking, sharing.	IP
Affiliation	affiliation	Belonging and connection.	IP
Assent	assent	Agreement markers (yes, okay).	IP
Perception and Time			
Perception	Perception / Percept	Sensory language (see, hear, feel).	ID
Motion	motion	Movement and change.	ID
Space	space / Espacio	Spatial reference.	ID
Past Focus	focuspast	Past tense orientation.	–
Present Focus	focuspresent / Present	Present tense orientation.	–
Future Focus	focusfuture	Future-oriented language.	–
Conversational and Punctuation			
Conversation	Conversation	Informal spoken discourse markers.	–
Nonfluencies	nonflu	Filled pauses and hesitations (um, uh).	–
All Punctuation	AllPunc	Overall punctuation density.	–
Comma	Comma	Comma frequency.	–
Apostrophe	Apostro	Apostrophe frequency.	–
Other Punctuation	OtherP	Punctuation other than periods and commas.	–

A Human-in-the-Loop Pipeline for Multi-Class SRL Classification

Andreas Kissmehl
KTH Royal Institute of Technology
Stockholm, Sweden
kismehl@kth.se

Olga Viberg
KTH Royal Institute of Technology
Stockholm, Sweden
oviberg@kth.se

Olov Engwall
KTH Royal Institute of Technology
Stockholm, Sweden
engwall@kth.se

Richard Lee Davis
KTH Royal Institute of Technology
Stockholm, Sweden
rldavis@kth.se

ABSTRACT

Annotation of big data is a nontrivial task for learning analytics scholars. Educational constructs drawing from learning psychology, such as self-regulated learning, require theory-grounded domain expertise and cannot be easily inferred from surface features, making annotation time-intensive and difficult to scale. Recent work aims to address this issue by using large language models (LLMs) to annotate data. While this approach can increase the amount of annotated data, evidence suggests that LLMs tend to overgeneralize, based on the examples included in the prompt cases, and misclassify due to a lack of context understanding. To address this, we investigate how human-in-the-loop machine learning strategies can be used to maximize the impact of human annotation by focusing efforts on a subset of data that would benefit most from labeling or relabeling. Utilized data include human-labeled, machine-labeled, and unlabeled student prompts from diverse educational contexts. We compare two active learning methods, loss weighting and their combination on the task of multi-class classification of student-generated text in AI-chatbot interactions. Experiments with a multilingual DeBERTa-based classifier indicate that combining human annotation with selective correction of weak labels can improve performance compared to a baseline trained on available annotated data annotated by humans and LLMs. The human-in-the-loop pipeline mitigates misalignment in labels produced by LLMs, providing a path forward to more responsible and rigorous use of LLMs in educational data annotation tasks.

Keywords

Human-in-the-loop learning, active learning, annotation quality, large language models, self-regulated learning

Andreas Sebastian Kissmehl, Olga Viberg, Olov Engwall, and Richard Lee Davis. A Human-in-the-Loop Pipeline for Multi-Class SRL Classification. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 400–407. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040133

1. INTRODUCTION

Educational constructs such as self-regulated learning (SRL), motivation, and help-seeking (e.g.,[20, 21, 11]) are widely used as proxies for studying students' learning behaviors. However, these constructs are inherently latent and not directly observable in student interactions; instead, they must be inferred through a theoretically grounded interpretation. Researchers therefore need to operationalize them through observable indicators, such as discourse patterns, behavioral traces, or interaction features, connecting abstract theoretical constructs to empirical data. This process is necessarily interpretative and theory-laden, making annotation less a technical labeling exercise and more an expert-driven analytical task. Producing high-quality annotated datasets is thus essential but also highly time-intensive, costly, and difficult to scale, particularly when expertise in both learning theory and specific educational contexts is required [10]. The strong dependence on expert judgment and contextual understanding further limits the feasibility of substituting expert annotators with crowdsourced approaches without risking reduced validity and theoretical fidelity [15].

These challenges are becoming more pronounced as students increasingly engage with AI conversational agents such as Claude in higher education [7]. Such interactions generate large volumes of rich process data that offer unprecedented opportunities to examine how learning unfolds in AI-mediated contexts. However, the scale, heterogeneity, and context sensitivity of these interaction logs intensify existing annotation challenges. Without robust theoretically grounded annotation frameworks, translating these large-scale student-AI interaction datasets into meaningful insights about learning processes remains highly complex and not straightforward.

To address these challenges, scholars have increasingly explored the potential of LLMs (see, e.g., [17, 1]) to support the annotation of educational data. Their capacity to process large volumes of text suggests a promising pathway toward scaling annotation efforts that have traditionally relied on intensive expert labor. However, a central concern remains how limited expert time and resources can be allocated so that LLM-assisted annotation increases the quantity of la-

beled data without compromising quality, theoretical validity, or interpretability [29]. Emerging research indicates that LLMs can facilitate large-scale annotation while reducing reliance on time-consuming expert coding. However, the reliability and theoretical fidelity of LLM-generated labels remain highly sensitive to prompting strategies, contextual framing, and task specification [9, 23]. Consequently, efficiency gains often risk coming at the expense of validity and alignment with underlying learning theories. These tensions suggest that fully automated annotation is unlikely to be sufficient for theory-driven educational research. Instead, *hybrid human–AI approaches* that strategically combine machine scalability with expert oversight may offer a more viable path forward [9, 19]. The key challenge is therefore not simply automation, but the effective orchestration of human expertise within LLM-supported workflows.

In response, we investigate a human-in-the-loop (HITL) annotation pipeline designed to support large-scale coding of educational datasets while maintaining theoretical rigor. The proposed pipeline integrates active learning techniques (see Section 2.2) to identify uncertain, potentially incorrect, or missing labels and prioritize them for expert review. In parallel, it incorporates loss-weighting strategies that reduce the influence of lower-confidence LLM-generated labels during model training. Together, these mechanisms aim to improve annotation efficiency while preserving validity and theoretical coherence. We evaluate the individual and combined contributions of these components using SRL classification [21] as a case study. SRL provides a particularly suitable testbed because it involves theoretically grounded constructs (e.g., effort regulation and elaboration) that are not directly observable and therefore require interpretative coding. Although the empirical evaluation focuses on SRL, the proposed workflow is intended to generalize to other theory-driven qualitative annotation tasks in education and related domains. This work is guided by the following **research question:**

How much expert involvement is needed for a human-in-the-loop LLM coding pipeline to improve annotation quality over an automated baseline?

This paper makes three **main contributions**:

- a human-in-the-loop pipeline to support a high-quality, hybrid LLM-human coding strategy for SRL prompt classification,
- a comparative evaluation of the effectiveness of the respective components of this pipeline on a dataset,
- practical implications for shifting expert labor from comprehensive and time-intensive coding to targeted supervision and revision of LLM-based coding systems.

2. RELATED WORK

2.1 LLMs for Qualitative Analysis

Recent research has increasingly explored the use of LLMs as tools for qualitative analysis in educational research. This work spans applications such as codebook development [1, 3], deductive coding [18], and the annotation of textual data [9, 29, 32]. Across these studies, LLMs are typically positioned as supports for specific stages of qualitative workflows, with the goal of improving analytical efficiency while maintaining the validity and theoretical grounding of qualitative interpretations.

Evidence from this literature suggests that LLM-supported qualitative coding is most reliable when applied to relatively well-defined and less theoretically complex constructs. As construct complexity increases, model performance becomes more sensitive to how coding tasks are framed and specified. Prompting strategies such as few-shot prompting [17], which combine construct definitions with annotated examples, have therefore been explored to improve alignment between LLM outputs and coding schemes. Under carefully specified conditions and within particular domains, such approaches have enabled LLMs to achieve annotation performance comparable to that of humans [18, 4, 17].

However, limitations become more apparent when LLMs are applied to complex, theory-driven constructs. Prior work reports tendencies toward overgeneralization and mislabeling in such settings, as models may rely on surface-level linguistic cues rather than theoretically grounded categories when assigning codes [17].

These challenges raise concerns about the validity and interpretability of fully automated qualitative coding in theory-intensive educational research.

To mitigate these issues, scholars have begun proposing HITL approaches, in which expert annotation is selectively integrated into LLM-supported workflows [3]. Typically, a subset of the data is manually coded to serve as a ground-truth reference for evaluating LLM-generated labels. Iterative comparisons between human annotations and model outputs can then identify disagreement cases that warrant targeted expert review. Insights from these reviews are used to refine prompts, clarify codebook definitions, and improve alignment between LLM-generated labels and theoretically grounded coding schemes [3].

Despite these advances, important challenges remain. For example, while strong performance on SRL classification has been achieved when models are trained on fully human-coded datasets [33], they often fail to generalize across subject domains or languages. This implies that high accuracy may come at the cost of repeatedly producing large, consistently annotated datasets for each new context. In contrast, more general LLM-based approaches offer greater transferability but may introduce additional noise into generated labels [34].

These tensions highlight the need for approaches that enable systematic validation and refinement of LLM-generated labels without relying on fully human-coded datasets. While prior work has explored HITL strategies primarily through prompt engineering [15] and iterative feedback [3], comparatively little research has examined how LLM-generated annotations can be systematically validated and integrated into theory-driven qualitative coding pipelines at scale.

2.2 Active Learning and Loss Weighting

Active Learning offers a promising approach to maximize the impact of limited human supervision in LLM-based coding tasks. Rather than uniformly reviewing all available in-

stances, an iterative HITL approach is used in which the model actively selects data for annotation based on an uncertainty measurement. A common approach is uncertainty-based sampling, which selects samples with low prediction confidence for human (re)labeling [26].

This selection process is typically embedded within a cycle of model training, evaluation, and targeted relabeling, allowing the training set to be refined progressively.

Hybrid methods that integrate LLM-generated labels with HITL active learning methods are a relatively recent development [24, 14]. These hybrid approaches treat the LLM as a noisy annotator that can scale up to label large amounts of data, while focusing human effort on the relabeling of samples that are likely to have the largest impact on classifier performance. While these methods have shown positive impacts on the performance of general NLP tasks such as topic classification and semantic similarity [14], they have not yet been systematically evaluated in educational contexts.

A complementary approach for managing annotation quality is *loss weighting*, which explicitly accounts for differences in label reliability during model training. Loss weighting assigns increased influence to annotations from more reliable coders (e.g., a human expert), while down-weighting labels generated automatically by LLMs or the model itself. This approach makes it possible to benefit from large volumes of weakly labeled data without allowing noisier labels to dominate the training process [30].

Hybrid LLM-HITL active learning methods, alone or combined with loss weighting, have not been systematically evaluated on the classification of latent, theory-driven educational constructs. Our work aims to fill this gap by evaluating how these methods can reduce annotation costs under a limited budget by focusing expert effort on informative instances and appropriately integrating labels of differing quality, as measured by their impact on the accuracy of a mDeBERTa-v3-base model [8] trained on data processed through different stages of the pipeline.

2.3 SRL for AI-Mediated Learning

To analyze how learners regulate their learning when interacting with generative AI chatbots, the SRL theoretical lens [36, 22] may be adopted. SRL guides learning through metacognitive skills in three main phases. The forethought phase represents the planning stage of learning, in which learners outline goals for a study session. It is followed by the performance phase, where the planned activities are executed. The final phase is self-reflection, in which learners review their performance in relation to the goals and adjust the behavior [36]. To operationalize this three-phase model in the setting of generative AI chatbot use in computer science education, the SRL framework by Pintrich et al. [22] may be adapted [31, 2]. This results in seven SRL categories for the classification of user prompts: rehearsal (adjusted), organization, elaboration, help-seeking, effort regulation, critical thinking, and non-SRL behavior. Table 1 provides definitions for each category.

3. METHODS

Against this background, this study addresses the problem of prioritizing samples for human annotation to identify and correct incorrect labels generated by a weak reviewer (e.g., an LLM) under a fixed budget for human supervision and given an initial set of human-annotated reference labels.

The problem was formulated as a multi-class classification problem with seven mutually exclusive classes, where each instance is assigned at most one categorical label. This dataset reflects a real-world, low-resource scenario, characterized by strong class imbalance, with some classes being represented only by a few labels.

3.1 Datasets

The study draws on multiple datasets representing different cultural contexts, educational settings, and subject domains at higher education institutions in Sweden [31], Germany [25], and South Korea [6]. The subject areas covered include foundational programming, scientific writing, and chemistry. The selected datasets consist of end-to-end, multi-turn dialogues between individual students and an LLM. The data was collected during self-study activities such as homework assistance, reflection, and exam preparation. Each dataset was collected under distinct instructional conditions: in Sweden, 10 students used ChatGPT without explicit instructions, generating 2445 messages. In Germany, 212 students used an unspecified LLM chatbot, generating 2335 messages, and in South Korea, 131 students interacted with a GPT-3.5 through a writing platform (RECIPE [5]), generating 1913 messages. The language used to communicate with the LLM was English in 4179 messages, German in 1600, Swedish in 567, and Korean in 70 (counted after the removal of duplicates). Although the data include complete dialogic interactions, the analysis treats individual student prompts as the unit of analysis to ensure comparability across datasets. Prompts were used verbatim, maintaining orthographic and grammatical errors.

In addition to these empirical datasets, a synthetic dataset was created to address rarely represented SRL constructs. Specifically, the *rehearsal* construct occurred only three times across the empirical data. To supplement this limitation, additional examples were generated using an iterative multi-shot prompting approach with the OpenAI GPT-5.2-2025-12-11 model following Litake et al. [16]. All synthetically generated instances were manually reviewed before inclusion. The prompt used for data generation is provided on OSF. The codebook for the project can be found in Table 1.

3.2 Datasets and Preparation

The different datasets were joined by maintaining the original dataset name, the student prompt, and, if present, the SRL label. Except for the Swedish dataset, the datasets do not include SRL annotations. In addition, we make use of a dataset reported in a master's thesis [37], which contains chat transcripts from self-study activities of Swedish STEM students together with LLM-generated SRL codes. The labels were normalized across datasets, and the source of each label was encoded based on the originating dataset as human-labeled, LLM-labeled, or unlabeled. The data cleaning and preparation consist of four subsequent steps:

1. Text cleaning: All prompts were cleaned before being

Table 1: Codebook of the theoretical lens used.

Construct	Description	Example
Rehearsal	Addresses drill-and-practice behavior, where students repeat learning material to memorize it. In prompt-based interaction, rehearsal may be reflected by repeated or highly similar prompts across learning sessions, including quiz-like activities where the student receives questions from a chatbot.	*"hi, i would like to review about the results of my quiz. I got wrong in this question [...]"*
Organization	Prompts in which students request summarization, support for structuring and editing content, such as spelling correction or writing assistance.	*"Then, what is the repeted word in my essay?"* or *"make this paragraph coherence and better for the grammar[...]"*
Elaboration	Prompts where students ask for additional information to better understand a concept by making connections between ideas.	*"What does Counter do?"* or *"What do linearly recursive, end-recursive, and non-linearly recursive mean?"*
Help-seeking	Prompts in which students directly request solutions or explicit guidance to solve a task.	*"def f4(m, n): if m == 0: return n + 1 elif n == 0: return f4(m - 1, 1) else: return f4(m - 1, f4(m, n - 1)) y = f4(1, 1) What is the return value? Number of calls? What is the recursion type?"*
Effort regulation	Prompts in which students ask for additional resources or explanations aimed at persisting with learning tasks.	*"For example, if my work's limitation is that it only works at the specific domain (graphic design), how can I write my limitation well?"*
Critical thinking	Prompts where students evaluate information, question assumptions, or reflect on the validity of responses.	*"No this is definitively wrong."* or *"Should it not be 5 due to our prior result?"*
Non-SRL behavior	Prompts that cannot reliably be coded as rehearsal, organization, elaboration, help-seeking, effort regulation, or critical thinking. This includes greetings and acknowledgments.	*"nope, thats it. thanks"* or *"Hello"* or *"Great, thanks"*.

further processed. This includes Unicode normalization, removal of control characters, leading and trailing whitespace, line breaks, and the removal of boilerplate tags and phrases originating from accessibility-related user interface components of the chatbot (e.g., file upload prompts).

2. Label normalization: The existing labels were normalized with, e.g., different capitalization, manually reviewed, and joined into canonical labels.

3. Duplicate handling: Rows with missing prompts were removed from the dataset. Exact duplicates, which are defined as a row with an identical combination of text, label, and label sources, were collapsed into a single instance. Rows with identical text but conflicting labels were merged by keeping the human annotation if present. In the case of conflicting LLM-generated labels, the rows were collapsed and marked as unlabeled, dropping the LLM lables .

4. ID Generation and Language Detection: In the last preprocessing step, a UUID was generated for each row using the cleaned text and the dataset name. Using *langid* the language of each prompt was detected, limited to the languages present in the datasets (Swedish, German, English, and Korean).

3.3 Baseline Model

As a baseline, the multilingual mDeBERTa-v3-base model [8] was fine-tuned as a text classifier for SRL label prediction at the prompt level using the Hugging Face Transformers library with the standard sequence classification architecture. Individual student prompts were used as the unit of analysis, with a single categorical SRL label per input. Following an initial hyperparameter tuning, inputs were tokenized using the corresponding pretrained tokenizer and truncated to a maximum sequence length of 128 tokens, with dynamic padding applied during batching. Fine-tuning was performed for five epochs using an AdamW-style optimizer with a learning rate of 3×10^{-5}, a weight decay of 0.01, and a warm-up ratio of 0.06, following a cosine learning-rate schedule. A per-device batch size of two with gradient accumulation over eight steps resulted in an effective batch size of sixteen. All trained checkpoints were retained and manually inspected for analysis, rather than selecting a single checkpoint based on an automated validation criterion. Following Sokolova and Lapalme [27], the evaluation metric used was $F1_{macro}$ due to the class imbalance.

3.4 Loss Weighting

To prioritize human-annotated instances x_i^{Hu}, a per-instance loss weight was introduced. Each training instance x_i is assigned a strictly positive weight $w_i > 0$. Human-annotated samples receive unit weight, while LLM-labeled samples are either assigned a fixed downweighting factor $w_{LLM} < 1$, treated as a tuned hyperparameter, or weighted individually using w_i^{LLM}. A normalization $\sum_i w_i$ with ℓ_i as the standard cross-entropy loss for x_i, limits the effect of weighting to the relative contribution of individual samples, leaving the overall gradient scale unchanged.

$$\mathcal{L} = \frac{\sum_{i=1}^{N} w_i, \ell_i}{\sum_{i=1}^{N} w_i}.$$

3.5 Active Learning for Label Acquisition

Active learning was applied exclusively to the unlabeled subset of the dataset. At each iteration of the label acquisition (LA), an uncertainty score was computed for all unlabeled instances using predictive entropy following [26]. The 50 instances with the highest entropy values were selected for manual annotation. The newly annotated samples were added to the training set, and the model was retrained using the same architecture and hyperparameter configuration as in the baseline setup. This procedure was repeated in iterative cycles until the annotation budget of 200 labels was exhausted.

3.6 Active Learning for Label Correction

To address the varying LLM annotation quality, a weighting approach was applied and evaluated. It is assumed that LLM-labeled instances x_i^{LLM} located close to clusters of human-annotated instances x_i^{Hu} in the model embedding space are more likely to be correct than instances that are further away, as discussed by Kontonatsios et al. [12]. Such instances were prioritized for label correction (LC) by human review.

This approach was implemented using the cosine similarity. For each instance x_i with one label set L_i, an embedding $e_i \in \mathbb{R}^d$ was computed in a shared d-dimensional embedding space using the mDeBERTa transformer encoder [8]. For each SRL label l, a normalized centroid c_l was defined as the mean embedding of all x_i^{Hu} instances assigned to the label l. These centroids were then used to define per-instance weights.

$$w_i^{LLM} = \max\left(\max_{l \in L_i} \exp\left(-\frac{1 - e_i^\top c_l}{\sigma_l}\right), \varepsilon\right),$$

where σ_l describes the median cosine similarity between x_i^{Hu} of label l and its centroid. Human-annotated instances are assigned unit weight ($w_i = 1$), and all weights are lower-bounded by a small constant$w_i \leftarrow \max(w_i, \varepsilon)$.

To identify candidates for human review, the w_i metric was combined with similarity-based indicators defined as

$$\delta = d_{\text{nearest}} - d_{\text{self}}$$

where d_{self} is the cosine similarity between x_i^{LLM} and the centroid c_l that corresponds to the assigned label, and d_{nearest} is the similarity to the closest centroid of a non-assigned label. A larger value of δ indicates a candidate for mislabeling. Based on the w_i metric, a review flag was assigned when w_i was in the lowest decile and prioritized by δ.

3.7 Evaluation Study

An evaluation study was concluded to assess the effectiveness of loss weigting, active learning through label acquisition (LA), and active learning for label correction (LC). For this purpose, after the initial cleaning described in section 3.2, a baseline dataset was constructed from the existing human-labeled instances x_i^{Hu}, the machine-labeled instances x_i^{LLM}, and unlabeled instances $x_i^{\emptyset}$. The resulting dataset includes 6416 samples with 836 x_i^{Hu}, 2016 x_i^{LLM} and 3564 $x_i^{\emptyset}$.

This dataset was used to train the baseline model and was iteratively extended in batches of 50 instances through additional human annotation and re-annotation of x_i^{LLM} instances. In total, this process resulted in ten distinct datasets summarized in Table 2.

An initial split was defined using only the $\boldsymbol{x_i^{\text{Hu}}}$ instances for the test and validation sets, while for the training set, the x_i^{LLM} and remaining $\boldsymbol{x_i^{\text{Hu}}}$ were used. The split between training, validation, and test sets was 80/10/10, while ensuring that all classes were present in the validation and test datasets. We first randomly select the latter sets and use the rest of the data to fill up the training set. The training set was iteratively improved by adding human labels to unlabeled instances (LA approach) or correcting LLM-labeled instances (LC approach), while the validation and test sets were unchanged.

Table 2: Dataset composition across annotation strategies: label acquisition (LA) and label correction (LC). (50–200 = human additional annotations or reviews).

	Dataset	$\boldsymbol{x_i^{\text{LLM}}}$	$\boldsymbol{x_i^{\text{Hu}}}$	$\boldsymbol{x_i^{\emptyset}}$
	Baseline	2016	836	3564
Iteration 1	LA 50	2016	886	3514
Iteration 2	LA 100	2016	936	3464
Iteration 3	LA 150	2016	986	3414
Iteration 4	LA 200	2016	1039	3364
Iteration 1	LC 50	1966	886	3564
Iteration 2	LC 100	1916	936	3564
Iteration 3	LC 150	1866	986	3564
Iteration 4	LC 200	1816	1036	3564
	Combined	1816	1236	3164

4. RESULTS

As indicated in Table 3, a macro-F1 score of 0.328 and an accuracy of 0.352 are achieved for the baseline model. For the baseline dataset, instance-based loss weighting w_i leads to improved performance compared to the unweighted, but this is not the case for the augmented datasets LA-200 and LC-197. For these, a fixed assignment weight, determined via grid search, instead results in higher performance. The weight increased with the number of iterations during which labels were added or corrected, indicating that the dataset quality improved.

Table 3: Performance across iterations, strategies, and weighting (none, per-instance, and best constant $w^\star$).

Iteration	Dataset	Loss weighting	Macro F1	Macro Recall	Accuracy
	Baseline	none	0.328	0.323	0.352
	Baseline	per ins.	0.350	0.377	0.383
	Baseline	$w^\star = 0.4$	0.371	0.396	0.379
1	LA-50	none	0.365	0.400	0.380
2	LA-100	none	0.382	0.407	0.393
3	LA-150	none	0.327	0.353	0.352
4	LA-200	none	0.371	0.395	0.380
4	LA-200	per ins.	0.329	0.349	0.440
4	LA-200	$w^\star = 0.3$	**0.402**	**0.418**	0.490
1	LC-50	none	0.324	0.355	0.419
2	LC-100	none	0.320	0.344	0.407
3	LC-150	none	0.350	0.345	0.457
4	LC-197	none	0.366	0.391	0.453
4	LC-197	per ins.	0.334	0.366	0.430
4	LC-197	$w^\star = 0.9$	0.375	0.398	0.467
4	Combined	none	0.386	0.405	0.494
4	Combined	per ins.	0.326	0.354	0.448
4	Combined	$w^\star = 0.9$	0.395	0.412	**0.498**

For active learning through LA, macro-F1 scores increase across successive LA iterations up to LA-200, except for a performance drop for LA-150, for which $F1_{Macro}$ falls below the baseline, likely reflecting sensitivity to sampling effects, class imbalance, and label noise [26]. For LA, instance-based loss weighting does not provide additional gains, whereas a fixed loss weight resulted in the highest overall $F1_{Macro} = 0.402$ for LA-200.

Similar results are observed for active learning through LC,

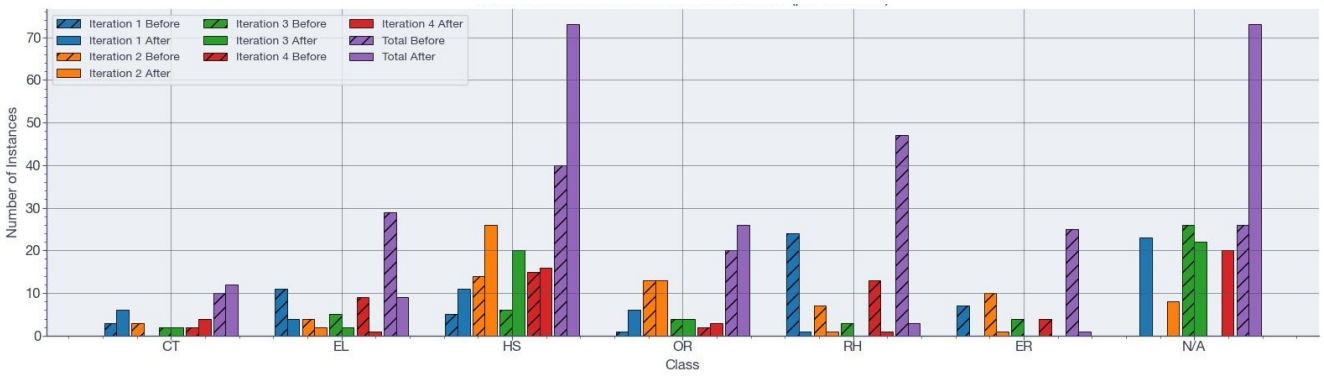


Figure 1: Class-wise label distribution for each LC Iteration and in Total, with Before/After bars indicating the labels among the 50 reviewed instances for that iteration. CT = Critical Thinking, EL = Elaboration, HS = Help Seeking, OR = Organization, RH = Rehearsal, ER = Effort Regulation, and N/A = no applicable SRL label.

outperforming the baseline from LC-150 and up. As in the LA setup, instance-based weighting performs worse than no weighting, and a fixed loss weight performs the best. Across the four LC iterations, 197 instances (out of 200, three samples were inadvertently removed due to a procedural oversight but were identified during the final write-up) were selected for review. Of these, 120 instances (60.9%) were identified as having been incorrectly labeled and were corrected. The distribution of corrected labels indicates an overgeneralization within the LLM-generated labels, with SRL labels having been assigned by the LLM, although the content did not align with any construct in the codebook. The absolute number of *Non-SRL* was hence drastically increased (+48) after label correction. In addition, the numbers of *Help seeking* (+34), *Organization* (+7), and *Critical Thinking* (+2) labels were also increased, while those of *Rehearsal* (-44), *Effort Regulation* (-25), and *Elaboration* (-20) were decreased. This pattern was consistent across all relabeling iterations and also corresponds to the shares in the newly labeled data in the LA iterations. The number of identified labeling errors decreased across successive iterations, from 42 (out of 50) in the first iteration to 28 in the second, 24 in the third, and 26 in the final iteration. Figure 1 summarizes the class-wise label distribution before and after reviews.

The combination of 200 newly labeled instances and 197 revised instances results in slightly higher overall accuracy compared to either the LA or LC configurations alone. A fixed loss weight provides performance close to the best-performing configuration, while instance-based weighting again performs poorly in $F1_{Macro}$.

5. DISCUSSION

LLMs have the potential to contribute meaningfully to scaling the labor-intensive process of qualitative coding of educational data when positioned as provisional labelers. Their outputs function as 'weak' and potentially noisy signals that require systematic human review rather than serving as trusted sources of annotation [35]. Addressing the research question, performance, naturally, normally increases with progressive human addition or correction of labels. In our experiment, the best results were obtained when adding 200 more human labels (LA-200) or combining addition and correction (Combined). Adding or reviewing fewer labels did not guarantee outperforming the baseline, as shown by LA-150, LC-50 and LC-100 (no weights). Reviewing 200 LLM annotations (10% of LLM labels) or adding 200 human annotations (24% increase) to a dataset that still consists of twice as many LLM-labeled and thrice as many non-labeled instances is a reasonable effort to improve performance (respectively, +11.6% and +13.1% in $F1_{Macro}$). HITL expert involvement is hence both necessary and effective in complementing or correcting LLM-generated labels. These findings extend prior work on utilizing LLMs for qualitative coding tasks [3, 17] with an approach to address limitations of overgeneralization.

Consistent patterns in how the proposed pipeline affects model performance were revealed. Active learning through LA shows overall performance gains under a fixed annotation budget, even if fluctuations occur in the LA-150 iteration, while active learning through LC improved results compared to the baseline from LC-150 and up. Without loss weighting, LC-197 is at par with LA-200 (0.366 vs. 0.371), suggesting that, under a fixed human annotation budget, correcting existing 'weak' labels can be as effective as expanding the dataset (note that 120 – not 200 – labels were corrected in LC-197). The higher label quality reduces the need for strong downweighting of LLM-generated labels ($w* = 0.9$ for LC vs. $w* = 0.3$ for LA).

Fixed loss weighting improves performance and indicates that LLM-generated labels can provide useful information if their imperfection is taken into account. Instance-based weighting worsened performance, suggesting that it relies on similarity estimates in a learned embedding space optimized for classification rather than geometric accuracy, making cosine similarity sensitive to regularization and feature scaling effects [28]. As a result, the corresponding weighting signals can become unstable during training. Distance-weighted cosine similarity is suggested to mitigate these effects [13].

This study has several limitations related to the scope of the empirical evaluation. The analysis is restricted to a single domain, which limits the generalizability of the findings to other domains. In addition, the evaluation considers only a single model architecture. While the proposed pipeline

explicitly addresses label noise and class imbalance, its effectiveness for rare constructs remains constrained by the availability of reliable human annotations. Furthermore, the study does not explicitly account for human factors such as reviewer fatigue or inter-rater reliability, which may influence the robustness of HITL annotation processes. Finally, we acknowledge that the use of existing datasets from different cultural and task contexts may limit the generalizability due to cross-context differences that may lead to different student–LLM interaction patterns.

Future work should examine the applicability of the proposed pipeline across additional domains, modeling settings, and annotation budgets. More generally, the interaction between automated labeling, human review, and data quality warrants further investigation beyond fully supervised annotation regimes. To further validate LLM-generated labels, the use of cosine similarity as an additional signal alongside inter-rater reliability (kappa) should be explored, which may help assess label consistency and improve robustness in diverse contexts. Such pipelines can be embedded into existing qualitative coding workflows to support targeted human review, building on prior work [3, 34].

The findings indicate that the effectiveness of LLM-supported annotation in educational research depends on structured human participation. Although LLM-generated labels provide a helpful boost, loss weighting combined with explicit mechanisms to involve human experts for label correction provides additional value. Improving the quality of existing weak labels via human involvement can be similarly effective as acquiring new labels. Taken together, these results show that a human-in-the-loop pipeline that treats LLM-generated labels as 'weak' signals and prioritizes targeted human correction can scale theory-driven annotation under a fixed annotation budget while preserving theoretical rigor.

Ethics Statement: In this study, we followed the ethical guidelines provided by the Swedish Ethical Review Authority (https://etikprovningsmyndigheten.se/en/).

Acknowledgements: This work has been in part supported by the EECS school (KTH) within the project: "Think for Yourself: Adaptive AI Technologies for Learning with LLMs in STEM Education". This work was also partially funded by an unrestricted gift from Google.

6. REFERENCES

[1] A. Barany, N. Nasiar, C. Porter, A. F. Zambrano, A. L. Andres, D. Bright, M. Shah, X. Liu, S. Gao, J. Zhang, et al. Chatgpt for education research: Exploring the potential of large language models for qualitative codebook development. In *International conference on artificial intelligence in education*, pages 134–149. Springer, 2024.

[2] G. Cheng, D. Zou, H. Xie, and F. L. Wang. Exploring differences in self-regulated learning strategy use between high- and low-performing students in introductory programming: An analysis of eye-tracking and retrospective think-aloud data from program comprehension. *Computers and Education*, 208:104948, Jan. 2024.

[3] Z. O. Dunivin. Scaling hermeneutics: a guide to qualitative coding with llms for reflexive content analysis. *EPJ Data Science*, 14(1):28, Apr. 2025.

[4] A. Ganesh, C. Chandler, S. D'Mello, M. Palmer, and K. Kann. Prompting as panacea? a case study of in-context learning performance for qualitative coding of classroom dialog. In *Proceedings of the 17th International Conference on Educational Data Mining*, page 835–843, 2024.

[5] J. Han, H. Yoo, Y. Kim, J. Myung, M. Kim, H. Lim, J. Kim, T. Y. Lee, H. Hong, S.-Y. Ahn, and A. Oh. Recipe: How to integrate chatgpt into efl writing education. In *Proceedings of the Tenth ACM Conference on Learning @ Scale*, L@S '23, page 416–420, New York, NY, USA, 2023. Association for Computing Machinery.

[6] J. Han, H. Yoo, J. Myung, M. Kim, T. Y. Lee, S.-Y. Ahn, and A. Oh. RECIPE4U: Student-ChatGPT interaction dataset in EFL writing education. In N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13666–13676, Torino, Italia, May 2024. ELRA and ICCL.

[7] K. Handa, D. Bent, A. Tamkin, M. McCain, E. Durmus, M. Stern, M. Schiraldi, S. Huang, S. Ritchie, S. Syverud, K. Jagadish, M. Vo, M. Bell, and D. Ganguli. Anthropic education report: How university students use claude, 2025.

[8] P. He, J. Gao, and W. Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *CoRR*, abs/2111.09543, 2021.

[9] Z. He, S. Naphade, and T.-H. K. Huang. Prompting in the dark: Assessing human performance in prompt engineering for data labeling when gold labels are absent. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–33, 2025.

[10] K. Huang, R. Ferreira Mello, C. Pereira Junior, L. Rodrigues, M. Baars, and O. Viberg. That's what roberta said: Explainable classification of peer feedback. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 880–886, 2025.

[11] S. A. Karabenick and J.-L. Berger. Help seeking as a self-regulated learning strategy. *Applications of self-regulated learning across diverse disciplines: A tribute to Barry J. Zimmerman*, 1:237–261, 2013.

[12] G. Kontonatsios, A. J. Brockmeier, P. Przybyła, J. McNaught, T. Mu, J. Y. Goulermas, and S. Ananiadou. A semi-supervised approach using label propagation to support citation screening. *Journal of Biomedical Informatics*, 72:67–76, Aug. 2017.

[13] B. Li and L. Han. Distance weighted cosine similarity measure for text classification. In H. Yin, K. Tang, Y. Gao, F. Klawonn, M. Lee, T. Weise, B. Li, and X. Yao, editors, *Intelligent Data Engineering and Automated Learning – IDEAL 2013*, page 611–618, Berlin, Heidelberg, 2013. Springer.

[14] M. Li, T. Shi, C. Ziems, M.-Y. Kan, N. Chen, Z. Liu, and D. Yang. CoAnnotating: Uncertainty-guided work

allocation between human and large language models for data annotation. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1487–1505, Singapore, Dec. 2023. Association for Computational Linguistics.

[15] T. Li, D. Sree, and T. Ringenberg. Assessing crowdsourced annotations with llms: Linguistic certainty as a proxy for trustworthiness. In *Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities*, pages 191–201, 2025.

[16] O. Litake, B. H. Park, J. L. Tully, and R. A. Gabriel. Constructing synthetic datasets with generative artificial intelligence to train large language models to classify acute renal failure from clinical notes. *Journal of the American Medical Informatics Association*, 31(6):1404–1410, June 2024.

[17] X. Liu, J. Zhang, A. Barany, M. Pankiewicz, and R. S. Baker. Assessing the potential and limits of large language models in qualitative coding. In *International Conference on Quantitative Ethnography*, pages 89–103. Springer, 2024.

[18] J. McClure, D. Smyslova, A. Hall, and S. Jiang. Deductive coding's role in ai vs. human performance. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 809–813, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[19] K. Misiejuk, R. Kaliisa, and J. Scianna. Augmenting assessment with ai coding of online student discourse: A question of reliability. *Computers and Education: Artificial Intelligence*, 6:100216, 2024.

[20] E. Panadero. A review of self-regulated learning: Six models and four directions for research. *Frontiers in psychology*, 8:422, 2017.

[21] P. R. Pintrich. The role of motivation in promoting and sustaining self-regulated learning. *International journal of educational research*, 31(6):459–470, 1999.

[22] P. R. Pintrich, D. A. Smith, T. Garcia, and W. J. McKeachie. *A Manual for the Use of the Motivated Strategies for Learning Questionnaire (MSLQ)*. National Center for Research to Improve Postsecondary Teaching and Learning, Ann Arbor, MI., 1991. ERIC Number: ED338122.

[23] I. Robinson and J. Burden. Framing the game: How context shapes llm decision-making. *arXiv preprint arXiv:2503.04840*, 2025.

[24] H. Rouzegar and M. Makrehchi. Enhancing text classification through LLM-driven active learning and human annotation. In S. Henning and M. Stede, editors, *Proceedings of the 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 98–111, St. Julians, Malta, Mar. 2024. Association for Computational Linguistics.

[25] A. Scholl and N. Kiesler. Data: Analyzing Chat Protocols of Novice Programmers Solving Introductory Programming Tasks with ChatGPT, 5 2024. Accessed: 2026-05-05.

[26] B. Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Springer International Publishing, Cham, 2012.

[27] M. Sokolova and G. Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, July 2009.

[28] H. Steck, C. Ekanadham, and N. Kallus. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 887–890, New York, NY, USA, May 2024. Association for Computing Machinery.

[29] L. Tavakoli and H. Zamani. Reliable annotations with less effort: Evaluating llm-human collaboration in search clarifications. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, pages 92–102, 2025.

[30] J. Terven, D.-M. Cordova-Esparza, J.-A. Romero-González, A. Ramírez-Pedraza, and E. A. Chávez-Urbiola. A comprehensive survey of loss functions and metrics in deep learning. *Artificial Intelligence Review*, 58(7):195, Apr. 2025.

[31] O. Viberg, J. Wong, Y. Feldman-Maggor, N. Dunder, and C. D. Epp. Chatting with code: Exploring llms as learning partners in programming education. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education*, pages 453–461, Cham, 2025. Springer Nature Switzerland.

[32] C. Y. Wang and J. J. H. Lin. Utilizing artificial intelligence to support analyzing self-regulated learning: A preliminary mixed-methods evaluation from a human-centered perspective. *Computers in Human Behavior*, 144:107721, July 2023.

[33] J. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using large language models to detect self-regulated learning in think-aloud protocols. In B. PaaÄŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 157–168, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[34] J. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using large language models to detect self-regulated learning in think-aloud protocols. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 157–168, 2024.

[35] Y. Zhang, S. A. Khan, A. Mahmud, H. Yang, A. Lavin, M. Levin, J. Frey, J. Dunnmon, J. Evans, A. Bundy, S. Dzeroski, J. Tegner, and H. Zenil. Exploring the role of large language models in the scientific method: from hypothesis to discovery. *NPJ Artificial Intelligence*, 1(1):14, Aug. 2025.

[36] B. J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory Into Practice*, 41(2):64–70, May 2002. Publisher: Routledge _eprint: https://doi.org/10.1207/s15430421tip4102_2.

[37] S. Özdere. Chatwise: Adaptive support for students' self-regulated learning in generative ai–mediated environments. Master's thesis, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden, Dec. 2025.

A Three-GenAI Consensus Model for Deductive Analysis of Response Data

Amruth N. Kumar
Ramapo College of NJ
Mahwah, New Jersey
USA
amruth@ramapo.edu

Renée McCauley
College of Charleston
Charleston, South Carolina
USA
mccauleyr@cofc.edu

Bonnie K. MacKellar
St. John's University
New York, New York
USA
mackellb@stjohns.edu

Tammy VanDeGrift
University of Portland
Portland, Oregon
USA
vandegri@up.edu

ABSTRACT

In deductive content analysis, text data is categorized into previously developed themes. We present a three-GenAI consensus model for deductive analysis of text data. In the model, the same prompt and data are provided to three GenAIs, and their agreement with each other is used to cluster data to facilitate deductive coding by researchers. First, data is split into trial and corpus sets. Trial data is used to iteratively revise the prompt provided to the three GenAIs. Next, the revised prompt is provided to the three GenAIs to deductively code the larger corpus data. Based on the responses of the three GenAIs, data is clustered into three groups: all three GenAIs agree, only 2 GenAIs agree, and all three GenAIs disagree. The data in the three clusters is separately coded by researchers, with the GenAI coding in each cluster used to narrow coding choices for the data in that cluster. To illustrate the use of the three-GenAI consensus model, we used it to deductively analyze 496 student responses to a self-directedness survey that had been previously coded by researchers. We calculated that the researchers would have saved nearly 70% of coding time if they had used the three-GenAI consensus model first. Deductive analysis of text data is labor-intensive, involving multiple researchers and multiple iterations. The three-GenAI consensus model enables researchers to redistribute time from exhaustively analyzing all data to closely reviewing only the data that does not clearly fit any of the themes of deductive analysis.

Keywords

Qualitative analysis, Deductive analysis, Generative AI, GenAI, Three-GenAI model

Amruth Kumar, Renée McCauley, Bonnie MacKellar, and Tammy Vandegrift. A Three-GenAI Consensus Model for Deductive Analysis of Response Data. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 408–414. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039843

1. INTRODUCTION

Qualitative content analysis is used to explore meanings, themes, and patterns in text, allowing researchers to understand social reality in a subjective, but scientific manner [25]. Research questions in qualitative investigations aim to probe the "what", "why", "who", "when", "where", and "how" of social interactions or human behavior, rather than quantifying occurrences or measurements to test hypotheses [10, 14]. Qualitative content analysis can be inductive or deductive [2]. In inductive (bottom-up) methods, researchers identify and describe themes that emerge during an iterative process of analyzing bodies of text. In deductive (top-down) methods, researchers use previously developed sets of themes or models or theories as a coding scheme/book to analyze large bodies of text, attempting to fit the data into existing themes. Deductive coding can be binary (whether a theme exists or not) or multi-level (identify one of several themes).

Qualitative analysis techniques are iterative in nature, involving multiple passes through the data to build a deeper, more refined understanding of the subject matter [19]. For example, in deductive coding, data analysis can result in the refinement of existing themes as well as the identification of new themes, which can lead to the recoding of already coded data. For large data sets, this iterative process is labor-intensive [20], and involves the time-consuming work of abstraction and interpretation [11].

Research teams doing qualitative data analysis must collaboratively review notes and do close readings of text [17]. Many of the software tools available for qualitative data analysis (such as ATLAS, Dedoose, NVivo) assist with data organization only and not with the data interpretation done by the research team. Furthermore, they are based on a subscription model that may not be within reach for all researchers. With the recent advent of powerful generative AI tools (GenAI) that can quickly find patterns in large bodies of text, researchers now have a new tool to conduct qualitative analysis. A drawback of GenAI is that they are prone to hallucinate, which reduces the reliability of their responses.

In order to reliably use GenAIs for deductive analysis, we propose a three-GenAI consensus model that uses three GenAIs

instead of one, uses the consensus in their responses instead of raw responses, and uses the consensus to narrow choices for deductive coding by researchers. We present the savings in time realized using this model and present a case study to illustrate its use and highlight its savings.

2. RELATED WORK

Qualitative analysis has been used extensively in computing education research. It has been used to identify themes in code interviews [9], student reflection papers [3], survey responses [6], online forum discussions [15], transcripts of think-aloud interviews [16, 23] and semi-structured interviews [8], among others. Typically, qualitative analysis involves processing text data, and involves multiple researchers engaged in a labor-intensive iterative process. As a result, qualitative analysis is hard to scale to larger data sets because of the time, cost and complexity involved.

Machine learning (ML) techniques have been used to process educational data (e.g., see a systematic literature review in [5]). Techniques used to analyze text data include latent semantic analysis [22], bag-of-words [7], etc. GenAI improves upon these traditional ML techniques in that it does not require significant training.

Given that GenAI has been publicly available for just over three years, the use of GenAI for qualitative coding is new. GenAIs have been used for both inductive and deductive coding. In one study, ChatGPT was used to help in codebook development during an inductive-deductive process [21]. In another study, ChatGPT-4 was found to be valuable for inductively developing codebooks, but human involvement in the process was deemed essential [1].

For deductive coding, prompts based on codebook were found to be more effective than those based on examples - combining GenAI (GPT-3) with expert-drafted codebooks achieved fair to substantial agreement with expert-coded results [24]. In another study, codebooks designed for researchers were found to need iterative refining for use with GenAIs [4]. For binary deductive coding, zero-shot prompting, i.e., not providing examples in the prompt, yielded good results for well-defined constructs, but was not as effective when the constructs were context-dependent [12, 13].

We propose a model to use three GenAIs for deductive coding. In the model, we propose to use a prompt based on the codebook designed for researchers. We use it in zero-shot mode, although this is not required for the model to work. The model narrows coding choices and helps researchers redistribute time from exhaustively analyzing all data to closely reviewing only the data that does not clearly fit any of the themes of deductive analysis.

3. THE THREE-GENAI CONSENSUS MODEL

In deductive qualitative analysis or coding, researchers use a previously developed set of themes, called a codebook, to identify the theme in each data item. In this text, we will refer to a data item as a statement, and a theme as a category.

We propose to use three GenAIs to aid in deductive multi-level coding, in what we refer to as the *three-GenAI consensus model*, as follows:

1. Shuffle the data and split it into a trial set (10-15% of the data) and corpus.
2. Draft a codebook-based prompt. Use an iterative cycle of deductively coding trial data using the three GenAIs, followed by revising the prompt provided to them, until the deductive coding of trial data is satisfactory in terms of inter-rater reliability with researchers.
3. Use the revised prompt to have the three GenAIs code the statements in the corpus data.

For both trial and corpus data, the consensus among the three GenAIs is used to split the data into three clusters:

- *Unanimous*, the set of statements that all three GenAIs code/categorize the same;
- *2-agree*, the set of statements that exactly two GenAIs categorize the same - this excludes the statements in unanimous cluster;
- *3-disagree*, the set of statements that all three GenAIs categorize differently.

The three clusters are used by researchers for deductive coding as follows:

- Unanimous: Researchers first conduct binary coding of each statement to check whether it belongs in the category identified unanimously by all three GenAIs. If not, then and only then do they consider the remaining categories for coding the statement;
- 2-agree: Researchers first consider the two categories identified by the three GenAIs: one by two concurring GenAIs and the other by the dissenting GenAI. If neither category applies to the statement, they consider the remaining categories for coding the statement.
- 3-disagree: Researchers first consider the three categories identified by the three GenAIs and consider the rest of the categories only if these three do not apply to the statement.

The utility of this model depends on whether the correct category of a statement is among the one/two/three categories identified by the three GenAIs in the three clusters.

In the three-GenAI consensus model, note that:

1. Three GenAIs are used instead of one to reduce the effect of hallucinations on the outcome.
2. The consensus in the responses of the three GenAIs is leveraged.

3. The clusters obtained based on the degree of consensus among the three GenAIs are used to reduce the number of categories researchers first consider when coding statements.

4. The final categorization of statements is still done by researchers and not GenAIs. The three-GenAI model is used to facilitate the qualitative coding done by researchers.

We will next present a case study of using the three-GenAI consensus model for multi-level deductive coding. In order to illustrate its feasibility and merit, we will use it to verify coding previously done by researchers. The case study will provide an example of using the model and its utility in facilitating deductive coding in practical settings.

4. CASE STUDY

For our case study, we used data from a study of the self-directed disposition of undergraduate computing students [18]. The study was conducted between 2022 and 2024 in five computing courses at four higher education institutions in the US. Data was collected from 264 students who consented to participate in the study. The study was approved by the institutional review boards of all four institutions (IRB-FY2022-53, IRB-FY2022-6, Protocol# 2021-08-005, and IRB-627).

In the study, students were presented the following definition of being self-directed: "Learn new tools/techniques on your own to complete a task, even when the tool/technique is only minimally used or discussed in class, and you may not receive additional credit just for learning it." They were then asked to respond to an open-ended question: "Describe an example of you being self-directed when completing this assignment. Otherwise, describe the circumstances that prevented you from being self-directed." Students were asked to respond to this question after multiple assignments in a course.

A total of 496 responses were collected and deductively coded using the codebook shown in Table 1 developed by researchers [18]. This coding by researchers (henceforth referred to as *prior coding*) was used to verify the correctness of coding of the three-GenAI consensus model and for calculating the time saved by using it.

We used three GenAIs to which we had ready access: Claude Sonnet 4.5, ChatGPT 5.2 Pro and Gemini 3. The prompt statement provided to the GenAIs was structured as follows:

1. **Persona:** GenAI was asked to act as a "qualitative analysis expert".

2. **Goal:** The goal was to conduct deductive coding of statements written by undergraduate students in computing courses in response to the open-ended question listed earlier in this section.

3. **Context:** The definition of self-directed disposition given to students [18] was listed next.

4. **Instructions:** GenAI was asked to identify one of the first eight categories in the codebook (Table 1) for each statement.

Table 1: Codebook for *self-directed* disposition [18]

Category	Description
Utilizing external resources	Selecting additional material or reaching out to people to support one's own learning
Learning necessary material	Demonstrating a process to learn new material, including concepts, tools, techniques
Working independently	Doing the problem or task without assistance from others
Assessing oneself	Recognizing one's own capabilities, deficits, or lack of expertise, and appropriateness of learning strategies
Planning ahead	Planning actions before execution, either in terms of time or designing sub-steps
Applying useful techniques	Applying techniques that are helpful with learning and doing the work
Completing the assigned work	Achieving the goal of completing the work
Reviewing against expectations	Reviewing one's own actions and outcomes against provided expectations, guidelines, or goals
Not applied	Factors prevented student from being self-directed
Not pertinent	Insufficient information to indicate self-directed behavior

5. **Codebook:** The entire codebook was provided. GenAI was asked to consider the categories in the order listed in the codebook, as was done by researchers during their coding.

6. **Constraints:** GenAI was further instructed to do the following:

   (a) If a statement belonged to more than one category, identify the dominant category as determined by intent.

   (b) If a statement did not contain any category, code it as "Not pertinent".

   (c) If a statement explained why self-directed was not applied, code it as "Not applied".

   (d) Ignore the word "self-directed" in a statement unless the statement is about not being self-directed.

7. **Purpose:** GenAI was told that its analysis would be used by computing education researchers to summarize how students understood self-directed disposition in undergraduate computing courses.

The prompt was zero-shot—no examples were provided for the categories—to avoid introducing potential biases based on those examples.

The 496 responses were shuffled and split into 74 trial statements (15%) and 422 corpus statements. The same prompt was provided to all three GenAIs and they were instructed to deductively code the 74 trial statements. Based on the responses of the three GenAIs, we clustered the 74 trial statements into unanimous, 2-agree and 3-disagree, as shown in Table 2:

- Unanimous cluster consisted of 52 trial statements (70.3% of trial data). The three GenAIs identified the same category as prior coding for 42 of the 52 statements

Table 2: Agreement among the three GenAIs

Data	Total	Unanimous	2-Agree	3-Disagree
Trial	74	52 (70.3%)	18 (24.3%)	4 (5.4%)
Corpus	422	289 (68.5%)	109 (25.8%)	24 (5.7%)
Total	496	341 (68.8%)	127 (25.6%)	28 (5.6%)

Table 3: Trial Statement Clusters

Cluster	Total	Same as Prior Coding
Unanimous	52	42 (80.8%)
2-Agree	18	6 + 5 (61.1%)
3-Disagree	4	1 (25%)
Total	74	54 (73%)

(80.8%), as shown in Table 3. The inter-rater reliability between the three GenAIs and prior coding was substantial (Cohen's kappa = 0.7508).

- 2-agree cluster contained 18 trial statements (24.3% of trial data). As shown in Table 3, the two concurring GenAIs identified the same category as prior coding for 6 statements and the dissenting GenAI identified the same category as prior coding for 5 more. In 61.1% of the 2-agree cases, the correct category was one of the two identified by the GenAIs.
- Only 4 out of 74 trial statements were in the 3-disagree cluster. For one of the 4 statements, one of the three GenAIs identified the same category as prior coding, as shown in Table 3.

Given the large percentage of trial statements in unanimous cluster (70.3%) and the substantial inter-rater reliability of coding in this group (0.7508), we chose not to iteratively improve the prompt. We provided the same prompt to the three GenAIs and instructed them to deductively code the 422 corpus statements. The cluster sizes for corpus statements are shown in Table 2.

- Unanimous cluster consisted of 289 corpus statements (68.5% of corpus data). The three GenAIs identified the same category as prior coding for 238 of the 289 statements (82.4%), as shown in Table 4. The inter-rater reliability between the three GenAIs and prior coding was substantial, with Cohen's kappa of 0.7678.
- 2-agree cluster contained 109 corpus statements (25.8% of corpus data). As shown in Table 4, the two concurring GenAIs identified the same category as prior coding for 57 statements and the dissenting GenAI identified the same category as prior coding for 32 more. In 81.7% of the 2-agree cases, the correct category was one of the two identified by the GenAIs.
- Only 24 out of 422 corpus statements were in the 3-disagree cluster. For 20 out of the 24 statements (83.3%), one of the three GenAIs identified the same category as prior coding, as shown in Table 4.

Table 4: Corpus Statement Clusters

Cluster	Total	Same as Prior Coding
Unanimous	289	238 (82.4%)
2-Agree	109	57 + 32 (81.7%)
3-Disagree	24	20 (83.3%)
Total	422	347 (82.2%)

Note that in all three corpus clusters, the correct category was one of those identified by the GenAIs for at least 80% of the statements.

For the statements for which the GenAIs did not identify the prior coding category, we conducted an additional validation step: we had two researchers independently re-identify the category, and compared their 1 - 2 (for 2 researchers) categories against prior coding category as well as the 1-3 categories identified by GenAIs in unanimous/2-agree/3-disagree clusters. The category identified by the two researchers was one of the 1-3 categories identified by GenAIs instead of the prior coding category in 52 - 61% of the statements in unanimous cluster (the range based on the two researchers' independent coding), 74 - 89% of 2-agree cluster and 71 - 100% of 3-disagree cluster. This suggests that the statements for which the three GenAIs did not identify the prior coding category were harder to categorize for researchers too.

In the next section, we will calculate the time saved by using the three-GenAI consensus model for qualitative coding.

5. TIME SAVINGS

Assume the data set contains s statements (data items), the codebook contains c categories (themes), and during multi-level deductive coding, it takes one unit of time to consider one category for one statement. Therefore, full-fledged deductive coding of s statements takes $s * c$ units of time.

If n_u, n_a and n_d are the number of statements in unanimous, 2-agree and 3-disagree clusters, respectively:

$$s = n_u + n_a + n_d$$

If p_u is the percentage of statements in unanimous cluster whose GenAI categorization is correct, the time taken t_u to categorize all the statements in the cluster is

$$t_u = n_u * (p_u * 1 + (1 - p_u) * c)$$

For p_u percentage of statements in unanimous cluster, researchers need to consider only one category. For the remaining $(1 - p_u)$ percentage of statements, after confirming that the category chosen by the GenAIs is incorrect, researchers must consider the remaining $(c - 1)$ categories.

Similarly, if p_a is the percentage of statements in 2-agree cluster for which the correct category is one of the two picked by the three GenAIs, the time taken t_a to categorize all the statements in the cluster is:

$$t_a = n_a * (p_a * 2 + (1 - p_a) * c)$$

Finally, if p_d is the percentage of statements in 3-disagree cluster for which the correct category is one of the three

picked by the three GenAIs, the time taken t_d to categorize all the statements in the cluster is:

$$t_d = n_d * (p_d * 3 + (1 - p_d) * c)$$

The time needed to deductively code using the three-GenAI consensus model, as a percentage of the time needed to code without the model is:

$$\frac{t_u + t_a + t_d}{s * c}$$

The percentage reduction in time is given by:

$$\frac{n_u * p_u * (c-1) + n_a * p_a * (c-2) + n_d * p_d * (c-3)}{s * c} \quad (1)$$

The larger the number of categories c, and the greater the percentages p_u, p_a, and p_d, the larger the savings in time. Note that the savings are *per researcher*, and can be substantial considering that typically, three or more researchers are involved in deductive coding.

In the case study in Section 4, combining trial and corpus data, we have from Table 2:

$$n_u = 341, \quad n_a = 127, \quad n_d = 28, \quad s = 496$$

From Table 1, we have $c = 10$. From Tables 3 and 4, we have:

$$p_u = \frac{42 + 238}{52 + 289} = 0.82$$

$$p_a = \frac{11 + 89}{18 + 109} = 0.79$$

$$p_d = \frac{1 + 20}{4 + 24} = 0.75$$

Note that p_u, p_a, and p_d are all large: at least 75%, one of the requirements for large savings in time. If the three-GenAI consensus model had been used for prior coding of the case study dataset, the time spent on coding could have been reduced by 69.88% *per researcher* involved in prior coding.

6. DISCUSSION

We proposed the three-GenAI consensus model for multi-level deductive coding of qualitative data. We used a codebook and prior coding of a dataset of 496 statements to demonstrate the use of the model for deductive coding and the savings in coding time that can be realized using the model. In the future, we plan to start with a codebook, use the three-GenAI model to cluster data into unanimous, 2-agree and 3-disagree, and then deductively code statements in the three clusters.

The percentage reduction in time (expression 1, Section 5) is a theoretical maximum. Using the model requires the following preparatory steps, all of which take additional time and effort: designing the prompt, splitting dataset into trial and corpus, using trial data to iteratively refine the prompt, using the refined prompt to deductively code corpus data, and tabulating the responses of three GenAIs into unanimous, 2-agree and 3-disagree clusters for both trial and corpus data. However, for large data sets, the time and effort needed for these preparatory steps will be largely offset by the time saved by researchers coding the data using the three-GenAI consensus model.

An advantage of the model is that researchers can reallocate some of the time saved coding statements in unanimous and 2-agree clusters to carefully code the statements in 3-disagree cluster. Statements in the 3-disagree cluster are typically the statements that are harder to code because they land on the boundary between categories. Prior research has found that the statements GenAI finds difficult to code are the same as those that researchers find difficult to code [12].

The unique features of the three-GenAI consensus model are: in order to counter the hallucinatory potential of GenAIs, three GenAIs are used instead of one; the consensus in their responses is leveraged; and the clusters obtained based on the degree of consensus are used to reduce the number of categories researchers need to consider when coding data. The final coding is still done by researchers; GenAIs ease their task so that researchers can spend more time on statements that are harder to code.

The Three-GenAI consensus model could have saved nearly 70% of deductive coding time in the case study. However, the case study involved coding statements that were responses to a single specific survey question. Therefore, the context of the data was both limited and clear, which may have played to the strengths of the three-GenAI consensus model. In contrast, deductive coding is often applied to large and ill-structured text such as interview transcripts, reflection papers, and forum discussions in which the context is large and diffuse. The applicability and benefits of using the three-GenAI consensus model to such data remain to be explored.

We considered only one category per statement. When multiple categories applied to a statement, we considered only the dominant category. If we were to consider *all* the categories in each statement, qualitative coding using the three-GenAI consensus model would have to consider sets of categories instead of individual categories. Unanimous, 2-agree and 3-disagree clusters would be based on set intersection, complicating the application of the model.

We used a zero-shot prompt. A significant performance improvement has been observed going from zero-shot to one-shot prompt [24]. We plan to evaluate the improvement in inter-rater reliability and time savings of the model with one-shot prompt in the future.

In another study with a different data set, we observed how the stochastic nature of GenAIs affects their behavior when used for deductive coding: when three GenAIs were instructed to code the same data set with the same prompt twice, a week apart, the inter-rater reliability (Cohen's kappa) between the two runs was substantial, but not perfect: 0.82 for Claude Sonnet 4.5, 0.78 for ChatGPT 5 Pro and 0.74 for Gemini 2.5 Flash. Given this observation, it may be beneficial to pick the category identified by a GenAI for a statement as the best of n deductive coding runs of the GenAI.

ChatGPT offered an excellent critique of the inter-rater reliability readiness of the codebook (Table 1). Although the three-GenAI consensus model does not include codebook improvement as part of the process, we would like to enhance the model in the future by using trial data to improve the codebook before refining the prompt.

7. ACKNOWLEDGMENTS

Partial support for this work was provided by the National Science Foundation under grants DUE-2216121, 2216031, 2215970 and 2215166. The authors thank Mihaela Sabin and Stephanos Matsumoto for their contribution to prior coding.

8. REFERENCES

[1] A. Barany, N. Nasiar, C. Porter, A. F. Zambrano, A. L. Andres, D. Bright, M. Shah, X. Liu, S. Gao, J. Zhang, S. Mehta, J. Choi, C. Giordano, and R. S. Baker. Chatgpt for education research: Exploring the potential of large language models for qualitative codebook development. In A. M. Olney, I.-A. Chounta, Z. Liu, O. C. Santos, and I. I. Bittencourt, editors, *Artificial Intelligence in Education*, pages 134–149, Cham, 2024. Springer Nature Switzerland.

[2] A. J. Bingham. From data management to actionable findings: A five-phase process of qualitative data analysis. *International Journal of Qualitative Methods*, 22:1–11, 2023.

[3] M. Celepkolu and K. E. Boyer. Thematic analysis of students' reflections on pair programming in cs1. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, SIGCSE '18, page 771–776, New York, NY, USA, 2018. Association for Computing Machinery.

[4] Z. Dunivin. Scaling hermeneutics: a guide to qualitative coding with llms for reflexive content analysis. *EPJ Data Sci.*, 14(28).

[5] Z. Ersozlu and I. Taheri, Sona and· Koch. A review of machine learning methods used for educational data. *Education and Information Technologies*, 29:22125–22145, 2024.

[6] M. Fowler, D. H. Smith IV, B. Chen, and C. Zilles. "i don't gamble to make my livelihood": Understanding the incentives for, needs of, and motivations surrounding open educational resources in computing. In *Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1*, ICER '23, page 430–443, New York, NY, USA, 2023. Association for Computing Machinery.

[7] M. G. Graff, D. Moctezuma, and E. S. Téllez. Bag-of-word approach is not dead: A performance analysis on a myriad of text classification challenges. *Natural Language Processing Journal*, 11, 2025.

[8] I. Hou, O. Man, K. Hamilton, S. Muthusekaran, J. Johnykutty, L. Zadeh, and S. MacNeil. 'all roads lead to chatgpt': How generative ai is eroding social interactions and student learning communities. In *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2025, page 79–85, New York, NY, USA, 2025. Association for Computing Machinery.

[9] S. Kannam, Y. Yang, A. Dharm, and K. Lin. Code interviews: Design and evaluation of a more authentic assessment for introductory programming assignments. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*, SIGCSETS 2025, page 554–560, New York, NY, USA, 2025. Association for Computing Machinery.

[10] W. M. Lim. What is qualitative research? an overview and guidelines. *Australasian Marketing Journal*, 33(2):199–229, 2024.

[11] B.-M. Lindgren, B. Lundman, and U. H. Graneheim. Abstraction and interpretation during the qualitative content analysis process. *International Journal of Nursing Studies*, 108:103632, 2020.

[12] X. Liu, A. F. Zambrano, R. S. Baker, A. Barany, J. Ocumpaugh, J. Zhang, M. Pankiewicz, N. Nasiar, and Z. Wei. Qualitative coding with gpt-4: Where it works better. *Journal of Learning Analytics*, 12(1):169–185, Mar. 2025.

[13] X. Liu, J. Zhang, A. Barany, M. Pankiewicz, and R. S. Baker. Assessing the potential and limits of large language models in qualitative coding. In Y. J. Kim and Z. Swiecki, editors, *Advances in Quantitative Ethnography*, pages 89–103, Cham, 2024. Springer Nature Switzerland.

[14] P. Mayring. Qualitative content analysis. *Forum: Qualitative Social Research*, 1(2), 6 2000.

[15] P. Mulgund, R. Singh, M. Gillenson, Y. Li, and C. Oliver. "not my first choice": Why software testing struggles to attract early-career professionals. In *Proceedings of the 2025 Computers and People Research Conference*, SIGMIS-CPR '25, New York, NY, USA, 2025. Association for Computing Machinery.

[16] H. Reichert, S. Sthapit, B. T. Tabarsi, A. Limke, T. Price, and T. Barnes. Experience helps, but it isn't everything: Exploring causes of affective state in novice programmers. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2*, SIGCSE 2024, page 1792–1793, New York, NY, USA, 2024. Association for Computing Machinery.

[17] S. M. Renz, J. M. Carrington, and T. A. Badger. Two strategies for qualitative content analysis: An intramethod approach to triangulation. *Qualitative Health Research*, 28(5):824–831, 2018.

[18] M. Sabin, A. N. Kumar, B. K. MacKellar, R. McCauley, T. VanDeGrift, and S. Matsumoto. The self-directed disposition: What computing students say. In *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2025, page 107–113, New York, NY, USA, 2025. Association for Computing Machinery.

[19] J. Saldaña. *The Coding Manual for Qualitative Researchers*. Sage, 2013.

[20] S. A. Silverio, J. Hall, and J. Sandall. Time and qualitative research: Principles, pitfalls, and perils. In *The Sixth Annual Qualitative Research Symposium*, 2020.

[21] R. Sinha, I. Solola, H. Nguyen, H. Swanson, and L. Lawrence. The role of generative ai in qualitative research: Gpt-4's contributions to a grounded theory

analysis. In *Proceedings of the 2024 Symposium on Learning, Design and Technology*, LDT '24, page 17–25, New York, NY, USA, 2024. Association for Computing Machinery.

[22] K. Wantanabe. Latent semantic scaling: A semisupervised text analysistechnique for new domains and languages. *Communicataion Methods and Measures*, 15(2):81–102, 2021.

[23] J. Whalley, A. Settle, and A. Luxton-Reilly. A think-aloud study of novice debugging. *ACM Trans. Comput. Educ.*, 23(2), 2023.

[24] Z. Xiao, X. Yuan, Q. V. Liao, R. Abdelghani, and P.-Y. Oudeyer. Supporting qualitative analysis with large language models: Combining codebook with gpt-3 for deductive coding. In *Companion Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23 Companion, page 75–78, New York, NY, USA, 2023. Association for Computing Machinery.

[25] Y. Zhang and B. M. Wildemuth. *Applications of Social Research Methods to Questions in Information and Library Science*, chapter Qualitative Analysis of Content. Libraries Unlimited, 2009.

An Ensemble Stacking with Perception-aware Meta-learning (ESPM) for Apprenticeship Dropout Prediction in France

Duaa Baig
INSA Lyon, CNRS,
Université Claude Bernard
Lyon 1, LIRIS, UMR5205,
69621 Villeurbanne, France,
and Effet B SAS,
69100 Villeurbanne, France
duaa.baig@insa-lyon.fr

Diana Nurbakova
INSA Lyon, CNRS,
Université Claude Bernard
Lyon 1, LIRIS, UMR5205,
69621 Villeurbanne, France
diana.nurbakova@insa-lyon.fr

Sylvie Calabretto
INSA Lyon, CNRS,
Université Claude Bernard
Lyon 1, LIRIS, UMR5205,
69621 Villeurbanne, France
sylvie.calabretto@insa-lyon.fr

Baba Mbaye
Effet B SAS,
69100 Villeurbanne, France
baba@effetb.com

ABSTRACT

The contract termination remains a persistent challenge in vocational education and training (VET), yet existing dropout prediction models overlook the tri- stakeholder dynamics inherent to apprenticeships, i.e., involving the apprentice, workplace master, and training provider. We propose **ESPM** (Ensemble Stacking with Perception-aware Meta-learning), a two-stage architecture that combines gradient boosting base learners with a neural meta-layer encoding stakeholder presence patterns and perception gaps. For interpretability, we introduce the Engagement Asymmetry Index (EAI) and an Engagement-Performance Decomposition module. Evaluated on 3,511 apprenticeship contracts (143, 983 skill evaluations) from the French VET system, ESPM achieves an F1 of 72.21% and a recall of 71.92%, outperforming nine baselines. Interpretability analysis reveals that employer-related features dominate predictions (52.2%) and engagement features outweigh performance features (56.1% vs. 43.9%), challenging the conventional focus on academic grades for dropout prediction.

Keywords

Dropout Prediction, AI in Education, Educational Data Mining, Vocational Education, Apprenticeship Training

1. INTRODUCTION

VET systems play an important role in addressing the demands of the labor market by developing skilled workforces. Apprenticeship, which combines theoretical and practical training, has emerged as an effective structure to facilitate the transition from education to employment across numerous countries [8, 20]. France offers two main forms of VET or work-study programs (fr. *formation en alternance*): (1) Apprenticeship (fr. *Apprentissage*), i.e., employment contract combining vocational training with paid work experience; and (2) Professionalization Contract (fr. *Contrat de Professionnalisation*), i.e., employment contract targeting specific groups. The apprenticeship system has witnessed a significant rise in popularity, with over 800,000 active contracts recorded in 2023, reflecting a massive increase over the last few years [13]. Despite this rise, the contract termination rate in apprenticeships is concerning, with approximately 30% of contracts ending prematurely at the national level [14]. In France, the apprenticeship system is governed by frameworks like the National Directory of Professional Certifications (fr. *Répertoire National des Certifications Professionnelles*, *RNCP*), which serves as a key pathway for youth employment, with over 500,000 contracts annually. This growth was driven by the 2018 reform, which increased financial incentives, simplified administrative procedures, expanded eligibility criteria, and created industry-led funding mechanisms [16]. French national agency *France Compétences*[1], established by the same law that governs the VET and apprenticeship. Its key functions include: regulating training quality and costs, managing the RNCP, distributing funding for apprenticeships and VET, ensuring that qualifications meet labor market needs, and evaluating or monitoring the effectiveness of the system. Digital training booklets (fr. *livrets de formation*) like Campus skill, Kwark, Sheldon skills, STUDEA, track apprentice progress against these competency frameworks, providing a structured record of skills acquisition throughout the training program.

In general, apprenticeship programs have been demonstrated to be an effective method to prepare youth for the labor market through a combination of classroom instruction and

Duaa Baig, Diana Nurbakova, Sylvie Calabretto, and Baba Mbaye. An Ensemble Stacking with Perception-aware Meta-learning (ESPM) for Apprenticeship Dropout Prediction in France. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 415–421. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039699

[1] https://www.francecompetences.fr/

workplace training. However, contract rupture or dropout has been a major problem in apprenticeships, which affects both learners and training organizations. The dropout rates in apprenticeship programs are significantly high and range between 10% and 20%, depending on the sector and program level. The high dropout rate has negative economic consequences beyond individual disappointment. Employers invest their resources in training and supervision, but they face disruption in operational continuity while generating recruitment costs [12]. The providers or training institutions face difficulties in managing program quality metrics, and experience reduced funding tied to completion rate as well [3]. The apprentices or learners often lead to an extended period of unemployment, diminished motivation, and long-term career setbacks due to contract termination [9]. Previous ML studies for dropout prediction have focused on learners in a higher education context and considered several individual-level factors such as academic performance, demographic data, and social and economic aspects. In contrast, apprenticeship education is a triad (tri-stakeholder) structure, which involves (1) apprentice (student) who learns skills and has personal characteristics, (2) master (employer) who provides workplace supervision, task allocation, and practical skills development, and (3) training provider (tutor) who delivers lessons in the classroom, sets the curriculum, and provides institutional support. The concept of apprenticeship inherently involves multi-stakeholder contexts where the learner and other stakeholders may hold asymmetric engagement or have varying levels of expectations. This difference may increase the risk of students dropping out. However, current studies of dropout prediction often neglect these dynamics and adopt the perspective of a single stakeholder, either the learner or the institution. By narrow framing and focusing only on immediate context, several issues may be overlooked, which include (1) critical interactions, (2) perception gaps encountered between apprentices, employers, or providers, (3) engagement asymmetries where stakeholders participate at different intensities, and (4) misalignment between training content and workplace needs.

Our study addresses these gaps by proposing the Ensemble Stacking with Perception-aware Meta-learning (ESPM), a two-stage ensemble approach that models tri-stakeholder dynamics through: (1) a neural meta-learner incorporating perception gap and stakeholder presence features, and (2) two interpretability modules, the EAI computing seven engagement-derived features for SHAP analysis, and Engagement - Performance decomposition that is responsible for separating the predictive signal into engagement-versus-performance and stakeholder-specific contributions. The approach is evaluated on one of the French training booklets' real-world datasets that contains 3,511 apprenticeship triads encompassing 143,983 skill evaluations from a VET platform. The proposed work is evaluated and compared with nine top baseline models, among which ESPM achieves the superior F1 (72.21%) and recall (71.92%). Beyond performance gain, ESPM provides interpretability through EAI-enhanced SHAP analysis, revealing that employer features contribute 52.2% among all stakeholders, engagement features outweigh performance metrics (56.1% vs 43.9%), and students with engagement patterns "apprentice-only" show 41.7% dropout rates, nearly 5× higher than triads where engagement pattern is complete.

The **major contributions** of this paper are fourfold: (1) a domain-specific feature set incorporating RNCP certification details, tri-stakeholder evaluations, and perception gap features, (2) ESPM's architecture, which fuses base learners with domain-specific encoders for stakeholder presence and perception gaps, (3) EAI and decomposition modules, enabling granular interpretability, and (4) empirical validation on real-world apprenticeship data, demonstrating that disengagement between stakeholders occurs before performance decline. The rest of the paper is structured as follows: Section 2 discusses a literature review. In section 3, we share the details of the dataset and preprocessing. Section 4 presents the detailed methodology. In 5, we share the details of the experimental setup. Section 6 highlights the results, and section 7 concludes the paper.

2. LITERATURE REVIEW

The ML application to dropout prediction has experienced significant growth. The use of gradient boosting methods consistently demonstrates strong performance on structured educational data. [5] adopted gradient boosting to identify GPA and credit hours to retain students, while [2] achieved more than 0.85 AUC score using an XGB model on higher education data. In [18], the study reinforces these findings by demonstrating that LightGBM and CatBoost with Optuna hyperparameter tuning outperformed the traditional approaches. The authors in [15] have also achieved state-of-the-art results by incorporating the ADASYN resampling approach with CatBoost on Moodle log data. In a recent meta-synthesis by [4], the authors analyzed 70 studies and 666 dropout variables. They confirm that prior research is mainly focused on individual-level dropout features while neglecting the learning environments of the workplace, which is a critical gap in the context of apprenticeship.

Some recent studies also confirm that gradient boosting methods often achieve similar or exceed the performance as compared with deep learning-based approaches while using tabular datasets [17, 10]. On the other side, ensemble methods, specifically the stacking architecture where a meta-learner is used to combine base model predictions [19], have shown promising results for capturing complementary strengths of the model. Recent work studied a neural meta-learner that learns non-linear combination functions [11], but these approaches remain domain-independent without incorporating task-specific knowledge into their architecture. [21] introduced a transformer-based stacking approach that uses an attention mechanism to dynamically integrate base classifier predictions. The research studies that specifically address dropout in apprenticeships remain sparse. The authors in [9] found that factors like prior education, training sector, and age have a significant effect on the likelihood of non-completion in England. [12] concluded in their work that employer training quality significantly affects retention in Switzerland. In France, [1] highlights the variation across qualification levels but notes the limited availability of data capturing the experience of apprenticeship. A recent report [7] emphasizes the urgency of this problem. They documented that in England, approximately 40% apprentices fail to complete their courses. The concept of perception gaps has been studied in educational psychology [6], but their integration into predictive models is limited. In our work, we address these gaps through four major contribu-

tions. First, we develop tri-stakeholder features to capture engagement patterns from the perspectives of all three stakeholders, which directly overcomes the challenge identified by [4] regarding the workplace learning environment gap. Second, we introduce a perception-aware neural meta-learner that encodes stakeholder presence patterns, perception gaps, and domain knowledge directly into the model architecture. Third, we propose the EAI in dropout prediction as a separate interpretability module that enables SHAP to decompose predictive contributions across stakeholders. Fourth, we develop an Engagement-Performance Decomposer to distinguish engagement and performance-related features, revealing that dissengagement precedes performance decline.

3. DATASET

The French apprenticeship system operates as a form of alternating training, where the contracts span is typically 12-36 months. During training, the learner spends most of their time in the workplace. Skill evaluations by stakeholders are a core component of monitoring throughout the contract. Along with three primary stakeholders (Student, Master, and Tutor) in each contract, there is another stakeholder in some contracts (less consistently implemented), called the LEA (Livret Électronique d'Apprentissage) manager, who oversees the digital apprenticeship portfolio. During the program, each stakeholder assesses the apprentice's competencies across various skill domains by recording scores.

Data Collection and Pre-Processing: Our dataset is obtained from one digital booklet system for apprenticeship management and is subject to confidentiality and data protection agreements. The study analyzes fully anonymized data, collected as part of standard educational practice and handled with institutional data protection policies. All personally identifiable information (PII) related to students, institutions, and workplaces is removed before analysis. The study uses secondary analysis of de-identified data and does not require institutional review board (IRB) approval under our institution's guidelines for educational research. The anonymized records of apprenticeship contracts initiated between 2020 and 2023. The raw data contains 143,983 individual skill evaluation records encompassing 3,529 unique triads (contracts), 6,179 unique skills, and 2,953 unique students. The unique skills contained 407 distinct skill categories, which we consolidated into 10 broad skill domains (e.g., technical skills, communication, professional attitude, etc.) through keyword-based mapping on French skill descriptions to reduce dimensionality while preserving domain-specific predictive signals. In data cleaning, we dropped the records where the apprentice ID or skill evaluation score is missing. The overall stats of the dataset after cleaning are presented in Table 1. Individual evaluations were aggregated to the contract level. All the contracts are strictly ordered by start date within each student to prevent data leakage.

Overall Features: The raw features in the dataset are mainly related to demographic, contract, and skill evaluations. The Table 2 summarizes the overall 53 features used in our experiments, organized by category, stakeholder group, and data source. The features belong to three different sources: (1) *Raw* features used directly from the database, (2) *Scraped* features obtained by web scraping the RNCP[1] registry to fetch RNCP meta-data like total skill blocks/skill counts,

Table 1: Summary of the Apprenticeship Evaluation Dataset

Category	Details
Dataset size	
Total skill evaluation records	143,550
Apprenticeship triads (contracts)	3,511
Unique apprentices/students	2,935
Unique skills	6,130
Original skill categories	407
Mapped skill categories	10
Contract characteristics	
Repeat-apprentice triads	1,070
Triads with prior rupture history	116 (3.3%)
Dropout contracts	367 (10.5%)
Completed contracts	3,144 (89.5%)
Class imbalance ratio	∼8.6:1
Evaluation distribution	
Apprentice (self): triads/evaluations	2,815 / 77,872
Tutor: triads / evaluations	252 / 5,192
Master: triads / evaluations	2,507 / 55,651
LEA/Manager: triads/evaluations	154 / 4,835

Table 2: Summary of Feature Groups by Category and Source

Category	Group	Count	Source
Contract & Demo.	Shared	5	Raw
Student History	Apprentice	2	Raw
Skill Evaluation	Shared	6	Raw
Self-Evaluation	Apprentice	5	Raw
Master Evaluation	Employer	5	Raw
Perception Gap	Shared	1	Raw
Skill Categories	Shared	10	Raw
Stakeholder Engagement	Shared	5	Raw
RNCP Certification	Provider	8	Scraped/Eng.
Interaction Features	Shared	6	Engineered
Total	—	**53**	—

and number of organizations that offer any RNCP, etc., and (3) *Engineered* features computed by transforming the raw features. The group column refers to the features ownership within the triad stakeholders, including *Apprentice* features that purely belong to student history and self-assessment, *Employer* features refer to workplace master evaluations, *Provider* features represent training institute and RNCP related data, and *Shared* features apply across all stakeholders to capture cross interactions. This group distribution or stakeholder mapping is defined for the ESPM to compute asymmetry patterns and perception gaps. The target variable *broken_contract* is binary (0 = retained, 1 = dropout) and is separate from 53 input features. The original dataset contains more than 54 columns, from which we exclude all the non-predictive identifiers, leaky and sparse features with coverage below 25% (including tutor and LEA evaluation statistics), as in most of the triads, tutor and LEA evaluations are missing. Despite low coverage, binary engagement indicators of both stakeholders are retained in the features as the presence/absence signal of their evaluation, which carries meaningful information regarding triad completeness.

4. PROPOSED APPROACH

The ESPM employs a novel two-stage architecture to predict dropout in apprenticeship. ESPM leverages the complementary strengths of multiple gradient boosting algorithms while incorporating domain-specific knowledge of tri-stakeholder dynamics that are inherent to apprenticeship. The model is

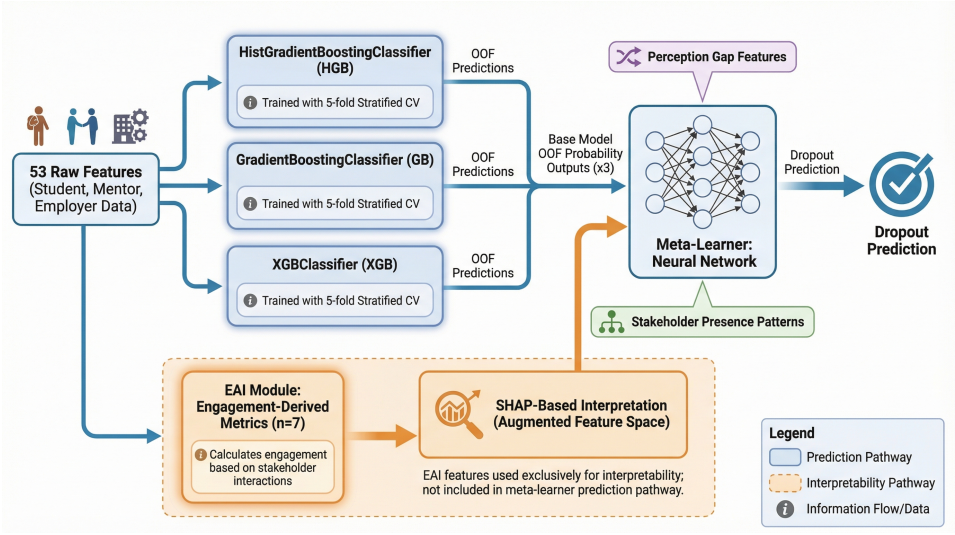


Figure 1: Stage 1 (S1): Base model ensemble (HGB, GB, XGB) generates OOF predictions. Stage 2 (S2): Perception-aware meta-learner combines predictions with domain-specific features. The EAI module (orange box) operates separately for interpretability.

designed using two complementary components: (1) a **two-stage prediction module** that combines the gradient boosting ensembles with a perception-aware neural meta-learner, and (2) an **Engagement Asymmetry Index (EAI) module**, which helps to provide interpretability through engagement analysis. The overall architecture of the ESPM is presented in Figure 1.

ESPM uses separate modules for prediction and interpretability. The prediction module is designed using a two-stage stacking architecture. In stage 1 (S1), an ensemble of gradient boosting models trains on the complex feature interactions to learn from the complete feature set. In stage 2 (S2), a neural network-based meta-learner combines base models' predictions with presence and perception-aware features to generate final dropout probabilities. Separately, we also computed EAI engagement metrics by using the EAI module, which is exclusively for SHAP-based analysis and risk stratification. The EAI features are not fed into the meta-layer for predictions to ensure clean separation between prediction accuracy and model interpretability. The design and module separation reflect our finding that EAI features are valuable for understanding the dynamics of stakeholders but do not contribute to predictive performance (see ablation results in section 6.1).

4.1 S1 - Base Model Ensemble

In ensemble stacking, we used three gradient boosting classifiers with class weighing, including (1) Histogram-based Gradient Boosting (HGB), (2) Gradient Boosting (GB), and (3) Extreme Gradient Boosting (XGB), which serve as base learners. The selection of classifiers is based on their complementary learning characteristics and their ability to efficiently handle imbalanced classes. The diversity of base models helps to capture different patterns of the data. Each base model is trained using 5-fold stratified nested cross-validation to generate out-of-fold (OOF) predictions, which help to avoid data leakage. For fold k, model trained on folds $\{1, \ldots, K\} \setminus \{k\}$ to generate predictions for samples in fold k. This produces an $n \times 3$ matrix of OOF predictions, where each entry is the predicted dropout probability P (dropout) that later serves as meta-features for stage 2. The ensemble produces a prediction matrix $\mathbf{P} \in \mathbb{R}^{n \times 3}$ where each row represents probabilities generated from all three base models.

4.2 S2 - Perception-Aware Meta-Learning

The meta-learner is a feed-forward neural network structure that combines base model predictions with additional domain-specific features to encode stakeholder dynamics. Unlike generic stacking approaches that combine base predictions, ESPM processes three distinct input streams via specialized encoders, including base predictions, perception gaps, and a presence mask. The **base predictions** are the OOF probability outputs of base models (XGB, GB, HGB), processed through a prediction encoder (Linear $\rightarrow$ ReLU $\rightarrow$ Dropout). The **perception gaps** are the 3 raw input features, extracted from the original 53 feature set (1. self_vs_master_gap (apprentice vs master perception gap): mean self evaluation - mean master evaluation (e.g., apprentice rates their skills at 4.2/5.0, while master rates them at 3.0/5.0, the gap is +1.2, indicating overconfidence). 2. fe_gap_magniture: the absolute value of the perception gap ($|1.2| = 1.2$, where a large value (>1.0) regardless of direction, indicates strong misalignment). 3. fe_self_master_interaction: product of self_mean and master_mean) that are passed through a gap encoder (Linear $\rightarrow$ Tanh $\rightarrow$ Dropout) to make the meta-layer aware of perception gaps. The **presence mask** is a binary vector [1 (always true for apprentice), has_master_eval (employer participation flag), has_provider_eval (institutional engagement flag)] that indicates the participation of stakeholders in the evaluation, processed through a pattern encoder (Linear $\rightarrow$ ReLU) to capture the completeness of the triad. The three representations are concatenated to generate a combined vector. The generated vector representation is passed to an attention layer that produces weights over the three base models, which helps to enable context-dependent model combination. Finally, the predictor network produces the final logit by processing the concatenation of raw base

predictions and encoded predictions through a feed-forward network (Linear → ReLU → Dropout → Linear). The architecture allows the meta-learner to use perception gap and presence context to weight base model results, while maintaining access to raw and encoded prediction information.

4.3 EAI Module for Interpretability

We introduce the EAI to identify the imbalanced participation level of each stakeholder group in the triad. In step 1, we compute the **stakeholder engagement score**, for s, where $s \in \{\text{apprentice, employer, provider}\}$. We compute a normalized score in two steps, **a.** Sum relevant (engagement) features of the stakeholder from the original 53 feature set.

$$\text{raw}_s = \sum_{f \in F_s} x_f.$$

The features include all three stakeholders (1) Apprentice (Fs): self_count (self-evaluation frequency), has_self_ eval (participation flag), self_mean (average self-score). (2) Employer (Fs): master_count (master evaluation frequency), has_master_ eval (participation flag), master_mean (average master score). (3) Provider (Fs): has_tutor_eval (tutor participation), has_lea_ eval (LEA manager participation). **b.** Apply min-max normalization across all contracts

$$e_s = \frac{\text{raw}_s - \min(\text{raw}_s)}{\max(\text{raw}_s) - \min(\text{raw}_s)}$$

This produces engagement scores in $[0, 1]$. In step 2, **Asymmetry Calculation**, it uses the vector of engagement score $\mathbf{e} = [e_{\text{app}}, e_{\text{emp}}, e_{\text{prov}}, e_{\text{lea}}]$ to compute EAI as the coefficient of variation

$$\text{EAI} = \frac{\sigma(\mathbf{e})}{\mu(\mathbf{e}) + \epsilon}$$

where σ denotes the standard deviation and μ denotes mean, and $\epsilon = 0.01$ prevents division by zero. The higher EAI score indicates strong asymmetry or one-sided engagement, while the value closer to zero shows balanced engagement across stakeholders. In step 3, we derive **interaction features** from the engagement scores: app_emp_sync $= 1 - |e_{\text{app}} - e_{\text{emp}}|$ (measuring apprentice-employer alignment), eng_completeness (proportion of stakeholders with engagement > 0.1), and eng_gap $= \max(\mathbf{e}) - \min(\mathbf{e})$ (capturing the range of engagement). The computed seven features are used for SHAP-based interpretability analysis, enabling decomposition of stakeholder-wise feature importance and engagement-versus-performance categories.

4.4 Meta-Learner Training

The perception-aware meta-learner is trained using BCE (binary-cross-entropy) loss with class weighing to handle the imbalance ratio. Early stopping is also applied with patience to prevent overfitting. The patience hyperparameter is tuned in the range 5 to 20 during optimization.

5. EXPERIMENTAL SETUP

All the experiments are implemented using Python 3.12, PyTorch 2.1, scikit-learn 1.4, and the Optuna hyper-parameter tuning is utilized for all baselines and the proposed approach, with 15 trials each. To evaluate ESPM, we compared it with **nine baseline classifiers**, including Logistic Regression (LR), Random Forest (RF), Balanced Random Forest (BRF), GB, HGB, XGB, LightGBM (LGB), CatBoost (CB), and AdaBoost (AB). All the models are evaluated using 5-fold stratified nested cross-validation: for each fold, the data split of 80% train and 20% test is performed. To avoid data leakage from repeat triads, folds are grouped by student ID to ensure that all contracts from the same student remain in the same fold to prevent student-level history (e.g., previous_ruptures_count) from leaking into the test set. The fixed random seed (42) is also used to ensure identical splits across all experiments. To evaluate the performance and considering the class imbalance issue, we have used multiple complementary evaluation metrics, Area under ROC curve (AUC) mean for ranking ability, average F1 score (threshold-optimized), average Precision and Recall for imbalanced classification. Considering the importance of identifying at-risk students, recall is prioritized as missing an at-risk student has a higher cost than a false alarm.

Table 3: Performance comparison (mean across 5 folds)

Model	AUC	F1	Precision	Recall	Specificity
ESPM	**0.9253**	**0.7221**	**0.7300**	**0.7192**	**0.9675**
HGB	0.9259	0.6978	0.7360	0.6730	0.9701
XGB	0.9255	0.6900	0.6846	0.7026	0.9611
GB	0.9235	0.7070	0.7338	0.6835	0.9710
LGB	0.9223	0.7001	0.7071	0.6946	0.9662
CB	0.9207	0.7076	0.7320	0.6866	0.9701
RF	0.9188	0.6651	0.6449	0.7025	0.9513
BRF	0.9136	0.6513	0.6164	0.7054	0.9452
AB	0.9074	0.6701	0.6815	0.6621	0.9634
LR	0.8812	0.5828	0.6016	0.5832	0.9500

6. RESULTS AND DISCUSSION

The Table 3 illustrates the overall performance of ESPM and the baseline models. ESPM achieves the highest F1 (0.7221) and recall (0.7192), while maintaining strong precision (0.7300) and specificity (0.9675). It correctly identifies 71.9% of at-risk students compared to 67.3% by the best baseline HGB (as per AUC), representing a 6.8% relative improvement in recall. While the F1 improvement appears modest, its practical impact is significant in apprenticeship. With the highest recall, ESPM identifies additional at-risk students across the 5-fold evaluation, and considering the high cost of dropout, even a small gain in recall justifies the added architectural complexity. For the CFA managing thousands of contracts annually, such improvements could prevent additional dropouts, providing meaningful improvement for both institutional outcomes and student success. The improvement comes with a favorable trade-off, as ESPM achieves higher recall at the cost of slightly lower precision in a context where missing at-risk students carries a higher cost than unnecessary monitoring. Noticing baseline models' pattern, gradient boosting methods (HGB, XGB, GB, LGB, CB) cluster tightly in AUC (0.9207-0.9259), confirming literature findings that these implementations achieve similar accuracy on well-engineered features. LR substantially underperforms (AUC=0.8812), indicating that non-linear interactions matter for this task.

6.1 Ablation Study

To understand the contribution of each component, we performed systematic ablation studies comparing four ESPM variants: (1) ESPM (proposed), (2) ESPM-GapsOnly (re-

Table 4: Ablation study results

Variant	F1	Recall	AUC
ESPM	**0.7221**	**0.7192**	**0.9253**
ESPM-PresenceOnly	0.7207	0.7029	0.9252
ESPM-GapsOnly	0.7171	0.7166	0.9245
ESPM-Baseline	0.7200	0.7165	0.9244

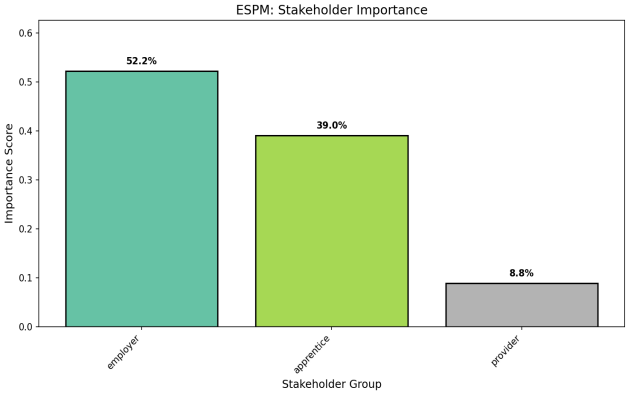


Figure 2: Stakeholder-wise Overall Feature Importance

moves stakeholder presence mask), (3) PresenceOnly (removes perception gap features), and (4) ESPM-Baseline (removes both perception gaps and presence). The study shows that ESPM configuration achieves superior F1 (0.7221) and recall (0.7192). The overall comparative results are shown in Table 4. ESPM's domain-specific design adds value beyond generic stacking by explicitly encoding tri-stakeholder structure directly into the meta-learner. Removing presence mask and perception gaps declines performance and indicates that these components capture patterns that base models alone miss. Results also show that combined features contribute to superior performance as compared to either alone. The results show that explicitly encoding apprenticeship structure through perception gaps and presence patterns improves predictions beyond what generic ensemble learning achieves.

6.2 Interpretability Analysis

SHAP analysis is applied on the 60-feature space (53 raw + 7 EAI features) to identify which stakeholder features contribute most to the prediction task. As the ESPM meta-layer is not directly interpretable, we use a separate HGB classifier as a surrogate model, because it achieves nearly identical performance to ESPM. HGB is trained on 60 feature set solely for interpretability analysis using SHAP TreeExplainer. Analysis results in Figure 2 show that employer-related features are 52.2% of total predictive features, while the stakeholder importance for apprentice is 39.0%, and Provider features are 8.8%. It shows that workplace dynamics/employer features (52.2%) exceed apprentice and provider combined, with master_count (evaluation frequency) as the top predictor. This suggests that monitoring employer engagement is the primary risk signal. The low provider contribution (8.8%) partly reflects sparse coverage (7.2% tutor evaluation rate), but it also indicates that institution-related features are less diagnostic than workplace interactions.

Table 5: Top Engagement (Eng.) vs Performance (Perf.) Features based on SHAP Importance (Imp.) Score

Eng. Features	Score	Perf. Features	Score
master_count	0.156	master_mean	0.089
total_eval_count	0.142	overall_skill_mean	0.072
self_count	0.128	self_vs_master_gap	0.065

Features Decomposition Methodology: To enable stakeholder-specific and engagement-vs-performance analysis, we train a separate HGB classifier on the 60-feature augmented space (53 original features + 7 EAI-derived metrics) and apply SHAP TreeExplainer. Features are categorized into two groups: Engagement features (23 features) and Performance features (30 features). The percentage contribution is computed as:

$$\text{Category}_{\%} = \frac{\sum_{f \in \text{Category}} |\text{SHAP}(f)|}{\text{Total}_{|\text{SHAP}|}} \times 100$$

Engagement vs. Performance Decomposition: The decomposition of features into two main categories, engagement (evaluation counts, participation) versus performance (skill scores, grades), reveals that the engagement features are 56.1% of the predictive signal and the remaining 43.9% are performance features. This feature importance split challenges conventional assumptions that dropout predictions should focus on academic grades, but in our study, active stakeholders' evaluations matter slightly more than what scores they assign. In Table 5, it is presented that the topmost important engagement feature has 75% higher SHAP importance than the most important performance feature. It shows that stakeholder disengagement precedes performance decline, so monitoring evaluation frequency in apprenticeship provides earlier warning signals. It tells that the dropout systems should alert when evaluation rates decline, not just when scores fall. The top 10 features by SHAP importance scores also reveal the dominance of engagement over performance.

EAI Feature Contribution: The seven engagement-derived metrics are computed by the EAI module for SHAP analysis. The importance rankings of these metrics reveal which aspects of stakeholders' asymmetry most strongly predict dropout. The overall EAI Feature Rankings by SHAP importance are (1) Engagement Gap (0.397), the difference between the most and least engaged stakeholders, (2) Apprentice Engagement (0.336), apprentice participation level. (3) Apprentice - Employer Sync (0.235), alignment between apprentice and employer, (4) Employer Engagement (0.171), employer participation level, and (5) EAI (0.159), overall imbalance coefficient. The engagement gap emerges as the most important EAI metric. This suggests that the imbalance magnitude matters more than which specific stakeholder is disengaged. An apprenticeship with one highly engaged stakeholder but two disengaged faces elevates risk regardless of who is engaged. The engagement pattern analysis is presented in Table 6. Students in "apprentice-only" pattern (where only the apprentice evaluates, without employer participation) show 41.7% dropout rate, nearly 5× higher than the minimal baseline (8.5%). This validates the critical importance of employer engagement. The EAI module transforms raw evaluation counts into meaningful

Table 6: Engagement Pattern Analysis

Pattern	Apprentice	Employer	Provider	Dropout Rate	Sample Size
Apprentice-Only	High	None	None	41.7%	72
Employer-Only	None	High	None	23.1%	255
Apprentice-Employer	High	High	None	28.6%	35
Full Triad	High	High	Present	40.0%	5
Minimal	Low	Low	None	8.5%	3144

stakeholder narratives.

7. CONCLUSION AND LIMITATIONS

This paper presents the ESPM approach for dropout predictions in apprenticeship. By modeling the tri-stakeholder dynamic using perception gap features and stakeholder presence mask pattern, ESPM achieves superior performance in terms of F1, Recall, and interpretability compared to nine baseline models. To achieve interpretability, the separate EAI module is used, which enables decomposition of predictive signal by stakeholder and engagement-vs-performance categories. The analysis reveals that engagement features contribute 52.2% and outweigh the performance features 43.9%. The overall findings show that stakeholder disengagement precedes performance decline, challenging conventional grade / performance-focused dropout prediction. Our findings have some implications beyond apprenticeship. First, the dominance of engagement over performance challenges conventional grade-focused learning analytics assumptions for predictions. Second, perception gap features offer a generalizable approach to model expectations misalignment in any feedback-intensive domain (employee reviews or peer assessments, etc.). Third, the tri-stakeholder framework can generalize to other triad contexts (internships or collaborative projects, etc.) as well. The study has some limitations as well, including limited generalizability to other institutes or countries, as our data originates from a single formation (CFA). Another limitation is the sparse coverage of tutor and LEA evaluations, which limits our ability to fully model provider engagement. The dominance of employer features (52.2%) may reflect feature availability bias, as employer evaluations are recorded more often than provider evaluations, which can inflate SHAP importance. The finding that engagement outweighs performance should therefore be interpreted with caution since evaluation frequency is linked to the contract duration and student persistence, where longer contracts generate more records.

8. REFERENCES

[1] D. Abriac, R. Rathelot, and R. Sanchez. L'apprentissage, entre formation et insertion professionnelles. *Formations et emploi*, pages 57–74, 2009.

[2] L. Aulck, N. Velagapudi, J. Blumenstock, and J. West. Predicting student dropout in higher education. *arXiv preprint arXiv:1606.06364*, 2016.

[3] S. Billett. *Vocational education: Purposes, traditions and prospects*. Springer Science & Business Media, 2011.

[4] S. Böhn and V. Deutscher. Dropout from initial vocational training–a meta-synthesis of reasons from the apprentice's point of view. *Educational Research Review*, 35:100414, 2022.

[5] D. Delen. A comparative analysis of machine learning techniques for student retention management. *Decision Support Systems*, 49(4):498–506, 2010.

[6] D. Dunning, K. Johnson, J. Ehrlinger, and J. Kruger. Why people fail to recognize their own incompetence. *Current directions in psychological science*, 12(3):83–87, 2003.

[7] S. Field. A world of difference: International apprenticeship policy and lessons for england, 2025.

[8] A. Fuller and L. Unwin. Learning as apprentices in the contemporary uk workplace: creating and managing expansive and restrictive participation. *Journal of Education and work*, 16(4):407–426, 2003.

[9] L. Gambin and T. Hogarth. Factors affecting completion of apprenticeship training in england. *Journal of Education and Work*, 29(4):470–493, 2016.

[10] L. Grinsztajn, E. Oyallon, and G. Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520, 2022.

[11] G. Ke, Z. Xu, J. Zhang, J. Bian, and T.-Y. Liu. Deepgbm: A deep learning framework distilled by gbdt for online prediction tasks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 384–394, 2019.

[12] J. Mohrenweiser and U. Backes-Gellner. Apprenticeship training: for investment or substitution? *International Journal of Manpower*, 31(5):545–562, 2010.

[13] A. Plé. L'apprentissage en 2023. Dares résultats, no. 72, DARES – Direction de l'Animation de la Recherche, des Études et des Statistiques, 2024.

[14] A. Plé. L'apprentissage en 2024. Dares résultats, no. 3, DARES – Direction de l'Animation de la Recherche, des Études et des Statistiques, 2026.

[15] M. Rebelo Marcolino, T. Reis Porto, T. Thompsen Primo, R. Targino, V. Ramos, E. Marques Queiroga, R. Munoz, and C. Cechinel. Student dropout prediction through machine learning optimization: insights from moodle log data. *Scientific Reports*, 15(1):1–16, 2025.

[16] République Française. Loi n° 2018-771 du 5 septembre 2018 pour la liberté de choisir son avenir professionnel. `https://www.legifrance.gouv.fr/loda/id/JORFTEXT000037367660`, 2018.

[17] R. Shwartz-Ziv and A. Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90, 2022.

[18] A. Villar and C. R. V. de Andrade. Supervised machine learning algorithms for predicting student dropout and academic success: a comparative study. *Discover Artificial Intelligence*, 4(1):2, 2024.

[19] D. H. Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[20] S. C. Wolter and P. Ryan. Apprenticeship. In *Handbook of the Economics of Education*, volume 3, pages 521–576. Elsevier, 2011.

[21] X. Yang, Y. Zhao, and X. Chen. A novel transformer-based stacking ensemble method with multi-model integration for cancer classification. *PeerJ Computer Science*, 11:e3314, 2025.

Augmenting Student Profiles and Course Attributes with Large Language Models for Course Recommendation

Tianyuan Yang
Grad. School of ISEE, Kyushu University
yang.tianyuan.791@s.kyushu-u.ac.jp

Baofeng Ren
Grad. School of ISEE, Kyushu University
ren.baofeng.817@s.kyushu-u.ac.jp

Chenghao Gu
Grad. School of ISEE, Kyushu University
gu.chenghao.564@s.kyushu-u.ac.jp

Feike Xu
Grad. School of ISEE, Kyushu University
xu.feike.383@s.kyushu-u.ac.jp

Boxuan Ma
Faculty of Arts and Science, Kyushu University
boxuan@artsci.kyushu-u.ac.jp

Shin'ichi Konomi
Faculty of Arts and Science, Kyushu University
konomi@artsci.kyushu-u.ac.jp

ABSTRACT

Course recommender systems (CRSs) play an important role in helping students navigate large-scale online course platforms. However, effective course recommendation remains challenging due to sparse interaction data and the lack of explicit, learning-oriented representations on both the student and course sides. In this paper, we propose SORARec, a Student prOfile and couRse Attribute–aware Recommendation model that leverages large language models (LLMs) to enrich educational recommendations with learning-aware semantic information. Specifically, SORARec employs an LLM to infer structured student profiles and learning-relevant course attributes from historical enrollments and course metadata. These inferred semantics are encoded into dense embeddings and integrated with collaborative representations through a lightweight graph-based fusion mechanism. Unlike approaches that rely on LLMs for direct recommendation, SORARec uses LLMs only in an offline inference stage, preserving efficiency and scalability. Experiments on the real-world MOOC dataset demonstrate that SORARec consistently outperforms baselines under both general and cold-start settings. Ablation studies further confirm the complementary contributions of student profile and course attributes. The results indicate that jointly modeling student profiles and course attributes provides an effective and practical way to improve course recommendation performance in educational scenarios.

Keywords

Course Recommendation, data augmentation, large language models, educational data mining, cold-start recommendation

Tianyuan Yang, Ren Baofeng, Chenghao Gu, Feike Xu, Boxuan Ma, and Shinichi Konomi. Augmenting Student Profiles and Course Attributes with Large Language Models for Course Recommendation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 422–429. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039668

1. INTRODUCTION

Massive Open Online Courses (MOOCs) and online learning platforms provide students with unprecedented access to diverse educational resources. However, the rapid growth of course catalogs also makes it increasingly difficult for students to identify courses that align with their backgrounds, interests, and learning goals [14]. Course recommender systems (CRSs) have therefore become a key component of modern educational platforms, supporting students in making informed enrollment decisions [16].

Despite substantial progress, effective course recommendation remains challenging. Unlike general-purpose recommender systems, CRSs must account for learning-oriented factors such as prior knowledge, learning objectives, and course difficulty, in addition to historical preferences. As a result, recommendation quality critically depends on the availability of informative student profiles and course attributes. In practice, however, such information is often missing. Most students enroll in only a small number of courses, leading to severe data sparsity and frequent cold-start situations [4,17,37]. Under these conditions, interaction-based models struggle to infer students' learning needs reliably. Moreover, student characteristics (e.g., background knowledge and difficulty tolerance) are largely latent, while course representations often lack fine-grained semantic and pedagogical attributes, resulting in recommendations that may be misaligned with students' abilities or goals [13, 15, 31].

To alleviate these issues, prior work has incorporated side information, content features, or knowledge graphs into CRSs [3,29]. While effective to some extent, these approaches typically rely on manually curated metadata or domain-specific resources, limiting their scalability. Recently, Large Language Models (LLMs) have demonstrated strong capabilities in semantic understanding and knowledge inference, offering new opportunities for enriching representations in recommender systems. However, existing LLM-based CRSs often augment only one side of the recommendation process [31] or treat LLM outputs as unstructured textual features [30]. In addition, directly using LLMs for recommendation in-

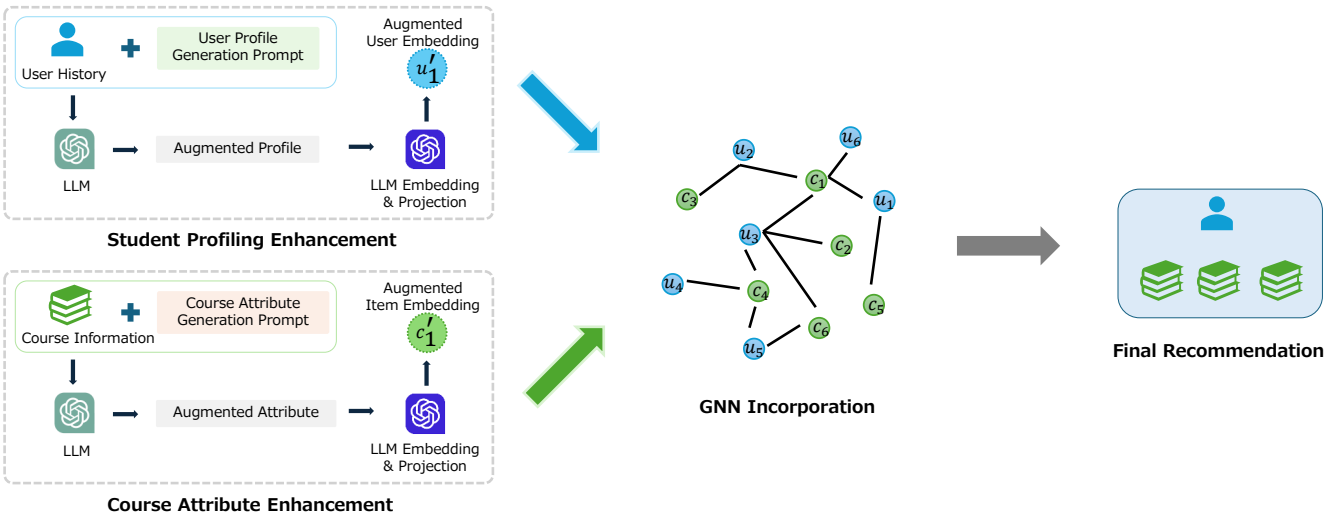


Figure 1: Overview of the proposed SORARec model.

curs high computational cost and does not consistently outperform lightweight neural models. Consequently, how to effectively leverage LLMs to address both representation insufficiency and data sparsity in educational recommendation remains an open question.

In this paper, we propose **SORARec**, a Student prOfile and couRse Attribute–aware course Recommendation model. SORARec employs LLMs to infer structured, learning-oriented student profiles and expressive course attributes from historical interactions and course metadata. These inferred semantics are encoded into dense embeddings and integrated with collaborative representations, complementing interaction data rather than replacing it. Extensive experiments on the real-world MOOC dataset show that SORARec consistently outperforms strong baselines, with particularly notable improvements in sparse and cold-start scenarios. Ablation studies further demonstrate the complementary contributions of student-side and course-side semantic information.

2. RELATED WORK

2.1 Course Recommendation

Course recommendation has been widely studied in educational settings to assist students in navigating large and diverse course catalogs. Early approaches primarily relied on content-based and collaborative filtering methods, which infer student preferences from historical enrollments or course similarity patterns [8, 18–20]. While effective and easy to deploy, these methods are often limited by sparse interaction data and their inability to capture learning-oriented factors.

To address these limitations, later studies introduced sequential, graph-based, and knowledge-aware models to capture course dependencies and student–course interactions [4, 21, 37]. Knowledge-enhanced and explainable CRSs further incorporate structured information such as prerequisite relations or course knowledge graphs to improve recommendation quality and transparency [29, 32]. However, most existing approaches still rely heavily on interaction data or manually curated features, making them vulnerable to data sparsity and cold-start scenarios, and limiting their scalability in real-world educational platforms.

2.2 LLMs for Recommendation and Education

LLMs have demonstrated strong semantic understanding and reasoning capabilities, and have recently been explored in recommender systems to alleviate data sparsity and enrich representations [2, 26]. Existing LLM-based recommendation approaches typically use LLMs to simulate user behaviors, generate auxiliary interactions, or produce preference signals and explanations [7, 25, 27]. While effective, these methods often incur high inference costs or treat LLM outputs as unstructured textual signals.

In parallel, LLMs have been applied in educational contexts for content generation, feedback, and learning support [1, 12, 28]. Recent studies have begun incorporating LLMs into educational recommendation, for example by generating textual recommendations or augmenting student–course representations under sparse conditions [22, 36]. However, the use of LLMs to construct explicit, structured student profiles and learning-relevant course attributes, and to integrate them efficiently into conventional course recommendation models, remains limited. This gap motivates our proposed SORARec framework.

3. THE PROPOSED MODEL

Figure 1 illustrates the overall workflow of SORARec. Given historical student–course interactions and course metadata, SORARec first employs an LLM to infer structured student profiles and learning-relevant course attributes. The inferred profiles and attributes are then encoded into semantic embeddings and aligned with the collaborative embedding space. Next, these semantic representations are propagated over the student–course interaction graph and integrated with interaction-based embeddings to obtain enhanced student and course representations. Finally, SORARec predicts

student–course preferences based on the fused representations and generates personalized course recommendations.

3.1 LLM-based Student Profile and Course Attribute Inference

SORARec employs a LLM as an auxiliary inference module to construct structured student profiles and learning-relevant course attributes. In educational platforms, explicit information about students' learning goals, background knowledge, or difficulty preferences, as well as fine-grained course attributes such as conceptual coverage and pedagogical difficulty, is often unavailable or inconsistently recorded. This information deficiency, combined with sparse interaction data, poses a major challenge for accurate and personalized course recommendations. To address this issue, SORARec leverages the reasoning capability of LLMs to infer missing or implicit attributes from available textual evidence and transforms them into dense semantic representations for downstream recommendation.

Student Profile Inference: For each student u, we collect the historical interaction set $\mathcal{H}_u = \{i_1, i_2, \ldots, i_{|\mathcal{H}_u|}\}$, where each interacted course provides indirect evidence of the student's latent learning characteristics. Then, SORARec prompts an LLM to infer a structured student profile based on the semantic patterns of the courses previously taken.

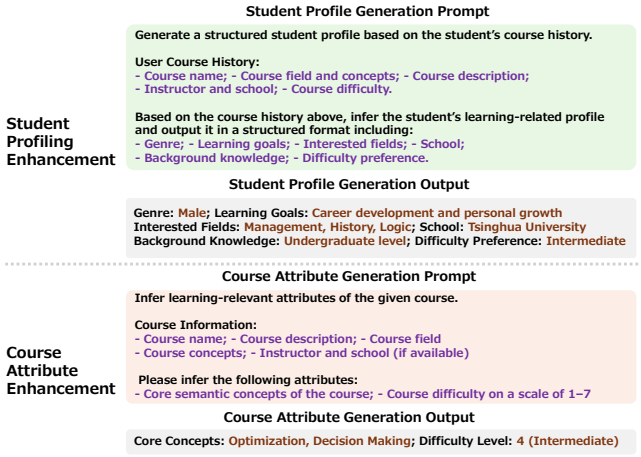


Figure 2: Illustration of the LLM-based semantic inference process in SORARec. The top part shows how student profiles are inferred from historical course enrollments to capture learning goals, background knowledge, and difficulty preferences. The bottom part shows how learning-relevant course attributes, including semantic concepts and difficulty levels, are generated from course metadata.

As illustrated in Fig. 2, the prompt consists of a task description, textual information extracted from the student's enrolled courses (e.g., course titles, topics, and fields), and a strict output schema that enforces a machine-readable format. The LLM is required to infer multiple educationally meaningful facets, such as learning goals, interested fields, background knowledge level, and difficulty preference. Let P_u^{prof} denote the constructed prompt for student u, The inferred student profile is obtained as:

$$A_u = \text{LLM}\left(P_u^{\text{prof}}\right),$$

where A_u is a structured textual description summarizing the student's latent learning characteristics inferred from historical interactions.

Course Attribute Inference: Similarly, for each course i, SORARec employs an LLM to infer learning-relevant course attributes from available course metadata. Course descriptions in online learning platforms are often written in free-form text and vary widely in granularity and quality, which limits their direct use in recommendation models. To obtain more structured and comparable representations, we prompt the LLM with course-specific information, including the course name, description, field, instructor, institution, and existing annotations when available, which is shown in Fig. 2.

The LLM is instructed to infer pedagogically meaningful attributes, with a particular focus on semantic concepts and course difficulty, which are critical for matching courses with students' backgrounds and learning objectives. Let P_i^{attr} denote the prompt constructed for course i. The inferred course attributes are defined as:

$$A_i = \text{LLM}\left(P_i^{\text{attr}}\right),$$

where A_i represents the LLM-generated textual description of the course's semantic and pedagogical characteristics.

Semantic Embedding of Inferred Profiles and Attributes: The inferred student profiles A_u and course attributes A_i are textual or structured outputs that cannot be directly consumed by standard recommendation models. As shown in Fig. 1, SORARec converts these inferred attributes into dense semantic representations using an embedding encoder. Formally, the embedding vectors are computed as:

$$\mathbf{f}_{A,u} = \text{Emb}\left(A_u\right), \quad \mathbf{f}_{A,i} = \text{Emb}\left(A_i\right),$$

where $\text{Emb}(\cdot)$ denotes the embedding function and $\mathbf{f}_{A,u}, \mathbf{f}_{A,i} \in \mathbb{R}^d$ are the resulting student-side and course-side semantic vectors.

By encoding inferred attributes into continuous representations, SORARec enables the integration of high-level educational semantics into the recommendation process while maintaining a lightweight downstream architecture. Notably, the LLM is only used in an offline inference stage, and all subsequent recommendation learning operates on the embedded representations. The utilization of these side information embeddings in the recommendation model is described in the following subsection.

3.2 Side Information Incorporation

After obtaining the LLM-inferred student profiles and course attributes, the key challenge is how to effectively incorporate this information into the course recommendation process. SORARec integrates the inferred student-side and course-side representations with interaction-based learning, enabling the model to jointly exploit collaborative signals

from historical enrollments and the semantic characteristics specific to students and courses.

Let $\mathbf{e}_u \in \mathbb{R}^d$ and $\mathbf{e}_i \in \mathbb{R}^d$ denote the base embeddings of student u and course i, respectively, learned from interaction data. From the LLM-based inference module described in Section 3.2, we obtain semantic representations $\mathbf{f}_{A,u}$ and $\mathbf{f}_{A,i}$, which encode the inferred student profiles and course attributes. The inferred semantic representations are first aligned to the recommender embedding space via a projection operation:

$$\overline{\mathbf{f}_u} = \mathrm{Proj}\left(\mathbf{f}_{A,u}\right), \quad \overline{\mathbf{f}_i} = \mathrm{Proj}\left(\mathbf{f}_{A,i}\right),$$

where $\overline{\mathbf{f}_u}$ and $\overline{\mathbf{f}_i}$ denote the projected semantic embedding obtained by mapping the LLM-inferred representation into the recommender embedding space, $\mathrm{Proj}(\cdot)$ maps the LLM-inferred embeddings from the semantic space to the collaborative embedding space. $\mathrm{Proj}(\cdot)$ is instantiated as a linear transformation, i.e., $\mathrm{Proj}(\mathbf{x}) = \mathbf{x}\mathbf{W}$ with a trainable projection matrix $\mathbf{W}$, and ℓ_2-normalization is applied to mitigate scale discrepancies between heterogeneous representations. Although dimensionally aligned, the projected semantic features do not explicitly encode collaborative structure. To inject interaction-aware context, SORARec further propagates the projected features over the student–course interaction graph $\mathcal{G}$:

$$\tilde{\mathbf{f}}_u = \mathrm{GNN}\left(\overline{\mathbf{f}_u}, \mathcal{G}\right), \quad \tilde{\mathbf{f}}_i = \mathrm{GNN}\left(\overline{\mathbf{f}_i}, \mathcal{G}\right),$$

where $\tilde{\mathbf{f}}_u$ and $\tilde{\mathbf{f}}_i$ represent the context-enhanced semantic embedding after injecting collaborative information through message passing on the interaction graph, and $\mathrm{GNN}(\cdot)$ denotes a message-passing operation that aggregates neighborhood information from $\mathcal{G}$. Finally, the context-enhanced semantic features are incorporated into the base embeddings via a residual-style fusion strategy:

$$\mathbf{h}_u = \mathbf{e}_u + \alpha \cdot \frac{\tilde{\mathbf{f}_u}}{\left\|\tilde{\mathbf{f}_u}\right\|_2}, \quad \mathbf{h}_i = \mathbf{e}_i + \alpha \cdot \frac{\tilde{\mathbf{f}_i}}{\left\|\tilde{\mathbf{f}_i}\right\|_2},$$

where α controls the contribution of semantic side information. The resulting representations $\mathbf{h}_u$ and $\mathbf{h}_i$ jointly capture collaborative signals and learning-relevant semantics, and are subsequently used for preference estimation in the recommendation task.

3.3 Recommendation Objective and Optimization

After incorporating LLM-inferred student profiles and course attributes into the collaborative representations, SORARec performs interaction prediction based on the final student and course embeddings. Given the fused representations $\mathbf{h}_u$ and $\mathbf{h}_i$ obtained in Section 3.3, the preference score between student u and course i is computed via inner product:

$$\hat{y}_{u,i} = \mathbf{h}_u^\top \mathbf{h}_i.$$

Following common practice in implicit-feedback recommendation scenarios, we adopt the Bayesian Personalized Ranking (BPR) [23] loss to optimize the model. The BPR objective encourages the predicted score of an observed interaction to be higher than that of an unobserved one. Specifically, for each student u, we construct training triplets (u, i, j), where i denotes a course that the student has enrolled in, and j denotes a course with no observed interaction. The loss function is defined as:

$$\mathcal{L}_{\mathrm{BPR}} = -\sum_{(u,i,j)\in\mathcal{D}} \log \sigma\left(\hat{y}_{u,i} - \hat{y}_{u,j}\right),$$

where $\sigma(\cdot)$ denotes the sigmoid function and $\mathcal{D}$ is the set of sampled training triplets. The overall objective function includes an ℓ_2 regularization term to prevent overfitting:

$$\mathcal{L} = \mathcal{L}_{\mathrm{BPR}} + \lambda \|\Theta\|_2^2,$$

where Θ represents all trainable parameters in SORARec, including embedding matrices and projection weights, and λ controls the regularization strength. The model is optimized using mini-batch stochastic gradient descent with the Adam [9] optimizer. During inference, SORARec ranks candidate courses for each student based on the predicted scores $\hat{y}_{u,i}$ and recommends the top-K courses with the highest scores.

4. EXPERIMENT SETUP

4.1 Datasets

We evaluate SORARec on a publicly available MOOC dataset, MOOCCube [35], which is a large-scale dataset containing 199,199 users, 706 courses, and 672,853 user-course interaction records. Each course is accompanied by textual metadata, including course titles, detailed descriptions, and manually annotated concepts. Following common practice, we preprocess both datasets by removing users with fewer than five interactions and courses with fewer than ten interactions to ensure data quality and stable evaluation. After preprocessing, the statistics of the resulting dataset are summarized in Table 1.

Table 1: Statistics of the MOOCCube dataset

Dataset	Users	Courses	Interactions	Interactions Density
MOOCCube	33,838	645	263,522	1.2074%

4.2 Metrics

Following prior work on course recommendation [3, 34, 37], we evaluate the recommendation performance using four widely adopted top-K ranking metrics: Precision, Recall, Hit Ratio (HR), and Normalized Discounted Cumulative Gain (NDCG). These metrics assess the quality of the ranked recommendation lists from complementary perspectives, including accuracy, coverage, and ranking quality. For all metrics, higher values indicate better recommendation performance.

4.3 Baselines

To evaluate the effectiveness of SORARec, we compare it with a set of representative baseline methods for course recommendation, covering classical collaborative filtering, factorization-based models, neural recommenders, graph-based approaches, and knowledge-enhanced models. Specifically, we include:

- **Item-based CF** [24], a memory-based collaborative filtering method that recommends courses based on item similarity computed from historical interactions.

Table 2: Performance comparison on the MOOCCube dataset

	K = 5				K = 10			
Model	Precision@5	Recall@5	HR@5	NDCG@5	Precision@10	Recall@10	HR@10	NDCG@10
CF	0.0525	0.1464	0.2326	0.1118	0.0344	0.1890	0.3005	0.1280
MF	0.0341	0.0973	0.1617	0.0655	0.0328	0.1839	0.2923	0.0989
LightFM	0.0492	0.1442	0.2304	0.0950	0.0419	0.2395	0.3651	0.1319
NeuMF	0.0609	0.1719	0.2763	0.1269	0.0469	0.2622	0.3978	0.1608
LightGCN	0.0893	0.2470	0.3782	0.2012	0.0625	0.3433	0.5014	0.2372
KEAM	0.0768	0.2018	0.2651	0.1605	0.0495	0.2581	0.3439	0.1819
SORARec	**0.1021**	**0.2874**	**0.4389**	**0.2740**	**0.0724**	**0.4015**	**0.5711**	**0.3274**

- **Matrix Factorization** [10], a matrix factorization model that learns latent representations of students and courses from the interaction matrix.
- **LightFM** [11], a hybrid recommendation model that combines collaborative filtering with side information; we report results under its standard BPR loss.
- **NeuMF** [6], a neural collaborative filtering model that integrates generalized matrix factorization and multilayer perceptrons to capture nonlinear interactions.
- **LightGCN** [5], a graph-based collaborative filtering model that propagates embeddings over the student–course interaction graph using simplified graph convolutions.
- **KEAM** [33], a knowledge-enhanced course recommendation model that incorporates course concepts through an autoencoder-based architecture.

All baseline models are trained and evaluated under the same data splits and evaluation protocol. Hyperparameters are carefully tuned on the validation set, and the best-performing configurations are reported on the test set.

4.4 Training Details

All models were implemented using PyTorch 2.1.2 and trained on a single NVIDIA RTX 4060 Ti GPU. For the MOOCCube dataset, user–course interaction records were randomly split into training, validation, and test sets with a ratio of 8:1:1. For SORARec, LLM-based inference was conducted offline. Student profiles and course attributes were generated using GPT-4o-mini, and the textual outputs were encoded with the text-embedding-3-small model, producing 1536-dimensional semantic embeddings. These embeddings were projected into a 64-dimensional latent space to align with interaction-based representations. All models were evaluated under the same data splits and experimental settings.

5. RESULTS

5.1 Overall Performance Comparison

Table 2 presents the overall performance comparison on the MOOCCube dataset. The main observations are summarized as follows: (1) SORARec consistently achieves the best performance across all evaluation metrics at both $K = 5$ and $K = 10$. This demonstrates the effectiveness of jointly incorporating LLM-inferred student profiles and course attributes into the recommendation process. (2) The improvements indicate that learning-aware semantic representations help SORARec better capture students' latent learning needs and course suitability, which are difficult to infer from sparse interaction data alone. (3) Compared with strong collaborative and knowledge-enhanced baselines, SORARec still yields clear performance gains, highlighting the importance of simultaneously modeling both student-side and course-side information in educational recommendation. (4) Overall, these results confirm that SORARec provides a robust solution for MOOC recommendation by alleviating data sparsity and improving representation quality.

5.2 Ablation Study

To examine the contribution of LLM-inferred student profiles and course attributes in SORARec, we conduct an ablation study by selectively removing semantic information from different sides of the model. Specifically, we compare the following variants:

- **SORARec w/o SP**: This variant removes the student profiles and retains only course attributes.
- **SORARec w/o CA**: This variant removes the course attributes and retains only student profiles.
- **SORARec w/o SP&CA**: This variant removes both student profiles and course attributes, reducing the model to an interaction-only variant.

These comparisons help demonstrate the importance and impact of each component in the overall SORARec model performance. Based on the results presented in Figure 3, we observe that (1) The SORARec model consistently achieves the best performance across all metrics, demonstrating that jointly modeling student profiles and course attributes is more effective than relying on either side alone. (2) Removing either student profiles or course attributes leads to noticeable performance degradation, indicating that both components contribute meaningful and complementary information. (3) The performance drop is slightly larger when removing student profiles than when removing course attributes, suggesting that student-side semantics play a particularly important role in educational recommendation. This aligns with the nature of learning scenarios, where students' background knowledge, learning goals, and difficulty preferences strongly influence course suitability. (4) The interaction-only variant (w/o SP & CA) performs the worst, highlighting the limitation of relying solely on sparse enrollment sig-

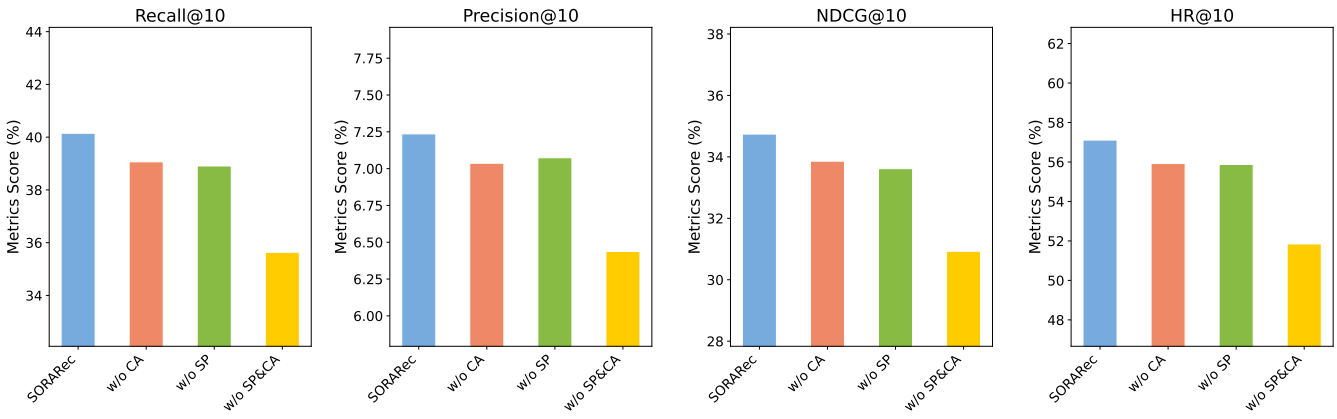


Figure 3: The effectiveness of each component in SORARec.

Table 3: Cold-Start recommendation performance comparison on the MOOCCube dataset

	K = 5		K = 10		K = 15		K = 20	
Model	HR@5	NDCG@5	HR@10	NDCG@10	HR@15	NDCG@15	HR@20	NDCG@20
Item-based CF	0.0614	0.0570	0.1933	0.0971	0.2495	0.1118	0.3086	0.1259
Matrix Factorization	0.1064	0.0618	0.1996	0.0902	0.2824	0.1129	0.3450	0.1266
LightFM	0.2259	0.1442	0.3012	0.1689	0.3494	0.1809	0.3765	0.1862
NeuMF	0.2172	0.1371	0.3140	0.1690	0.3763	0.1848	0.4248	0.1962
LightGCN	0.3062	0.2128	0.3984	0.2406	0.4631	0.2577	0.5184	0.2707
KEAM	0.3398	0.2585	0.4095	0.2811	0.4446	0.2904	0.4689	0.2962
SORARec	**0.3612**	**0.2736**	**0.4543**	**0.3036**	**0.5064**	**0.3174**	**0.5457**	**0.3267**

nals. Overall, this ablation study confirms that student profiles and course attributes are both indispensable components of SORARec, and the results further support the importance of explicitly modeling learning-oriented semantics on both sides of the recommendation process.

5.3 Cold-Start Scenario Recommendation

Following the same cold-start setting adopted in prior work [31], we focus on the user cold-start scenario on the MOOCCube dataset by selecting users with no more than five historical interactions. The experimental results are reported in Table 3. The main findings are summarized as follows: (1) SORARec consistently achieves the best performance under cold-start conditions across all Top-K settings. In particular, SORARec outperforms all baselines in both HR and NDCG, demonstrating its strong robustness when user interaction histories are extremely limited. (2) The performance advantage of SORARec becomes more evident as K increases. At $K = 10$, 15, and 20, SORARec shows clear and stable improvements over strong baselines such as LightGCN and KEAM, indicating its superior ability to rank relevant courses higher for cold-start users. (3) Compared with interaction-driven models, SORARec benefits from LLM-inferred student profiles and course attributes, which provide learning-aware semantic signals beyond sparse behavioral data. These semantic representations effectively compensate for missing interactions and enable more reliable preference estimation in cold-start scenarios. (4) Overall, the results confirm that SORARec is particularly well-suited for user cold-start recommendation. By jointly leveraging collaborative information and inferred educational semantics, SORARec substantially improves recommendation performance under sparse data conditions. While all models benefit from sufficient interaction histories in the general setting (Section 5.1), cold-start evaluation provides a clearer view of how effectively learning-aware semantic representations can compensate for missing behavioral signals.

6. CONCLUSIONS

In this paper, we proposed SORARec and explored the use of large language models to enhance course recommendation by enriching both the student side and the course side with inferred semantic information. Specifically, we leveraged LLMs to generate structured student profiles from historical learning behaviors and to extract learning-relevant attributes from course descriptions, and incorporated these signals into the recommendation process. Experimental results show that augmenting student and course representations with LLM-inferred information leads to consistent performance improvements over baseline methods, particularly in sparse interaction settings. Rather than treating LLMs as end-to-end recommenders, our findings suggest that their strength in this context lies in supporting the understanding and modeling of students and courses, thereby complementing data-driven recommendation models.

However, this work has several limitations pointing to directions for future research. First, the current framework focuses on augmenting student profiles and course attributes, but does not explicitly address interaction-level completion.

A natural extension is to allow LLMs to predict potential liked and disliked courses by jointly considering student profiles and course attributes, thereby enriching the interaction space under sparse or cold-start conditions. Second, LLM-generated content is inherently subject to hallucination and uncertainty, which may introduce noise into the recommendation process. Future work will explore mechanisms to mitigate these effects, such as constrained generation, consistency checking across multiple prompts, and verification against existing course information. Finally, our evaluation relies on offline recommendation metrics, which cannot fully reflect the educational validity of the generated profiles and recommendations. We therefore plan to conduct user studies involving students and instructors, asking them to assess the accuracy, plausibility, and usefulness of the inferred student profiles, course attributes, and predicted preferences. Such human-centered evaluation will be crucial for understanding the practical value and limitations of LLM-augmented course recommendation in real educational settings.

7. ACKNOWLEDGMENTS

This work was supported by JST SPRING, Grant Number JPMJSP2136, JSPS KAKENHI Grant Numbers JP20H00622, JP24K20903 and JP25K03230.

8. REFERENCES

[1] A. Barany, N. Nasiar, C. Porter, A. F. Zambrano, A. L. Andres, D. Bright, M. Shah, X. Liu, S. Gao, J. Zhang, et al. Chatgpt for education research: exploring the potential of large language models for qualitative codebook development. In *International conference on artificial intelligence in education*, pages 134–149. Springer, 2024.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[3] J. Frej, N. Shah, M. Knezevic, T. Nazaretsky, and T. Käser. Finding paths for explainable mooc recommendation: A learner perspective. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 426–437, 2024.

[4] W. Gao, Q. Liu, Z. Huang, Y. Yin, H. Bi, M.-C. Wang, J. Ma, S. Wang, and Y. Su. Rcd: Relation map driven cognitive diagnosis for intelligent education systems. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 501–510, 2021.

[5] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.

[6] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

[7] F. Huang, Y. Bei, Z. Yang, J. Jiang, H. Chen, Q. Shen, S. Wang, F. Karray, and P. S. Yu. Large language model simulator for cold-start recommendation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pages 261–270, 2025.

[8] X. Jing and J. Tang. Guess you like: course recommendation in moocs. In *Proceedings of the international conference on web intelligence*, pages 783–789, 2017.

[9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[11] M. Kula. Metadata embeddings for user and item cold-start recommendations. In T. Bogers and M. Koolen, editors, *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems (RecSys 2015), Vienna, Austria, September 16-20, 2015.*, volume 1448 of *CEUR Workshop Proceedings*, pages 14–21. CEUR-WS.org, 2015.

[12] J. Lin, E. Chen, Z. Han, A. Gurung, D. R. Thomas, W. Tan, N. D. Nguyen, and K. R. Koedinger. How can i improve? using gpt to highlight the desired and undesired parts of open-ended responses. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 236–250, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[13] B. Ma, M. A. Z. Khan, T. Yang, A. Polyzou, and S. Konomi. Evaluating the effectiveness of large language models for course recommendation tasks. In *International Conference on Computers in Education*, 2025.

[14] B. Ma, M. Lu, Y. Taniguchi, and S. Konomi. Courseq: the impact of visual and interactive course recommendation in university environments. *Research and practice in technology enhanced learning*, 16:1–24, 2021.

[15] B. Ma, Y. Taniguchi, and S. Konomi. Course recommendation for university environments. *International educational data mining society*, 2020.

[16] B. Ma, T. Yang, and B. Ren. A survey on explainable course recommendation systems. In N. A. Streitz and S. Konomi, editors, *Distributed, Ambient and Pervasive Interactions*, pages 273–287, Cham, 2024. Springer Nature Switzerland.

[17] H. Ma, J. Zhu, S. Yang, Q. Liu, H. Zhang, X. Zhang, Y. Cao, and X. Zhao. A prerequisite attention model for knowledge proficiency diagnosis of students. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4304–4308, 2022.

[18] R. Morsomme and S. V. Alferez. Content-based course recommender system for liberal arts education. *International educational data mining society*, 2019.

[19] S. Morsy, G. Karypis, et al. Will this course increase or decrease your gpa? towards grade-aware course recommendation. *Journal of Educational Data Mining*, 11(2):20–46, 2019.

[20] J. Naren, M. Z. Banu, and S. Lohavani.

Recommendation system for students' course selection. In *Smart Systems and IoT: Innovations in Computing: Proceeding of SSIC 2019*, pages 825–834. Springer, 2020.

[21] A. Polyzou, A. N. Nikolakopoulos, and G. Karypis. Scholars walk: A markov chain framework for course recommendation. *International Educational Data Mining Society*, 2019.

[22] J. Rao, C. Borchers, and J. Lin. Coursera-rec: Explainable moocs course recommendation using rag-facilitated llms. 2024. Accessed: 2025-06-22.

[23] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, pages 452–461, 2012.

[24] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295, 2001.

[25] J. Wang, H. Lu, J. Caverlee, E. H. Chi, and M. Chen. Large language models as data augmenters for cold-start item recommendation. In *Companion Proceedings of the ACM Web Conference 2024*, pages 726–729, 2024.

[26] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022.

[27] W. Wei, X. Ren, J. Tang, Q. Wang, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang. Llmrec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 806–815, 2024.

[28] L. Wu, Z. Zheng, Z. Qiu, H. Wang, H. Gu, T. Shen, C. Qin, C. Zhu, H. Zhu, Q. Liu, et al. A survey on large language models for recommendation. *World Wide Web*, 27(5):60, 2024.

[29] T. Yang, B. Ren, C. Gu, B. Ma, T. He, and S. Konomi. Towards better course recommendations: integrating multi-perspective meta-paths and knowledge graphs. In *Proceedings of the 15th international learning analytics and knowledge conference*, pages 137–147, 2025.

[30] T. Yang, B. Ren, C. Gu, B. Ma, and S. Konomi. Leveraging gpt for concept generation and knowledge graph construction in educational recommender systems. In *Teaching and Learning in the Generative Artificial Intelligence Age*, pages 75–96. Springer, 2026.

[31] T. Yang, B. Ren, C. Gu, F. Xu, B. Ma, and S. Konomi. Enhancing course recommendation with llm-generated concepts: A unified framework for side information integration. *Big Data and Cognitive Computing*, 9(12):311, 2025.

[32] T. Yang, B. Ren, B. Ma, T. He, C. Gu, and S. Konomi. Boosting course recommendation explainability: A knowledge entity aware model using deep learning. In *Proceedings of the 32nd International Conference on Computers in Education, ICCE2024*, pages 360–366. Asia-Pacific Society for Computers in Education, 2024.

[33] T. Yang, B. Ren, B. Ma, M. A. Z. Khan, T. He, and S. Konomi. Making course recommendation explainable: A knowledge entity-aware model using deep learning. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 658–663, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[34] Y. Yang, C. Zhang, X. Song, Z. Dong, H. Zhu, and W. Li. Contextualized knowledge graph embedding for explainable talent training course recommendation. *ACM Trans. Inf. Syst.*, 42(2), sep 2023.

[35] J. Yu, G. Luo, T. Xiao, Q. Zhong, Y. Wang, W. Feng, J. Luo, C. Wang, L. Hou, J. Li, Z. Liu, and J. Tang. MOOCCube: A large-scale data repository for NLP applications in MOOCs. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3135–3142, Online, July 2020. Association for Computational Linguistics.

[36] X. Yu, Q. Mao, X. Wang, Q. Yin, X. Che, and X. Zheng. Cr-lcrp: Course recommendation based on learner–course relation prediction with data augmentation in a heterogeneous view. *Expert Systems with Applications*, 249:123777, 2024.

[37] J. Zhang, B. Hao, B. Chen, C. Li, H. Chen, and J. Sun. Hierarchical reinforcement learning for course recommendation in moocs. In *Proc. AAAI conf artificial intelligence*, volume 33(1), pages 435–442, 2019.

Cross-Context Urgency Detection in MOOC Forums

Shreya Singhal
Indian Institute of Technology, Madras
singhalshreya201@gmail.com

ABSTRACT

Urgency detection in online course forums aims to identify posts that need a timely instructor response. A key challenge is cross-context generalization: models trained on Western MOOC forums may not work well in non-Western settings where help-seeking norms and urgency cues differ. This study trains an urgency model on a Western MOOC dataset and evaluates it on a second Western benchmark and a new Indian dataset of 305 forum posts (two-coder weighted κ=0.7308). Using a 3-level actionability mapping (non-actionable, TF-actionable, instructor-actionable), the Western-trained model performs worse in the Indian setting than in the Western benchmark (Stanford ρ=0.381 vs. Indian ρ=0.292) and retrieves few instructor-actionable posts under limited review (Indian recall@50=0.194). Class balancing yields a small gain (Indian ρ=0.306), while adding a small amount of labeled Indian training data improves performance on a held-out Indian test split (ρ=0.401; recall@50=0.511) without materially changing Western benchmark dataset results.

Keywords

urgency detection, discussion forums, MOOCs, cross-context generalization, class imbalance

1. INTRODUCTION

Discussion forums are a primary channel for learner support in MOOCs and other large-scale online courses. Learners use them to ask questions, report technical issues, and seek clarification, but at scale they also create a recurring bottleneck where teaching teams must decide what to respond to first. Some posts describe time-sensitive blockers (e.g., access problems, deadline constraints, exam logistics, or escalating confusion) where delayed help can meaningfully affect learning progress. This motivates urgency detection: prioritizing posts that need faster staff attention under limited capacity.

Educational data mining has increasingly treated forum discourse as actionable data, enabling instructor-facing support tools that go beyond participation summaries. Prior work has studied closely related prioritization problems such as predicting when an instructor is likely to intervene in a discussion thread, using a combination of language cues and discussion structure [5,6].

A key concern is generalization. Urgency is shaped not only by what is being asked, but also by how learners express time pressure and needs which are patterns that can vary across platforms, course designs, and broader communication norms. Most models are built and validated within relatively narrow settings, which makes it unclear whether they remain reliable when deployed in new contexts. This matters because the highest-urgency, instructor-actionable cases are often the most operationally important to detect yet may be sparse in training data.

This paper frames urgency detection explicitly as a cross-context robustness problem. It extends the work of Švábenský et al. [12] and evaluates whether urgency models trained in a Western MOOC context transfer to (i) a second Western benchmark dataset and (ii) a non-Western MOOC forum context, and it examines two pragmatic strategies for improving reliability in the new setting: incorporating a small amount of labeled target-context data and addressing class imbalance for instructor-actionable posts.

We address the following research questions:

RQ1: How well do urgency detection models trained on Western MOOC forum data generalize to a non-Western MOOC forum context?

RQ2: How does class imbalance affect model performance under cross-context transfer, especially for instructor-actionable posts?

RQ3: To what extent does adding a small amount of labeled non-Western data to the training data improve model performance?

2. RELATED WORK

2.1 Urgency detection in MOOC discussion forums

Urgency detection is typically framed as predicting which learner posts need timely staff attention in large-scale online courses. Much of the early work used the Stanford MOOCPosts benchmark [1] and simplified the 7-point urgency construct into a binary "urgent vs. not urgent" decision, highlighting that truly urgent cases are rare and easy to miss in large volumes of routine posts [2]. This line of work positions urgency as an instructor-support problem: models are valuable only insofar as they help teaching teams prioritize limited response capacity.

More recent work has emphasized that strong performance on a single benchmark can overstate real-world usefulness when course topics, learner populations, or platform norms change. Švábenský et al. [12] made this concern explicit by evaluating urgency prediction under dataset shift and by treating urgency as an ordered construct rather than only a binary cutoff, clarifying that generalization is a central challenge for deployment.

Urgency detection is also related to the broader problem of predicting instructor intervention in MOOC discussion forums. Chaturvedi et al. [6] studied whether instructor intervention could be predicted from MOOC forum activity and showed that intervention prediction depends on signals beyond the post text itself, including course and discussion context. Chandrasekaran et al. [5] similarly examined instructor intervention across MOOC forum threads and highlighted the role of forum structure and

Shreya Singhal. Cross-Context Urgency Detection in MOOC Forums. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 430–436. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039889

instructional context in determining when staff responses are needed. These studies show that actionability may not be a property of post text alone. Whether a post requires a teaching assistant or instructor response can depend on the course design, available support channels, learner expectations, and local instructional practices.

2.2 From feature engineering to embedding-based models

Modeling approaches for urgency detection have broadly followed trends in NLP for educational text: from lexical feature baselines to representation learning with neural embeddings. For example, Guo et al. [7] proposed a deep learning approach for identifying urgent posts in MOOC forums, and Khodeir [8] used BERT-based text representations with a recurrent architecture for urgency classification. These approaches aim to capture meaning beyond surface word counts and to be more tolerant of noisy forum writing.

A parallel strand incorporates additional signals beyond raw text, such as post metadata or learner/context cues, motivated by the idea that urgency is not only "what is said" but also "when and how it is asked." Alrajhi et al. [3] combined textual and numerical features for predicting urgent responses, and Yu et al. [13] explored Bayesian deep learning to estimate instructor intervention need with uncertainty, reflecting interest in decision support settings where false negatives on high-priority posts are costly.

2.3 Generalizability, class imbalance, and evaluation gaps

Two persistent issues recur across the literature: (i) limited cross-context generalization and (ii) severe class imbalance. Urgent posts are typically a small minority in Western MOOC benchmarks [2], which can encourage models that achieve good average error while still failing on the cases that matter most operationally. This motivates both evaluation protocols that stress robustness under shift [12] and training strategies that explicitly address imbalance. For example, Alrajhi et al. [4] focus on urgency detection under imbalance and study methods intended to improve detection of the minority class.

Finally, there is increasing recognition that "stronger" NLP architectures do not automatically translate into better educational utility without careful evaluation. Work on broader educational forum-post classification emphasizes systematic comparisons and robustness-focused metrics rather than assuming newer models dominate by default [9,10]. This motivates evaluations that reflect instructor-facing workflows (e.g., whether high-priority posts appear near the top of a ranked list), not only aggregate error.

Building directly on the cross-dataset framing in Švábenský et al. [12], this paper targets the remaining gap between benchmark performance and deployment-relevant reliability under context shift by evaluating transfer to a non-Western forum setting and by testing pragmatic strategies that can be applied when only limited target-context labels are available.

3. RESEARCH METHODS

This section describes the data and procedures used to train and evaluate predictive models of discussion-forum post urgency. This paper's methodology adapts the full pipeline from Towards Generalizable Detection of Urgency of Discussion Forum Posts [12] including the 7-point ordinal urgency construct, coder protocol, lightweight NLP preprocessing, and cross-dataset generalizability evaluation. It then extends it in two ways that align with the motivation and gaps: (1) evaluation on an additional non-Western MOOC forum dataset labeled on the same scale, and (2) two deployment-relevant strategies intended to improve robustness under context shift: low-resource target-domain adaptation and class-imbalance mitigation.

3.1 Data Collection and Properties

This paper reuses the datasets used by Švábenský et al. [12]: (1) UPenn MOOC forum-post dataset (used here as the main training dataset), and (2) the Stanford MOOCPosts dataset (used here as a separate Western test dataset). In addition, this paper introduces a new Indian MOOC forum-post dataset to test how well models trained on Western MOOC posts work in a different context.

UPenn MOOC forum posts (Western training dataset). Following Švábenský et al. (2023), the training data consists of forum posts from nine UPenn MOOCs offered between 2012–2015. The original dataset construction sampled 500 posts per course and removed posts that were non-English or contained only symbols, only math, or only links, resulting in 3,503 posts from 2,882 students. Each post includes a student ID, timestamp, and post text. Typos and grammatical errors were kept as written.

Stanford MOOCPosts (Western test dataset). Performance is evaluated on the Stanford MOOCPosts dataset, a benchmark dataset introduced by Agrawal and Paepcke [1]. It contains 29,604 anonymized discussion-forum posts collected from 11 Stanford MOOCs. The 11 courses are grouped into three broad areas (Humanities/Sciences, Medicine, and Education) with approximately 10,000 posts per area. This benchmark is used here as a separate Western test dataset, consistent with its use for generalization testing in Švábenský et al. [12].

Indian MOOC forum posts (Non-Western dataset). A new dataset of 305 forum posts was collected from the Discourse forum of an online MOOC degree program in India. Posts were scraped from discussions occurring between 2024-2025 and were collected with the following fields: post ID, title, timestamp, and course ID. The raw text was cleaned by removing HTML and excluding posts that were not in English. The final sample includes posts drawn from four data science–related courses (Programming in Python; Statistics for Data Science; Business Data Management; Strategies for Professional Growth), with roughly 75 posts sampled per course. This dataset is used to evaluate performance in a new setting and to support later experiments that add a small amount of Indian training data.

Table 1 summarizes the three datasets, including their sources, number of courses, number of posts, and collection period.

Table 1. Summary of datasets used in this study

Dataset	Source	Courses	Posts	Time span
UPenn MOOC forums	Švábenský et al. [12]	9	3,503	2012–2015
Stanford MOOCPosts	Agrawal & Paepcke [1]	11	29,604	2013-2014
Indian MOOCPosts	Discourse forum of a MOOC degree program in India	4	305	2024–2025

3.2 Data Anonymization

The UPenn and Stanford datasets were used in their already de-identified forms [12], without additional anonymization. For the Indian dataset, personally identifying information was redacted during scraping using the GPT-4-0613 model-based procedure in Singhal et al. [11]. Each post was processed individually using the same prompt structure, which instructs the model to redact specific categories of PII and preserve the original text formatting. It was further checked during the labeling phase by two coders.

3.3 Data Labeling

This study adopts the urgency construct and labeling protocol used in [12] and then refines it. Urgency is defined as the degree to which an instructor response is needed quickly, expressed on a 7-point ordinal scale (1 = not urgent at all, 7 = extremely urgent) [12]. The same scale descriptions are used to ensure direct comparability across datasets and settings. Ordinal scores correspond to the following anchors:

1: No reason to read the post

2: Not actionable, read if time

3: Not actionable, may be interesting

4: Neutral, respond if spare time

5: Somewhat urgent, good idea to reply; a teaching assistant might suffice

6: Very urgent, good idea for the instructor to reply

7: Extremely urgent, instructor definitely needs to reply

For the Indian dataset, posts are labeled using the same scale and training structure as [12]: coders complete structured practice on held-out examples, then label posts independently. Inter-rater reliability is assessed using weighted Cohen's κ (linear weights), consistent with the ordinal nature of the scale [12]. In this study, the Indian dataset was labeled by 2 coders and achieved $\kappa = 0.7308$. Disagreements were resolved via adjudication (discussion to consensus) to produce a single final label per post.

After the first round of labeling, it was observed that the urgency labels are imbalanced in both UPenn and Stanford datasets, with substantially more posts at the lower and middle urgency levels and comparatively fewer posts at the highest urgency levels (6–7). This skew is important because the rare high-urgency cases are the most instructionally actionable, yet they are underrepresented in the data. On the other hand, the Indian dataset contained a much higher proportion of high-urgency posts.

To align the prediction task with an instructor-facing triage workflow and reduce sparsity at the upper end of the urgency scale, the original 7-point urgency annotations were mapped into three actionability categories, as shown in Table 2. The 7-point scale was used for annotation and comparability with prior work, while the 3-level actionability labels were used for all model training and evaluation reported in this paper. This mapping preserves the key operational distinction in the original scale: whether a post is non-actionable, teaching-assistant-actionable, or instructor-actionable. The resulting 3-level scheme provides more stable categories for modeling and evaluation under severe class imbalance, and all reported metrics should therefore be interpreted on this 1–3 actionability scale rather than on the original 1–7 urgency scale.

Table 2. Distribution of actionability labels derived from the 7-point urgency scale.

Label	Meaning	UPenn (N=3,503)	Stanford (N=29,604)	Indian (N=305)
1	Non-actionable (1–3)	2,830 (80.8%)	21,806 (73.7%)	78 (25.6%)
2	TF-actionable (4–5)	623 (17.8%)	5,313 (17.9%)	134 (43.9%)
3	Instructor-actionable (6–7)	50 (1.4%)	2,485 (8.4%)	93 (30.5%)

3.4 Data Automated Pre-Processing

A lightweight preprocessing pipeline matching Švábenský et al. [12] was used to reduce superficial variation while retaining the linguistic cues that may signal urgency. The following steps were applied in order:

- Convert text to lowercase.
- Replace non-alphanumeric characters with spaces (preserving letters and digits).
- Normalize whitespace (remove duplicates).
- Remove common English stopwords.
- Apply stemming to reduce inflected forms to a shared root.

Spelling and grammar correction were not performed intentionally, as these artifacts may differ systematically across contexts.

3.5 Model Training and Cross-Validation

This section describes the shared model-training and validation procedure used to answer RQ1 (cross-context generalization), RQ2 (class-imbalance mitigation), and RQ3 (low-resource target-domain adaptation). Following the design principle established in Sections 3.1–3.4, the modeling pipeline holds the base learner and representation fixed to the strongest-performing configuration in [12] and varies only the training procedure for imbalance handling (RQ2) and training data composition (RQ3).

3.5.1 Model and feature representation

Urgency is modeled using support vector regression (SVR) with an RBF kernel, trained on Universal Sentence Encoder v4 (USE) embeddings of post text. SVR hyperparameters are left at default settings, consistent with the baseline configuration in [12]. Each post is embedded using USE v4 (TensorFlow Hub) to obtain a fixed-length dense representation, which is used directly as the input feature vector for SVR. The purpose of this study is not to establish a new state-of-the-art model for urgency detection, but to test cross-context robustness under a controlled and comparable modeling pipeline. For this reason, the USE + SVR configuration is held fixed across experiments. This design makes it possible to attribute changes in performance primarily to evaluation context, class-imbalance mitigation, and target-context supervision rather than to changes in model architecture.

3.5.2 Cross-validation protocol (student-level grouping)

Model selection and in-domain validation use 10-fold student-level cross-validation implemented with GroupKFold, where the grouping variable is the student identifier. This ensures that all

posts from a given student occur in a single fold, preventing leakage across train and validation splits. For all cross-validation results, metrics are computed on the held-out fold and then averaged across folds.

For any setting in which training data combines UPenn and Indian examples (RQ3), student identifiers from the Indian dataset are offset by a large constant prior to grouping to avoid accidental ID collisions across datasets.

3.5.3 Training and evaluation settings by research question

RQ1: Cross-context generalization (Western → non-Western).

For RQ1, the model is trained using the full UPenn training dataset. Ten-fold student-level cross-validation is computed on UPenn. After cross-validation, the model is refit on the full UPenn training set and evaluated separately on the two external datasets: Stanford MOOCPosts (Western benchmark) and the Indian dataset (non-Western setting). External evaluations are performed without combining the two test datasets.

RQ2: Class-imbalance mitigation (in-fold SMOTE).

RQ2 evaluates whether mitigating label imbalance improves performance while holding the learner and representation fixed. In this condition, SMOTE oversampling is applied in the USE embedding space to rebalance the refined 3-level labels (Section 3.3). Oversampling is performed only on the training split within each cross-validation fold; validation folds remain untouched. Concretely, within each GroupKFold split, SMOTE is applied to the training fold to generate synthetic minority examples, SVR is trained on the resampled training fold, and evaluation is performed on the original validation fold.

Because minority-class counts vary across folds, the SMOTE neighborhood parameter is set adaptively per fold. If any class has fewer than two examples in the training fold, SMOTE is not well-defined; in that case, oversampling is skipped for that fold and SVR is trained on the unresampled training fold.

After cross-validation, a final RQ2 model for external testing is trained by applying SMOTE once to the full UPenn training set (again in embedding space) and fitting SVR on the resulting oversampled set. This model is then evaluated separately on Stanford MOOCPosts and the Indian dataset. No resampling is applied to any test set.

RQ3: Low-resource target-domain adaptation.

For RQ3, the goal is to test whether adding a small amount of labeled target-domain data improves performance in the Indian setting. The Indian dataset is split 50/50 into an Indian-train subset and an Indian-test subset using a stratified split over the refined 3-level urgency labels (Section 3.3) to preserve label proportions across the two halves. The training pool is then formed by concatenating all UPenn training examples with the Indian-train subset. Ten-fold student-level GroupKFold is run on this combined training pool. After cross-validation, the model is refit on the full combined training pool and evaluated separately on Stanford MOOCPosts and the held-out Indian-test subset.

3.6 Model Generalizability Evaluation

Model generalizability is evaluated by measuring performance under dataset shift and low-resource target-data inclusion, aligned with RQ1–RQ3. All reported results are computed on held-out data only (validation folds in cross-validation and external test sets). No resampling or rebalancing is applied to any validation or test split.

Performance is summarized using root mean squared error (RMSE) and Spearman's rank correlation (ρ) between predicted urgency and gold labels. RMSE captures average prediction error magnitude, while Spearman's ρ measures ordinal agreement and is appropriate for urgency as an ordered construct. Predictions are treated as continuous regression outputs and are not rounded prior to computing either metric.

Since the intended use case is prioritization under limited staff time, evaluation additionally includes top-k precision and recall for the highest actionability level. For each evaluation split, posts are ranked by predicted urgency (descending). "Instructor-actionable" corresponds to Level 3 in the refined label scheme (Section 3.3). Precision@k is the fraction of the top-k ranked posts that are instructor-actionable and Recall@k is the fraction of all instructor-actionable posts that appear in the top-k. Unless otherwise noted, results report k = 10 and 50 to reflect small and moderate review budgets.

For training settings that use the UPenn dataset, in-domain performance is estimated via 10-fold student-level cross-validation (Section 3.5). RMSE, Spearman's ρ, and top-k precision/recall are computed on each held-out fold and then averaged across folds.

Generalizability across contexts is assessed through external evaluation on Stanford MOOCPosts and the Indian dataset. For RQ1, a model trained on UPenn is evaluated separately on Stanford and on the full Indian dataset. For RQ2, the class-balancing condition (SMOTE applied during training, Section 3.5) is evaluated separately on Stanford and on the full Indian dataset. For RQ3, the low-resource target-data condition is evaluated on Stanford and on the held-out Indian-test split (with the Indian 50/50 split defined in Section 3.5). Across conditions, results are reported for (i) UPenn cross-validation (when applicable), (ii) Stanford test performance, and (iii) Indian test performance, using RMSE, Spearman's ρ, and the top-k instructor-actionable precision/recall metrics defined above.

4. RESULTS AND DISCUSSION

This section reports findings for RQ1–RQ3 using the refined 3-level actionability labels from Section 3.3 (1 = non-actionable, 2 = TF-actionable, 3 = instructor-actionable). Overall predictive performance is summarized with RMSE and Spearman's ρ. To connect performance to an instructor-facing workflow, triage utility is summarized using precision@10 (concentration of instructor-actionable posts in the top 10) and recall@50 (coverage of instructor-actionable posts when reviewing the top 50).

Table 3. Predictive performance and triage utility across research questions (instructor-actionable = label 3)

RQ	Training setup	CV (RMSE / ρ)	Stanford (RMSE / ρ)	Indian (RMSE / ρ)	Indian p@10	Indian r@50
RQ 1	UPenn only	**0.3832 / 0.4465**	**0.5895 / 0.3812**	**1.0061 / 0.2918**	0.40	0.194
RQ 2	UPenn + in-fold SMOTE	**0.3892 / 0.4345**	**0.5900 / 0.3719**	**0.9822 / 0.3064**	0.60	0.237

RQ	Training setup	CV (RMSE / ρ)	Stanford (RMSE / ρ)	Indian (RMSE / ρ)	Indian p@10	Indian r@50
RQ3	UPenn + Indian-train (50%)	**0.4028 / 0.4891**	**0.5888 / 0.3805**	**0.6932 / 0.4012**	0.600	0.511

All results are computed using the refined 3-level actionability labels, not the original 7-point annotation scale. SVR predictions are continuous and are not rounded before computing RMSE or Spearman's ρ. Triage metrics are computed for label 3 (instructor-actionable) only. The Indian column is evaluated on the full Indian dataset for RQ1–RQ2 (N = 305; instructor-actionable = 93) and on the held-out Indian-test split for RQ3 (N = 153; instructor-actionable = 47).

4.1 RQ1: Cross-context generalization

The RQ1 results show that the UPenn-trained model transfers better to the Stanford benchmark than to the Indian dataset. On Stanford, the model achieves RMSE = 0.5895 and Spearman's ρ = 0.3812. On the Indian dataset, performance drops to RMSE = 1.0061 and Spearman's ρ = 0.2918. The triage-oriented results show a similar pattern: Indian precision@10 is 0.40 and recall@50 is 0.194, meaning that the model retrieves only a small share of instructor-actionable posts within a limited review budget.

One likely reason for this drop is the difference in actionability distributions. In the UPenn training data, instructor-actionable posts are rare, so the model learns urgency patterns from a setting where high-actionability cases are uncommon. In the Indian dataset, instructor-actionable posts are much more frequent and may also be expressed through different linguistic and pragmatic cues. As a result, the model is being asked to prioritize a different mix of learner needs than the one represented in its training data. This suggests a context-shift explanation rather than a purely "harder dataset" explanation.

These results should be interpreted as evidence of cross-context sensitivity, not as evidence of culture alone. The Indian dataset differs from the Western benchmarks along several dimensions, including platform, institutional setting, course design, learner population, and actionability distribution. The lower transfer performance suggests that a model trained in one benchmark setting may not fully capture the local cues that define instructor actionability in another.

4.2 RQ2: Class-imbalance mitigation

The RQ2 results show that SMOTE produces a small but limited improvement in the Indian setting. Compared with the UPenn-only model, the SMOTE model lowers Indian RMSE from 1.0061 to 0.9822 and increases Spearman's ρ from 0.2918 to 0.3064. The triage-oriented metrics also improve as precision@10 rises from 0.40 to 0.60 and recall@50 increases from 0.194 to 0.237. These results suggest that class rebalancing helps the model move more instructor-actionable posts toward the top of the ranked list, which is useful for an instructor-facing triage workflow.

However, the improvement is limited. The SMOTE model still performs substantially worse on the Indian dataset than on the Stanford benchmark, so class imbalance cannot be the whole explanation for the transfer gap. SMOTE can rebalance the source training distribution, but it cannot introduce target-context cues that are missing from the UPenn data, such as how learners in the Indian setting signal urgency, what kinds of requests are common, or how problems are framed in that course and platform environment. In other words, increasing exposure to rare instructor-actionable cases helps, but it does not necessarily solve cross-context generalization.

This distinction matters because urgency detection is ultimately a triage task. A false positive may ask staff to review a lower-priority post, but a false negative may leave an instructor-actionable issue unresolved. The increase in precision@10 is therefore encouraging because it means the top of the ranked list contains more instructor-actionable posts. At the same time, recall@50 remains low, meaning that many instructor-actionable posts are still not retrieved within a realistic review budget. This is why top-k precision and recall are important alongside RMSE and Spearman's ρ, because aggregate score accuracy can miss whether the model is helping staff find the posts that most need timely attention.

4.3 RQ3: Low-resource target-domain adaptation.

The RQ3 results show that adding a small amount of labeled Indian data produces the strongest improvement in the Indian setting. Compared with the UPenn-only model, the UPenn + Indian-train model lowers Indian RMSE from 1.0061 to 0.6932 and increases Spearman's ρ from 0.2918 to 0.4012. The triage metrics also improve substantially as precision@10 rises from 0.40 to 0.60 and recall@50 increases from 0.194 to 0.511. This means that the adapted model retrieves more than half of the instructor-actionable posts in the held-out Indian test set within the top 50 ranked posts.

These gains suggest that even limited target-context supervision can help the model learn local patterns of actionability. Adding Indian training examples may help the model distinguish posts that can be handled by a teaching assistant from those that require instructor attention in this specific course and platform environment. This is especially important because the Indian dataset has a much larger share of instructor-actionable posts than the UPenn training data. The target-context examples therefore help recalibrate what high actionability looks like in the new setting.

At the same time, this result should be interpreted carefully. The RQ3 Indian result is evaluated on the held-out Indian test split rather than the full Indian dataset, so it is not directly identical to the RQ1 and RQ2 Indian evaluations. However, the direction and size of the improvement are still practically meaningful. The main implication is that local calibration may be more useful than source-only rebalancing when deploying urgency models in new educational contexts. The result does not imply that every institution needs a large labeled dataset before deployment. Rather, it suggests that even modest target-context supervision can improve triage utility when the target setting differs from the source training context.

4.4 Comparison with Existing Work

The results extend prior work on MOOC forum urgency detection in a more specific way than simply testing another dataset. Almatrafi et al. [2] framed urgent MOOC posts as a "needle in a haystack" problem: high-priority posts are rare but missing them can matter for learner support. That framing fits the UPenn training data used in this study, where instructor-actionable posts are extremely sparse. However, the Indian dataset complicates this assumption because instructor-actionable posts are not rare in the same way. This difference matters because a model trained in a source context where high-actionability cases are unusual may be

poorly calibrated for a target context where such cases are much more common. Thus, the present findings suggest that urgency detection is not only a minority-class detection problem; it is also a distribution-shift problem.

This distinction also sharpens the contribution relative to Švábenský et al. [12]. Their work showed that urgency prediction should be evaluated under dataset shift and treated as an ordered construct rather than only as a binary urgent/not-urgent classification task. The present study follows that logic by retaining a comparable urgency construct and modeling pipeline but tests a different kind of shift. The UPenn-to-Stanford comparison evaluates transfer between Western MOOC benchmarks, while the UPenn-to-Indian comparison evaluates transfer to a setting that differs in platform, institutional context, learner population, and actionability distribution. The larger degradation on the Indian dataset shows that benchmark-to-benchmark generalization may understate the difficulty of deploying urgency models in substantially different educational settings.

The findings also connect urgency detection to the broader literature on instructor intervention. Chaturvedi et al. [6] modeled instructor intervention in MOOC forums using signals from forum activity and course context, and Chandrasekaran et al. [5] examined intervention prediction across forum threads, highlighting the role of forum structure and instructional context. The present results support this broader interpretation of actionability. A post's need for instructor attention is not determined by its wording alone; it also depends on the course design, available support structures, and local expectations about who should respond. This helps explain why a model can perform reasonably on Stanford while degrading on the Indian dataset, even though all datasets contain MOOC forum posts. The relevant shift is not simply geographic or cultural, but instructional and operational.

The SMOTE results further qualify prior work on imbalance in urgency detection. Alrajhi et al. [4] focus on class imbalance as a central challenge for automatic urgency detection. The present results support the importance of that issue, since SMOTE improves the concentration of instructor-actionable posts near the top of the ranked list. At the same time, the improvement is limited: rebalancing the UPenn training data does not make the model transfer well to the Indian setting. This suggests that imbalance mitigation can improve exposure to rare source-context labels but cannot replace target-context information when the cues, frequency, or meaning of actionability differ across settings.

Finally, the results align with broader educational forum classification work by Sha et al. [9,10], who caution against evaluating educational text classifiers only through narrow model-performance comparisons. The present study makes a similar point from a deployment perspective. The main issue is not whether USE + SVR is the best possible architecture, but whether a model that appears useful under one evaluation context remains useful when the instructional context changes. By holding the model fixed, this study shows that changes in training composition and target context can substantially affect both ordinal prediction quality and triage utility. The contribution is therefore not a new classifier, but evidence that urgency detection systems need target-context validation, actionability-aware evaluation, and metrics that reflect how teaching teams review and prioritize forum posts.

4.5 Opportunities for Future Work

A first direction is to strengthen the generalization claims through broader and more realistic validation. This includes evaluating a fuller transfer matrix across UPenn, Stanford, Indian, and additional datasets to separate source-dataset, target-dataset, platform, and regional/contextual effects. It also includes collecting additional target-context datasets across multiple institutions, disciplines, and platforms, including potentially multilingual settings. Because the Indian dataset in this study is small and concentrated in data science-related courses, broader validation is needed before making stronger claims about non-Western MOOC contexts more generally. Future work should also test sensitivity to different sampling and split strategies and evaluate whether adaptation learned in one target context transfers to another.

In parallel, the triage framing can be made more decision-relevant by aligning k with actual staffing constraints, reporting class-wise false positives and false negatives, assigning higher costs to missed instructor-actionable posts, and evaluating the stability of actionability thresholds across contexts via calibration. A second direction is to move from "regression that is evaluated as triage" toward methods that are explicitly optimized for triage. This includes learning-to-rank or cost-sensitive objectives that prioritize instructor-actionable retrieval, alternatives to SMOTE that avoid synthetic embedding points, and richer adaptation strategies that leverage unlabeled target posts. Future work should also compare additional model families, including ordinal regression, random forests, transformer-based classifiers, and uncertainty-aware models. Finally, error analyses that categorize missed instructor-actionable posts by pedagogical function, and small instructor-in-the-loop studies that measure time saved and perceived usefulness, would connect offline metrics to instructional outcomes and clarify which improvements matter most in practice.

5. CONCLUSION

This paper extends prior work on MOOC forum-post urgency prediction by evaluating cross-context generalization in a setting that includes both Western benchmarks and a new non-Western forum dataset, while retaining a comparable modeling pipeline to isolate the effect of context shift. The results show that models trained on Western MOOC posts can generalize reasonably to another Western benchmark but degrade substantially when transferred to the Indian context, particularly on the instructor-actionable cases that are most important for instructional triage. Because the Indian dataset differs from the Western benchmarks along multiple dimensions, these findings should be interpreted as evidence of cross-context sensitivity rather than as proof of a purely cultural effect. Incorporating triage-oriented evaluation further clarifies this gap by highlighting how differences in ranking behavior under constrained review budgets can matter even when global performance metrics appear similar.

Across the robustness strategies examined, low-resource target-domain adaptation by adding a small amount of labeled target-context data provides the largest and most operationally meaningful improvements, outperforming class-imbalance mitigation via oversampling. Taken together, these findings suggest that reliable urgency triage across diverse educational contexts will likely require at least modest target-domain supervision and evaluation that reflects instructor workflows, rather than relying on source-only training and global fit metrics alone. For EDM systems intended to guide instructor attention, benchmark performance should therefore be accompanied by target-context validation, triage-oriented metrics, and explicit attention to missed instructor-actionable posts. Future work should broaden validation across additional contexts and move toward triage-optimized training and calibration so that actionability decisions remain stable and useful when deployed.

6. ACKNOWLEDGMENTS

The author thanks Švábenský et al. for making the UPenn MOOC forum-post dataset available for research use and for providing the methodological foundation for this study. The author also thanks the IIT Madras BS in Data Science and Applications program for granting permission to use the Indian forum data. The author is grateful to Hyejoo Yun and Jaidev Deshpande for their contributions to the initial conceptualization of the project.

ChatGPT was used for editing and language refinement during preparation of this paper; all research design, analysis, interpretation, and final content decisions remain the responsibility of the author.

7. REFERENCES

[1] A. Agrawal and A. Paepcke. The Stanford MOOC Posts Dataset, 2014. http://infolab.stanford.edu/~paepcke/stanfordMOOCForumPostsSet.tar.gz.

[2] O. Almatrafi, A. Johri, and H. Rangwala. Needle in a haystack: Identifying learner posts that require urgent response in MOOC discussion forums. Computers & Education, 118:1–9, 2018. https://doi.org/10.1016/j.compedu.2017.11.002.

[3] L. Alrajhi, K. Alharbi, and A. I. Cristea. A Multidimensional Deep Learner Model of Urgent Instructor Intervention Need in MOOC Forum Posts. In V. Kumar and C. Troussas, editors, Intelligent Tutoring Systems, pages 226–236, Cham, 2020. Springer International Publishing. https://doi.org/10.1007/978-3-030-49663-0_27.

[4] Alrajhi, L., Alamri, A., Pereira, F.D. et al. Solving the imbalanced data issue: automatic urgency detection for instructor assistance in MOOC discussion forums. User Model User-Adap Inter 34, 797–852 (2024). https://doi.org/10.1007/s11257-023-09381-y

[5] M. K. Chandrasekaran, M.-Y. Kan, B. C. Y. Tan, and K. Ragupathi. Learning Instructor Intervention from MOOC Forums: Early Results and Issues. In Proceedings of the 8th International Conference on Educational Data Mining, pages 218–225, 2015. https://doi.org/10.48550/arXiv.1504.07206

[6] S. Chaturvedi, D. Goldwasser, and H. Daumé III. Predicting Instructor's Intervention in MOOC forums. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers, pages 1501–1511, 2014. https://doi.org/10.3115/v1/P14-1141.

[7] S. X. Guo, X. Sun, S. X. Wang, Y. Gao, and J. Feng. Attention-Based Character-Word Hybrid Neural Networks With Semantic and Structural Information for Identifying of Urgent Posts in MOOC Discussion Forums. IEEE Access, 7:120522–120532, 2019. https://doi.org/10.1109/ACCESS.2019.2929211.

[8] N. A. Khodeir. Bi-GRU Urgent Classification for MOOC Discussion Forums Based on BERT. IEEE Access, 9:58243–58255, 2021. https://doi.org/10.1109/ACCESS.2021.3072734.

[9] L. Sha, M. Raković, Y. Li, A. Whitelock-Wainwright, D. Carroll, D. Gašević, and G. Chen. Which Hammer Should I Use? A Systematic Evaluation of Approaches for Classifying Educational Forum Posts. International Educational Data Mining Society, 2021. https://eric.ed.gov/?q=ED615664&id=ED615664.

[10] L. Sha, M. Raković, J. Lin, Q. Guan, A. Whitelock-Wainwright, D. Gašević, and G. Chen. Is the Latest the Greatest? A Comparative Study of Automatic Approaches for Classifying Educational Forum Posts. IEEE Transactions on Learning Technologies, XX:1–14, 2022. https://doi.org/10.1109/TLT.2022.3227013.

[11] S. Singhal, A. F. Zambrano, M. Pankiewicz, X. Liu, C. Porter, and R. S. Baker. De-Identifying Student Personally Identifying Information with GPT-4. In Proceedings of the 17th International Conference on Educational Data Mining, pages 559–565, 2024. https://doi.org/10.5281/zenodo.12729884.

[12] V. Švábenský, R. S. Baker, A. Zambrano, Y. Zou, and S. Slater. Towards Generalizable Detection of Urgency of Discussion Forum Posts. In M. Feng, T. Käser, and P. Talukdar, editors, Proceedings of the 16th International Conference on Educational Data Mining, pages 302–309, Bengaluru, India, 2023. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115790.

[13] J. Yu, L. Alrajhi, A. Harit, Z. Sun, A. I. Cristea, and L. Shi. Exploring Bayesian Deep Learning for Urgent Instructor Intervention Need in MOOC Forums. In A. I. Cristea and C. Troussas, editors, Intelligent Tutoring Systems, pages 78–90, Cham, 2021. Springer International Publishing. https://doi.org/10.1007/978-3-030-80421-3_10.

Estimating IRT Parameters from Text via an ICC-Based Supervised and Contrastive Loss

Yoshiki HORI Takumi ITAMIYA Iwao TANUMA
Emiko YOSHIHARA
DWANGO Co., Ltd., Tokyo, Japan
{yoshiki_hori, takumi_itamiya, iwao_tanuma, emiko_yoshihara}@dwango.co.jp

ABSTRACT

Item response theory (IRT) is a modeling framework that defines the probability of a correct response through an item characteristic curve (ICC), which is a function of examinee ability and item characteristics. However, since IRT-based estimation typically requires large-scale pretesting with many examinees, question difficulty estimation from text (QDET) has been proposed to estimate item characteristics directly from text. While prior QDET models designed the objective function to align each parameter separately, we instead focus on the shape similarity of ICCs. In addition, we introduce an approach that leverages not only labeled IRT data but also auxiliary information available on real-world e-learning platforms. This allows us to learn from both items with estimated parameters and items whose parameters have not been estimated, using difficulty ordering information such as curriculum level.

We propose two loss terms that exploit both labeled and unlabeled items while reflecting the relationship between item parameters and the ICC. First, since the ICC can be approximated by the cumulative distribution function of a normal distribution, the similarity between ICCs can be formulated as the similarity between normal distributions. Second, we introduce a contrastive loss that leverages relative difficulty information for unlabeled items by formulating an inequality between the normal distributions induced by the ICCs.

Experiments on real-world data from an e-learning platform show that our method improves binary cross-entropy by 2.02% over an MSE-based baseline when predicting student response correctness on unseen items.

Keywords

Item Response Theory, Contrastive Learning, Natural Language Processing, Neural Network

Yoshiki Hori, Takumi Itamiya, Iwao Tanuma, and Emiko Yoshihara. Estimating IRT Parameters from Text via an ICC-Based Supervised and Contrastive Loss. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 437–445. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039821

1. INTRODUCTION

In recent years, the widespread adoption of online learning platforms has enabled the large-scale collection of learning logs and immediate feedback. Accordingly, personalized learning technologies that leverage data to assess learners' proficiency levels and progress, and to provide optimal questions from within the platform by considering item characteristics such as difficulty, have garnered significant attention [1, 21]. Consequently, identifying item characteristics within the platform is one of the critical challenges for the realization of personalized learning.

To measure item characteristics, item response theory (IRT) [12, 17, 26] is a widely adopted statistical framework that estimates item parameters, such as difficulty and discrimination, from students' response data and represents the probability of a correct response through an item characteristic curve (ICC) based solely on these item properties and student ability. The IRT framework has the advantage of enabling stable measurement of ability scores that are independent of test difficulty and student group characteristics, and it has been adopted in various fields, ranging from psychological experiments to large-scale tests such as the SAT and TOEFL.

In applying IRT frameworks, pretest measurement of item characteristics is typically conducted. To obtain stable item parameter estimates, it is necessary to secure a sufficiently large and ability-diverse sample of students and to ensure that they are adequately motivated to engage with the items. Consequently, for platforms containing a large volume of items, measuring the characteristics of every item through pretesting poses a significant challenge in terms of both temporal and financial resources.

To address these resource-intensive requirements, research on question difficulty estimation from text (QDET) [6], which leverages natural language processing (NLP) to estimate item characteristics directly from text, has become active.

In existing approaches of QDET, item parameters are typically predicted by formulating the task as a regression problem. For example, in the two-parameter logistic (2PL) model, a standard model in IRT, each item has two parameters, difficulty and discrimination, and these parameters are commonly predicted by minimizing the sum of squared errors for each parameter.

Since these two parameters collectively define the functional form of the ICC, it is more principled to formulate an ICC-based optimization task rather than individually minimizing the loss function for each parameter. Such an approach enables the inference of item characteristics aligned with learners' response accuracy, which is expected to capture the interplay between difficulty and discrimination, thereby improving overall predictive performance.

Furthermore, regression-based QDET is typically formulated as a supervised learning problem. Consequently, the training is restricted to items for which item characteristics, such as difficulty and discrimination, have been previously measured through IRT. However, in practical applications, even when explicit item difficulty or discrimination labels are absent, auxiliary data providing implicit signals are often available. For example, if problem sets are labeled as "beginner" or "advanced," it is natural to assume that items in the advanced set possess higher difficulty.

The contributions of this work are as follows.

- **Proposal of a new ICC-based supervised loss for QDET** : We propose a novel supervised loss term to jointly characterize the difficulty and discrimination parameters. This loss function is designed based on the discrepancy between latent variable distributions, focusing on the shape of the ICC.
- **Proposal of a new contrastive loss using unlabeled data for QDET** : We propose a novel contrastive loss term to incorporate unlabeled item texts. This loss term leverages auxiliary information to utilize items with unknown parameters, directly contributing to improved predictive performance.
- **Performance evaluation using real-world student response data**: We evaluate the effectiveness of our proposed loss terms by comparing them with conventional methods in terms of their ability to predict students' correct and incorrect responses on an actual e-learning platform.

Since our focus is on the loss function rather than a specific model architecture in this study, the experiments are conducted using relatively simple models. However, we expect them to be broadly applicable to QDET using 2PL models by replacing the training objective in existing text-to-parameter models.

2. RELATED WORK

In this work, we aim to estimate IRT item parameters, specifically difficulty and discrimination, directly from text by leveraging both supervised and contrastive learning. To ground our approach, we first review the foundations of IRT and the current state of QDET. Subsequently, we provide an overview of contrastive learning frameworks.

2.1 Item Response Theory

IRT is a modeling framework that predicts the probability of an examinee's correct response using parameters such as difficulty, discrimination, and student ability [12, 17, 26]. The item characteristic curve (ICC) describes the relationship between the probability of a correct response and the student's ability for given item parameters. Because of their interpretability and flexibility, IRT models are widely adopted in various fields, ranging from psychological experiments to large-scale standardized tests such as SAT and TOEFL.

In IRT, the one-parameter logistic (1PL) and two-parameter logistic (2PL) models are standard formulations [17, 26].

The 1PL model, also called the Rasch model, considers only item difficulty and student ability. The probability of a correct response by student j to item i is given by:

$$\sigma(D(\theta_j - b_i)), \tag{1}$$

where the sigmoid function $\sigma(x)$ is defined as $(1 + \mathrm{e}^{-x})^{-1}$. θ_j denotes the ability of the student j, and b_i denotes the difficulty of the item i. As b_i increases, the probability of a correct response to the item decreases; conversely, as θ_j increases, the student's probability of a correct response increases.

D is referred to as a scaling constant, typically set to 1.702 to ensure the following approximation holds [8]:

$$\int_{-\infty}^{x} \mathcal{N}(t|0,1)dt \simeq \sigma(Dx). \tag{2}$$

The 2PL model additionally accounts for the item discrimination parameter a_i. For student j, the probability of a correct response to item i is formulated as:

$$\sigma(Da_i(\theta_j - b_i)), \tag{3}$$

where a_i is a parameter that determines the slope of the curve. As item discrimination a_i increases, the probability of a correct response becomes more sensitive to the difference between θ_j and b_i, resulting in a steeper ICC. Consequently, a_i is an indicator for assessing how effectively an item differentiates between students of varying ability levels.

2.2 Question Difficulty Estimation from Text

To obtain stable item parameter estimates in IRT, it is necessary to secure a sufficiently large and ability-diverse sample of students while ensuring that they are adequately motivated to engage with the items. Consequently, the application of IRT in practice requires extensive field testing, which demands substantial temporal and financial resources. To address this challenge, QDET has been proposed as an approach that leverages NLP to infer item characteristics, such as difficulty, directly from text; for a comprehensive review, see [6].

In early QDET research, the prevailing approach involved applying feature engineering to item text and constructing regression models based on the resulting features. A representative example is R2DE [4, 5], which leverages TF-IDF [28] features to estimate IRT item parameters, specifically difficulty and discrimination.

In recent years, pre-trained language models (PLMs) such as BERT [10] and GPT [25] have emerged as one of the dominant approaches in NLP due to their ability to effectively

model complex linguistic contexts. Given their superior performance over conventional feature-based methods, PLM-based approaches have also been applied to QDET. For instance, Benedetto et al. demonstrated that using BERT and DistilBERT to predict IRT parameters outperforms traditional feature-driven models [3]. In addition, Lalor et al. proposed an approach that utilizes multiple PLMs as artificial crowds to generate responses for unlabeled items [13]. This framework enables the zero-shot estimation of IRT parameters based purely on model-generated responses, thus eliminating the need for labeled training data.

2.3 Contrastive Learning

Contrastive learning is a training framework that enhances model performance and representations by contrasting positive and negative samples. It has been widely adopted across various domains, including NLP, recommender systems, and computer vision [11, 24, 27].

In the field of educational technology, contrastive learning has been incorporated into several Knowledge Tracing (KT) studies, which aim to track students' proficiency over time. For instance, CL4KT generates negative samples by flipping answer correctness, enabling data augmentation grounded in contrastive learning [15]. Additionally, Q-MCKT demonstrates that the inclusion of contrastive learning during training improves prediction accuracy for items with sparse response data [29].

In recent years, one of the most commonly used loss functions in contrastive learning is InfoNCE [20, 22], which aims to maximize the difference between the scores of positive and negative samples. This loss function, also referred to as NT-Xent [9], is defined as follows:

$$-\mathbb{E}_{s_+}\left[\log \frac{\exp(s_+/\tau)}{\exp(s_+/\tau)+\sum_{s_-}\exp(s_-/\tau)}\right] \tag{4}$$

where τ is the temperature parameter, and $s_\pm$ are the scores of the positive and negative samples, respectively. When only a single negative sample is used, the loss function simplifies to:

$$-\mathbb{E}_{s_\pm}\left[\log\sigma\left(\frac{s_+ - s_-}{\tau}\right)\right], \tag{5}$$

which is referred to as the pairwise logistic loss or BPR loss [7, 27].

One of the advantages of contrastive learning is that a model can be trained without fully labeled data, provided that relative relationships between positive and negative samples are available. Consequently, it is extensively adopted in scenarios with limited labeled data. On many educational platforms, it is not uncommon for the volume of unlabeled items to far exceed those with labels, due in part to the high cost of obtaining labeled item parameters (e.g., difficulty and discrimination). Therefore, contrastive learning can be expected to be useful in such cases.

Despite this, the role of contrastive learning in IRT and QDET remains underexplored, as existing research has predominantly relied on supervised methods using labeled item parameters [2–5].

3. METHODOLOGY

We propose an ICC-based loss that combines supervised and contrastive terms for learning item parameters from text. The overall loss function $\mathcal{L}$ is defined as follows:

$$\mathcal{L} = \lambda\mathcal{L}_{\mathrm{CS}} + (1-\lambda)\mathcal{L}_{\mathrm{con}}, \tag{6}$$

where $\mathcal{L}_{\mathrm{CS}}$ and $\mathcal{L}_{\mathrm{con}}$ represent the supervised and contrastive loss terms, respectively. The hyperparameter λ balances the contributions of these two terms. The detailed definitions of $\mathcal{L}_{\mathrm{CS}}$ and $\mathcal{L}_{\mathrm{con}}$ are provided in the following subsections.

3.1 Supervised Loss

The item difficulty b_i and item discrimination a_i are inseparable parameters that jointly determine the ICC and the probability of a correct response. Therefore, instead of focusing on b_i and a_i as separate scalar values, we focus on the probability distribution of the latent variable z_i defined by these parameters. This approach enables us to directly optimize the joint distribution, yielding parameter estimates that better reflect the actual response behavior.

By defining the latent variable z_i as a random variable following a normal distribution, $z_i \sim \mathcal{N}(b_i, a_i^{-2})$ [16], the 2PL ICC in Eq. (3) can be approximated using the relationship established in Eq. (2):

$$\sigma(Da_i(\theta_j - b_i)) \simeq \Pr\{\theta_j > z_i \mid z_i \sim \mathcal{N}(b_i, a_i^{-2})\}. \tag{7}$$

Since z_i is a sample from a normal distribution with the difficulty parameter as its mean, it can be interpreted as the realized difficulty faced by examinees, accounting for noise due to factors such as testing conditions.

We define the supervised loss $\mathcal{L}_{\mathrm{CS}}$, which optimizes the predicted item parameters $\hat{b}_i$ and $\hat{a}_i$ so that the induced latent variable distribution $\mathcal{N}(z \mid \hat{b}_i, \hat{a}_i^{-2})$ is close to the target distribution $\mathcal{N}(z \mid b_i, a_i^{-2})$. Specifically, we employ the Cauchy-Schwarz divergence (CSD) [23], which serves as a measure of the discrepancy between two probability distributions. The supervised loss $\mathcal{L}_{\mathrm{CS}}$ is defined as follows:

$$\mathcal{L}_{\mathrm{CS}} = \mathbb{E}_{i\in\mathcal{D}_{\mathrm{CS}}}\left[L_i^{\mathrm{CS}}\right], \tag{8}$$

$$\begin{aligned} L_i^{\mathrm{CS}} &= -\log \frac{\int \mathcal{N}(z \mid \tilde{b}_i, \tilde{a}_i^{-2})\mathcal{N}(z \mid b_i, a_i^{-2})\mathrm{d}z}{\sqrt{\int \mathcal{N}(z \mid b_i, a_i^{-2})^2\mathrm{d}z \int \mathcal{N}(z \mid \tilde{b}_i, \tilde{a}_i^{-2})^2\mathrm{d}z}} \\ &= \frac{(b_i - \tilde{b}_i)^2}{2(a_i^{-2} + \tilde{a}_i^{-2})} + \frac{1}{2}\log\frac{a_i^{-2} + \tilde{a}_i^{-2}}{2a_i^{-1}\tilde{a}_i^{-1}}, \end{aligned} \tag{9}$$

where L_i represents the discrepancy for a single item i. Here, $\mathcal{D}_{\mathrm{CS}}$ is the set of items whose difficulty and discrimination are known.

We also experimented with other closed-form distributional discrepancies (e.g., Kullback–Leibler divergence and Hellinger distance). Among them, CSD yielded the best performance in our experiments(Appendix B).

3.2 Contrastive Loss

We denote (i_+, i_-) as an ordered pair of items where the relative difficulty order is determined by auxiliary information, such as metadata tags or learning course titles. The

contrastive loss $\mathcal{L}_{\text{con}}$ is defined based on the negative log-likelihood of the relationship $z_+ > z_-$, formulated as follows:

$$\mathcal{L}_{\text{con}} = \mathbb{E}_{i_\pm \in \mathcal{D}_{\text{con}}} \left[L_{i_\pm}^{\text{con}} \right] \tag{10}$$

$$L_{i_\pm}^{\text{con}} = -\log \Pr\{z_+ > z_- \mid z_\pm \sim \mathcal{N}(\tilde{b}_{i_\pm}, \tilde{a}_{i_\pm}^{-2})\} \tag{11}$$

$$= -\log \int_0^\infty \mathcal{N}(x \mid \tilde{b}_{i_+} - \tilde{b}_{i_-}, \tilde{a}_{i_+}^{-2} + \tilde{a}_{i_-}^{-2}) \mathrm{d}x \tag{12}$$

$$\simeq -\log \sigma \left(D \frac{\tilde{b}_{i_+} - \tilde{b}_{i_-}}{\sqrt{\tilde{a}_{i_+}^{-2} + \tilde{a}_{i_-}^{-2}}} \right). \tag{13}$$

Here, $\mathcal{D}_{\text{con}}$ is the set of item pairs for which only the relative order of difficulty is known. In Eq. (13), we used the approximation given by Eq. (2).

4. EXPERIMENTS

We conduct experiments to clarify the following two points.

- The supervised loss $\mathcal{L}_{\text{CS}}$ yields more accurate predictions of correct responses compared to mean squared error (MSE).
- Adding the contrastive loss $\mathcal{L}_{\text{con}}$ improves model performance.

4.1 Dataset

Table 1: Dataset sizes used in this study

(a) training and validation

	Supervised	Contrastive	
		pair	unique
English	763	90,531	3,069
Mathematics	745	43,756	4,979
Japanese	398	100,687	1,242
Social Studies	118	85,567	10,169
Total	2,024	263,271	19,459

(b) Evaluation response logs

	log	unique item	unique student
English	33,595	75	457
Mathematics	39,017	156	415
Japanese	39,418	114	451
Social Studies	25,118	179	319
Total	137,148	524	1,642

As experimental data, among the materials available on ZEN Study[1], an e-learning platform for students, we selected learning materials designed for junior high and high school students. We focused on the four subjects for which a dedicated pretest was conducted: English, Mathematics, Japanese, and Social Studies. Since the platform manages its materials in HTML format, we performed preprocessing by removing non-text elements (e.g., images and hyperlinks), eliminating duplicate and highly similar materials, and converting the content to Markdown.

[1] `https://www.nnn.ed.nico`

The sizes of the datasets used in our experiments are shown in Table 1. To evaluate the model's performance on unseen items, the supervised training and evaluation datasets are disjoint.

From this supervised dataset, we held out 10% of the items from training directly and used them only for hyperparameter tuning and early stopping. The item characteristics used as supervised training data were estimated under a Bayesian 2PL IRT framework, with normal priors placed on item difficulty and learner ability and a log-normal prior on item discrimination to enforce positivity.

For the contrastive training data, we leveraged auxiliary information available on the e-learning platform (e.g., metadata tags and course titles). Specifically, we selected candidate pairs of items from similar topics, such as those belonging to the same chapter or whose titles became identical after stripping prefix/suffix tags and numbers. From these candidates, we used only pairs whose metadata, such as course names, tags, or title prefixes/suffixes, clearly indicated a difference in difficulty between the two items , excluding those with comparable difficulty levels. Pairs with comparable difficulty levels in the metadata were excluded. To address the data imbalance across subjects, the sampling probabilities during training were adjusted to ensure that each subject (e.g., English, Mathematics, Japanese, and Social Studies) was sampled with equal probability. The sizes of these supervised and contrastive datasets are summarized in Table 1a.

In addition, we conducted an independent evaluation test administration to assess the model's performance; the sizes of this evaluation dataset are summarized in Table 1b. The "log" column indicates the number of student-item response records, while the "item" and "student" columns show the counts of unique items and unique students, respectively. If the same student participated in multiple subjects, they were treated as distinct students for each subject.

4.2 Model Architecture

As in conventional QDET, the model is based on neural networks and is designed to jointly predict item difficulty and discrimination from question text. As illustrated in Figure 1, the architecture consists of two primary components: a text encoder and a feed-forward network (FFN). To highlight the impact of the loss terms, we adopt a simple model architecture so that performance differences can be attributed primarily to the training objective.

4.2.1 Text-Encoder

For each item, the Markdown text is input to the encoder to compute a d_{enc}-dimensional text embedding. We utilized a pre-trained language model (PLM) as the text encoder and kept its parameters frozen while training the remaining components. Specifically, we employed Gemini Embedding [14] with its default output dimension of $d_{\text{enc}} = 3{,}072$.

4.2.2 Feed-Forward Network

Given the text embedding $\boldsymbol{e}_i$, we employ an FFN to estimate the item difficulty $\tilde{b}_i$ and discrimination $\tilde{a}_i$:

$$\boldsymbol{h}_i = \mathrm{ReLU}(W\boldsymbol{e}_i + \boldsymbol{w}_0) \tag{14}$$

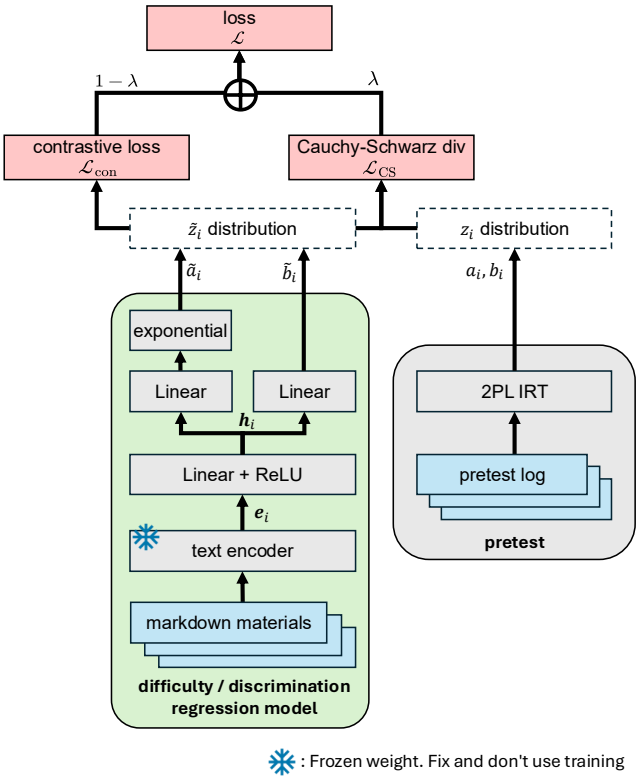


Figure 1: Overview of the experiments. The text encoder is kept frozen, and only the feed-forward network (FFN) is trained. We estimate the distribution of z_i using the regression model and a pre-test, and use it to define the loss function.

$$\tilde{b}_i = \boldsymbol{w}_b^T \boldsymbol{h}_i + w_{b0} \qquad \tilde{a}_i = \exp(\boldsymbol{w}_a^T \boldsymbol{h}_i + w_{a0}) \tag{15}$$

where $W \in \mathbb{R}^{d_h \times d_{\text{enc}}}$, $\boldsymbol{w}_0 \in \mathbb{R}^{d_h}$, $\boldsymbol{w}_b, \boldsymbol{w}_a \in \mathbb{R}^{d_h}$, and $w_{b0}, w_{a0} \in \mathbb{R}$ are learnable parameters. We set the hidden dimension to $d_h = 64$. ReLU denotes the rectified linear unit activation function, defined as $\text{ReLU}(x) = \max\{x, 0\}$ [19]. As the discrimination parameter must be positive, an exponential transformation is applied to the corresponding FFN output.

4.3 Evaluation

To benchmark the effectiveness of our proposed CSD-based supervised loss $\mathcal{L}_{\text{CS}}$, we compare against a baseline model that employs MSE of the difficulty and discrimination parameters as the loss function. Furthermore, as an ablation study to verify the effectiveness of the contrastive loss $\mathcal{L}_{\text{con}}$, we also compare it with a model that removes the contrast loss term. We also introduce a baseline that applies the pairwise logistic loss (Eq. 5) only to the difficulty parameter; in this setting, the temperature τ is fixed at D^{-1}.

As the primary evaluation metric, we calculate the binary cross-entropy (BCE) over the evaluation response logs, between the observed student correctness and the predicted response probability. The predicted probability for student j on item i is computed via Eq. (3) using the predicted item parameters $(\tilde{a}_i, \tilde{b}_i)$ and the student ability θ_j estimated from pretests:

$$\mathbb{E}_{i,j}[y_{ij} \log \sigma(D\tilde{a}_i(\theta_j - \tilde{b}_i)) + (1 - y_{ij}) \log(1 - \sigma(D\tilde{a}_i(\theta_j - \tilde{b}_i)))]. \tag{16}$$

Here, $y_{ij} \in \{0, 1\}$ denotes the observed correctness of student j on item i. To provide further insight into parameter estimation accuracy, we also evaluate the performance using the Root Mean Squared Error (RMSE) for difficulty and discrimination, as well as the CSD for each item.

4.4 Implementation Details

The learning rate was set to 1×10^{-4}, and we used the AdamW optimizer [18]. The batch size was set equal to the total size of the supervised training dataset. Training used early stopping, and patience set to 200 epochs. The hyperparameter λ was tuned via grid search over the range $\{0.1, 0.2, \ldots, 0.9\}$.

5. RESULT

To validate the effectiveness of $\mathcal{L}_{\text{CS}}$ and $\mathcal{L}_{\text{con}}$, we compute the BCE for student response correctness on the evaluation dataset, and measure the CSD and RMSE for each item. These results are summarized in Table 2. The hyperparameters λ reported are those that achieved the lowest Average BCE during our grid search. The "pretest" column shows the BCE computed using item parameters directly estimated from student response logs, which serves as a theoretical performance ceiling for the models.

First, we examine the performance of our proposed CSD loss compared to the conventional MSE loss. The CSD loss achieved better performance than the MSE loss in almost all metrics, and this trend was observed regardless of whether contrastive learning was used. Compared to the baseline MSE value (0.5987 ± 0.0008), switching the supervised loss to CSD yielded (0.5882 ± 0.0003), corresponding to a 1.75% relative improvement in BCE; here, values are reported as the mean $\pm$ standard deviation over five independent runs. A paired t-test was conducted to compare the average BCE, yielding a p-value of 8×10^{-7} and the null hypothesis was rejected. Improvements were observed across all four subjects—English, Mathematics, Japanese, and Social Studies. In terms of BCE, the CSD-based model outperformed the MSE-based model in every subject.

Next, we evaluate the impact of contrastive learning. Regardless of the supervised loss employed (CSD or MSE), incorporating our proposed contrastive loss (Eq. (13)) improved performance on all metrics except RMSE-a. Furthermore, the proposed contrastive loss outperformed the pairwise-logistic loss (Eq. (5)), in many metrics including BCE. In particular, under the baseline MSE supervised loss, introducing our contrastive objective improved BCE from 0.5987 ± 0.0008 to 0.5972 ± 0.0008, corresponding to a 0.25% relative improvement. A paired t-test was conducted to compare the average BCE, yielding a p-value of 2×10^{-3} and the null hypothesis was rejected.

However, we did not observe a clear improvement in RMSE-a. With the CSD supervised loss, we speculate that this is because our objectives emphasize aspects directly related to student correctness, rather than regions where the discrimi-

Table 2: Evaluation metrics (BCE, CSD, RMSE-b, and RMSE-a) on the test set for each loss term. The best metrics for each loss term are highlighted in bold. The "pretest" column serves as a reference, representing the BCE computed using the item parameters directly estimated from the pretest response logs (ground truth). For the contrastive loss columns, "a, b" and "b" designate the variations defined by Eq. (13) and Eq. (5), respectively, while "nothing" denotes the absence of the contrastive term.

supervised loss		CSD			MSE			pretest
contrastive loss		a, b	b	nothing	a, b	b	nothing	
	λ	0.9	0.9	1.0	0.8	0.9	1.0	-
BCE	English	**0.5567**	0.5592	0.5600	0.5725	0.5745	0.5757	0.5213
	Mathematics	**0.5569**	0.5576	0.5599	0.5696	0.5705	0.5727	0.5145
	Japanese	0.6167	0.6187	**0.6162**	0.6280	0.6288	0.6278	0.5608
	Social Studies	0.6160	**0.6148**	0.6164	0.6187	0.6174	0.6187	0.5612
	Average	**0.5866**	0.5876	0.5882	0.5972	0.5978	0.5987	0.5394
CSD	English	0.0811	**0.0799**	0.0883	0.0827	0.0837	0.0883	-
	Mathematics	**0.0863**	0.0869	0.0898	0.0982	0.0987	0.1018	-
	Japanese	0.1205	0.1212	**0.1200**	0.1244	0.1244	0.1245	-
	Social Studies	0.0942	0.0953	**0.0925**	0.1014	0.1012	0.1024	-
	Average	**0.0955**	0.0958	0.0976	0.1017	0.1020	0.1043	-
RMSE-b	English	0.6190	**0.6066**	0.6553	0.6272	0.6326	0.6473	-
	Mathematics	0.5919	**0.5883**	0.6028	0.6315	0.6298	0.6416	-
	Japanese	0.8491	**0.8384**	0.8736	0.8327	0.8333	0.8423	-
	Social Studies	0.6788	**0.6733**	0.6926	0.6879	0.6871	0.7000	-
	Average	0.6847	**0.6766**	0.7061	0.6948	0.6957	0.7078	-
RMSE-a	English	0.5628	0.5645	0.5515	0.4981	0.4963	**0.4956**	-
	Mathematics	0.4606	0.4627	0.4594	0.4281	**0.4265**	0.4285	-
	Japanese	0.5551	0.5555	0.5411	0.5152	0.5092	**0.5076**	-
	Social Studies	0.3740	0.3842	0.3904	**0.3663**	0.3748	0.3767	-
	Average	0.4881	0.4917	0.4856	0.4519	**0.4517**	0.4521	-

nation parameter a becomes excessively large or small. For contrastive learning, we expected our proposed loss (Eq. (13)), which explicitly incorporates both a and b, to yield a smaller RMSE$-a$ than the pairwise-logistic loss (Eq. (5)) that learns only b; however, this was not observed in our experiments. This point requires more in-depth investigation in future work. This point requires more in-depth investigation in future work, particularly regarding which types of contrastive item pairs contribute to better generalization.

Overall, our proposed two loss terms resulted in the highest predictive accuracy across the evaluated metrics, underscoring the effectiveness of the proposed framework for the QDET task. Compared to the conventional MSE baseline without contrastive learning (BCE: 0.5987±0.0008), our proposed loss combination (CSD with the proposed contrastive loss) achieved a BCE of 0.5866±0.0007, corresponding to a 2.02% relative improvement.

6. LIMITATIONS AND FUTURE WORK

Our study focuses on a 2PLM-based neural regression model for estimating item difficulty and discrimination directly from question texts. Consequently, it remains to be explored whether our proposed method is applicable to other IRT formulations, such as the 1PL or 3PL models, and whether such extensions would yield similar benefits. In addition to neural regression models, QDET encompasses diverse approaches, including those leveraging artificial crowds [13]. Integrating our method with such approaches remains a challenge for future work.

Furthermore, it remains to be verified whether this approach can contribute to the realization of personalized learning and the enhancement of the student learning experience. Consequently, the practical utility of the proposed method for educational applications should be substantiated through further experimental studies on real-world e-learning platforms.

7. CONCLUSION

This study proposes two loss terms for estimating question difficulty and discrimination from text, with a focus on the ICC and contrastive learning. First, we introduced a supervised loss term designed to align the model's latent variable distribution with that of the training data. Second, leveraging relative difficulty orders derived from auxiliary information, we proposed a contrastive loss term based on the order

of latent variables.

Using real-world data from an e-learning platform, the experiments confirmed that the proposed loss terms outperform baseline methods in predicting student response correctness, achieving a 2.02% improvement in BCE. These results demonstrate the potential of the method to reduce the costs associated with large-scale pretesting and to contribute to the reliable estimation of item parameters.

8. ACKNOWLEDGMENTS

The authors would like to acknowledge the support of DWANGO Co, Ltd. and KADOKAWA DWANGO Educational Institute for this work.

APPENDIX

A. DERIVATION EQUATIONS

A.1 Derivation cumulative distribution function from the 2PL ICC

Equation (7) can be rewritten as

$$\Pr\{\theta_j > z_i \mid z_i \sim \mathcal{N}(b_i, a_i^{-2})\} \tag{17}$$

$$= \Pr\left\{ \frac{z_i - b_i}{a_i^{-1}} < a_i(\theta_j - b_i) \,\middle|\, z_i \sim \mathcal{N}(b_i, a_i^{-2}) \right\} \tag{18}$$

$$= \Pr\{t < a_i(\theta_j - b_i) \,|\, t \sim \mathcal{N}(0,1)\} \tag{19}$$

$$= \int_{t < a_i(\theta_j - b_i)} \mathcal{N}(t \mid 0,1)\, dt \tag{20}$$

$$\simeq \sigma\big(D a_i(\theta_j - b_i)\big). \tag{21}$$

A.2 Derivation of the supervised loss

We define the per-item supervised loss by the Cauchy–Schwarz divergence:

$$L_i[p_i(z), q_i(z)] = -\log \frac{\int p_i(z) q_i(z)\, dz}{\sqrt{\left(\int p_i(z)^2\, dz\right)\left(\int q_i(z)^2\, dz\right)}}. \tag{22}$$

$$p_i(z) = \mathcal{N}(z \mid b_i, a_i^{-2}), \qquad q_i(z) = \mathcal{N}(z \mid \tilde{b}_i, \tilde{a}_i^{-2}). \tag{23}$$

The following identity holds for the integral of the product of two Gaussian densities.

$$\int \mathcal{N}(x|\mu_1, \sigma_1^2)\mathcal{N}(x|\mu_2, \sigma_2^2) dx \quad = \frac{\exp\left(-\frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}\right)}{\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}} \tag{24}$$

Therefore, the integrals in Eq.(22) can be evaluated as follows:

$$\int p_i(z) q_i(z) dz = \frac{\exp\left(-\frac{(b_i - \tilde{b}_i)^2}{2(a_i^{-2} + \tilde{a}_i^{-2})}\right)}{\sqrt{2\pi(a_i^{-2} + \tilde{a}_i^{-2})}} \tag{25}$$

$$\int p_i(z)^2 dz = \frac{a_i}{2\sqrt{\pi}} \qquad \int q_i(z)^2 dz = \frac{\tilde{a}_i}{2\sqrt{\pi}} \tag{26}$$

Substituting these expressions into Eq. (22), we obtain the closed-form expression

$$L_i[p_i(z), q_i(z)] = \frac{(b_i - \tilde{b}_i)^2}{2(a_i^{-2} + \tilde{a}_i^{-2})} + \frac{1}{2} \log \frac{a_i^{-2} + \tilde{a}_i^{-2}}{2 a_i^{-1} \tilde{a}_i^{-1}} \tag{27}$$

.

A.3 Derivation of the contrastive loss

$$\Pr\{z_+ > z_- \mid z_\pm \sim \mathcal{N}(b_\pm, a_\pm^{-2})\} \tag{28}$$

$$= \Pr\{x > 0 \mid x \sim \mathcal{N}(b_+ - b_-, a_+^{-2} + a_-^{-2})\} \tag{29}$$

$$= \int_0^\infty \mathcal{N}(x | b_+ - b_-, a_+^{-2} + a_-^{-2}) dx \tag{30}$$

$$= \int_{-\frac{b_+ - b_-}{a_+^{-2} + a_-^{-2}}}^{\infty} \mathcal{N}(t|0,1) dt \tag{31}$$

$$= \int_{-\infty}^{\frac{b_+ - b_-}{a_+^{-2} + a_-^{-2}}} \mathcal{N}(t|0,1) dt \tag{32}$$

$$\simeq \sigma\left(D \frac{b_+ - b_-}{a_+^{-2} + a_-^{-2}}\right) \tag{33}$$

B. ALTERNATIVE SUPERVISED LOSSES

Before fixing CSD as our supervised objective, we conducted a preliminary experiment to determine which metric is most suitable for measuring the similarity between the latent-variable distributions. As candidate metrics, we selected divergences that admit closed-form expressions between two Gaussian distributions, so that they can be incorporated into training without numerical approximation. Specifically, we compared CSD with the Kullback–Leibler (KL) divergence and the Hellinger distance, while keeping the rest of the training framework unchanged. Table 3 summarizes the resulting BCE on the evaluation set for each supervised loss, combined with the three contrastive settings considered in the main paper. The results show that CSD consistently achieved the lowest BCE across all three contrastive settings.

Table 3: Additional comparison of supervised losses on the evaluation BCE. The contrastive settings "a, b", "b", and "nothing" follow the definitions used in Table 2.

	supervised loss			
contrastive loss	CSD	KL	Hellinger	MSE
a, b	**0.5866**	0.5934	0.5892	0.5972
b	**0.5876**	0.5950	0.5914	0.5978
nothing	**0.5882**	0.5957	0.5894	0.5987

The KL divergence and Hellinger distance used in this comparison are given below:

$$\mathrm{KL} = \log \frac{\tilde{a}}{a} + \frac{a^2}{2}\left\{\tilde{a}^{-2} + (\tilde{b} - b)^2\right\} - \frac{1}{2} \tag{34}$$

$$\text{Hellinger} = 1 - \sqrt{\frac{2a\tilde{a}}{a^2 + \tilde{a}^2}} \exp\left\{-\frac{(b - \tilde{b})^2 a^2 \tilde{a}^2}{4(a^2 + \tilde{a}^2)}\right\} \tag{35}$$

References

[1] A. Alkhatlan and J. Kalita. Intelligent tutoring systems: A comprehensive historical survey with recent developments. *International Journal of Computer Applications*, 181(43):1–20, Mar 2019.

[2] G. Aradelli. Transformers for question difficulty estimation from text, 2020.

[3] L. Benedetto, G. Aradelli, P. Cremonesi, A. Cappelli, A. Giussani, and R. Turrin. On the application of transformers for estimating the difficulty of multiple-choice

questions from text. In *Proceedings of the 16th workshop on innovative use of NLP for building educational applications*, pages 147–157, 2021.

[4] L. Benedetto, A. Cappelli, R. Turrin, and P. Cremonesi. Introducing a framework to assess newly created questions with natural language processing. In *International conference on artificial intelligence in education*, pages 43–54. Springer, 2020.

[5] L. Benedetto, A. Cappelli, R. Turrin, and P. Cremonesi. R2DE: a nlp approach to estimating irt parameters of newly generated questions. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 412–421, 2020.

[6] L. Benedetto, P. Cremonesi, A. Caines, P. Buttery, A. Cappelli, A. Giussani, and R. Turrin. A survey on recent approaches to question difficulty estimation from text. *ACM Computing Surveys*, 55(9):1–37, 2023.

[7] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.

[8] G. Camilli. Teacher's corner: origin of the scaling constant d= 1.7 in item response theory. *Journal of educational and behavioral statistics*, 19(3):293–295, 1994.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[11] T. Gao, X. Yao, and D. Chen. SimCSE: Simple contrastive learning of sentence embeddings. In M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.

[12] R. K. Hambleton, H. Swaminathan, and H. J. Rogers. *Fundamentals of item response theory*, volume 2. Sage, 1991.

[13] J. P. Lalor, H. Wu, and H. Yu. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4249–4259, 2019.

[14] J. Lee, F. Chen, S. Dua, D. Cer, M. Shanbhogue, I. Naim, G. H. Ábrego, Z. Li, K. Chen, H. S. Vera, et al. Gemini embedding: Generalizable embeddings from gemini. *arXiv preprint arXiv:2503.07891*, 2025.

[15] W. Lee, J. Chun, Y. Lee, K. Park, and S. Park. Contrastive learning for knowledge tracing. In *Proceedings of the ACM web conference 2022*, pages 2330–2338, 2022.

[16] F. M. Lord. *Applications of Item Response Theory to Practical Testing Problems*. Lawrence Erlbaum Associates, 1980.

[17] F. M. Lord and M. R. Novick. *Statistical Theories of Mental Test Scores*. Addison-Wesley, Reading, MA, 1968.

[18] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, pages 1–18, 2019. Published as a conference paper at ICLR 2019.

[19] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[20] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[21] Z. Papamitsiou and A. A. Economides. Learning analytics and educational data mining in practice: A systematic literature review of empirical evidence. *Journal of Educational Technology & Society*, 17(4):49–64, 2014.

[22] B. Poole, S. Ozair, A. Van Den Oord, A. Alemi, and G. Tucker. On variational bounds of mutual information. In *International conference on machine learning*, pages 5171–5180. PMLR, 2019.

[23] J. C. Principe, D. Xu, and J. W. Fisher, III. Information-theoretic learning. In S. Haykin, editor, *Unsupervised Adaptive Filtering, Volume I: Blind Source Separation*, pages 265–319. John Wiley & Sons, New York, NY, 2000.

[24] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[25] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018. OpenAI Technical Report.

[26] G. Rasch. *Probabilistic Models for Some Intelligence and Attainment Tests*. Studies in mathematical psychology. University of Chicago Press, 1980.

[27] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, page 452–461, Arlington, Virginia, USA, 2009. AUAI Press.

[28] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[29] H. Zhang, Z. Liu, C. Shang, D. Li, and Y. Jiang. A question-centric multi-experts contrastive learning framework for improving the accuracy and interpretability of deep sequential knowledge tracing models. *ACM Transactions on Knowledge Discovery from Data*, 19(2):1–25, 2025.

Fair When Sparse, Accurate When Rich: RL-Gated Conversational Fusion for Temporally Equitable Knowledge Tracing

Fan Zhang
University of Florida
f.zhang1@ufl.edu

Rui Guo
University of Miami
rui.guo@miami.edu

Hai Li
University of Florida
li.ha@ufl.edu

ABSTRACT

Knowledge Tracing (KT) models often underperform for students with short interaction histories, creating a temporal equity gap between early- and late-stage learners. We tackle this issue by fusing behavioral traces with dialogue-derived signals using a lightweight reinforcement-learning (RL) gate that adaptively reweights evidence across the learning trajectory. To make temporal disparities explicit, we evaluate by student-level practice-volume cohorts (deciles from 10% to 100%) and report ACC/AUC within cohorts and overall. Across both backbones, the RL-gated variant consistently improves aggregate accuracy and AUC while reducing across-cohort variance, yielding flatter, more reliable cohort curves relative to a behavioral-only baseline and a naïve feature-augmented model without gating. These findings suggest a practical design principle for next-generation knowledge tracing systems: when interaction histories are sparse, semantically rich auxiliary signals can stabilize prediction, and as histories accumulate, the model can shift back toward behavioral evidence. We discuss limitations particularly the reliance on rich interaction data, and outline directions for testing on public datasets with sufficient depth or proxy signals, as well as for exploring stronger equity objectives and interpretable gating.

Keywords

Knowledge Tracing; Temporal Equity; Reinforcement Learning; Conversational Features; LLM; DKT; SAKT.

1. INTRODUCTION

Knowledge Tracing (KT) has become a cornerstone of modern educational technology [10], especially in the development of intelligent tutoring systems [34]. KT models predict students' knowledge states based on their learning interactions [10]. Computer-supported education systems rely on KT to personalize learning experiences and optimize instructional decisions [17, 21]. However, recent research reveals a concerning pattern: existing KT models, including classical Bayesian Knowledge Tracing (BKT) and Deep Knowledge Tracing (DKT), often fail to achieve equitable tutoring across different student populations, exhibiting systematic bias against certain groups [13, 29]. While early-stage prediction methods have shown effectiveness in some domains via statistical approaches such as survival analysis [14], KT presents temporal fairness challenges that extend beyond traditional demographic disparities. A critical, under-explored dimension of fairness involves the differential treatment of early-stage learners with limited interaction histories versus experienced learners with extensive practice records. This temporal bias manifests as less accurate predictions for early-stage students due to sparse sequential data, leading to suboptimal recommendations during crucial initial phases; this phenomenon corresponds to the well-known *cold-start* problem in both knowledge tracing and recommender systems [7, 9]. Conversely, experienced learners benefit from rich histories that enable more precise knowledge estimation, creating a "rich-get-richer" effect in which students who need the most support receive the least accurate guidance.

To address temporal fairness in KT, we present a three–tier comparative framework evaluated on two standard backbones (DKT and SAKT). The Baseline relies on behavioral traces only (skill, correctness, optionally difficulty/response time). The +Features model augments the baseline with time–aligned conversation–derived signals (dialogue–rubric vectors and unit–level mastery indicators) designed to avoid temporal leakage. The +Features+RL model adds a gating mechanism trained with policy–gradient updates to adaptively weight behavioral versus conversational streams across a learner's trajectory. We operationalize temporal equity via *percent–progress bins* (10%, 20%, ..., 100%), explicitly targeting more reliable predictions for early–stage learners while preserving overall performance [5, 10, 20, 21, 26]. This framing is especially relevant as tutoring systems become more conversational. In newer AI-mediated environments, dialogue is part of the instructional process itself, which makes dialogue-integrated KT increasingly practical even if it remains inapplicable to many legacy benchmark datasets [2, 11, 15].

Our contributions are threefold. (1) We formulate a fairness–aware optimization for KT that treats disparity across progress bins as an explicit signal, enabling adaptive fusion of heterogeneous evidence through an RL gate. (2) We integrate conversation–derived features with behavioral traces using leak–conscious, step–aligned fusion, offering a

Fan Zhang, Rui Guo, and Hai Li. Fair When Sparse, Accurate When Rich: RL-Gated Conversational Fusion for Temporally Equitable Knowledge Tracing. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 446–452. International Educational Data Mining Society (2026).
© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039877

practical path to richer KT without compromising temporal validity. (3) We provide a systematic comparison across backbones and variants and visualize progress–aligned performance with cohort–size overlays, demonstrating qualitatively flatter, more equitable curves for the RL–gated model while maintaining competitive aggregate accuracy and AUC. Detailed metrics and variant comparisons are reported in the Results section.

2. RELATED WORK

2.1 Knowledge Tracing

Classical KT models, most notably BKT, represent mastery as a latent state updated from response histories with parameters for initialization, learning, slip, and guess [10]. Subsequent extensions account for learner heterogeneity and temporal dynamics, including individualized BKT [31] and time-aware variants that model spacing and forgetting [22, 23]. Neural KT reframes KT as sequence modeling: DKT uses recurrent encoders to predict next-step correctness from interaction sequences [21], while attention-based architectures such as SAKT capture relevance among past items and skills through query–key attention [20, 33].

Two practical challenges motivate our work. First, modern KT often relies primarily on correctness patterns and student–item interactions, which can underutilize rich side information such as problem text, hints, or conversational evidence [1]. Second, directly injecting auxiliary signals risks temporal leakage if features are not carefully aligned to the prediction horizon (e.g., using future turns of a dialogue to predict the present step) [12, 24]. Our setting exposes both issues: we have post-problem dialogues that are potentially informative, but any benefit hinges on leak-conscious alignment and on a mechanism that can adapt how much the model should rely on these signals at different stages of a learner's trajectory. Moreover, KT systems face a form of the cold-start problem analogous to recommender systems—predictions are less reliable for students with sparse histories [7, 9]. We therefore complement standard overall metrics with a progress-aware analysis to make such temporal effects explicit.

2.2 Fairness in Educational AI and KT

A growing literature examines disparate outcomes in educational AI [8, 32]. In KT specifically, studies have surfaced systematic gaps across student subgroups and contexts, raising fairness concerns for data-driven personalization [13, 29]. Some work explores modeling choices aimed at mitigating disparities (e.g., time-augmented BKT) [6], while others document construct-related effects such as reading-ability gaps in math KT or platform/socioeconomic biases in second-language KT [25, 27].

Our perspective focuses on temporal equity: comparable predictive performance for learners at early vs. late stages. In conventional training, aggregate objectives naturally overweight data-rich trajectories; learners with short histories precisely those who need the most reliable guidance—can receive noisier predictions. We operationalize this dimension by grouping students into deciles based on total practice volume (10%–100%) and reporting performance by cohort. This differs from demographic fairness (which we do not analyze here) but is practically salient for tutorial decision making: reliable early-stage estimation can prevent compounding errors, while later stages may benefit more from accumulated behavioral evidence. Our equity lens thus aims at flattening cohort-wise performance curves, not enforcing strict parity at the expense of accuracy, and it aligns with the cold-start framing noted above [7, 9].

2.3 Reinforcement Learning in Educational AI

Reinforcement learning (RL) has been widely studied for sequencing and recommendation problems in education, treating personalization as a sequential decision process [3, 19]. Applications include curriculum sequencing and learning-path optimization as well as broader decision-making frameworks [4, 18, 28]. Much of this work, however, optimizes a single objective (e.g., learning gain, engagement) and gives limited attention to fairness-aware architectural adaptation.

In contrast, we use RL not to select content but to adapt model fusion. Concretely, a lightweight gate produces a convex weight that fuses behavioral and conversation/unit streams, with a policy-gradient objective [26] that balances predictive utility and a temporal-equity penalty defined over practice-volume cohorts. This framing has three practical advantages. First, it is model-agnostic: the gate can sit atop standard KT backbones (e.g., DKT, SAKT) without architectural overhaul. Second, it directly targets equity at the level of the evaluation protocol (deciles by practice volume), avoiding a mismatch between training goals and assessment. Third, it supports history-adaptive reliance on auxiliary signals: conversational features can carry more weight where behavioral evidence is scarce and recede when long histories suffice. Our empirical results (Section Results) suggest that such equity-aware fusion can reduce across-cohort variance while maintaining overall accuracy, providing a cautious but actionable path toward temporally robust KT.

3. METHOD

3.1 Data Description

We analyze logs from a deployed middle–school mathematics platform in which each problem–solving attempt may be followed by an optional chatbot conversation. The corpus spans approximately one academic term (about eight months) and contains 2,678 students, 1,605 items (three difficulty bands), and 10,877 problem–solving instances linked to up to 31 skills/competencies, with an overall correctness of about 47%. Each step records an item/skill identifier, binary correctness, and elapsed time; when present, the post–problem dialogue consists of multiple turns between the student and the chatbot (on average, about 1.47 post–problem conversations per attempt and about 7.97 turns). These characteristics provide the empirical context and are consistent with prior descriptions of the same platform.

The data analyzed in this study were derived from the ALTER-Math project [30]. The underlying human-subjects study was reviewed and approved by the Institutional Review Board (IRB) at the University of Florida (IRB202301838). All research procedures involving human participants in the source study were conducted in accordance with the ethical standards of the University, and informed consent was obtained from all participants prior to their inclusion in the study.

For the present analysis, only de-identified learner log data and post-problem dialogue data were used.

To assess how performance varies with the amount of prior history, we group *students* by their total number of attempted problems and form ten learner cohorts at the deciles: 10%, 20%, ..., 100%. Unless otherwise noted, we report Accuracy (ACC) and AUC *within each cohort* and overall. This cohorting emphasizes reliability for learners with *sparse* vs. *rich* practice histories and is used consistently across all experiments.

3.2 LLM–Derived Dialogue Features

Beyond behavioral traces, we incorporate features distilled from the post–problem conversation. Feature extraction is conducted *offline* using an instruction–tuned LLaMA–family model, Llama-3-8B-Instruct, with decoding parameters temperature = 0, top-p = 0.9, and max_tokens = 128, running on a single A100–80GB GPU [16]. The resulting numeric features are then *time–aligned* to the KT sequence so that features used at step t reflect only information available up to (and including) t, preventing temporal leakage. We use two groups of dialogue features:

1. Rubric vector (fixed–length): aggregated indicators that summarize the conversation signals most relevant to KT—e.g., knowledge–point mentions/coverage, clarification/help–seeking, confusion/uncertainty cues, correctness–related evidence, and engagement tone—computed per solved item from the dialogue that follows that item.
2. Unit–level mastery (scalar per step): an LLM–based verdict on whether the learner evidences mastery of the curricular unit associated with the current item.

Both groups are exported as numeric features and concatenated as an auxiliary stream; missing dialogues at a step are zero–imputed for the rubric vector with a neutral default for mastery.

3.3 Models

We instantiate two standard knowledge–tracing backbones and build three variants on each:

DKT encodes the interaction sequence with a recurrent network and outputs a behavioral logit $z_t^{(\mathrm{beh})}$ for step t [21]. SAKT aggregates relevant past items via self–attention to produce $z_t^{(\mathrm{beh})}$ [20].

1. **Baseline (behavioral only)** uses the behavioral stream (skill×correctness) with available covariates (difficulty, response time).
2. **+Features** concatenates the *rubric vector* and the unit-mastery scalar as an auxiliary stream $X_t^{(\mathrm{conv})}$ that is strictly step–aligned (forward–only).
3. **+Features+RL** adds a lightweight *gating* mechanism that produces a convex weight $\alpha_t \in [0,1]$ each step and fuses streams:

$$z_t \;=\; \alpha_t\, z_t^{(\mathrm{beh})} \;+\; \big(1-\alpha_t\big)\, z_t^{(\mathrm{conv})}, \qquad \hat{p}_t \;=\; \sigma(z_t).$$

3.4 Fairness–Aware Reinforcement–Learning Gate

The gate is trained with policy–gradient updates [26] to preserve overall predictive utility while discouraging steep disparities across learner cohorts (10–100% by practice volume). Concretely, we maximize the objective

$$\mathcal{J} = \underbrace{\mathbb{E}\big[\mathrm{Perf}(\hat{p}_t, r_t)\big]}_{\text{overall utility}} - \lambda_{\mathrm{fair}} \cdot \underbrace{\Phi\big(\{\mathrm{Perf}_c\}_{c=1}^{10}\big)}_{\text{temporal–equity penalty}} + \lambda_{\mathrm{ent}} \cdot \underbrace{\mathbb{H}[\alpha_t]}_{\text{entropy}}.$$

where Perf denotes a differentiable surrogate aligned with ACC/AUC, Φ penalizes large performance gaps/variance across the ten practice–volume cohorts, and $\mathbb{H}$ regularizes the gate to avoid premature collapse. Intuitively, the gate can rely more on conversational evidence for learners with short histories and gradually shift toward behavioral evidence as history accumulates, aiming for flatter, more equitable cohort curves without sacrificing aggregate accuracy.

3.5 Training and Evaluation

All variants minimize masked cross–entropy over valid steps. Unless otherwise noted, we keep the same data splits, batch size, and training epochs across backbones/variants to enable controlled comparison.[1] We report Accuracy (ACC) and AUC *within each learner cohort* (10–100% by practice volume) and overall. Plots overlay a dashed right axis indicating the number of students per cohort to contextualize variance across groups with fewer vs. more interactions. Results are reported for DKT and SAKT under identical cohort definitions and evaluation code.

4. RESULTS

4.1 Setup for Temporal-Equity Evaluation

All models are evaluated by percent progress (10%, 20%, ..., 100%) computed per learner from the normalized position t/T. For each bin and overall we report Accuracy (ACC) and AUC. Plots include a dashed right axis showing cohort size per bin (students when available; otherwise interactions), enabling visual inspection of stability vs. noise across the trajectory.

4.2 Aggregate Trends Across Backbones

Figures 1 a-d show per-bin AUC/ACC for DKT and SAKT. In both backbones, +Features+RL yields the strongest curves overall and particularly in early-to-mid progress bins where histories are short. The dashed right axes indicate that bins with fewer data points (or fewer students) have more volatile performance; even there, +Features+RL maintains a consistent advantage.

Table 1 summarizes percent-bin statistics (mean, SD, min, max, median) by model/variant. On DKT, +Features+RL improves mean AUC from 0.5985 (baseline) to 0.6094 and mean ACC from 0.5738 to 0.5877. On SAKT, +Features+RL

[1]In our runs we used a batch size of 64 and trained for 12 epochs per setting.

Table 1: Percent-progress bin statistics (mean ± SD, with min/max and median) by backbone and variant.

Backbone	Variant	ACC_{mean}	ACC_{SD}	ACC_{min}	ACC_{max}	ACC_{med}
DKT	baseline	0.5738	0.0253	0.5184	0.6064	0.5777
DKT	+features	0.5777	0.0242	0.5285	0.6165	0.5749
DKT	+features+RL	*0.5877*	*0.0205*	*0.5609*	*0.6212*	*0.5877*
SAKT	baseline	0.5596	0.0321	0.4887	0.6013	0.5715
SAKT	+features	0.5677	0.0291	0.5000	0.6005	0.5744
SAKT	+features+RL	*0.5872*	*0.0231*	*0.5527*	*0.6171*	*0.5880*

Backbone	Variant	AUC_{mean}	AUC_{SD}	AUC_{min}	AUC_{max}	AUC_{med}
DKT	baseline	0.5985	0.0320	0.5398	0.6445	0.6031
DKT	+features	0.5921	0.0272	0.5444	0.6312	0.5930
DKT	+features+RL	*0.6094*	*0.0261*	*0.5664*	*0.6507*	*0.6059*
SAKT	baseline	0.5792	0.0396	0.4829	0.6155	0.5969
SAKT	+features	0.5754	0.0315	0.5159	0.6113	0.5900
SAKT	+features+RL	*0.6027*	*0.0225*	*0.5666*	*0.6349*	*0.6066*

Notes. (i) Statistics are computed over ten student-level practice-volume cohorts (deciles: 10%, 20%, . . . , 100%). For each cohort we evaluate ACC and AUC, then summarize across cohorts (mean, SD, min, max, median). (ii) *Underlined italics* mark the best value within each row group (per backbone across variants): for *means/medians/max*, higher is better; for *SD*, lower is better (flatter, more equitable across cohorts); for *min*, higher indicates stronger worst-cohort performance. (iii) The `+features+RL` variant consistently achieves the highest means and lowest SDs across both backbones, suggesting the RL gate boosts overall quality while reducing dispersion across practice-volume cohorts. (iv) Min/max arise from different cohorts (early vs. late); interpret alongside SD and cohort sizes (shown in the figures' dashed right axes) to understand variability under sparse vs. rich histories.

raises mean AUC from 0.5792 (baseline) to 0.6027 and mean ACC from 0.5596 to 0.5872. Notably, +Features (without RL) shows mixed effects on AUC (slightly below baseline on both backbones), indicating that naive fusion of conversational/unit features can be suboptimal; the RL gate appears necessary to realize consistent gains.

4.3 Variance Reduction and Curve Flattening

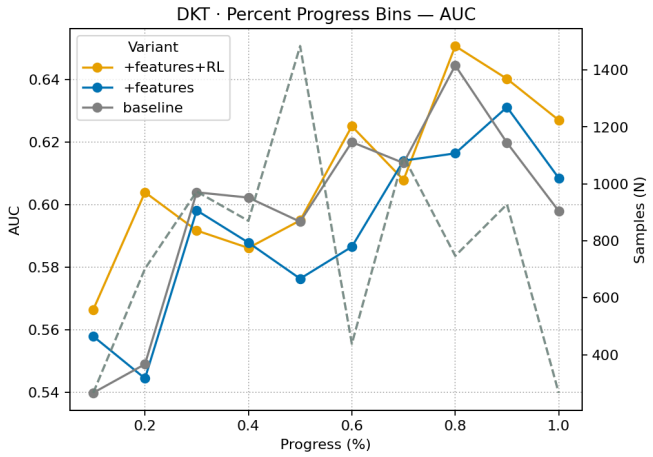
Beyond mean improvements, +Features+RL consistently reduces across-bin variance, suggesting flatter, more equitable trajectories. For DKT, ACC SD decreases from 0.0253 (baseline) to 0.0205 (+Features+RL), and AUC SD from 0.0320 to 0.0261. For SAKT, ACC SD drops from 0.0321 to 0.0231, and AUC SD from 0.0396 to 0.0225. This aligns with the design goal: attenuate the early/late performance gap by adapting the reliance on conversational vs. behavioral evidence over time.

5. CONCLUSION AND DISCUSSION

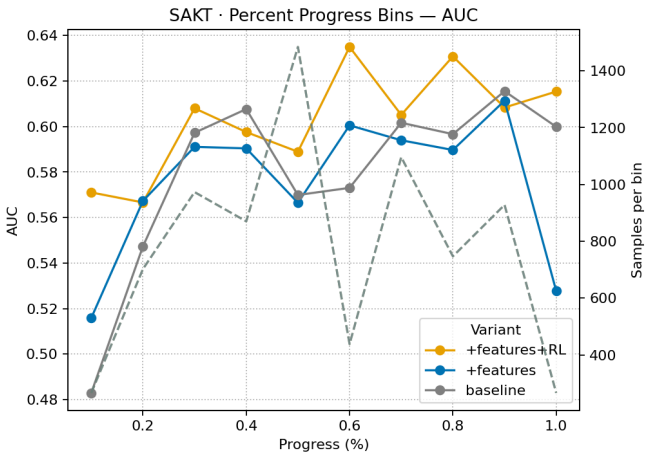
This work examined temporal equity in KT by augmenting behavioral inputs with dialogue-derived features and by learning an adaptive fusion policy via a lightweight RL gate. Under two standard backbones (DKT, SAKT) and a cohorting scheme that groups students by practice volume (10–100% deciles), the RL-gated variant consistently yielded higher aggregate performance (ACC/AUC) and lower dispersion across cohorts, producing flatter cohort-wise curves. Taken together, these results suggest a practical route to reduce performance variability between sparse- and rich-history learners without discarding established KT architectures. At the same time, these findings should be interpreted in relation to prior work on fairness in KT rather than as a standalone fairness claim: prior studies have shown that KT systems can yield inequitable performance across learners and contexts [1], and recent work has explored mitigating such disparities by modifying temporal assumptions within the KT model itself, such as time-augmented BKT [6]. Our results extend this line of work from a different angle. Rather than redesigning the KT backbone itself, we show that a lightweight prediction-stage fusion mechanism can improve reliability when learner histories are sparse while remaining compatible with established KT architectures.

The immediate broader impact of this architecture is therefore not that every legacy ITS can adopt post-problem dialogue features tomorrow, but that it provides a deployable fusion template for settings in which learning evidence extends beyond correctness sequences alone. This is consistent with broader KT research showing that student–item correctness traces often underutilize richer side information such as hints, problem text, or conversational evidence [1]. From a fairness perspective, this also matters because more information does not automatically translate into more equitable modeling; the educational AI literature has repeatedly emphasized that fairness depends on how evidence is represented and used, not merely on whether additional signals are available [5]. In that sense, the present architecture should be read as a practical mechanism for selectively using auxiliary evidence under conditions of evidence scarcity, rather than as a universal retrofit for all KT deployments.

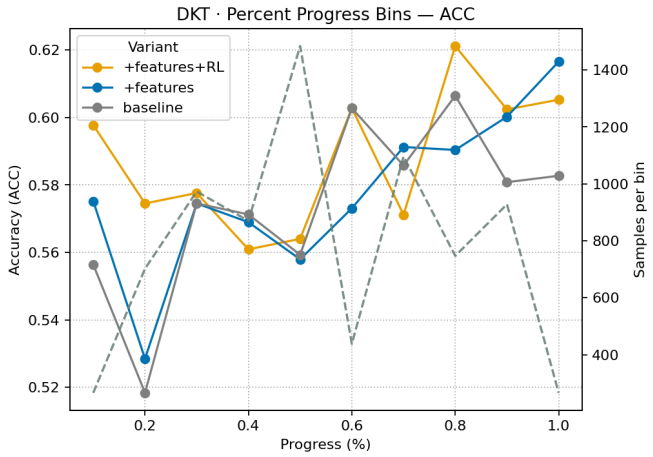
Beyond aggregate metrics, the cohort view offered diagnostic value. We observed that adding dialogue features naïvely (without gating) did not uniformly help—in some cases AUC was unchanged or slightly reduced—indicating that conversational evidence is heterogeneous and history-dependent. This point is important in light of prior fairness work: just as inequities in KT do not disappear automatically through



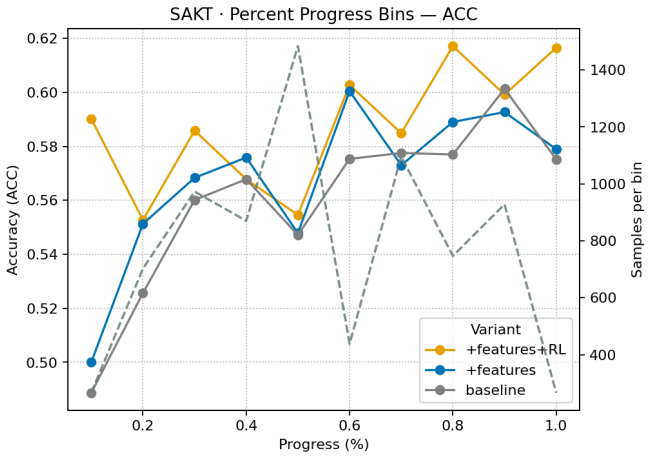
(a) DKT — AUC (cohort size dashed on right axis)



(b) SAKT — AUC (cohort size dashed on right axis)



(c) DKT — ACC (cohort size dashed on right axis)



(d) SAKT — ACC (cohort size dashed on right axis)

Figure 1: Percent–progress evaluation across learner cohorts formed by student-level practice volume (deciles: 10%, 20%, . . . , 100%). X-axis: cohort by practice volume (10% → 100%). Left Y-axis: performance metric (AUC in panels (a,b); ACC in panels (c,d); range 0–1). Right Y-axis (dashed line): the number of valid prediction samples per bin. (Number of students). Curves compare Baseline, +Features, and +Features+RL; reading them together with the dashed cohort-size trace clarifies reliability under sparse (lower-decile) vs. rich (upper-decile) histories.

model complexity alone [13, 29], our results suggest that they also do not disappear simply by appending more features. Instead, auxiliary conversational evidence appears useful only when it is carefully aligned and selectively weighted. The RL gate appeared to address this by modulating reliance on conversational vs. behavioral signals along the trajectory; early cohorts (where histories are short) tended to benefit more from the additional stream, whereas later cohorts did not require as strong a contribution. In that sense, the equity-aware fusion objective functioned less like a global regularizer and more like a schedule for when and how to use each information source.

We view these findings as pointing to a possible path rather than a definitive solution. In particular, the approach remains intentionally simple: a standard backbone, a compact set of dialogue features (rubric + unit mastery), and a single gating mechanism with a fairness term. This simplicity made the behavior easier to reason about and the gains more attributable, but it also leaves room for complementary improvements (e.g., calibration, per-cohort operating points, or interpretable gating analyses). It also distinguishes our contribution from prior efforts that target fairness by modifying the KT model itself [6]. As such, our study is best interpreted as evidence that adaptive, equity-aware fusion is a viable design principle for KT, not as a claim that any particular architecture is optimal. Accordingly, the present contribution should be read less as a dataset-specific feature recipe and more as a transferable fusion principle for modern tutoring environments that record richer interaction traces than correctness sequences alone [1].

6. LIMITATIONS

First, our dataset contains substantial interaction volume per learner and includes post-problem dialogues from a deployed tutoring setting. This richness likely made it feasible to learn a useful fusion policy: the model could observe enough variation to adjust weights between behavioral

and conversational signals. In contrast, many public KT datasets contain shorter sequences and no conversational channel, which limits how far our specific feature design and gating policy can be generalized as is. We therefore refrained from reporting results on standard public benchmarks that lack comparable signals, to avoid an apples-to-oranges comparison.

Second, our evaluation emphasizes temporal equity through practice-volume cohorts (student-level deciles). Although this targets a practically salient dimension of fairness (supporting learners with limited histories), it is only one slice. We did not analyze demographic fairness, course-level heterogeneity, or domain transfer, nor did we exhaust architectural variants of the gate (e.g., hierarchical or skill-conditional gating). The present results should therefore be read as context-constrained: they demonstrate that equity-aware fusion can stabilize cohort-wise performance on this dataset and setup, but external validity remains to be tested.

In the future, we plan to (i) explore public datasets with sufficient interaction depth or proxied contextual signals (e.g., textual rationales, hint requests) to stress-test the approach beyond our setting; (ii) examine calibration and thresholding per cohort to complement AUC/ACC; (iii) probe the learned gate (feature attributions, skill-conditional patterns) for interpretability; and (iv) evaluate alternative equity objectives (e.g., explicit worst-cohort optimization or variance-constrained training) and more expressive fusion architectures. These steps will clarify when equity-aware fusion provides the largest benefit and how it can be safely adopted in broader KT deployments.

References

[1] G. Abdelrahman, Q. Wang, and B. Nunes. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11):1–37, 2023.

[2] K. Academy. Meet khanmigo: Khan academy's ai-powered teaching assistant & tutor. *Khanmigo. ai*, 2024.

[3] M. M. Afsar, T. Crump, and B. Far. Reinforcement learning based recommender systems: A survey. *ACM Computing Surveys*, 55(7):1–38, 2022.

[4] S. Amin, M. I. Uddin, A. A. Alarood, W. K. Mashwani, A. Alzahrani, and A. O. Alzahrani. Smart e-learning framework for personalized adaptive learning and sequential path recommendations using reinforcement learning. *IEEE Access*, 11:89769–89790, 2023.

[5] R. S. Baker and A. Hawn. Algorithmic bias in education. *International journal of artificial intelligence in education*, 32(4):1052–1092, 2022.

[6] J. Barrett, A. Day, and K. Gal. Improving model fairness with time-augmented bayesian knowledge tracing. In *Proceedings of the 14th Learning Analytics and Knowledge Conference (LAK)*, pages 46–54, 2024.

[7] I. Bhattacharjee and C. Wayllace. Cold start problem: An experimental study of knowledge tracing models with new students. In *International Conference on Artificial Intelligence in Education*, pages 425–432, Cham, 2025. Springer Nature Switzerland.

[8] S. Bulathwela, M. Pérez-Ortiz, C. Holloway, M. Cukurova, and J. Shawe-Taylor. Artificial intelligence alone will not democratise education: On educational inequality, techno-solutionism and inclusive tools. *Sustainability*, 16(2):781, 2024.

[9] D. Cai, S. Qian, Q. Fang, J. Hu, and C. Xu. User cold-start recommendation via inductive heterogeneous graph neural network. *ACM Transactions on Information Systems*, 41(3):1–27, 2023.

[10] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[11] M. Cukurova, M. Mavrikis, R. Luckin, J. Clark, and C. Crawford. Interaction analysis in online maths human tutoring: The case of third space learning. In *international conference on artificial intelligence in education*, pages 636–643. Springer, 2017.

[12] A. Darvishi, H. Khosravi, S. Sadiq, and D. Gašević. Incorporating ai and learning analytics to build trustworthy peer assessment systems. *British Journal of Educational Technology*, 53(4):844–875, 2022.

[13] S. Doroudi and E. Brunskill. Fairer but not fair enough: On the equitability of knowledge tracing. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK)*, pages 335–339, 2019.

[14] M. J. Fard, P. Wang, S. Chawla, and C. K. Reddy. A bayesian perspective on early stage event prediction in longitudinal data. *IEEE Transactions on Knowledge and Data Engineering*, 28(12):3126–3139, 2016.

[15] A. Ganguly, N. Mehjabin, A. Malik, and A. Johri. Conversational ai agents in education: An umbrella review of current utilization, challenges, and future directions for ethical and responsible use. *AI and Ethics*, 6(1):72, 2026.

[16] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al. The llama 3 herd of models, 2024.

[17] M. Khajah, R. Wing, R. V. Lindsey, and M. C. Mozer. Integrating latent-factor and knowledge-tracing models to predict individual differences in learning. In *Proceedings of the International Conference on Educational Data Mining (EDM)*, pages 99–106, 2014.

[18] P. Kulkarni. *Reinforcement and Systemic Machine Learning for Decision Making*. John Wiley & Sons, 2012.

[19] B. Memarian and T. Doleck. A scoping review of reinforcement learning in education. *Computers and Education Open*, 6:100175, 2024.

[20] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. In *12th International Conference on Educational Data Mining, EDM 2019*, pages 384–389. International Educational Data Mining Society, 2019.

[21] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[22] Y. Qiu, Y. Qi, H. Lu, Z. A. Pardos, and N. T. Heffernan. Does time matter? modeling the effect of time with bayesian knowledge tracing. In *Proceedings of the International Conference on Educational Data Mining (EDM)*, pages 139–148, 2011.

[23] I. Šarić-Grgić, A. Grubišić, and A. Gašpar. Twenty-five years of bayesian knowledge tracing: A systematic review. *User Modeling and User-Adapted Interaction*, 34(4):1127–1173, 2024.

[24] A. Shibani, S. Knight, and S. Buckingham Shum. Contextualizable learning analytics design: A generic model and writing analytics evaluations. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK)*, pages 210–219, 2019.

[25] F. Stinar, H. Lee, C. Belitz, N. Nasiar, S. Fancsali, S. Ritter, H. Almoubayyed, R. Baker, J. Ocumpaugh, and N. Bosch. Fairness of bayesian knowledge tracing for math learners of different reading ability. In *Proceedings of the 18th International Conference on Educational Data Mining (EDM)*, pages 170–181, 2025.

[26] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2018.

[27] W. Tang, G. Chen, S. Zu, and J. Luo. Fair knowledge tracing in second language acquisition. *arXiv preprint arXiv:2412.18048*, 2024.

[28] X. Tang, Y. Chen, X. Li, J. Liu, and Z. Ying. A reinforcement learning approach to personalized learning recommendation systems. *British Journal of Mathematical and Statistical Psychology*, 72(1):108–135, 2019.

[29] S. Tschiatschek, M. Knobelsdorf, and A. Singla. Equity and fairness of bayesian knowledge tracing. *arXiv preprint arXiv:2205.02333*, 2022.

[30] W. Xing, Y. Song, C. Li, Z. Liu, W. Zhu, and H. Oh. Development of a generative ai-powered teachable agent for middle school mathematics learning: A design-based research study. *British Journal of Educational Technology*, 56(5):2043–2077, 2025.

[31] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized bayesian knowledge tracing models. In *International Conference on Artificial Intelligence in Education (AIED)*, pages 171–180. Springer, 2013.

[32] F. Zhang, W. Xing, C. Li, and Y. Jiang. Fair ai in educational predictions: A multi-group fairness approach using reinforcement learning. *The Internet and Higher Education*, page 101074, 2026.

[33] X. Zhang, J. Zhang, N. Lin, and X. Yang. Sequential self-attentive model for knowledge tracing. In *International Conference on Artificial Neural Networks*, pages 318–330. Springer, 2021.

[34] Y. Zhou, Z. Lv, S. Zhang, and J. Chen. Disentangled knowledge tracing for alleviating cognitive bias. In *Proceedings of the ACM Web Conference*, pages 2633–2645, 2025.

Federated Code-Aware Knowledge Tracing for Privacy-Preserving Student Performance Forecasting

Anurata Prabha Hridi
North Carolina State University
aphridi@ncsu.edu

Muntasir Hoq
North Carolina State University
mhoq@ncsu.edu

Seyyedali Hosseinalipour
University at Buffalo–SUNY
alipour@buffalo.edu

Rajeev Sahay
UC-San Diego
r2sahay@ucsd.edu

Bita Akram
North Carolina State University
bakram@ncsu.edu

ABSTRACT

Learning to program involves repeated trial-and-error attempts, leading to substantial heterogeneity in learning behavior across student cohorts and course offerings. Code-aware knowledge tracing (KT) models aim to capture how students' programming knowledge evolves over time. However, most existing KT studies rely on data from individual courses, where student interactions are often sparse and insufficient for modeling robust performance patterns. Centralizing student data across classrooms or institutions to address this limitation is often constrained by the risk of privacy breach of sensitive code submission data, limiting the impact of existing KT methods. Federated learning (FL) provides a privacy-preserving alternative by enabling collaborative model training without sharing raw data, and recent work has shown that model personalization can further improve federated KT performance. However, the interaction between model personalization and code representation has not been systematically studied. In this paper, we present a comparative analysis of code-aware KT representations under personalized federated learning (pFL) framework, using two semesters of CodeWorkout data treated as separate clients. Our results show that personalization consistently improves predictive accuracy under federated training, but the magnitude and consistency of these improvements depend on how student code is represented. In particular, code representations that capture richer structural information and syntax-aware encodings exhibit more reliable performance gains and promise more stable learning trajectories across assignments and semesters, especially for more complex programming tasks. These findings provide practical insights for EDM researchers and practitioners on how representation choice shapes the effectiveness of personalization when deploying privacy-preserving knowledge tracing models in programming education.

Keywords

Knowledge tracing, federated learning, model personalization, student performance, predictive modeling.

Anurata Prabha Hridi, Muntasir Hoq, Seyyedali Hosseinalipour, Rajeev Sahay, and Bita Akram. Federated Code-Aware Knowledge Tracing for Privacy-Preserving Student Performance Forecasting. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 453–461. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039681

1. INTRODUCTION

With the growing reliance on learning management systems (LMSs) for the organization and delivery of instructional activities and assessments, time-ordered logs of student interactions with the digital learning environment have become available at scale. Learning analytics leverages these records to track student learning over time, and predict future performance. One prominent approach, known as knowledge tracing (KT) [37], estimates students' latent knowledge states over time and has long been used to uncover learning trajectories, support practice recommendations, and inform adaptive instructional interventions.

Due to their data-driven nature, KT methods benefit from larger and more diverse student interaction datasets, where aggregating student data across courses and semesters enables models to capture nuanced learning behaviors and their evolution over time. However, while such data centralization/pooling can improve model performance through leveraging diverse data, it raises significant privacy concerns. Specifically, regulations like the family educational rights and privacy act (FERPA) [1] restrict student data sharing across institutions to prevent the disclosure of sensitive information. At the same time, while individual/localized training of KT models within a single course avoids these privacy concerns, it often leads to less than optimal performance due to limited data scale and diversity. This limitation is further exacerbated by the nature of student interaction data itself, which is often sparse and unevenly distributed across assignments: some problems receive only a small number of attempts or exhibit limited behavioral variation, constraining the model's ability to learn robust representations of student knowledge and performance, particularly for less frequently attempted or more complex tasks.

Federated learning (FL) has gained traction as a promising distributed machine learning (ML) paradigm to address these challenges [24, 17]. FL allows collaborative model training among multiple data-owning entities, namely *clients*, without requiring the exchange of raw data. Instead, clients train models locally and share only model/gradient parameters with a central server, which aggregates them to form a *global model* that is redistributed for subsequent training

rounds. As a result, by design, FL preserves data privacy while allowing models to benefit from knowledge learned from distributed data sources. In practice, however, data sourced from multiple clients often exhibit substantial heterogeneity - particularly evident in educational domains due to differences in cohort characteristics, instructional methods, or infrastructural contexts. Fairness and correctness considerations being central concerns in education, therefore, makes such one-size-fits-all models less effective. To address this issue, recent work has explored *personalization* strategies in FL, which adapt the global model to client-specific data and have been shown to improve both predictive performance and fairness. As a result, FL with personalization has attracted growing attention in educational applications [3, 33, 35, 14, 21, 13, 22, 43, 39], particularly for tasks such as KT [41, 5, 6].

Yet, programming education introduces unique challenges for KT in FL or simply, *federated KT*. Specifically, as learning happens through iterative code submissions to open-ended programming problems, student-generated code serves as the primary sequential evidence of learning progress, making code representation a central factor in modeling student knowledge. To cater to such distinct needs in programming education, KT models are, therefore, often code-aware, incorporating representations of student code submissions to better capture learning processes. However, while personalized federated KT has been studied, the interaction between code representation choices and personalization of federated models remains underexplored. In particular, it is unclear how different code-aware KT models behave under federated training, and how personalization impacts representation-specific strengths and weaknesses of the model. To address this gap, this paper investigates how code representation shapes the behavior and performance of code-aware KT models trained under personalized federated learning (pFL).

1.1 Research Questions

We extend three code-aware KT models: Code-DKT [38], SANN-DKT [12], and srcML-DKT [27], to FL settings to enable privacy-preserving performance prediction across semesters. To account for cross-semester heterogeneity, we incorporate pFL strategies and empirically analyze their effects across different code representations. To our knowledge, this work is the first study that employs pFL on code representation to provide data privacy and better predictive modeling for code-aware KT methods. The summary of our contributions lies in the answers to the following questions:

RQ1: Do relative performance differences arising from code representation choices in KT models persist under privacy-preserving pFL?

RQ2: To what extent does the personalization aspect of pFL impact cross-semester prediction accuracy in code-aware KT?

2. RELATED WORK

2.1 Modeling Student Learning with KT

Prior work has shown that students engage with programming practice systems in distinct ways, exhibiting varied behavioral patterns such as exploration and persistence [30]. At the same time, programming learning involves mastering a set of underlying skills or knowledge components (KCs) that are reflected across exercises and instructional materials [40]. Understanding how these behaviors relate to the acquisition and application of knowledge over time has been a central focus of learning analytics [37], motivating the development of knowledge tracing (KT) models. Early KT models were probabilistic, most notably Bayesian Knowledge Tracing (BKT) [8], which represents learning as a hidden Markov process and estimates the probability that a student has mastered a skill as they engage with successive tasks. Extensions of BKT introduced individualized parameters to account for variation in prior knowledge and learning rates across students [44]. Related approaches, such as Performance Factors Analysis (PFA) [28] and Learning Factors Analysis (LFA) [4], used logistic regression over prior successes and failures to estimate knowledge states. While influential, these models struggle to capture complex learning dynamics involving multiple interdependent skills and individualized learning trajectories.

The introduction of deep learning marked a major shift in knowledge tracing research. Deep Knowledge Tracing (DKT) [29] showed that recurrent neural networks can capture temporal patterns in student behavior better, leading to notable gains in predictive accuracy. Subsequent work built on this idea by incorporating memory mechanisms, attention, and richer representations. Examples include Dynamic Key-Value Memory Networks (DKVMN) [47] and Self-Attentive Knowledge Tracing (SAKT) [26], as well as later models that combine multiple architectural components to improve scalability and performance [32, 31, 18]. More recent approaches have further expanded the KT landscape by introducing relational and contrastive perspectives, such as graph-based and problem-solving–oriented models [25, 42, 16, 48]. Together, these advances have substantially improved the modeling of student learning behavior.

2.2 Potential of pFL in Code-aware KT

While KT has been effective in many educational settings, open-ended programming courses pose distinct challenges. In such courses, learners often submit multiple iterations of source code, and correctness alone provides only a partial signal of learning. Each attempt contains structural and semantic details that reflects how students reason about problems, debug errors, and refine their solutions over time. The emergence of code-aware KT models prompted moving beyond just correctness labels by explicitly modeling student code. Code-DKT uses transformed code embeddings to represent semantic details of student submissions [38], while SANN-DKT applies attention mechanism to emphasize code parts that contribute most to prediction [12]. srcML-DKT takes this further by allowing incomplete or buggy codeblocks to incorporate syntactic and semantic evolution of student code [27]. Collectively, these studies demonstrate that incorporating structural, semantic, and contextual information from code leads to a more nuanced and fine-grained understanding of programming knowledge acquisition. Additional work has explored alternative representations and objectives, including error-based features for capturing misconceptions (Error-DKT)[36], explainable KT models providing cognitive feedback (HELP-DKT)[23], methods that improve interpretability through automatically generated knowledge components (KCGen-KT)[9] or knowledge graphs (KGNN-KT)[45], and question-centered

approaches that incorporate student-generated questions to support generalization (SQKT)[19].

As efforts to improve KT in programming education continue, they also reveal a key limitation: code-aware KT models typically require large and diverse datasets to perform reliably. In practice, data from a single course or semester are often insufficient, motivating the aggregation of student interaction and code data across offerings or institutions. However, such data centralization/pooling, especially raw code submissions, raises serious privacy concerns and is frequently restricted by institutional policies and regulations. FL has therefore emerged as a promising alternative, enabling collaborative model training without sharing raw data [24]. Prior work has explored federated KT in domain-generic contexts, showing that pFL can improve performance and fairness across heterogeneous clients [41, 5, 6]. However, the interaction between domain-specific code representation and pFL remains largely understudied. Specifically, existing federated KT methods do not examine how different code-aware representations behave under federated training, nor whether personalization impacts their accuracy. To address this gap, our work systematically studies how different code-aware KT representations respond to pFL, providing insights into predictive performance across decentralized programming education settings.

3. METHODOLOGY

3.1 Datasets

In this study, we will analyze two datasets from CodeWorkout, an online platform to practice Java programming.[1] The platform allows instructors to design and deliver programming courses, while providing students with the opportunity to complete exercises and receive immediate feedback [10]. CodeWorkout records detailed logs of student submissions, with scores assigned based on the number of test cases passed. Students may submit multiple solutions for the same problem, allowing us to get chronological updates of their iterative problem-solving processes. The evaluation of student knowledge in these datasets spans a range of programming concepts, including method declarations, loops, arrays, arithmetic operations, logical expressions, data types, conditionals, and nested conditional statements.

Table 1: Dataset details

Semester	Students	Attempts	Compilable	Correct	Incorrect
Spring 19	258	131203	45096	10343	34753
Fall 19	506	360176	125578	24680	100898

Upon its use in an introductory programming (CS1) course at a public university in the United States during the Spring 2019 and Fall 2019 semesters, two datasets were formed containing anonymized submissions for 50 programming problems across five assignments `caughtSpeeding, redTicket, countCode, sum13, and canBalance`) and no accompanying demographic information. [2] We obtained these publicly available datasets from DataShop [20]. Each submission was graded on a continuous scale from 0 to 1 based on the proportion of test cases passed, and for modeling purposes, we categorized the outcomes into binary groups: correct (scored 1 out of 1) and incorrect submissions (failing at least one test case resulted in a 0). Table 1 shows details of the Spring 2019 and Fall 2019 datasets. We split each dataset into training, validation, and test sets in a stratified manner to account for class imbalance, using a 80:10:10 ratio.

[1]https://codeworkout.cs.vt.edu

[2]https://pslcdatashop.web.cmu.edu/Files?datasetId=3458

3.2 KT Formulation

For knowledge tracing, we use deep knowledge tracing (DKT), a well-established model that along with its variants has performed reasonably well when applied on the CodeWorkout dataset [38]. In DKT formulation, a student's learning history is represented as a sequence of attempts $\{x_1, x_2, \ldots, x_T\}$, where each attempt x_t at time step t corresponds to a problem-correctness pair (q_t, a_t). Here, $q_t \in \{1, 2, 3, \ldots, M\}$ denotes the problem identifier for M total problems and a_t can be 0 or 1, indicating incorrectness and correctness, respectively. Each interaction is one-hot encoded as a binary vector $x_t \in \{0, 1\}^{2M}$, where $x_{q_t+M(1-a_t)}$ is considered 1, and the rest of the entries are set to 0. In code-aware DKT methods, each interaction is additionally associated with a code representation z_t, extracted from the student's submitted programs. They are further transformed into vectorized code representations $\{z_1, z_2, \ldots, z_T\}$, which are concatenated with interaction embedding to form an augmented input vector $[x_t; z_t]$, as shown in Figure 1. These inputs are processed sequentially by a long short-term memory (LSTM) model that maintains an internal memory state to capture both short- and long-term temporal dependencies across attempts. The LSTM hidden state provides a latent representation of the student's knowledge and is updated after each interaction. Starting with h_0 that represents the model's belief about the student's knowledge before seeing any data, h_t tracks the evolving knowledge of the student across time t, which can be expressed as $h_t = \text{LSTM}([x_t; z_t], h_{t-1})$, where $x_t = \text{embedding}(q_t, a_t)$. At each time step t, the model outputs a vector of predicted correctness probabilities over all problems $\{y_1, y_2, \ldots, y_T\}$, where each entry denotes the predicted likelihood that the student will answer the corresponding problem correctly. The predicted correctness for the next interaction, a_{t+1}, is obtained by selecting the entry of y_t associated with the next attempted problem q_{t+1}, which is used for performance prediction.

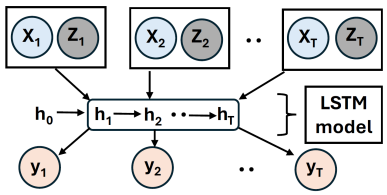


Figure 1: Schematic diagram of code-aware DKT. Across T timesteps, each interaction at t is a concatenation of problem-correctness encoding x_t and student code vector z_t, which are processed sequentially along with latent hidden state h_t to predict correctness probabilities y_t for future attempts.

3.3 FL Framework

The vanilla **Federated Averaging (FedAvg)** algorithm [24], combined separately with each of the three KT techniques, is used as the set of baseline methods in this work. This architecture is presented within the blue box in Figure 2, where,

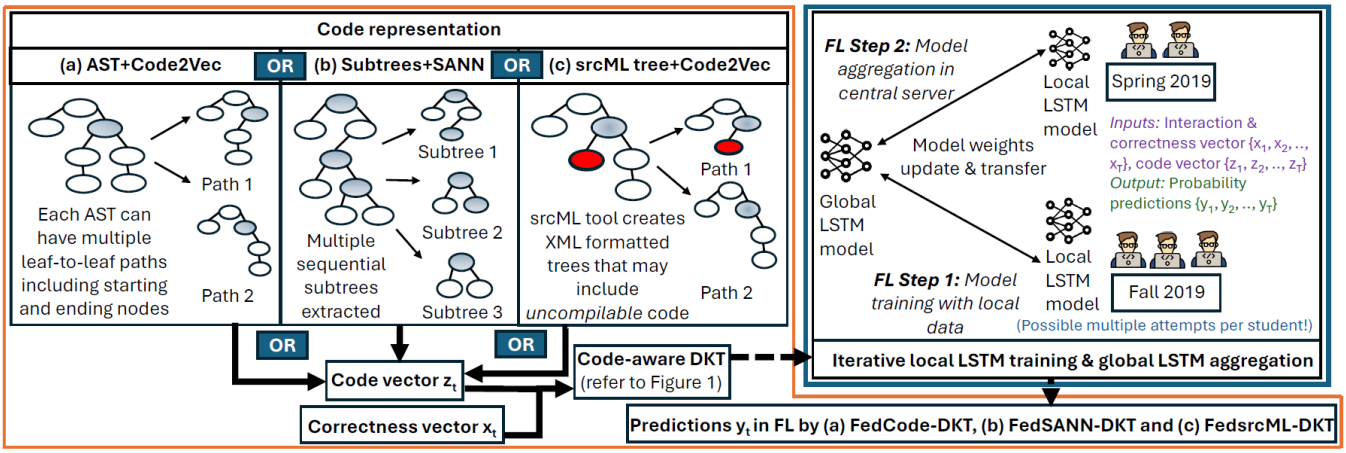


Figure 2: Detailed pipeline of code-aware DKT with FL: orange box contains the entire architecture while blue box shows FL framework. Three code representation strategies are considered. In (a) and (c), gray nodes illustrate how code paths are constructed from an abstract syntax tree by Code2Vec, whereas red nodes in (c) indicate uncompilable code blocks captured by srcML. In (b), subtrees are extracted by SANN from the AST based on gray nodes. Student code submissions are transformed into fixed-length vectors z_t, concatenated with correctness vectors x_t and used to train semester-specific LSTM-based DKT models locally. Model parameters are then aggregated via FedAvg to form separate federated models that predict the probability of future student attempts across semesters y_t.

data from academic semesters are treated as belonging to distinct *clients*, reflecting a realistic scenario, where student interaction data are sourced from different course offerings.

A central server coordinates the federated training process across these clients through iterative rounds of local training and global aggregation. Let K denote the number of clients (semesters), and let $\mathcal{D}_k$ represent the dataset associated with client k, with size $n_k = |\mathcal{D}_k|$. To start each global aggregation τ, the server broadcasts the current global model parameters $\theta^{(\tau)}$ to all clients. Each client initializes its local model with these parameters $\theta_k^{(\tau),0} \leftarrow \theta^{(\tau)}$ and performs local stochastic gradient descent (SGD) updates on mini-batches sampled from its own dataset. Specifically, during each global aggregation τ, client k updates its local parameters by minimizing the empirical loss function ℓ for $e_{\max}$ local SGD steps, where the final local model is:

$$\theta_k^{(\tau+1)} \leftarrow \theta^{(\tau)} - \alpha \sum_{i=0}^{e_{\max}-1} \nabla \ell\big(\theta_k^{(\tau),i}, \mathcal{B}_k^{(\tau),i}\big), \tag{1}$$

where α denotes the learning rate and $\mathcal{B}_k^{(\tau),i}$ represents a mini-batch of size B sampled from client k's data at SGD iteration i. Subsequently, the client sends $\theta_k^{(\tau+1)}$ to the server, and the server aggregates the local models using weighted averaging to obtain the next global model as follows:

$$\theta^{(\tau+1)} = \sum_{k=1}^{K} \frac{n_k}{\sum_{k'=1}^{K} n_{k'}} \theta_k^{(\tau+1)}. \tag{2}$$

The resulting global model $\theta^{(\tau+1)}$ is then reshared with all clients to initiate the next training round. In our experiments, this federated optimization process is repeated for a total of $\mathcal{T} = 10$ global rounds (i.e., $\tau \in \{0, \cdots, 9\}$). All remaining FL hyperparameters are selected via grid search, with learning rates $\alpha \in \{0.001, 0.005\}$, mini-batch sizes $B \in \{128, 256\}$, and the number of local training steps $e_{\max} \in \{20, 40\}$.

We next apply this FL framework to three code-aware KT models, yielding the novel FedCode-DKT, FedSANN-DKT, and FedsrcML-DKT models, which are discussed below.

3.3.1 *FedCode-DKT*

Code-DKT builds upon the original DKT framework by augmenting the problem-correctness interaction x_t with a code representation z_t derived from the student's attempt logged, enabling the model to leverage information beyond binary correctness [38]. As illustrated in Figure 2(a), each code submission at time t is transformed into an abstract syntax tree (AST), which serves as the basis for extracting the corresponding code representation z_t. Code-DKT employs `Code2Vec` [2], a neural model embeds leaf-to-leaf paths in the AST, to transform ASTs into embedded vectors ready to be used for any ML model. Code2Vec is originally designed to predict Java method names, and its internal representations can be retrieved as code embeddings, providing a compact vectorized representation of the underlying program structure. Using separate node and path embedding layers, embeddings for each of the paths are generated as vectors. These vectors are then fed into the attention layer, where the correctness score x_t determines which path is important for the final prediction. Following softmax activation and concatenation of weighted sum, the aggregated vector of correctness probabilities is generated as the output z_t of the linear layer. Once the vectors are formed for the entire dataset for reported timestamps $1, 2, 3, \ldots, T$, they are fed as inputs to respective LSTM models (identified by semesters) along with the correctness scores for all T attempts. Each semester is locally trained with their respective data, where the global model is updated with the aggregated local model parameters. After a back-and-forth transfer for 10 iterations, the federated model outputs prediction probabilities.

3.3.2 *FedSANN-DKT*

SANN-DKT introduces a self-attention-based code representation strategy to highlight critical portions of the source

code that are most indicative of future performance [12]. It employs *subtree-based attention neural network (SANN)* [11] instead of ASTs, as demonstrated in Figure 2(b). Under the SANN framework, a single code submission is decomposed into a set of *node-specific, order-preserving subtrees* generated via depth-first, preorder traversal. Each subtree corresponds to a semantically meaningful code fragment, such as loops, conditional blocks, or method invocations. These subtrees are transformed into vector representations through two complementary embedding operations: *node embeddings*, which encode individual nodes within a subtree, and *subtree embeddings*, which capture subtree-level structural information. The node embeddings within each subtree are first aggregated, and the resulting vector is concatenated with the corresponding subtree embedding to form a unified subtree representation. The resulting set of subtree vectors is then passed through an attention mechanism that assigns higher weights to code fragments most informative of a student's understanding or programming concepts. The aggregated representation of weighted subtrees is originally fed into a neural network for classification, predicting whether a student is likely to succeed or struggle. In this work, however, we do not employ SANN in an end-to-end classification setting. Instead, we extract the intermediate code representations z_t produced by the fully connected layer following the attention mechanism and softmax normalization, and use these vectors along with correctness vector x_t as inputs to the LSTM model. *It is to be noted here that unlike Code2Vec, which is a pretrained encoding model, SANN directly learns from the same client populations that participate in federated training for knowledge tracing.* Therefore, to ensure that the resulting code representations reflect cohort-specific characteristics, we federate the SANN pipeline itself with the client data before the rest of the FL pipeline is combined.

3.3.3 FedsrcML-DKT

srcML-DKT extends this line of work by adopting `srcML` as the underlying code representation, enabling the modeling of *uncompilable* code submissions and thereby capturing syntactic errors that arise as students iteratively refine their solutions [27]. By replacing AST-based representations with srcML, as shown in Figure 2(c), this framework can parse code fragments that fail to compile and convert them into XML-based structures [7]. These representations can encode both syntactically correct and incorrect code by wrapping source text within XML tags that describe structural elements, while preserving erroneous or incomplete constructs textually. As a result, srcML-DKT incorporates *all* student submissions, rather than discarding non-compilable attempts. Following a pipeline analogous to Code2Vec, the srcML representations are subsequently transformed into fixed-length vector embeddings z_t, which are then provided as inputs, along with correctness vector x_t, to the LSTM-based KT model in FL setting. This design allows to model the evolution of student coding strategies across attempts while maintaining data privacy and retaining fine-grained syntactic information, even in the presence of compile errors.

3.4 Personalization of FL Models via pFL

Although both datasets originate from the same CS1 course and share identical feature definitions, the two semesters correspond to different student cohorts. Variations in prior experience, learning pace, and coding behavior lead to systematic shifts in interaction patterns and code structures, resulting in non-IID feature distributions across semesters, confirmed by statistically significant differences in a Kolmogorov–Smirnov (KS) test. Under such cohort-induced heterogeneity, a single global federated model may not perform uniformly across semesters, motivating the use of pFL to adapt to local data distributions. We adopted **Federated Adaptive Local Aggregation (FedALA)** [46], a pFL approach that extends vanilla FedAvg by introducing a personalized model synchronization mechanism at each client, as shown in Figure 3. This choice is motivated by FedALA's superior capability of sufficiently adapting to local settings than simple post-hoc fine-tuning of the federated model [13].

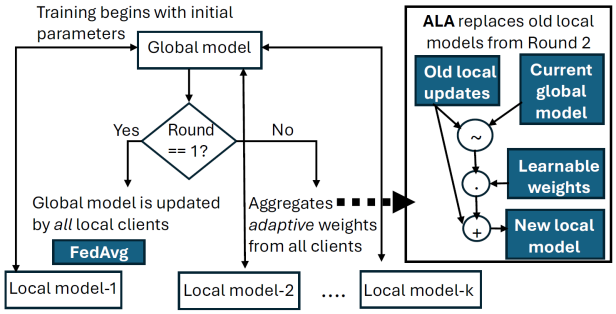


Figure 3: FedALA applies personalization on FedAvg.

While local training and global aggregation follow the standard FedAvg procedure, FedALA replaces direct overwriting of local models with an *adaptive local aggregation* (ALA) step. Let $\theta^{(\tau)}$ denote the global model parameters broadcast by the server at global aggregation round τ, and let $\theta_k^{(\tau-1)}$ denote the final local model parameters of client k obtained at the end of the previous round. Rather than initializing the local model as $\theta_k^{(\tau),0} \leftarrow \theta^{(\tau)}$, FedALA initializes the local parameters by adaptively combining the previous local model with the received global model:

$$\theta_k^{(\tau),0} = \theta_k^{(\tau-1)} + \left(\theta^{(\tau)} - \theta_k^{(\tau-1)}\right) \odot \lambda_k, \tag{3}$$

where λ_k is a learnable, element-wise aggregation weight matrix constrained to the interval $[0, 1]$, and $\odot$ denotes element-wise multiplication. Instead of initializing the local model with the received global model, this adaptive aggregation allows each client to selectively incorporate globally shared information while suppressing updates that are less relevant to its local data distribution. FedALA further exploits the hierarchical structure of deep neural networks by applying adaptive aggregation only to the top l layers of the model, while the remaining lower layers (assumed to encode more general representations) are directly inherited from the global model without modification. FedALA learns λ_k using a fraction ζ of each client's local dataset. This design balances global knowledge sharing with client-specific adaptation. We tune the FedALA hyperparameters through grid search, with $l \in \{1, 2\}$ and $\zeta \in [5, 100]$ with increments of 5.

4. RESULTS & DISCUSSIONS

To evaluate the prediction task, we considered all student attempts across five programming assignments (*A1-A5*) and addressed our two research questions (RQ), described below.

We measure predictive performance using area under curve (AUC): as programming datasets are highly imbalanced, AUC is more informative than accuracy, with 0.5 indicating random prediction and 1 indicating perfect discrimination. To answer RQ2, in addition to embedding-based representations, we report results for a non-embedding TF-IDF [34] method that represents code as weighted word-frequency vectors capturing surface-level patterns only.

RQ1: What does federated code representation performance difference mean in code-aware KT model performance? Figure 4 shows assignment-level AUC differences between Code-DKT and SANN-DKT and srcML-DKT and SANN-DKT under individual (i.e., sole local training) and federated settings, where positive values indicate higher AUC for SANN-DKT framework. As a reference, we used results reported by Hoq et al.[12] on Spring 2019 dataset, and examined whether SANN-DKT outperforming Code-DKT remains consistent under privacy-preserving FL. We replicated the same analysis on Fall 2019 dataset and the rank persisted. Further, we also implemented srcML-DKT and observed srcML-DKT performing better than SANN-DKT in most cases. Our results in Figure 4 top row indicate that FedSANN-DKT largely retains its code representation-driven performance advantage over FedCode-DKT, achieving significantly higher AUC values, confirmed by a paired Wilcoxon signed-rank test ($p < 0.05$). Bottom row indicates similar outcomes between FedsrcML-DKT and FedSANN-DKT, further demonstrating that FL does not fundamentally disrupt the performance advantages conferred by richer code representations. However, this phenomenon is not uniform across all assignments. We observe small, assignment-specific reversals in relative performance, primarily on simpler tasks where overall AUC is already high and solution strategies exhibit limited structural diversity. This effect is most evident for A1 in top-left subfigure, where near-saturated performance constrains the marginal benefit of SANN's neighborhood-based structural encoding over simpler path-based representations, making model rankings more sensitive to cohort-induced variation amplified through federated aggregation. In contrast, as assignments become conceptually more complex (A2–A5), and understanding code organization (i.e., combining multiple constructs and managing program control flow) is critical, representations that explicitly encode program structure more frequently outperform simpler path-based encodings under federated setting, resulting in $AUC_{FedSANN-DKT} > AUC_{FedCode-DKT}$. This pattern is consistent with the performance hierarchy reported in individual training settings [12], unveiling that FL tends to preserve code representation rank with observed variations influenced by task complexity and data sparsity. These findings motivate a closer examination of how personalization interacts with representation choice under FL, which we investigate next in the context of RQ2.

RQ2: How does model personalization affect federated code representation across semesters/clients? Personalization via FedALA substantially improves cross-semester predictive accuracy in all *code-aware* federated KT models, with gains varying by assignment difficulty and code representation, as shown in Figure 5. Although TF-IDF is not a focus of our method, its inclusion here underscores that personalization by itself is insufficient; meaningful and consistent improvements in federated settings emerge primarily when personalization

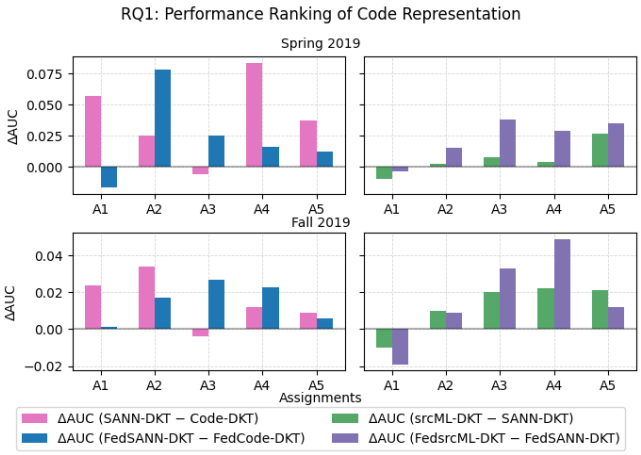


Figure 4: Performance gains ($AUC_{SANN-DKT} > AUC_{Code-DKT}$ [12]) are largely preserved in FL for Spring 2019 (small flips occur on simpler assignments where differences are tiny). Similar results are obtained in other cases.

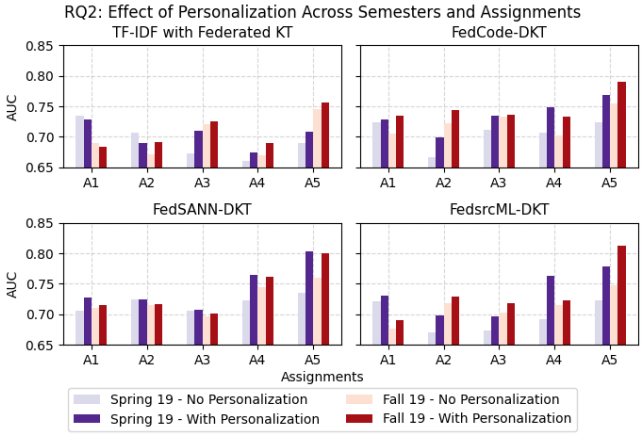


Figure 5: Though personalization impact on no-embedding technique TF-IDF was not reliably determined, personalization of FedCode-DKT, FedSANN-DKT and FedsrcML-DKT significantly outperform their baseline FL counterparts based on AUC differences (mean standard deviations: 0.0015, 0.008 and 0.0012, respectively) across semesters and assignments.

operates over embedding-based, code-aware representations. In contrast, embedding-based approaches show larger and more consistent personalization gains, with FedCode-DKT and FedSANN-DKT exhibiting improved alignment across semesters, particularly on later assignments. The effect is most pronounced for FedsrcML-DKT, where personalization leads to substantial and stable gains on structurally complex assignments (A4–A5) and reduces cross-semester performance gaps. A paired t-test shows that these sets of improvements achieved by FedALA over FedAvg are statistically significant ($p < 0.05$). These results suggest that though improvement magnitude varies based on task complexity, personalization does not degrade performance, instead it has the potential to better adapt to cohort-specific patterns in code-aware KT, while preserving privacy.

5. IMPLICATIONS FOR EDM & LIMITATIONS

Informed by our findings, personalized federated model generates actionable insights that instructors can use to identify students who may be struggling early in a privacy-preserving manner, particularly those showing repeated errors or inefficient coding patterns. These patterns reflect how students iteratively refine their problem-solving strategies over time through repeated attempts, errors, and code revisions.

However, two limitations should be noted despite promising results. First, our analysis is based on a single programming language Java. Since programming languages differ in syntax and structure, the observed performance trends may not generalize directly to other contexts such as Python or C++. Second, we model FL using two semesters of the same course from one institution. While there are some cohort differences, the overall learning environment remains similar. This limits the level of data heterogeneity compared to real-world federated settings that involve multiple institutions and more diverse student populations. Consequently, the extent to which the model performs consistently across diverse learner groups remains an open question. The close connection between fairness and model usefulness in human-centered applications [15] highlights the need for future evaluations on more diverse, cross-institutional datasets beyond predictive performance alone.

6. CONCLUSION & FUTURE WORK

In this paper, we studied how different code-aware knowledge tracing (KT) models behave under federated learning (FL), with a particular focus on code representation and model personalization. To this end, we present the first integration of personalized FL (pFL) with code embedding methods as an initial step toward privacy-preserving student code modeling and improved performance prediction. Using semester-level federated training on programming course data, we show that code representation type continues to influence predictive performance under FL, which pFL can further improve besides reducing cross-semester performance disparities through semester-specific adaptation. *From a pedagogical perspective, this framework enables equitable and privacy-preserving instructional support regarding code-level misunderstanding.* Our findings also indicate that retaining uncompilable code and incomplete attempts, which reflect authentic learning processes, leads to the strongest performance among the evaluated representations in pFL settings.

A key challenge in programming education datasets is the prevalence of incorrect submissions relative to correct ones, making it difficult for models to learn from the relatively few correct attempts. This limitation motivates future exploration of federated few-shot learning method to ensure that infrequent but informative demonstrations of understanding are not overlooked. Further, we plan to include features derived from student and system engagement data in our analysis, which our preliminary explorations suggest may capture insights of both coding and learning behavior. *In practice, this will allow instructors to distinguish between students who struggle conceptually and those who are actively engaging but iterating through errors only, enabling more targeted instructional support.* While our study focuses on a limited set of representations and course offerings, it highlights key design considerations for deploying privacy-preserving learning analytics in programming education, and motivates future research on representation-aware federated KT with programming-specific modeling choices, broader datasets, and richer FL personalization strategies.

7. ACKNOWLEDGMENT

This work was supported in part by the U.S. National Science Foundation (NSF) under Grant No. SaTC-2513164, ECCS-2512911, ECCS-2543754, and DRL-2418658.

References

[1] Family educational rights and privacy act (FERPA). https://epic.org/family-educational-rights-and-privacy-act-ferpa/, Online; accessed 19 Feb. 2025.

[2] U. Alon, M. Zilberstein, O. Levy, and E. Yahav. code2vec: Learning distributed representations of code. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–29, 2019.

[3] S. Bhattacharya, P. Vyas, S. Yarradoddi, B. Dasari, S. Sumathy, R. Kaluri, S. Koppu, D. J. Brown, M. Mahmud, and T. R. Gadekallu. Towards smart education in the industry 5.0 era: A mini review on the application of federated learning. In *2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pages 0602–0608. IEEE, 2023.

[4] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*, pages 164–175. Springer, 2006.

[5] Y.-W. Chu, S. Hosseinalipour, E. Tenorio, L. Cruz, K. Douglas, A. Lan, and C. Brinton. Mitigating biases in student performance prediction via attention-based personalized federated learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 3033–3042, 2022.

[6] Y.-W. Chu, S. Hosseinalipour, E. Tenorio, L. Cruz, K. Douglas, A. S. Lan, and C. G. Brinton. Multi-layer personalized federated learning for mitigating biases in student predictive analytics. *IEEE Transactions on Emerging Topics in Computing*, X:1–15, 2024.

[7] M. L. Collard, M. J. Decker, and J. I. Maletic. srcml: An infrastructure for the exploration, analysis, and manipulation of source code: A tool demonstration. In *2013 IEEE International conference on software maintenance*, pages 516–519. IEEE, 2013.

[8] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

[9] Z. Duan, N. Fernandez, A. B. L. Narayanan, M. Hassany, R. S. de Alencar, P. Brusilovsky, B. Akram, and A. Lan. Automated knowledge component generation and knowledge tracing for coding problems. *arXiv preprint arXiv:2502.18632*, 2025.

[10] S. H. Edwards and K. P. Murali. Codeworkout: short programming exercises with built-in data collection. In *Proceedings of the 2017 ACM conference on innovation and technology in computer science education*, pages 188–193, 2017.

[11] M. Hoq, S. R. Chilla, M. Ahmadi Ranjbar, P. Brusilovsky, and B. Akram. Sann: Programming code representation using attention neural network with optimized subtree extraction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 783–792, 2023.

[12] M. Hoq, A. Rao, R. Jaishankar, K. Piryani, N. Janapati, J. Vandenberg, B. Mott, N. Norouzi, J. Lester, and B. Akram. Automated identification of logical errors in programs: Advancing scalable analysis of student misconceptions. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 90–103, Palermo, Italy, July 2025. International Educational Data Mining Society.

[13] A. P. Hridi, M. Hoq, Z. Gao, C. Lynch, R. Sahay, S. Hosseinalipour, and B. Akram. Privacy-preserving distributed link predictions among peers in online classrooms using federated learning. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 276–288, Palermo, Italy, July 2025. International Educational Data Mining Society.

[14] A. P. Hridi, R. Sahay, S. Hosseinalipour, and B. Akram. Revolutionizing ai-assisted education with federated learning: a pathway to distributed, privacy-preserving, and debiased learning ecosystems. In *Proceedings of the AAAI Symposium Series*, volume 3, pages 297–303, 2024.

[15] A. P. Hridi and B. Watson. Are fair machine learning models more useful? In *International Conference on Human-Computer Interaction*, pages 38–53. Springer, 2024.

[16] T. Huang, X. Ou, H. Yang, S. Hu, J. Geng, J. Hu, and Z. Xu. Remembering is not applying: Interpretable knowledge tracing for problem-solving processes. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3151–3159, 2024.

[17] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

[18] M. A. Kia and A. Koliousis. Ackt: Self-attentive convolutional model for knowledge tracing. *Procedia Comput. Sci.*, 246(C):2723–2732, Jan. 2024.

[19] D. Kim, S. Kim, and Y. Jo. Knowledge tracing in programming education integrating students' questions. In W. Che, J. Nabende, E. Shutova, and M. T. Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27703–27718, Vienna, Austria, July 2025. Association for Computational Linguistics.

[20] K. R. Koedinger, R. S. Baker, K. Cunningham, A. Skogsholm, B. Leber, and J. Stamper. A data repository for the edm community: The pslc datashop. *Handbook of educational data mining*, 43:43–56, 2010.

[21] S. Lamsiyah, A. Nourbakhsh, S. El-Amrany, and C. Schommer. Privacy-preserving federated learning for student dropout prediction: Enhancing model transparency with explainable ai. In *International Conference on Artificial Intelligence in Education*, pages 324–332. Springer, 2025.

[22] E. Latif and X. Zhai. Privacy-preserved automated scoring using federated learning for educational research. In *International Conference on Artificial Intelligence in Education*, pages 135–143. Springer, 2025.

[23] Y. Liang, T. Peng, Y. Pu, and W. Wu. Help-dkt: an interpretable cognitive model of how students learn programming based on deep knowledge tracing. *Scientific Reports*, 12(1):4012, 2022.

[24] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In A. Singh and J. Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282, Fort Lauderdale, FL, USA, Apr. 2017. PMLR.

[25] H. Nakagawa, Y. Iwasawa, and Y. Matsuo. Graph-based knowledge tracing: Modeling student proficiency using graph neural network. In *IEEE/WIC/ACM International Conference on Web Intelligence*, WI '19, page 156–163, New York, NY, USA, 2019. Association for Computing Machinery.

[26] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. In C. F. Lynch, A. Merceron, M. Desmarais, and R. Nkambou, editors, *Proceedings of the 12th International Conference on Educational Data Mining*, pages 384–389. International Educational Data Mining Society, 2019.

[27] M. Pankiewicz, Y. Shi, and R. S. Baker. srcml-dkt: Enhancing deep knowledge tracing with robust code representations from srcml. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 541–548. International Educational Data Mining Society, July 2025.

[28] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis –a new alternative to knowledge tracing. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, page 531–538, NLD, 2009. IOS Press.

[29] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[30] A. Poh, A. P. Hridi, J. Barria-Pineda, P. Brusilovsky, and B. Akram. Example explorers and persistent finishers: Exploring student practice behaviors in a python practice system. In *Proceedings of the 9th Educational Data Mining in Computer Science Education (CSEDM) Workshop*, volume 4019 of *CEUR Workshop Proceedings*, Sicily, Italy, July 2025.

[31] S. Pu, G. Converse, and Y. Huang. Deep performance factors analysis for knowledge tracing. In *International Conference on Artificial Intelligence in Education*, pages 331–341. Springer, 2021.

[32] S. Pu, M. Yudelson, L. Ou, and Y. Huang. Deep knowledge tracing with transformers. In *International conference on artificial intelligence in education*, pages 252–256. Springer, 2020.

[33] Y. Qin, M. Li, and J. Zhu. Privacy-preserving federated learning framework in multimedia courses recommendation. *Wireless Networks*, 29(4):1535–1544, 2023.

[34] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, NY, 1988.

[35] D. Sengupta, S. S. Khan, S. Das, and D. De. Fedel: Federated education learning for generating correlations between course outcomes and program outcomes for internet of education things. *Internet of Things*, page 101056, 2024.

[36] M. Shaka, D. Carraro, and K. Brown. Error tracing in programming: a path to personalised feedback. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 330–342, 2024.

[37] S. Shen, Q. Liu, Z. Huang, Y. Zheng, M. Yin, M. Wang, and E. Chen. A survey of knowledge tracing: Models, variants, and applications. *IEEE Transactions on Learning Technologies*, 17:1858–1879, 2024.

[38] Y. Shi, M. Chi, T. Barnes, and T. Price. Code-DKT: A code-based knowledge tracing model for programming tasks. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, pages 50–61, Durham, United Kingdom, July 2022. International Educational Data Mining Society.

[39] R. Sinha and V. Jaglan. A federated learning approach for personalized online course recommendation using hybrid recommender strategies. In *2025 International Conference on Computing and Communications (COMPUTINGCON)*, pages 1–7. IEEE, 2025.

[40] S. Smith, A.-B. Lekshmi Narayanan, A. P. Hridi, R. Sampaio de Alencar, B. Akram, A. Hellas, J. Leinonen, P. Brusilovsky, and N. Norouzi. Knowledge component-driven alignment of cs1 textbooks and exercises. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V. 2*, pages 1533–1534, 2026.

[41] J. Wu, Z. Huang, Q. Liu, D. Lian, H. Wang, E. Chen, H. Ma, and S. Wang. Federated deep knowledge tracing. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, page 662–670, New York, NY, USA, 2021. Association for Computing Machinery.

[42] C.-K. Yeung. Deep-irt: Make deep learning based knowledge tracing explainable using item response theory. In C. F. Lynch, A. Merceron, M. Desmarais, and R. Nkambou, editors, *Proceedings of the 12th International Conference on Educational Data Mining*, pages 683–686. International Educational Data Mining Society, 2019.

[43] S. Yoneda, V. Svabensky, G. Li, D. Deguchi, and A. Shimada. Ranking-based at-risk student prediction using federated learning and differential features. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 289–302, Palermo, Italy, July 2025. International Educational Data Mining Society.

[44] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pages 171–180. Springer, 2013.

[45] D. Zhang, Q. Niu, T. Wang, Y. Hou, J. Wu, C. Zhang, and A. Stefanidis. Kgnn-kt: Enhancing knowledge tracing in programming education through llm-extracted knowledge graphs. In *International Conference on Intelligent Computing*, pages 137–147. Springer, 2025.

[46] J. Zhang, Y. Hua, H. Wang, T. Song, Z. Xue, R. Ma, and H. Guan. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11237–1fedl2p1244, 2023.

[47] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774, 2017.

[48] S. Zhao, C. Wang, and S. Sahebi. Transition-aware multi-activity knowledge tracing. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1760–1769. IEEE, 2022.

From Flat to Structural: Enhancing Automated Short Answer Grading with GraphRAG

Yucheng Chu
Michigan State University
chuyuch2@msu.edu

Haoyu Han
Michigan State University
hanhaoy1@msu.edu

Shen Dong
Michigan State University
dongshe1@msu.edu

Hang Li
Michigan State University
lihang4@msu.edu

Kaiqi Yang
Michigan State University
kqyang@msu.edu

Yasemin Copur-Gencturk
University of Southern California
copurgen@usc.edu

ABSTRACT

Automated short answer grading (ASAG) is critical for scaling educational assessment, yet large language models (LLMs) often struggle with hallucinations and strict rubric adherence due to a mismatch between generalized pre-training objectives and the task-specific requirements of grading. While Retrieval-Augmented Generation (RAG) can reduce hallucinations by providing access to relevant reference materials, standard flat vector retrieval treats knowledge as isolated fragments, failing to capture the structural relationships and multi-hop dependencies inherent in complex educational content. To explore whether incorporating structural information is beneficial for ASAG, we investigate Graph Retrieval-Augmented Generation (GraphRAG) as a general paradigm and systematically evaluate two representative instantiations, Microsoft GraphRAG and HippoRAG, on this task. These approaches differ in how they construct and traverse knowledge graphs, providing complementary perspectives on graph-based evidence retrieval. Experimental evaluations on a Next Generation Science Standards (NGSS) dataset demonstrate that structural retrieval outperforms standard RAG baselines across all metrics. Notably, the HippoRAG achieved substantial improvements in evaluating Science and Engineering Practices (SEP), confirming the superiority of structural retrieval in verifying the logical reasoning chains required for higher-order academic assessment.

Keywords

Large Language Model, Automated Grading,
Retrieval-Augmented Generation (RAG), GraphRAG

1. INTRODUCTION

Automated short answer grading (ASAG) has emerged as a critical technology for scaling educational assessment, aiming to evaluate student responses with the same accuracy and consistency as human instructors. While early approaches relied on keyword matching and latent semantic analysis, the advent of large language models (LLMs) has fundamentally transformed the landscape. LLMs possess remarkable semantic understanding and generation capabilities. These features allow them to not only interpret complex student answers but also, crucially, to understand and adhere to nuanced grading rubrics with human-level flexibility. However, despite their fluency, off-the-shelf LLMs frequently struggle with the strict constraints of academic grading. They are prone to "hallucinations" where plausible but incorrect justifications are generated. Furthermore, they often fail to adhere strictly to specific grading rubrics, relying instead on their generalized internal knowledge acquired during pre-training. This vast but generic training data often conflicts with specific course definitions or omits the precise constraints required by a local curriculum. To mitigate these limitations, recent works have integrated retrieval-augmented generation (RAG) into the grading pipeline. A prominent example of this advancement is the system proposed by Chu et al., which enhances LLM grading by grounding the model's decisions in a specific knowledge base [1]. Their approach constructs a repository of reference answers and scoring criteria, utilizing a dual retrieval strategy to fetch the most relevant grading standards for a given student response. By dynamically providing the LLM with the exact rubric criteria needed for a specific question, this RAG-based method significantly reduces hallucination and improves the alignment between automated scores and human judgment. The system effectively converts the grading task from an open-ended generation problem into a grounded reasoning task, where the model must support its evaluation with retrieved evidence.

Despite the improvements offered by standard RAG frameworks, important technical limitations remain when they are applied to complex educational content. Standard RAG systems typically rely on vector similarity search, which encodes text into high-dimensional semantic embeddings and retrieves chunks of information that are semantically related to the query. This flat retrieval method is widely adopted because it is simple and efficient. However, it treats knowledge as isolated fragments and does not explicitly represent the structural relationships among concepts, criteria, and examples. A key issue in ASAG is not only whether a relevant rubric chunk can be retrieved, but whether the retrieved ev-

Yucheng Chu, Haoyu Han, Shen Dong, Hang Li, Kaiqi Yang, Yasemin Copur-Gencturk, Joseph Krajcik, Namsoo Shin, and Hui Liu. From Flat to Structural: Enhancing Automated Short Answer Grading with GraphRAG. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 462–468. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039760

idence preserves the relational structure needed to evaluate a student's reasoning. For example, a rubric may describe a correct explanation using formal scientific terminology, such as "enzymatic denaturation leading to substrate-binding failure," while a student may write that "the enzyme stops working because heat changes its shape." A reference-anchored flat-RAG query can help retrieve the formal rubric language, and a student-anchored query can retrieve passages related to heat or reaction failure. However, both approaches still return independent text chunks rather than an explicit path connecting heat, structural change, binding failure, and reaction outcome. The grader must then infer the missing relationships from fragmented evidence. This context fragmentation can degrade automated grading, especially when the scoring decision depends on whether a student has connected multiple ideas in a scientifically valid way. When retrieval provides isolated facts but not their dependencies, the LLM may still rely on its pre-trained knowledge to infer the relationship among them. This reintroduces the risk of hallucination or inconsistent rubric adherence, even when some relevant text has been retrieved.

To address the inability of flat retrieval systems to capture deep semantic structure, this work introduces a novel grading framework based on graph retrieval-augmented generation (GraphRAG) [5]. Unlike standard RAG, which indexes information as independent vectors, GraphRAG organizes the reference material and grading rubrics into a structured *knowledge graph* (KG). In this architecture, specific concepts, criteria, and keywords act as nodes, while the logical relationships between them, such as hierarchy, causality, and prerequisites, are represented as edges. This graph-based structure allows the system to move beyond simple similarity matching. When evaluating a student's answer, our system does not merely fetch isolated text chunks; it traverses the knowledge graph to retrieve a connected subgraph of information. This enables the retrieval of "multi-hop" evidence, where the system can identify that concept A is relevant to the student's answer because it is causally linked to concept B, even if concept A was not explicitly mentioned in the initial query. By preserving the relational context of the reference material, our approach provides the LLM with a holistic view of the grading criteria, rather than fragmented snapshots. Our method is particularly advantageous for questions requiring higher-order thinking, where the correctness of an answer depends on the synthesis of multiple concepts. The following sections detail the construction of the grading knowledge graph, the graph traversal algorithms employed during the grading phase, and the comparative experimental results. Our evaluations on Next Generation Science Standards (NGSS) tasks demonstrate that this structural retrieval approach significantly outperforms standard RAG baselines, with the HippoRAG architecture achieving particularly substantial improvements in verifying the logical reasoning chains required for Science and Engineering Practices.

2. RELATED WORK

The development of automated short answer grading (ASAG) has evolved from simple lexical matching to complex semantic reasoning systems. This section reviews the historical progression of grading technologies, the recent integration of retrieval-augmented generation (RAG) to address hallucination, and the emerging paradigm of graph-based retrieval which forms the technical foundation of this work.

2.1 Automated Short Answer Grading

Early approaches to ASAG relied heavily on statistical and keyword-based methods. Techniques such as latent semantic analysis (LSA) and unigram overlap were employed to measure the similarity between a student's response and a reference answer [4, 7]. While computationally efficient, these methods operated on a "bag-of-words" assumption, ignoring word order and syntactic structure, which often led to the misclassification of answers that used correct terminology in incorrect contexts. The field advanced significantly with the introduction of pre-trained language models (PLMs) such as BERT. These models utilized self-attention mechanisms to capture contextual nuances, allowing for more accurate semantic matching. However, BERT-based systems were primarily discriminative, treating grading as a classification task (correct vs. incorrect) rather than generating interpretable feedback. The advent of large language models (LLMs) shifted the paradigm towards generative grading, where models could both score an answer and provide a justification. Despite this capability, generative LLMs suffer from a lack of domain specificity and a tendency to hallucinate, fabricating plausible but factually incorrect reasoning when the model's internal knowledge conflicts with specific course rubrics.

2.2 RAG in Education

To mitigate the hallucination and context limitations of off-the-shelf LLMs, researchers began adopting RAG [3]. RAG systems decouple the model's reasoning capabilities from its knowledge storage by retrieving relevant documents from an external corpus and injecting them into the model's context window. This approach is particularly vital in educational settings where grading criteria are rigid and course-specific. The work of Chu et al. represents the state-of-the-art in this domain [1]. Their system utilizes a dense vector retrieval mechanism to fetch specific reference answers and scoring rubrics relevant to a student's query. By grounding the LLM's generation in these retrieved artifacts, they successfully demonstrated a reduction in grading errors and improved alignment with human evaluators. However, standard RAG systems, including the one proposed by Chu et al., rely on "flat" retrieval strategies. They chunk documents into independent text segments and retrieve them based on vector similarity. This process treats knowledge as isolated fragments, often failing to retrieve information that is conceptually related but not explicitly similar in vector space, such as multi-step causal chains or hierarchical dependencies essential for grading complex questions.

2.3 GraphRAG

Graph retrieval-augmented generation (GraphRAG) has emerged as a solution to the limitations of flat retrieval by introducing structural awareness into the generation pipeline [6]. Unlike standard RAG, which views data as an unstructured collection of text chunks, GraphRAG organizes information into a knowledge graph (KG), where entities (concepts) are nodes and relationships (dependencies, causality) are edges. Microsoft Research's recent implementation of GraphRAG demonstrates that traversing these graph

structures allows LLMs to perform "global" reasoning over a dataset, connecting disparate pieces of information that standard vector search would miss [2]. In the context of grading, this allows a system to recognize that a student's answer regarding a specific biological function is correct not just because it matches a keyword, but because it correctly identifies the prerequisite process defined in the knowledge graph. By traversing the edges of the graph, the retrieval module can fetch a subgraph containing the question, the direct answer, and the underlying principles, providing the LLM with a structurally complete context for evaluation.

3. PROBLEM STATEMENT

While the integration of retrieval-augmented generation (RAG) into grading pipelines has significantly reduced the incidence of hallucination, current state-of-the-art systems remain constrained by their underlying retrieval mechanisms. The fundamental limitation lies in the mismatch between the "flat" data representation used by standard vector retrievers and the highly structured, relational nature of educational knowledge.

3.1 Limitations of Flat Vector Retrieval

Standard RAG systems, including the baseline framework proposed by Chu et al. [1], operate on the principle of semantic similarity matching. In this paradigm, the grading rubric and reference materials are segmented into discrete text chunks, $D = \{d_1, d_2, ..., d_n\}$, which are then encoded into high-dimensional vector embeddings. Given a student's answer as a query q, the system retrieves a subset of chunks based on a similarity metric, such as cosine similarity, expressed as $Score(q, d_i) = \cos(\mathbf{v}_q, \mathbf{v}_{d_i})$.

This approach relies on the assumption that the necessary grading criteria are semantically proximate to the student's answer in vector space. However, this assumption frequently fails in educational contexts where concepts are linked by logical dependencies rather than mere lexical overlap. A student's response might correctly discuss a downstream effect (concept C) of a biological process (concept A) without explicitly mentioning the intermediate mechanism (concept B). A flat retriever, searching for terms related to A and C, is likely to retrieve chunks containing only A or C. It often fails to retrieve the bridging chunk containing concept B because B is not semantically similar enough to a query embedding constructed from A and C. Consequently, the LLM is presented with fragmented evidence so that it knows the cause and the effect but misses the crucial explanatory link required to verify the student's reasoning.

3.2 The Challenge of Multi-Hop Reasoning

The core deficiency of flat retrieval in automated short answer grading is the inability to support multi-hop reasoning. Academic grading often requires verifying a chain of logic: verifying that a student understands that $X \rightarrow Y \rightarrow Z$. In a standard vector store, the relationship $X \rightarrow Y$ and $Y \rightarrow Z$ may reside in separate, non-adjacent chunks. To quantify the necessity of this reasoning, we conducted a preliminary analysis of the NGSS grading dataset. Our examination revealed that approximately 40% of the questions in the Science and Engineering Practices (SEP) dimension require the synthesis of information across at least two distinct conceptual nodes that are not co-located in the source text. For example, in Task 2 (Chemical Reactions in Soap Making), standard RAG frequently failed to grade responses correctly regarding the distinction between *descriptive* and *relational* explanations. The rubric required connecting "observation of property changes" to "evidence of new substance formation," and subsequently to "reasoning about chemical reaction mechanisms." Standard retrieval consistently fetched definitions of chemical reactions but missed the intermediate reasoning criteria that distinguish a descriptive listing of changes from a relational explanation connecting cause to effect. Without this structural retrieval, the LLM is forced to rely on its pre-trained internal parameters to infer these missing connections. This reintroduces the very risk of hallucination that RAG was intended to solve. Therefore, the central technical problem this work addresses is the "Context Fragmentation" problem: how to retrieve not just relevant isolated facts, but a connected subgraph of evidence that preserves the relational dependencies essential for validating complex student arguments.

4. METHODOLOGY

The proposed framework addresses the structural limitations of flat retrieval in automated grading by evaluating two distinct GraphRAG paradigms. Unlike standard RAG, which relies solely on vector similarity, these approaches explicitly model educational content as a structured network. This section details the distinct graph construction and retrieval strategies employed: the community-based summarization of Microsoft GraphRAG and the neurosymbolic associative memory of HippoRAG.

4.1 Community-Based Retrieval via Microsoft GraphRAG

To capture high-level thematic structures and hierarchical relationships within the course material, we employ the Microsoft GraphRAG framework. This approach moves beyond simple entity extraction by utilizing a modular graph construction technique. The system first segments the reference text and uses an LLM to identify entities and relationships, constructing a base knowledge graph. Crucially, it then applies the Leiden community detection algorithm to partition the graph into hierarchical clusters of related concepts. For each detected community, the system generates a natural language summary that synthesizes the key information contained within that cluster. During the grading phase, rather than retrieving raw text chunks, the system performs a "local search" that navigates these pre-computed community summaries. This ensures that when a student mentions a specific concept, the LLM receives a context window enriched with the broader thematic implications and neighbor-descriptions of that concept, providing a holistic view of the topic area (DCI) required for grading.

4.2 Neurosymbolic Retrieval with HippoRAG

Complementing the community-based approach, we employ HippoRAG to address the specific multi-hop reasoning requirements of the SEP dimension. HippoRAG utilizes a neurosymbolic approach inspired by the hippocampal indexing theory of human memory. Unlike the community-summary approach of Microsoft GraphRAG, HippoRAG maintains a direct link between graph nodes and the original source text chunks.The construction phase involves extracting po-

Table 1: Performance Comparison on NGSS Grading Tasks (Accuracy). Best results are highlighted in bold.

	NonRAG			RAG			HippoRAG			MS GraphRAG		
Task	**DCI**	**SEP**	**CCC**	**DCI**	**SEP**	**CCC**	**DCI**	**SEP**	**CCC**	**DCI**	**SEP**	**CCC**
Task 1	0.233	0.367	0.267	0.367	0.333	0.233	**0.919**	**0.811**	0.757	0.838	0.730	**0.811**
Task 2	0.478	0.043	0.239	0.435	0.065	0.196	**0.696**	**0.848**	**0.913**	0.652	0.674	0.870
Task 3	0.114	0.523	0.136	0.114	0.545	0.136	0.591	**0.523**	0.409	**0.750**	0.409	**0.727**
Task 4	0.304	0.283	0.500	0.304	0.261	0.478	0.500	**0.522**	**0.630**	**0.522**	0.457	0.609
Task 5	0.355	0.161	0.581	0.387	0.226	0.581	**0.806**	**0.645**	**0.839**	0.645	0.548	0.516
Task 6	0.848	0.478	0.283	0.848	0.500	0.304	0.848	**0.543**	**0.609**	**0.913**	0.391	0.413
Average	0.389	0.309	0.334	0.409	0.322	0.321	**0.727**	**0.649**	**0.693**	0.720	0.535	0.658

tential entities from the corpus and building a knowledge graph where edges represent co-occurrence or semantic dependency. During retrieval, the system employs the Personalized PageRank (PPR) algorithm. When a student submits a response, the key terms in the answer act as "seed nodes" on the graph. The PPR algorithm propagates activation probability from these seeds to their neighbors, effectively traversing the graph to identify relevant concepts via multi-hop paths. This allows the system to retrieve a subgraph of evidence that includes not only the concepts explicitly mentioned by the student but also the logically connected upstream prerequisites and downstream consequences, thereby reconstructing the full reasoning chain $A \rightarrow B \rightarrow C$ necessary for accurate assessment.

4.3 Graph-Grounded Grading

The final phase involves synthesizing the retrieved context (whether from community reports or associative subgraphs) into a coherent prompt for the grading LLM. The model is instructed to verify the student's response against this structured narrative. By explicitly grounding the evaluation in the retrieved relationships (edges) rather than just isolated facts (nodes), the system converts the grading task from open-ended generation into a constrained graph verification problem.

5. EXPERIMENTS

To validate the efficacy of structural retrieval in automated grading, we conducted a comprehensive evaluation using the standard benchmark dataset reported in Chu et al. [1]. We compared our GraphRAG-driven approaches against the state-of-the-art flat RAG baseline across multiple scientific grading tasks.

5.1 Experimental Setup

Dataset. We utilized the short answer grading dataset established in [1], which is aligned with the Next Generation Science Standards (NGSS). The dataset consists of student responses across six distinct assessment tasks (task 1 through task 6). Each task is graded along three multidimensional learning distinct dimensions. DCI (disciplinary core ideas) is the evaluation of specific scientific content knowledge. SEP (science and engineering practices) is the assessment of the student's ability to engage in scientific reasoning and argumentation. CCC (crosscutting concepts) is the evaluation of connections between different scientific domains. This multidimensional scoring makes the dataset particularly challenging, as it requires the grading system to verify not just factual recall (DCI) but also logical process (SEP) and synthesis (CCC).

Baselines and Models. We benchmarked our proposed methods against the standard RAG implementation described by Chu et al. [1], which utilizes flat vector retrieval to fetch scoring rubrics. We implement HippoRAG as our primary neurosymbolic retrieval approach that utilizes personalized PageRank to capture multi-hop dependencies. We also include Microsoft GraphRAG (local), which is an entity-centric approach that retrieves context based on the ego-graphs of key terms found in the student response. We also performed an ablation study labeled *-Background*, examining the performance when the retrieval is constrained or focused specifically on the background knowledge graph components.

5.2 Main Results

Table 1 presents the comparative performance of the baseline RAG system versus our GraphRAG implementations (including Microsoft GraphRAG and HippoRAG). The results shown in Table 1 demonstrate that incorporating structural graph information significantly enhances grading accuracy across all dimensions.

Analysis of Structural Retrieval. Our HippoRAG implementation achieved the highest average performance across all three metrics, with a substantial improvement in the SEP dimension (0.649 vs. 0.322 for baseline RAG). This confirms our hypothesis that flat retrieval struggles with the reasoning chains required for SEP. By traversing the graph, HippoRAG successfully retrieves the intermediate logical steps that students implicitly rely on, allowing the LLM to verify their reasoning process rather than just keyword matching. Microsoft GraphRAG (local) also outperformed the baseline in DCI (0.720 vs. 0.409) and CCC (0.658 vs. 0.321), showing that entity-centric graph context is highly effective for factual verification, though it is slightly less effective than HippoRAG for capturing the complex procedural dependencies in SEP.

5.3 Ablation Study: Impact of Background Knowledge

We further investigated the impact of different retrieval configurations by isolating the background knowledge components. Table 2 details the performance of the *-Background* variants for both graph methods.

Table 2: Ablation Study on Background Knowledge Configuration

	HippoRAG - Background			MS GraphRAG - Background		
Task	**DCI**	**SEP**	**CCC**	**DCI**	**SEP**	**CCC**
Task 1	0.838	0.649	0.811	0.541	0.757	0.730
Task 2	0.652	0.935	0.978	0.587	0.761	0.891
Task 3	0.818	0.386	0.773	0.795	0.455	0.750
Task 4	0.391	0.413	0.609	0.478	0.478	0.652
Task 5	0.548	0.548	0.452	0.613	0.581	0.452
Task 6	0.891	0.370	0.326	0.870	0.435	0.348
Average	0.690	0.550	0.658	0.647	0.578	0.637

The ablation results highlight a distinct trade-off. While the full HippoRAG system (average SEP 0.649) outperforms the background-only variant (average SEP 0.550), the MS GraphRAG (local) - Background configuration achieved a higher SEP score (0.578) than its standard counterpart (0.535). This suggests that for certain entity-centric retrieval methods, focusing the context explicitly on background knowledge can enhance the model's ability to evaluate scientific practices, potentially by reducing noise from less relevant rubric details. However, for factual consistency (DCI), the full graph integration consistently yields the best results.

6. DISCUSSION

The experimental results provide strong evidence that structural retrieval methods substantially outperform flat vector-based approaches for automated short answer grading. In this section, we analyze representative case studies that illustrate the specific mechanisms by which GraphRAG improves grading accuracy, particularly in scenarios where baseline methods fail systematically.

6.1 Case Study 1: No-RAG Failure in SEP Assessment

The most striking performance gap between methods occurs in the Science and Engineering Practices (SEP) dimension of Task 2 (Chemical Reactions in Soap Making). The No-RAG baseline achieved only 4.3% accuracy on this dimension, essentially random performance, while HippoRAG achieved 84.8% accuracy, representing a 19× improvement.

The Assessment Challenge. The SEP dimension requires the grader to distinguish between *descriptive* explanations (merely listing observations) and *relational* explanations (connecting observations to underlying scientific principles). Consider the following student response:

> *"Your Claim: Yes a chemical reaction did occur. Evidence: Salubility in water. Before: #1=Yes #2=No. After: #1=No #2=yes. Reasoning: The salubility changed in water."*

This response should receive a score of 0 (descriptive) because while the student correctly identifies a property change, they fail to *explain why* this change indicates a chemical reaction occurred. The reasoning merely restates the observation without connecting it to the formation of new substances with different properties.

Why No-RAG Fails. Without access to grading rubrics, the LLM must rely on its pre-trained knowledge to evaluate whether an explanation is "relational." The model's general understanding of chemical reactions leads it to accept any mention of property changes as evidence of scientific reasoning. For instance, when evaluating:

> *"Evidence: All the numbers on density, mass, and melting point are all different which means it created something new with new properties. Reasoning: If it makes something new with new properties, that means it's a chemical reaction."*

The No-RAG system incorrectly classifies this as relational (score 1) because it *sounds* like scientific reasoning. However, according to the rubric, this is still descriptive: the student is applying a memorized rule ("new properties = chemical reaction") rather than constructing a relational explanation that connects specific property changes to the molecular rearrangement process.

Systematic Misalignment. Our analysis revealed that the No-RAG baseline consistently exhibited a pattern of *over-acceptance*: the model's prior knowledge about what constitutes "good" scientific explanation conflicts with the specific rubric criteria. The rubric requires students to go beyond identifying changes to explaining the *mechanism* connecting those changes: a nuance that generic pre-training cannot capture. This explains why SEP accuracy (4.3%) was dramatically lower than DCI accuracy (47.8%) for the same task: factual content (DCI) aligns better with the model's training distribution, while procedural reasoning (SEP) requires task-specific criteria.

6.2 Case Study 2: GraphRAG Success through Structural Context

In contrast to the No-RAG failure, HippoRAG achieved 84.8% accuracy on Task 2 SEP by retrieving structured grading criteria that explicitly define the distinction between descriptive and relational explanations.

Retrieved Context Analysis. When processing the same student responses, HippoRAG retrieved interconnected passages that form a coherent grading framework:

1. **Assessment Criteria Node**: "A Level 0 response (coded as 0) shows incomplete understanding... The student may not correctly identify the reaction as chemical, confuse physical and chemical changes, or not recognize that new substances are formed with different properties."
2. **Level 0 Example Node**: "Yes a chemical reaction did occur. Salubility in water Before: #1=Yes #2=No After: #1=No #2=yes. The salubility changed in water." This response notes a property change but doesn't explain why this indicates a chemical reaction or mention new substances.
3. **Key Concepts Bridge Node**: "When evaluating student understanding of chemical reactions, look for: (1)

Recognition that new substances are formed, (2) Identification of evidence for chemical change, (3) Understanding that the reaction cannot be easily reversed, (4) Recognition that atoms are rearranged to form new molecules."

Multi-Hop Retrieval in Action. The critical advantage of HippoRAG lies in its ability to traverse the knowledge graph to retrieve the "Key Concepts Bridge Node" even when the student's response does not explicitly contain the terminology in that node. The Personalized PageRank algorithm propagates activation from the seed concepts ("chemical reaction," "property change") through the graph edges to reach the bridging criteria. This multi-hop traversal is precisely what flat retrieval cannot achieve: standard vector similarity would not retrieve the "Key Concepts" passage because it shares limited lexical overlap with the student's response.

Grounding the Decision. With this structured context, the LLM can perform a constrained verification rather than open-ended generation. The model observes that the student's response ("The salubility changed in water") matches the Level 0 example almost verbatim, and that the response lacks the key indicators defined in the bridge node (no mention of new substances, molecular rearrangement, or irreversibility). The grading decision becomes a graph verification task: does the student's reasoning path through the conceptual space match a valid path defined in the rubric graph?

6.3 Case Study 3: Distinguishing Relational from Descriptive Explanations

To further illustrate the nuanced distinction that GraphRAG enables, consider two student responses that differ subtly but critically:

Response A (Correctly graded as 0 - Descriptive):

> *"Claim: A chemical reaction did happen.*
> *Evidence: The lye reacted with the oil.*
> *Reasoning: Because the lye and coconut oil heated together to make a germ-killing solid."*

Response B (Correctly graded as 1 - Relational):

> *"Claim: When Mixing Coconut Oil With lye and heating them up a chemical reaction does occur.*
> *Evidence: The Melting Point of Coconut Oil is 27° C, The melting Point of lye is 318° C. The Melting point of the soap he created was 48° C. The Melting Point of Glycerol is 17° C.*
> *Reasoning: The melting point of Coconut Oil and lye before they were combined was very different than after they were mixed together with heat."*

Both responses identify property changes and conclude that a chemical reaction occurred. However, Response B provides *specific quantitative evidence* and *explicitly connects* the before/after property differences to support the claim. Response A merely describes what happened ("heated together to make a solid") without explaining *why* this transformation constitutes evidence of a chemical reaction.

GraphRAG's Discrimination Mechanism. HippoRAG's retrieved context includes the Level 1 criteria: "The student provides specific evidence about formation of new substances with different properties." The graph traversal also retrieves example Level 1 responses that demonstrate the expected level of specificity. By providing this contrastive context, both what constitutes Level 0 and Level 1, the LLM can accurately position Response A as descriptive (lacks specific property values) and Response B as relational (explicitly compares specific melting points before and after). The No-RAG and standard RAG baselines lack this contrastive grounding. Without explicit examples of both categories, these systems tend to accept any response that mentions property changes as "relational," leading to systematic overscoring.

6.4 Implications for Educational Assessment

These case studies reveal several important implications for the design of automated grading systems:

Task-Specific Criteria are Essential. The dramatic failure of No-RAG (4.3% SEP accuracy) demonstrates that LLMs cannot reliably apply task-specific grading standards from general knowledge alone. Even sophisticated models with extensive pre-training require explicit rubric context to distinguish nuanced categories like descriptive vs. relational explanations.

Structural Retrieval Enables Multi-Hop Reasoning. The success of HippoRAG on SEP tasks (84.8% vs. 4.3% for No-RAG) is directly attributable to its ability to retrieve conceptually related but lexically distinct passages. The "bridge nodes" that define grading criteria are often not semantically similar to student responses, making them invisible to flat vector retrieval but accessible through graph traversal.

Contrastive Examples Help Discrimination. GraphRAG's retrieval of both positive and negative examples (Level 0 and Level 1 responses) provides the LLM with decision boundaries rather than isolated definitions. This contrastive grounding is particularly important for grading tasks that require fine-grained categorical distinctions.

7. CONCLUSION

This work demonstrates that graph retrieval-augmented generation (GraphRAG) significantly outperforms standard flat vector retrieval for automated short answer grading. Our evaluation of Microsoft GraphRAG and HippoRAG on an NGSS-aligned dataset reveals substantial improvements across all grading dimensions, with HippoRAG achieving the most significant improvement in SEP accuracy. This dramatic gap occurs because SEP assessment requires distinguishing nuanced categorical distinctions that demand explicit rubric criteria, which is the context that graph-based multi-hop retrieval can provide but flat vector search cannot. Our findings suggest that for higher-order assessment tasks evaluating scientific reasoning, structural retrieval is essential for achieving alignment with human expert judgment. Future work will explore domain-specific educational knowledge graphs and personalized retrieval based on anticipated student misconceptions.

8. ACKNOWLEDGMENTS

This work was supported in part by the U.S. National Science Foundation under Grant Nos. 1813760, 2405483, 2234015, CNS-2321416, IIS-2212032, IIS-2212144, IIS-2504089, DUE-2234015, CNS-2246050, DRL-2405483, and IOS-2035472; the U.S. Department of Education under Grant No. S411B240024; the Gates Foundation; the Michigan Department of Agriculture and Rural Development; the U.S. Department of Commerce; Amazon Faculty Award; Meta; NVIDIA; Microsoft; and Snap Inc.

9. ADDITIONAL AUTHORS

Additional authors: Joseph Krajcik (Michigan State University, email: `krajcik@msu.edu`), Namsoo Shin (Michigan State University, email: `namsoo@msu.edu`), and Hui Liu (Michigan State University, email: `liuhui7@msu.edu`).

10. REFERENCES

[1] Y. Chu, P. He, H. Li, H. Han, K. Yang, Y. Xue, T. Li, J. Krajcik, and J. Tang. Enhancing LLM-based short answer grading with retrieval-augmented generation, 2025.

[2] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, and J. Larson. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

[3] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, and Q. Li. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6491–6501, 2024.

[4] P. W. Foltz and A. D. Wells. Automatically deriving readers' knowledge structures from texts. *Behavior Research Methods, Instruments, & Computers*, 31(2):208–214, 1999.

[5] H. Han, K. Guo, H. Shomer, Y. Wang, Y. Chu, H. Li, L. Ma, and J. Tang. Reasoning by exploration: A unified approach to retrieval and generation over graphs. *The Web Conference (WWW)*, 2025.

[6] H. Han, Y. Wang, H. Shomer, K. Guo, J. Ding, Y. Lei, M. Halappanavar, R. A. Rossi, S. Mukherjee, X. Tang, et al. Retrieval-augmented generation with graphs (graphrag). *arXiv preprint arXiv:2501.00309*, 2024.

[7] M. Mohler, R. Bunescu, and R. Mihalcea. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 752–762, 2011.

How Do Analytic Design Choices Shape MLLM-Based Detection of Non-Verbal Behaviors in 360-Degree Classroom Video

Yiqiu Zhou
University of Illinois Urbana-Champaign
United States
yiqiuz3@illinois.edu

Chris Palaguachi
University of Illinois Urbana-Champaign
United States
cwp5@illinois.edu

Jina Kang
University of Illinois Urbana-Champaign
United States
jinakang@illinois.edu

Luc Paquette
University of Illinois Urbana-Champaign
United States
luc.paquette@gmail.com

Noah Schroeder
University of Florida
United States
schroedern@ufl.edu

Zhenhailong Wang
University of Illinois Urbana-Champaign
United States
wangz3@illinois.edu

Yuji Zhang
University of Illinois Urbana-Champaign
United States
yujiz@illinois.edu

ABSTRACT

360-degree classroom video captures entire learning spaces using omnidirectional recording, preserving group-level context and reducing blind spots associated with fixed-view cameras. This affords advantages for analyzing non-verbal behaviors central to understanding collaborative interaction. However, manual annotation of such video remains time-consuming and difficult to scale. Multimodal large language models (MLLMs) offer a promising alternative: unlike traditional computer vision models that often struggle with the curvilinear distortion and viewpoint variation, MLLMs can leverage global context and semantic cues to interpret complex scenes. Yet applying MLLMs to 360-degree video requires decisions about how panoramic scenes are represented and how temporal information is sampled, factors known to influence model performance but rarely examined in educational contexts. This exploratory case study investigates how video extracted views (front-facing vs. top-down view) and temporal sampling density (1–4 frames per second) influence MLLM-based detection of non-verbal behaviors. We applied Molmo2-8B to 360-degree classroom recordings of small groups collaborative learning and evaluated detection performance across gaze, gesture, posture and tool usage behaviors. Results reveal that object-related behaviors (e.g., gaze at laptop, tool usage) were detected more reliably than social behaviors requiring inference about interpersonal coordination (e.g., gaze at other students). Projected view shaped detectability: top-down views supported more reliable detection of gaze- and tool-related behaviors by preserving spatial context. However, higher sampling rates did not consistently improve detection, suggesting that more frames are not always better. Together, these findings can inform 360-degree video preprocessing decisions when applying MLLMs to video analysis.

Keywords

Multimodal large language models, 360-degree video, Collaborative learning, Non-verbal behavior detection

1. INTRODUCTION

Understanding collaborative learning requires attention not only to what learners say or produce, but also to how they coordinate through non-verbal behavior [14]. Gaze, gesture, posture, and shared tool use play important roles in establishing joint attention, signaling understanding, and regulating turn-taking during group work [3, 20, 16]. Research in learning sciences has long recognized these behaviors as indicators of productive collaboration—moments when learners build shared understanding, negotiate meaning, or struggle to coordinate [40, 43]. Capturing these dynamics is essential for understanding how collaboration unfolds and for designing interventions that support effective group learning [15, 50]. Yet, non-verbal behaviors remain difficult to capture through interaction logs or textual data alone, making classroom video a central data source for examining collaborative learning processes [13, 25].

To capture these non-verbal dynamics in situ, researchers have relied on classroom video recordings. Traditional front-facing cameras require researchers to make early decisions about camera placement and viewing angle, which can re-

Yiqiu Zhou, Chris Palaguachi, Jina Kang, Luc Paquette, Noah Schroeder, Zhenhailong Wang, and Yuji Zhang. How Do Analytic Design Choices Shape MLLM-Based Detection of Non-Verbal Behaviors in 360-Degree Classroom Video. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 469–478. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039881

sult in occlusion or loss of information when learners move outside the camera's limited field of view [10]. These constraints are especially problematic in collaborative learning settings. Recent advances in video capture technology have sought to address these limitations. In particular, wide angle and 360-degree cameras enable capture of entire learning spaces, reducing blind spots and preserving group-level context across multiple learners [2, 39, 47]. Although 360-degree recording addresses some of these limitations, it introduces new challenges. Researchers need to reconstruct and reprocess video from distorted equirectangular projection, and manage the increased complexity of processing panoramic footage. These technical demands compound the already substantial cost of manual video annotation, which has long constrained the scale of research on collaborative learning, often limiting analysis to small samples or brief episodes [58]. Prior work has explored computer vision techniques to automate classroom video analysis, including methods for detecting gaze direction, gesture, and body posture [21, 62, 57]. These approaches have shown promise but typically rely on task-specific pipelines and can be sensitive to viewpoint variation and visual distortion in 360-degree video [6, 19].

Multimodal large language models (MLLMs), a class of vision-language models capable of processing video input, have recently emerged as a promising alternative [23, 59]. By integrating visual perception with natural language generation, MLLMs offer a unified interface for describing a wide range of behaviors, potentially lowering the technical barrier to video-based research [51]. However, existing research provides limited guidance for educational applications. Empirical evaluations of MLLMs primarily rely on standardized benchmarks developed outside educational domains [53, 56], offering limited insight into how MLLMs perform on the context-dependent behaviors that characterize learner interactions. Meanwhile, work on MLLMs in education has remained largely conceptual rather than empirically examining model performance in authentic learning settings [5]. There is also limited understanding of how analytic conditions shape MLLM performance, as model outputs are impacted not only by architecture but also by representation choices and dataset characteristics [11]. These factors are particularly consequential for 360-degree video, where researchers must make explicit decisions about how panoramic footage is projected, framed, and sampled before it can be processed by a model. Yet how such preprocessing decisions affect MLLM performance remains an open question.

This study addresses this question by empirically examining two analytic design choices: video extracted views (how the panoramic scene is projected and framed) and temporal sampling (how many frames are extracted per clip). Focusing on concrete, observable non-verbal behaviors relevant to collaborative learning, we investigate how these choices affect behavior detection performance in 360-degree classroom video. Rather than proposing a fully automated solution, this exploratory case study adopts a diagnostic perspective aimed at identifying the conditions under which MLLM-based analysis of 360-degree video is effective or limited. The following research questions guide this study:

RQ1: How does temporal sampling density affect MLLM-based detection of non-verbal behaviors in 360-degree classroom video?

RQ2: How does video extracted view affect MLLM-based detection of non-verbal behaviors?

2. LITERATURE REVIEW

2.1 360-Degree Video in Educational Research

Advances in 360-degree camera technology present new opportunities for using 360 cameras for teaching and learning [41, 17, 35]. 360-degree cameras capture entire learning environments from a single vantage point, recording omnidirectional views that can be navigated from multiple perspectives [17, 35]. Unlike traditional or multi-camera setups that require coordination among multiple devices, 360-degree cameras present explorable spherical scenes that allow researchers and educators to revisit and navigate the learning space, supporting a deeper examination of instructional practices, learners' engagement, and other social dynamics that occur in the classroom [39, 2]. 360-degree cameras are increasingly used in education for purposes ranging from exam proctoring [1] to immersive training environments, including surgical simulation [24], public speaking practice [44], and observation of teaching practices [48]. However, this flexibility introduces a distinct challenge during video processing. With traditional front-facing cameras, the viewing angle is determined at the time of recording. In contrast, 360-degree video defers this decision to the processing stage, requiring researchers to actively choose how to project and frame the spherical scene for analysis. These decisions are often guided by analytic priorities. Front-facing projections maximize visibility of facial expressions and gaze direction, whereas top-down views better capture spatial arrangements and interactions among group members at a shared workspace [33, 42]. These early choices about viewing angles influence ground-truth annotation and the inputs provided to automated models, making representation selection consequential for human and computational analysis.

These representational decisions compound a broader set of challenges limiting the use of 360-degree video in learning analytics. Processing such footage demands substantial computational resources [32], and standard computer vision models trained on conventional camera footage struggle with 360-degree recordings. Facial, body, and object recognition models often fail due to equirectangular distortion and occlusion common in omnidirectional footage [38]. These challenges point to the need for a systematic evaluation of both video processing methods and computational approaches for collaborative learning analysis.

2.2 Detecting Non-Verbal Behaviors

Early efforts to automate behavior detection have relied on computer vision models trained for specific tasks such as gaze estimation [62], pose detection [55], and gesture recognition [22]. While these approaches have shown promise in controlled settings, they require assembling separate pipelines for each behavior type. For instance, detecting gaze direction may require sequential application of face detection, landmark localization, head pose estimation, and gaze regression, each trained independently and requiring calibration [62]. This presents a substantial engineering burden. A further challenge is that most models are trained on images

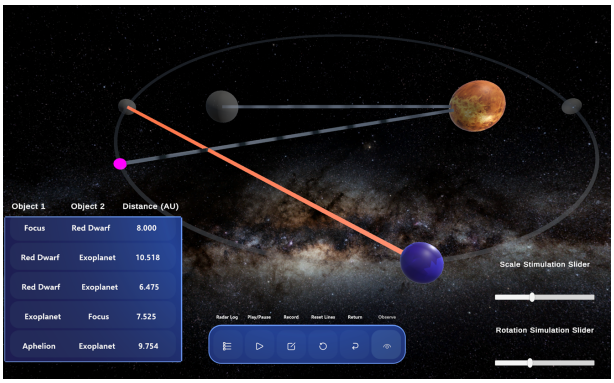


Figure 1: A web-based immersive science learning simulation.

from standard cameras, assuming consistent scale, frontal viewpoints, and minimal occlusion [57]. As a result, they often struggle to generalize to 360-degree video, where equirectangular projection introduces curvilinear distortion that warps object shapes and spatial relationships [45, 6, 54]. These limitations highlight the need for approaches that can handle the unique visual characteristics of 360-degree footage.

MLLMs offer several potential advantages. First, a single model can describe multiple behavior types without separate pipelines, substantially reducing engineering effort[51]. Second, MLLMs can leverage global context and semantic cues to produce partially grounded interpretations; this context-aware reasoning may support more robust detection of behaviors [60, 56]. Finally, MLLMs produce natural-language outputs that are directly interpretable. But MLLM performance is impacted by how video is represented for input and how temporal information is sampled [26]. Without empirical evidence on how these analytic choices influence behavior detection, it remains unclear when MLLMs can reliably support non-verbal behavior analysis. By identifying which behaviors are robust to variation in representation and temporal sampling, and which are highly sensitive to these choices, such work helps researchers calibrate their validation efforts, anticipating when MLLM outputs are likely to be more or less reliable, and where closer inspection or alternative approaches may be required.

3. METHODS

3.1 Context and Participants

Data were collected during group activities in an introductory curriculum and instruction course at a Midwestern University. Students engaged with a simulation designed to teach planetary motion and Kepler's laws (Figure 1), completing a structured group activity involving exploration, data collection, and reflection. A total of 24 out of the 39 students across two classroom sessions consented to participate in the research and worked in groups of two to four. This study focused on one four-student group as a case for examining analytic design choice. The group was selected because the video had high visual quality, all group members remained relatively visible during the focal period, and students worked around a shared workspace with two laptops and worksheets. These features made it possible to compare how the same collaborative activity was represented under different view and sampling conditions. Accordingly, the study should be interpreted as a case-based analysis of preprocessing and detection conditions. Ethical approval was obtained from the institutional review board. All participants provided informed consent for research use of the recordings. Video data were stored and analyzed under approved research procedures.

3.2 Video Data

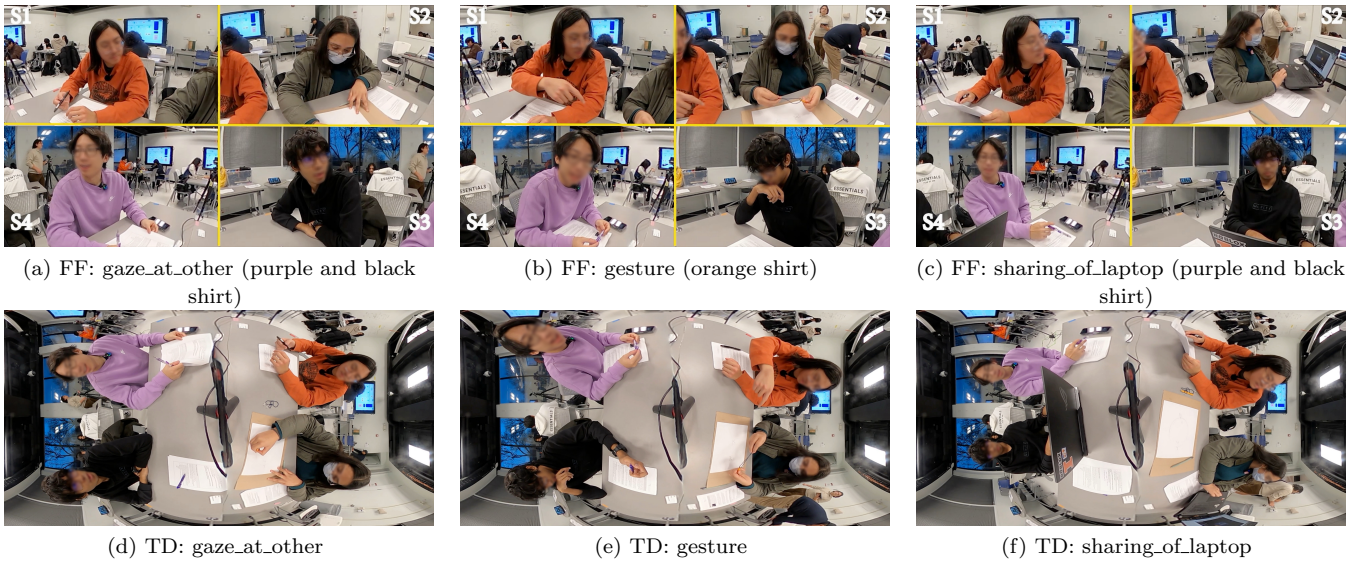
The selected group's video lasted approximately 40 minutes. Based on preliminary review, we focused our analysis on the first 20 minutes, after which cross-group interactions became infrequent. Raw `.360` files were converted to `.mp4` (ProRes) using GoPro Player with H.264 encoding and equirectangular projection. Adobe Premiere Pro v24.1 with a GoPro 360 plugin was then used to extract two viewing conditions (Figure 2). The front-face (FF) view was designed to isolate and maximize the exposure of individual faces to the camera lens, while the top-down (TD) view was designed to isolate and maximize the exposure of the group's physical interactivity around the table. These two views should not be understood as neutral representations containing identical visual information. Rather, they foreground different aspects of the same 360-degree recording. The FF view provides more direct access to faces and upper-body orientation, whereas the TD view more clearly exposes the table layout, shared artifacts, and relative positioning among students. The comparison therefore examines how view-specific information availability shapes MLLM-based detection. Final videos were rendered at 24 fps and 1080p resolution in `.mp4` format.

3.3 Behavior Annotation

Behavior selection was guided by prior work in collaborative learning and non-verbal interaction analysis [14]. The annotation protocol focused on observable, externally visible behaviors, excluding inferences about internal cognitive or affective states. The video was segmented into non-overlapping 10-second clips before annotation. For each clip, trained annotators assigned binary labels indicating whether each behavior in Table 1 was visibly present at any point during the clip. Two trained annotators independently coded the video corpus using the behavior taxonomy summarized in Table 1. Inter-rater agreement was calculated separately for each behavior category using Cohen's kappa over the 10-second clip-level binary labels. Disagreements were resolved through discussion before producing the final reference labels used for model evaluation. Inter-rater reliability exceeded Cohen's $\kappa > 0.75$ for all behavior categories.

3.4 MLLM-Based Behavior Detection

Model and Configuration. We employed Molmo2-8B at full precision (BF16) without quantization. This is an open-weight vision-language model designed for multimodal understanding [12]. Molmo2 integrates a Vision Transformer encoder with a decoder-only language model, using multi-headed attention pooling and an overlapping multi-crop strategy to capture fine-grained visual details with surrounding context. Rather than treating video as an unordered collection of frames, the model associates visual tokens with temporal context, supporting description of actions and events unfolding over short video segments. Molmo2 emphasizes



(a) FF: gaze_at_other (purple and black shirt)

(b) FF: gesture (orange shirt)

(c) FF: sharing_of_laptop (purple and black shirt)

(d) TD: gaze_at_other

(e) TD: gesture

(f) TD: sharing_of_laptop

Figure 2: Example video frames showing non-verbal behaviors in front-face (FF) and top-down (TD) views. Faces and identifying features are obscured for privacy.

Table 1: Behavior taxonomy and frequencies (N=121).

Behavior	Sub-behavior	Definition	Frequency
Gaze	Gaze_at_worksheet	Student looks at the worksheet.	110
	Gaze_at_laptop	Student looks at the laptop.	84
	Gaze_at_other_students	Student looks at another student's face or body.	20
	Gaze_away	Student looks away from their own group or the task.	13
Gesture	Gesture	Student shows gesture like pointing or other hand movement without interacting with objects.	26
Posture	Lean_forward	Student's upper body shifts towards the table.	104
	Headnod	Student moves their head up and down in a nodding motion.	21
	Lean_adjacent	Student shifts to be closer to the adjacent student.	64
Tool Usage	Usage_of_worksheet	Student is actively using the worksheet.	99
	Usage_of_laptop	Student is actively using the laptop.	74
	Sharing_of_laptop	Students share a laptop (i.e., a laptop is present in front of two students).	84

visual grounding through its ability to output 2D points corresponding to pixels supporting its answers. Molmo2-8B was selected for three reasons. First, it is open-weight with a fixed model checkpoint, ensuring exact reproducibility. Second, it achieves benchmark performance comparable to proprietary models such as GPT-4o [12]. Third, its 8B-parameter size allows inference on widely available hardware.The pre-trained model was used without fine-tuning as this work aims to assess baseline model behavior. Temperature was set to zero to ensure deterministic outputs. Inference was conducted on a single NVIDIA H200 GPU.

Video Segmentation. We adopted video-level rather than frame-level behavior detection to support identification of temporal behaviors such as posture changes. To implement this, videos were segmented into fixed-length 10-second clips, a duration that balances temporal context against computational constraints and aligns with the broader multimodal learning analytics project in which this study is situated. Fixed-length segmentation reflects realistic deployment scenarios and avoids reliance on event-based heuristics.

Prompt Design. Each prompt consisted of a system message and a task-specific text prompt. The prompt design draws on prior empirical work showing that explicit role assignment and task framing improve MLLM output quality for educational video analysis [49, 63]. The prompt instructed the model to: *"Act as an educational researcher specializing in qualitative and ethnographic video analysis. Your task is to observe the video clip and determine whether the following [BEHAVIOR-CATEGORY]-related behaviors are visually observable. Base your judgment only on visible, externally observable cues, and do not infer attention, intention, engagement, or internal states. If a behavior is ambiguous, partially occluded, or too subtle to identify with confidence, mark it as not detected (0)."* This was followed by behavior definitions for each category (see Table 1). Prompts required machine-parsable output and were held constant across all

experimental conditions.

Experimental Design. We employed a 2 × 4 factorial design, varying video representation (front-facing vs. top-down) and temporal sampling rate (1–4 FPS) while holding model configuration and prompts constant. MLLM outputs were then evaluated against human annotations. Due to space constraints, the main paper reports accuracy and F1 score while full results including precision and recall are provided in the supplementary materials at `https://anonymous.4open.science/r/EDM2026-4CC0/`. Because the dataset is imbalanced and several behavior categories were relatively infrequent (e.g., gaze away and headnod; Table 1), accuracy and F1 scores should be interpreted with caution. In particular, high accuracy may partly reflect base rates for low-frequency behaviors, while low or unstable F1 scores may reflect data sparsity as well as model difficulty. For this reason, full per-behavior precision and recall results are provided in the supplementary materials.

4. RESULTS

4.1 Effects of Temporal Sampling (RQ1)

To examine how temporal sampling density influenced MLLM-based detection within this specific setup, we compared model performance across four frame rates (1–4 FPS) under both views (see Table 2 and 3). Within the present 10-second clip setup and 1- 4 FPS range, increasing sampling density did **not** yield consistent improvements in detection performance.

Sustained behaviors (i.e., actions that persist over multiple seconds) demonstrated high and stable detection across all FPS conditions. `gaze_at_worksheet` maintained strong performance in both views (FF: F1 0.95–0.96; TD: F1 0.93–0.96), as did `lean_forward` in the TD view (F1 0.92–0.93). Tool usage behaviors all maintained high performance regardless of FPS: `usage_of_worksheet` (TD: F1 0.88–0.89), `usage_of_laptop` (TD: F1 0.89–0.90), and `sharing_of_laptop` (TD: F1 0.89–0.91). These results indicate that for sustained behaviors, lower sampling rates (1–2 FPS) are sufficient. In contrast, several behaviors proved persistently difficult to detect regardless of sampling density. `lean_adjacent` showed consistently low F1 scores in both views (FF: 0.00–0.17; TD: 0.00–0.06), with accuracy close to the base rate (53%). `headnod` exhibited similarly poor detection (FF: F1 0.00–0.09; TD: F1 0.00–0.36). Social gaze behaviors (i.e., `gaze_at_other` and `gaze_away`) also remained challenging across all sampling rates. For these behaviors, FPS variation did not meaningfully improve performance in this setup, suggesting that sampling density alone may not address detection challenges that arise from relational inference, occlusion, or low behavioral frequency.

Only a small number of behaviors showed sensitivity to sampling density. `gesture` in the FF view improved modestly (F1 0.12 at 1 FPS to 0.31 at 4 FPS), suggesting that transient hand movements may benefit from denser temporal coverage. However, this pattern did not hold in the TD view, where `gesture` peaked at 2 FPS (F1 0.39) and declined at higher rates (F1 0.21–0.23). This divergence might point to view-dependent effects on temporal sampling. Other behaviors similarly showed degraded performance at higher rates: `lean_forward` in the FF view declined from accuracy 0.88 and F1 0.93–0.94 at 1–2 FPS to accuracy 0.69–0.70 and F1 0.80–0.81 at 3–4 FPS. `gaze_at_laptop` (FF) decreased from F1 0.30 to 0.19 across the FPS range, and `sharing_of_laptop` (FF) peaked at 2 FPS (F1 0.86) before declining (F1: 0.71–0.73). Together, these findings suggest that uniform increases in temporal sampling do not reliably improve detection performance within this case and sampling range, and that the effects of sampling rate are both behavior- and representation-dependent.

Table 2: Accuracy across FPS for each behavior by view.

Feature	View	FPS1	FPS2	FPS3	FPS4
gaze_at_worksheet	FF	0.92	0.92	0.90	0.90
	TD	0.88	0.92	0.89	0.91
gaze_at_laptop	FF	0.43	0.42	0.37	0.38
	TD	0.91	0.91	0.81	0.90
gaze_at_other	FF	0.84	0.84	0.84	0.84
	TD	0.82	0.41	0.78	0.76
gaze_away	FF	0.85	0.64	0.89	0.89
	TD	0.88	0.73	0.89	0.87
gesture	FF	0.74	0.79	0.79	0.77
	TD	0.55	0.71	0.74	0.66
lean_forward	FF	0.88	0.88	0.70	0.69
	TD	0.86	0.86	0.88	0.88
headnod	FF	0.82	0.83	0.83	0.83
	TD	0.82	0.76	0.82	0.83
lean_adjacent	FF	0.49	0.51	0.47	0.47
	TD	0.48	0.48	0.47	0.47
usage_of_worksheet	FF	0.84	0.84	0.83	0.82
	TD	0.79	0.79	0.79	0.81
usage_of_laptop	FF	0.64	0.69	0.64	0.66
	TD	0.88	0.88	0.86	0.88
sharing_of_laptop	FF	0.72	0.82	0.69	0.70
	TD	0.88	0.86	0.85	0.86

4.2 Effects of Representation View (RQ2)

To examine how video representation influences MLLM-based detection, we compared performance between FF and TD views (see Tables 2 and 3). Performance differed across views, with the TD view generally providing visual information that supported more reliable detection, particularly for behaviors involving shared artifacts and spatial context.

The TD view showed stronger performance for object-related behaviors. `gaze_at_laptop` showed markedly stronger performance in the TD view (F1 0.84–0.93) compared to the FF view (F1 0.17–0.30). Similar patterns emerged for `usage_of_laptop` (TD: F1 0.89–0.90 vs. FF: F1 0.59–0.67) and `sharing_of_laptop` (TD: F1 0.89–0.91 vs. FF: F1 0.71–0.86). These behaviors require information about spatial relationships between learners and artifacts, which was more visible in the global scene layout preserved in top-down projections.

For highly salient sustained behaviors, both views achieved comparable performance. `gaze_at_worksheet` was detected reliably across representations (FF F1 0.95–0.96; TD F1 0.93–0.96), as was `usage_of_worksheet` (FF F1 0.89–0.91; TD F1 0.88–0.89). `lean_forward` performed similarly at

Table 3: F1 score across FPS for each behavior by view.

Feature	View	FPS1	FPS2	FPS3	FPS4
gaze_at_worksheet	FF	0.96	0.96	0.95	0.95
	TD	0.93	0.96	0.94	0.95
gaze_at_laptop	FF	0.30	0.29	0.17	0.19
	TD	0.93	0.93	0.84	0.92
gaze_at_other	FF	0.10	0.23	–	–
	TD	0.39	0.30	0.40	0.17
gaze_away	FF	0.25	0.24	–	–
	TD	0.30	0.33	0.44	0.11
gesture	FF	0.12	0.25	0.29	0.31
	TD	0.31	0.39	0.21	0.23
lean_forward	FF	0.93	0.94	0.81	0.80
	TD	0.92	0.92	0.93	0.93
headnod	FF	–	0.09	–	–
	TD	–	0.36	–	–
lean_adjacent	FF	0.06	0.17	–	–
	TD	0.03	0.06	–	–
usage_of_worksheet	FF	0.99	0.91	0.90	0.89
	TD	0.88	0.88	0.88	0.89
usage_of_laptop	FF	0.61	0.67	0.59	0.62
	TD	0.90	0.90	0.89	0.90
sharing_of_laptop	FF	0.75	0.86	0.71	0.73
	TD	0.91	0.89	0.89	0.89

lower sampling rates (FF F1 0.93–0.94; TD F1 0.92–0.93), although the FF view showed degradation at higher FPS. These behaviors appear sufficiently visually distinctive that representation choice has limited impact on detection. In contrast, socially oriented behaviors proved difficult to detect under both representations. `gaze_at_other` yielded low and unstable F1 scores in both views (TD: 0.17–0.40; FF: 0.10–0.23), as did `gaze_away` (TD: 0.11–0.44; FF: 0.24–0.25). `lean_adjacent` and `headnod` remained near zero across all conditions in both views. These behaviors require relational inference about interpersonal coordination, a challenge that neither representation resolved.

5. DISCUSSION

5.1 Temporal Sampling: When Less May Be More

Results for RQ1 show that higher frame sampling rates did not consistently improve detection performance,within the present 10-second clip setup and 1- 4 FPS range. This finding should not be interpreted as a general claim that fewer frames are always preferable. Rather, it suggests that the value of additional frames depends on the temporal characteristics of the target behavior, the length of the clip, and the model's visual token and context constraints. Within 10-second segments, denser sampling may add redundant frames when behaviors are sustained across multiple seconds. This pattern is consistent with recent findings showing that uniform increases in FPS can plateau or even degrade performance when adjacent frames are highly similar [34, 36]. Many collaborative learning behaviors, such as gazing at a worksheet or maintaining a forward-leaning posture, unfold over several seconds. At higher sampling rates (e.g., 3–4 FPS), the model processes many near-identical frames. This redundancy may dilute the model's effective attention under fixed token budgets and increase sensitivity to minor visual variations [34]. In contrast, transient behaviors may benefit from higher sampling rates to capture momentary behavioral cues. These findings therefore support behavior-aware sampling rather than uniformly increasing FPS.increasing FPS.

5.2 Affordances for Behavior Detection

Results for RQ2 indicate that video representation impacts which non-verbal behaviors can be reliably detected by video-based MLLMs. A key methodological caveat is that FF and TD views do not contain identical visual information. They are different input conditions that organize visibility in different ways. The TD view more clearly exposes the table layout, shared artifacts, and relative body positions, whereas the FF view provides more direct access to faces and upper-body cues but may obscure the broader workspace. Therefore, the observed differences should be interpreted as view-specific differences in information availability, rather than evidence that the model is inherently better suited to one projection type.

TD views supported more reliable detection of several gaze- and tool-related behaviors, particularly those involving shared artifacts or spatial coordination among learners. This advantage aligns with insights from group activity recognition, which emphasizes that many collaborative behaviors depend on the spatial configuration of multiple participants rather than isolated individual cues [7, 52].

Detecting such behaviors requires knowing whether multiple learners are oriented toward the same object, how close they are to one another, and how their bodies are arranged in space. TD views appear to support these inferences by preserving the global layout of the learning environment, including the relative positions of learners, shared artifacts, and workspaces. By reducing occlusion, normalizing scale differences across learners, and exposing shared reference frames, TD representations foreground scene-level context and inter-person relations that are difficult to reconstruct from frontal viewpoints alone. Similar advantages of global or overhead perspectives have been reported in prior work on collaborative activity recognition, where preserving spatial context improves robustness under occlusion and viewpoint variation [7, 28, 52].

FF views, while less effective overall in our evaluation, have demonstrated strengths in prior work using traditional computer vision approaches. They provide higher resolution of facial features, hand positions, and upper-body movements [4]. Prior research on classroom analytics suggests that such perspectives are well suited for detecting individual-level nonverbal cues, including head orientation, eye gaze, and affect detection [29, 38]. In the present study, however, these advantages did not translate to improved MLLM-based detection as FF views yielded weaker and less stable performance than TD views across most behavior categories. One possible explanation is that MLLMs and traditional computer vision pipelines leverage visual information differently: whereas specialized computer vision models can ex-

tract fine-grained facial features from frontal views, MLLMs may rely more heavily on global scene context and semantic cues that are better preserved in TD representations. Additionally, in multi-learner settings, FF views often obscure the broader context needed to infer group-level coordination [31]. Taken together, these findings suggest that no single representation is universally optimal. Representation choice should be guided by analytic goals and methods. Future work should examine whether combining FF and TD information through early or late fusion yields more robust detection [52].

5.3 Object-Related vs. Social Behaviors

One key finding of this study is that object-related behaviors—those involving direct interaction with physical materials such as worksheets and laptops—were detected more reliably than social behaviors requiring inference about interpersonal attention or coordination. For example, higher F1 scores and accuracy were observed for `gaze_at_laptop` and `gaze_at_worksheet` compared to `gaze_at_other`.

Object-related behaviors primarily require perceptual grounding: recognizing the presence of an artifact and identifying that a learner's gaze or action is directed toward it. Such judgments rely largely on visual pattern matching and benefit from model pretraining on large-scale object recognition data [37]. In contrast, social behaviors require relational inference across participants. Rather than identifying a single target, the model must integrate cues about spatial arrangement, body orientation, and temporal coordination [8, 62]. In 360-degree classroom video, this relational inference is further complicated by wide fields of view and spatial distortion.

These demands remain challenging for current MLLMs, which have been shown to exhibit systematic limitations in interpreting subtle relational aspects of human interaction [30]. These findings have practical implications for the design of MLLM-based analytics pipelines. Object-related behaviors may be more amenable to scalable automated analysis with comparably less human oversight, while social behaviors may require specialized models for detecting interpersonal coordination, such as eye contact and shared attention [18, 46, 28].

5.4 Limitations and Future Work

This study has several limitations. First, This study should be interpreted as a case-based analysis rather than a general evaluation across classroom contexts. The analysis was based on one four-student group, one task, one classroom configuration, and the first 20 minutes of a single 360-degree recording. The selected group was not randomly sampled; it was chosen because the video quality was high and the shared workspace made FF/TD comparisons analytically tractable. As a result, group-level factors such as seating arrangement, communication style, division of labor, technology-use patterns, and visibility of shared artifacts may have influenced the observed differences between FF and TD views. Second, evaluation was conducted using a single MLLM (Molmo2-8B), limiting generalizability to other vision–language models. Differences in architecture, pretraining data, and visual encoding strategies may substantially influence behavior detection performance [9]. Future work should test whether the observed patterns hold across other open-source and proprietary models. Third, the study focused on a specific taxonomy of non-verbal behaviors relevant to collaborative learning. While theoretically grounded, this taxonomy does not exhaustively capture all educationally meaningful behaviors. Third, the study examined only zero-shot inference without fine-tuning or prompt optimization. Although this reflects realistic use cases for many researchers, performance likely underestimates what could be achieved with task-specific adaptation. Techniques such as many-shot in-context learning or chain-of-thought prompting may improve detection accuracy [27, 61].

Beyond these limitations, future research could explore adaptive sampling strategies tailored to specific behavior dynamics, multi-view fusion approaches that leverage complementary perspectives, and fine-tuning methods that adapt pretrained MLLMs to educational contexts. Integrating MLLM-based video analysis with additional data streams—such as audio transcripts, interaction logs, or physiological sensors—also offers promising directions for advancing multimodal learning analytics.

References

[1] Y. Atoum, L. Chen, A. X. Liu, S. D. H. Hsu, and X. Liu. Automated online exam proctoring. *IEEE Transactions on Multimedia*, 19(7):1609–1624, July 2017.

[2] N. Balzaretti, A. Ciani, C. Cutting, L. O'Keeffe, and B. White. Unpacking the potential of 360-degree video to support pre-service teacher development. *Research on Education and Media*, 11(1):63–69, 2019.

[3] B. Barron. When smart groups fail. *Journal of the Learning Sciences*, 12(3):307–359, 2003.

[4] A. Behera, P. Matthew, A. Keidel, P. Vangorp, H. Fang, and S. Canning. Associating facial expressions and upper-body gestures with learning tasks for enhancing intelligent tutoring systems. *International Journal of Artificial Intelligence in Education*, 30(2):236–270, 2020.

[5] A. Bewersdorff, C. Hartmann, M. Hornberger, K. Seßler, M. Bannert, E. Kasneci, G. Kasneci, X. Zhai, and C. Nerdel. Taking the next step with generative artificial intelligence: The transformative role of multimodal large language models in science education. *Learning and Individual Differences*, 118:102601, 2025.

[6] M. Cao, C. Mou, F. Yu, X. Wang, Y. Zheng, J. Zhang, and B. Luo. Ntire 2023 challenge on 360° omnidirectional image and video super-resolution: Datasets, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1731–1745, 2023.

[7] W. Choi and S. Savarese. Understanding collective activities of people from videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1242–1257, 2013.

[8] E. Chong, N. Ruiz, Y. Wang, Y. Zhang, A. Rozga, and J. M. Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of

gaze and scene saliency. In *Proceedings of the European Conference on Computer Vision*, pages 383–398, 2018.

[9] F. Cocchi, N. Moratelli, D. Caffagni, S. Sarto, L. Baraldi, M. Cornia, and R. Cucchiara. Llava-more: A comparative study of llms and visual backbones for enhanced visual instruction tuning. *arXiv preprint*, arXiv:2503.15621, 2025.

[10] Z. Dai, A. McReynolds, and J. Whitehill. In search of negative moments: Multi-modal analysis of teacher negativity in classroom observation videos. In M. Feng, T. KÃ¤ser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 278–285, Bengaluru, India, July 2023. International Educational Data Mining Society.

[11] S. Danish, A. Sadeghi-Niaraki, S. U. R. Khan, L. M. Dang, L. Tightiz, and H. Moon. A comprehensive survey of vision-language models: Pretrained models, fine-tuning, prompt engineering, adapters, and benchmark datasets. *Information Fusion*, 126:103623, 2025.

[12] M. Deitke, C. Clark, S. Lee, R. Tripathi, Y. Yang, J. S. Park, M. Salehi, N. Muennighoff, K. Lo, L. Soldaini, J. Lu, T. Anderson, E. Bransom, K. Ehsani, H. Ngo, Y. Chen, A. Patel, M. Yatskar, C. Callison-Burch, A. Head, R. Hendrix, F. Bastani, E. VanderBilt, N. Lambert, Y. Chou, A. Chheda, J. Sparks, S. Skjonsberg, M. Schmitz, A. Sarnat, B. Bischoff, P. Walsh, C. Newell, P. Wolters, T. Gupta, K.-H. Zeng, J. Borchardt, D. Groeneveld, C. Nam, S. Lebrecht, C. Wittlif, C. Schoenick, O. Michel, R. Krishna, L. Weihs, N. A. Smith, H. Hajishirzi, R. Girshick, A. Farhadi, and A. Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 91–104, 2025.

[13] S. J. Derry, R. D. Pea, B. Barron, R. A. Engle, F. Erickson, R. Goldman, R. Hall, T. Koschmann, J. L. Lemke, and M. G. Sherin. Conducting video research in the learning sciences. *Journal of the Learning Sciences*, 19(1):3–53, 2010.

[14] I. Dey, M. M. Ko, and S. Puntambekar. Visualizing collaboration using multiple modalities. In J. Oshima, B. Chen, F. Vogel, and J. Järvelä, editors, *Proceedings of the 18th International Conference on (Computer-Supported) Collaborative Learning - (CSCL) 2025*, pages 664–666, 2025.

[15] P. Dillenbourg. *Collaborative Learning: Cognitive and Computational Approaches*. Pergamon, 1999.

[16] P. Dillenbourg, M. Baker, A. Blaye, and C. O'Malley. The evolution of research on collaborative learning. In E. Spada and P. Reimann, editors, *Learning in Humans and Machine: Towards an Interdisciplinary Learning Science*, pages 189–211. Elsevier, Oxford, 1996.

[17] M. Evens, M. Empsen, and W. Hustinx. A literature review on 360-degree video as an educational tool. *Journal of Computers in Education*, 10(2):325–375, 2023.

[18] L. Fan, W. Chen, P. Wei, W. Wang, and S.-C. Zhu. Inferring shared attention in social scene videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6460–6468, 2018.

[19] S. V. Georgakopoulos, K. Kottari, K. Delibasis, V. P. Plagianakos, and I. Maglogiannis. Pose recognition using convolutional neural networks on omni-directional images. *Neurocomputing*, 280:23–31, 2018.

[20] C. Goodwin. Action and embodiment within situated human interaction. *Journal of Pragmatics*, 32(10):1489–1522, 2000.

[21] M. Holmes, A. Latham, K. Crockett, and J. O'Shea. Near real-time comprehension classification with artificial neural networks. *IEEE Transactions on Learning Technologies*, 11(1):5–12, 2017.

[22] J. Hou, G. Wang, X. Chen, J.-H. Xue, R. Zhu, and H. Yang. Spatial-temporal attention res-tcn for skeleton-based dynamic hand gesture recognition. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.

[23] B. Huang, X. Wang, H. Chen, Z. Song, and W. Zhu. Vtimellm: Empower llm to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14271–14280, 2024.

[24] T. Huber, M. Paschold, C. Hansen, T. Wunderling, H. Lang, and W. Kneist. New dimensions in surgical training: Immersive virtual reality laparoscopic simulation exhilarates surgical staff. *Surgical Endoscopy*, 31(11):4472–4477, 2017.

[25] C. Jewitt, J. Bezemer, and K. O'Halloran. *Introducing Multimodality*. Routledge, 2016.

[26] M. Jia, Z. Qi, S. Zhang, W. Zhang, X. Yu, J. He, and L. Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision–language models. *arXiv preprint*, arXiv:2506.03135, 2025.

[27] Y. Jiang, J. Irvin, J. H. Wang, M. A. Chaudhry, J. H. Chen, and A. Y. Ng. Many-shot in-context learning in multimodal foundation models. *arXiv preprint*, arXiv:2405.09798, 2024.

[28] S. Li, Q. Cao, L. Liu, K. Yang, S. Liu, J. Hou, and S. Yi. Groupformer: Group activity recognition with clustered spatial-temporal transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13668–13677, 2021.

[29] F.-C. Lin, H.-H. Ngo, C.-R. Dow, K.-H. Lam, and H.-L. Le. Student behavior recognition system for the classroom environment based on skeleton pose estimation and person detection. *Sensors*, 21(16):5314, 2021.

[30] Z. Liu, P. Rabbani, V. Duddu, K. Fan, M. Lee, and Y. Huang. The social gaze of llms: A literature review of multimodal approaches to human behavior understanding. *arXiv preprint*, arXiv:2510.23947, 2025.

[31] E. A. Mosabbeb, K. Raahemifar, and M. Fathy. Multi-view human activity recognition in distributed camera sensor networks. *Sensors*, 13(7):8750–8770, 2013.

[32] R. Noël, D. Miranda, C. Cechinel, F. Riquelme, T. T. Primo, and R. Munoz. Visualizing collaboration in teamwork: A multimodal learning analytics platform for non-verbal communication. *Applied Sciences*, 12(15), 2022.

[33] S. Oviatt, J. Lin, and A. Sriramulu. I know what you know: What hand movements reveal about domain expertise. *ACM Trans. Interact. Intell. Syst.*, 11(1):1–26, Mar. 2021.

[34] J. Park, K. Ranasinghe, K. Kahatapitiya, W. Ryu, D. Kim, and M. S. Ryoo. Too many frames, not all useful: Efficient strategies for long-form video qa. *arXiv preprint*, arXiv:2406.09396, 2024.

[35] J. Qian, J. Shang, and L. Qin. A systematic scoping review of 360-degree videos in teacher education. *Journal of Research in Innovative Teaching & Learning*, 18(1):20–38, 2025.

[36] T. Qu, L. Tang, B. Peng, S. Yang, B. Yu, and J. Jia. Does your vision–language model get lost in the long video sampling dilemma? *arXiv preprint*, arXiv:2503.12496, 2025.

[37] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 8748–8763, 2021.

[38] R. J. Rajarathinam, C. Palaguachi, J. Kang, et al. 360-degree cameras vs traditional cameras in multimodal learning analytics: Comparative study of facial recognition and pose estimation. *Journal of Educational Data Mining*, 17(1):157–182, 2025.

[39] J. Reyna. The potential of 360-degree videos for teaching, learning and research. In *INTED2018 Proceedings*, pages 1448–1454. IATED, 2018.

[40] J. Roschelle and S. Teasley. The construction of shared knowledge in collaborative problem solving. In C. O'Malley, editor, *Computer-Supported Collaborative Learning*, pages 69–97. Springer, 1995.

[41] N. L. Schroeder, R. F. Siegle, and S. D. Craig. A meta-analysis on learning from 360° video. *Computers & Education*, 206:104901, 2023.

[42] D. Spikol, E. Ruffaldi, and M. Cukurova. Using multimodal learning analytics to identify aspects of collaboration in project-based learning. In *CSCL'17: The 12th International Conference on Computer Supported Collaborative Learning, Philadelphia, PA (June 18-20, 2017)*, volume 1, pages 263–270. International Society of the Learning Sciences., 2017.

[43] G. Stahl. *Group Cognition*. MIT Press, 2006.

[44] S. Stupar-Rutenfrans, L. E. Ketelaars, and M. S. van Gisbergen. Beat the fear of public speaking: Mobile 360 video virtual reality exposure training in home environment reduces public speaking anxiety. *Cyberpsychology, Behavior, and Social Networking*, 20(10):624–633, 2017.

[45] Y.-C. Su and K. Grauman. Learning spherical convolution for fast features from 360 imagery. In *Advances in Neural Information Processing Systems*, volume 30, pages 529–539, 2017.

[46] Ö. Sümer, P. Gerjets, U. Trautwein, and E. Kasneci. Attention flow: End-to-end joint attention estimation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 3316–3325. IEEE, 2020.

[47] H. Theelen, A. van den Beemt, and P. den Brok. Using 360-degree videos in teacher education. *Journal of Computer Assisted Learning*, 35(5):582–594, 2019.

[48] N. Walshe and P. Driver. Developing reflective trainee teacher practice with 360-degree video. *Teaching and Teacher Education*, 78:97–105, 2019.

[49] R. Whitehead, A. Nguyen, and S. Järvelä. Utilizing multimodal large language models for video analysis of posture in studying collaborative learning: A case study. *Journal of Learning Analytics*, 12(1):186–200, 2025.

[50] A. F. Wise and B. B. Schwarz. Visions of cscl. *International Journal of Computer-Supported Collaborative Learning*, 12(4):423–467, 2017.

[51] J. Wu, W. Gan, Z. Chen, S. Wan, and P. S. Yu. Multimodal large language models: A survey. In *Proceedings of the 2023 IEEE International Conference on Big Data (BigData)*, pages 2247–2256. IEEE, 2023.

[52] L. Wu, Q. Wang, M. Jian, Y. Qiao, and B. Zhao. A comprehensive review of group activity recognition in videos. *International Journal of Automation and Computing*, 18(3):334–350, 2021.

[53] M. Xu, M. Gao, Z. Gan, H.-Y. Chen, Z. Lai, H. Gang, K. Kang, and A. Dehghan. Slowfast-llava: A strong training-free baseline for video large language models. *arXiv*, 2024. arXiv:2407.15841.

[54] Y. Xu, Y. Dong, J. Wu, Z. Sun, Z. Shi, J. Yu, and S. Gao. Gaze prediction in dynamic 360 immersive videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5333–5342. IEEE, 2018.

[55] Y. Xu, J. Zhang, Q. Zhang, and D. Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in neural information processing systems*, 35:38571–38584, 2022.

[56] S. Yin, C. Fu, S. Zhao, K. Li, X. Sun, T. Xu, and E. Chen. A survey on multimodal large language models. *National Science Review*, 11(12):nwae403, 2024.

[57] J. Yu, A. C. P. Grassi, and G. Hirtz. Applications of deep learning for top-view omnidirectional imaging: A survey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6421–6433, 2023.

[58] C. Zahn, A. Ruf, and R. Goldman. Video data collection and video analyses in cscl research. In U. Cress, C. Rosé, A. F. Wise, and J. Oshima, editors, *International handbook of computer-supported collaborative learning*, pages 643–660. Springer, Cham, 2021.

[59] X. Zeng, X. Wang, T. Zhang, C. Yu, S. Zhao, and Y. Chen. Gesturegpt: Toward zero-shot free-form hand gesture understanding with large language model agents. *Proceedings of the ACM on Human-Computer Interaction*, 8(ISS):462–499, 2024.

[60] D. Zhang, X. Zheng, Z. Weng, Y. Lyu, D. P. Paudel, L. Van Gool, and X. Hu. Are multimodal large language models ready for omnidirectional spatial reasoning? *arXiv preprint*, arXiv:2505.11907, 2025.

[61] R. Zhang, B. Zhang, Y. Li, H. Zhang, Z. Sun, Z. Gan, and Y. Yang. Improve vision–language model chain-of-thought reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1662, 2025.

[62] Y. Zhong and S. H. Lee. Gazesymcat: A symmetric cross-attention transformer for robust gaze estimation under extreme head poses and gaze variations. *Journal of Computational Design and Engineering*, 12(3):115–129, 2025.

[63] Y. Zhou, J. Kang, and H. Nguyen. Can we trust large language models for video analysis: An exploration of hallucination in multimodal llms. 2025.

Identifying Items on Which Humans and Chatbots Diverge Using Differential Item Functioning

Licol Zeinfeld* Alona Strugatski* Ziva Bar-Dov
Ron Blonder Shelley Rap Giora Alexandron

Weizmann Institute of Science
*Equal contribution
{licol.zeinfeld, alona.faktor, ziva.bar-dov, ron.blonder, shelley.rap, giora.alexandron}@weizmann.ac.il

ABSTRACT

The rapid adoption of large language models (LLMs) in education raises profound challenges for assessment design. To adapt assessments to the presence of LLM-based tools, it is crucial to characterize the strengths and weaknesses of LLMs in a generalizable, valid and reliable manner. However, current LLM evaluations often rely on descriptive statistics derived from benchmarks, and little research applies theory-grounded measurement methods to characterize LLM capabilities relative to human learners in ways that directly support assessment design. Here, by combining educational data mining and psychometric theory, we introduce a statistically principled approach for identifying items on which humans and LLMs show systematic response differences, pinpointing where assessments may be most vulnerable to AI misuse, and which task dimensions make problems particularly easy or difficult for generative AI. The method is based on Differential Item Functioning (DIF) analysis – traditionally used to detect bias across demographic groups – together with negative control analysis and item-total correlation discrimination analysis. It is evaluated on responses from human learners and six leading chatbots (ChatGPT-4o & 5.2, Gemini 1.5 & 3 Pro, Claude 3.5 & 4.5 Sonnet) to two instruments: a high school chemistry diagnostic test and a university entrance exam. Subject-matter experts then analyzed DIF-flagged items to characterize task dimensions associated with chatbot over- or under-performance. Results show that DIF-informed analytics may provide a theory-grounded framework for understanding where LLM and human capabilities diverge, and highlight their value for improving the design of valid, reliable, and fair assessment in the AI era.

Keywords

Assessment, Generative AI, Differential Item Functioning

Licol Zeinfeld, Alona Strugatski, Ziva Bar-Dov, Ron Blonder, Shelley Rap, and Giora Alexandron. Identifying Items on Which Humans and Chatbots Diverge Using Differential Item Functioning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 479–486. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039815

1. INTRODUCTION

The rapid adoption of generative AI (GenAI[1]) tools in education has created both opportunities and risks. While these systems, particularly chatbots such as ChatGPT, can provide personalized explanations, feedback, and support for learners, their growing use also poses a profound threat to the validity of educational assessments [1, 4]. By making it exceptionally easy for students to generate fluent and convincing answers, GenAI lowers the barriers to academic dishonesty in ways that traditional safeguards struggle to address [1, 4, 10]. This risk is amplified in unproctored or remote environments, where monitoring is limited and misuse is harder to detect [27].

Importantly, the concern is no longer theoretical: recent studies document that students are already using GenAI for assignments and exams, directly undermining the validity and reliability of assessment results and raising urgent challenges for educational practice and policy [18, 23]. These changes point to a key observation: to ensure resistance to GenAI-related misconduct and to maintain validity and reliability in assessment contexts characterized by potentially heavy LLM use, it is necessary to understand where LLMs diverge from human learners and to characterize their capabilities and limitations across assessment tasks.

Current evaluations of LLM performance on assessment tasks are largely shaped by technical benchmarks (e.g., [3, 25, 14]). These benchmarks provide descriptive statistics that locate LLMs on various scales relative to human learners and provide evidence that LLMs are affected differently from humans by task features such as visual elements or sequential reasoning steps [24, 26]. Yet, little research has examined more generalizable ways of measuring LLM capabilities in relation to human learners. Assessment, by definition, is a proxy that seeks to generalize from sparse samples of student performance (e.g., across time or content). For human learners, the design and interpretation of such proxies have been refined through decades of assessment research and implementation. For GenAI agents, however, it remains far less clear what inferences assessments support.

Mature frameworks from educational measurement – such

[1]In this paper, we use GenAI to refer to generative AI in the broad sense; LLMs to refer specifically to language-oriented models; and chatbots to refer to conversational agents powered by LLMs.

as, but not limited to, Item Response Theory (IRT) [5] – may offer principled ways to assess the capabilities of GenAI. Recent studies [21, 22, 13] have shown that psychometric modeling can be extended beyond its traditional applications to capture systematic differences between human learners and chatbots, mainly for identifying GenAI. Within educational data mining, a central paradigm is that item-level analysis can reveal underlying cognitive processes, providing a foundation for more valid assessment and adaptive learning design [2, 11].

Combining these approaches, our work moves the center of attention from *examinees* to *items*, and its goal is identifying items on which chatbots and humans exhibit differential behavior. Identifying such items serves two important purposes. First, to better understand the dimensions that make tasks easy or difficult for LLMs relative to human learners. Second, to apply these insights to the design of assessments that are better adapted – in terms of validity, reliability, and fairness – to scenarios in which students may work (legitimately or not) with GenAI.

Within psychometrics, there is a collection of methods known as Differential Item Functioning (DIF). DIF refers to situations in which an assessment item functions differently for subgroups of learners distinguished by a characteristic unrelated to the construct being measured (typically a demographic characteristic). In DIF terminology, the baseline group is referred to as the *reference* group, while the group of interest is referred to as the *focal* group. DIF methods go beyond simply comparing overall performance between the reference and focal groups by controlling for student ability, *thereby distinguishing true item bias from general group-level performance differences*.

DIF analysis is widely used in assessment design to flag biased or poorly constructed items, as such items compromise the validity and fairness of the assessment [17]. For example, a mathematics item requiring advanced reading comprehension may disadvantage learners with equivalent math ability but weaker language skills (e.g., second language learners). Methods to detect DIF include non-parametric methods, such as Mantel–Haenszel [16], or parametric ones, such as those based on IRT or logistic regression [15].

The rationale of the current work is to examine whether DIF methods can be useful to identify items on which human learners and chatbots differ. The observation is that chatbots can be referred to as the 'focal' group, while humans are the 'reference' (baseline) group. This is formulated through the following research questions (RQs):

RQ1: *Can DIF techniques identify items that function differently for human learners and chatbots?*

RQ2: *What key characteristics do subject-matter experts identify in items on which chatbots exhibit differential performance compared to human learners?*

To study these questions, we developed, in a stepwise manner, a method that combines DIF analysis with additional item-level criteria and identifies items on which chatbots and human learners show differential behavior. This method was progressively refined and evaluated on human learners and GenAI responses to two assessment instruments taken from two very different contexts: a high school chemistry test administered as a formative assessment, and the quantitative section of a high-stakes psychometric entrance exam for higher education. The GenAI responses were generated using six chatbots (see the Methodology section for details). Following that, subject-matter experts conducted a qualitative analysis of the chemistry DIF items to identify key dimensions that may lead to the differential behavior.

The key contribution of this paper is proposing a theory-inspired and statistically robust method to identify items that exhibit differential behavior for chatbots relative to students, and demonstrating its effectiveness in providing item analytics to those who wish to incorporate GenAI-related considerations into assessment design.

2. PSYCHOMETRIC PRELIMINARIES

2.1 Item–Total Correlation and Rest Score

Item–total correlation (ITC) is defined as the correlation between the score on a single item and the *rest score* for that item, which is the aggregated performance across all the *other* items in the test (also named 'corrected ITC'). It assesses the consistency of an item with the rest of the test, providing a measure of *item discrimination* – how well the item distinguishes between examinees with high versus low overall performance on the test. ITC analysis is used in test design to improve validity and reliability. Following common practice, ITC values of 0.20 or higher are considered acceptable [20]; we adopt this threshold in the present study.

2.2 Differential Item Functioning (DIF)

DIF methods test whether an item functions differently across groups of respondents conditioned on the respondents' *abilities*. Here, 'ability' refers to a respondent's overall proficiency on the instrument and is measured using the per-item rest score, defined as

$$R_{i,-j} = \sum_{k \neq j} X_{ik},$$

where $R_{i,-j}$ is the sum of the respondent i's correct responses across all items excluding the target item j. We use $R_{i,-j}$ as an observed ability proxy throughout the analyses. In DIF terminology, the *reference group* is the baseline for comparison (here, humans), and the *focal group* is the group tested for differences (here, chatbots). For analysis, we denote each response as $X_{ij} \in \{0, 1\}$ for respondent i on item j, where 1 indicates a correct answer and 0 an incorrect one. Conceptually, an item exhibits DIF if two respondents with the same ability, but from different groups, have systematically different probabilities of answering it correctly. Throughout, we adopt a consistent direction convention: **POS**/**NEG** (positive/negative) indicates a chatbot advantage/disadvantage (higher/lower conditional probability of answering correctly than humans at the same ability level).

2.2.1 *Mantel–Haenszel DIF (MH-DIF).*

MH-DIF compares focal and reference groups within discrete strata of the ability proxy and aggregates these comparisons into a

common odds ratio (α_{MH}). Odds ratios less than one correspond to POS (chatbot advantage), while odds ratios greater than one correspond to NEG (chatbot disadvantage). Following ETS guidelines [29], effect sizes are categorized by the magnitude of $|\log(\alpha_{\mathrm{MH}})|$: Category A (*negligible*) if $|\log(\alpha_{\mathrm{MH}})| < \log(1.5)$, Category B (*moderate*) if $\log(1.5) \leq |\log(\alpha_{\mathrm{MH}})| < \log(2)$, and Category C (*large*) if $|\log(\alpha_{\mathrm{MH}})| \geq \log(2)$. MH-DIF is widely used for detecting *uniform DIF*, or differences that are stable across the ability range, and is particularly robust under stratification with limited sample sizes. For more details on MH-DIF, see [19].

2.2.2 Logistic Regression DIF (LR-DIF). LR-DIF provides a more flexible framework that models the probability of a correct response as a function of ability, group membership, and their interaction:

$$\text{logit}\, P(X_{ij} = 1) = \beta_0 + \beta_1 R_{i,-j} + \beta_2 G_i + \beta_3 (R_{i,-j} \times G_i),$$

where $G_i = 0$ for humans and $G_i = 1$ for chatbots, and $R_{i,-j}$ is the per-item rest score used as an ability proxy. Nested model comparisons allow us to test separately for *uniform DIF* (via the main group effect β_2) and *non-uniform DIF* (via the interaction term β_3). In cases where the interaction implies different directions across the ability range, we label the effect as *Sign-change DIF*. From the fitted logistic function we compute likelihood-ratio p-values (m_0 vs. m_1, m_1 vs. m_2), McFadden's $\Delta R^2 = R^2_{m_2} - R^2_{m_0}$ with $R^2_m = 1 - \ell(m)/\ell(m_{\mathrm{null}})$, and at-median odds ratios, which indicate how much more or less likely chatbots are to answer correctly relative to humans at the median ability level after trimming. We adopt $\Delta R^2 \geq 0.035$ as the threshold for a meaningful effect size [9], and use these measures together to detect and characterize DIF effects.

3. METHODOLOGY

This section describes how the psychometric measures above were applied to design a statistical method that identifies items that function differently for humans and chatbots (RQ1), and also the process through which DIF items were subjected to qualitative analysis by subject matter experts to characterize human–chatbot DIF behavior (RQ2).

3.1 Method Design

Task Definition and Modeling. Our task is to develop a statistically principled method that detects and characterizes multiple-choice items exhibiting systematic response differences between human learners and LLM-based chatbots. Adopting the DIF terminology, we refer to humans as the reference group and chatbots as the focal group.

Design Process. To develop a method for identifying true item-level differences between humans and chatbots, we followed a stepwise process, which is outlined in Fig. 1. Below, we briefly describe the key steps (the implementation and results are detailed in Section 4):

Step 1 – DIF Analysis: Given response data from humans and chatbots (see data description below), we applied the two DIF methods defined above (MH- and LR-DIF). These methods allow us to identify items where humans and chatbots with the same ability level differ in the probability of providing a correct response (here, *ability* refers to respondents' overall proficiency on the instrument). We interpret the items flagged in this step as candidate DIF items.

Step 2 – Negative Control Analysis: To control for false positives and validate the overall stability of the DIF analysis, we performed a negative control analysis (Placebo Test [7]). For that, Step 1 was reapplied to 50 null datasets per instrument, where the chatbot group was replaced with random samples of human responses (drawn without replacement). On each null dataset, both MH- and LR-DIF were rerun, and we recorded which items were flagged as DIF to compute their false-positive rates.

Step 3 – Applying Psychometric Diagnostics: Next, we analyzed the items flagged by the placebo test in Step 2 using psychometric validation measures, with the purpose of identifying whether these items have certain psychometric characteristics (e.g., low ITC) that can explain their instability and be used to filter them upfront.

3.2 Instruments and Data

We used two assessment instruments from two very different contexts. The first was a chemistry diagnostic test administered as a formative assessment activity in preparation for the high school matriculation exam. It consisted of 22 multiple-choice items answered by 931 students. The second was the quantitative section of a psychometric entrance exam used for higher-education admissions, containing 40 items with over 4,800 respondents. Both instruments featured multi-modal items with figures, images, and formulas.

To generate chatbot data, we collected responses from six LLM chatbots representing three distinct model families. For each family, we included both a previous version and the most recent release available at the time of study, specifically: OpenAI's GPT-4o and GPT-5.2; Google's Gemini 1.5 Pro and Gemini 3 Pro; and Anthropic's Claude 3.5 Sonnet and Claude 4.5. Via the chat models' web interfaces, we prompted each model to generate a list of final answers to the instrument, which we attached as a PDF file in the prompt. The models typically returned numbered answer lists, which we exported to CSV. This process was repeated 20 times per model, simulating 20 "artificial students" for each chatbot. Overall, for the analysis, we had 120 responses from the six chatbots pooled together. Runs were conducted in separate sessions to prevent memory effects, and variation across responses was introduced by using a non-zero default temperature setting and default parameters set. Since both assessments included multimodal content, it was important that all models supported visual inputs. When a model skipped an item or did not provide a valid option, the response was counted as incorrect (similar to how student responses were treated). The combined human and chatbot responses were then balanced by down-sampling the more abundant data source to construct a dataset at a 1:10 ratio.

The ability distributions of the chatbots and human learners on both instruments are presented in Fig. 2. The bimodal distribution of the chatbots' ability originates from the division between previous and recent chatbot versions (to examine the robustness of the method to model generations, we validated the method reported below also on a subset containing only the previous versions). Table 1

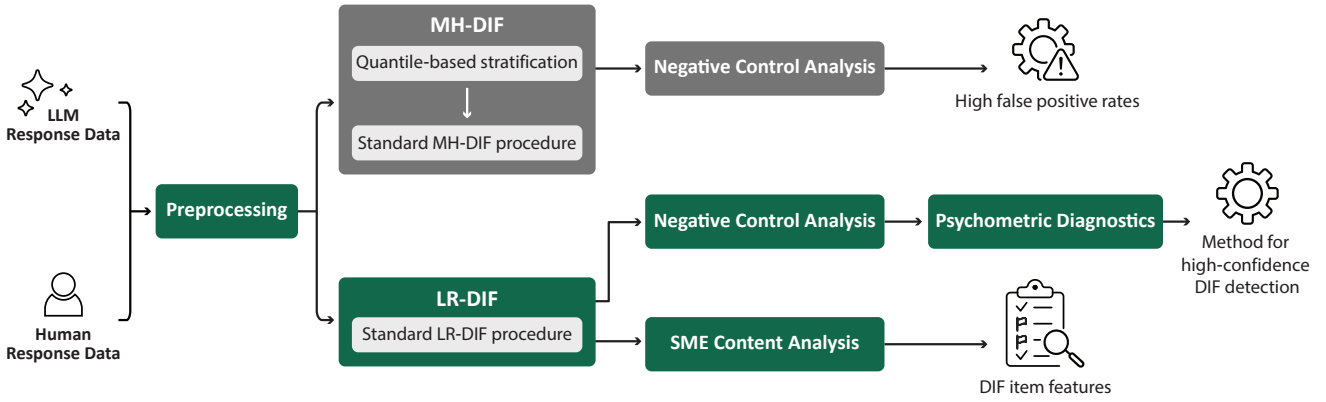


Figure 1: Implemented methodological framework for human–chatbot DIF analysis.

shows the fraction correct of each group on the chemistry items (which were further analyzed by the subject-matter experts). As shown, although the overall success rate of both groups is similar, the chatbots have higher success rates on most items. These sophisticated relations can be handled by the MH- and LR-DIF, as we later demonstrate.

3.3 Qualitative Analysis by Domain Experts

The qualitative analysis was conducted only for the items flagged for DIF by the LR-DIF method in the chemistry instrument. Two subject-matter experts participated in the review process. One expert leads the national team responsible for the annual analysis of the high-stakes chemistry matriculation examinations. The experts received the instrument together with a table summarizing the DIF properties of the flagged items (e.g., POS or NEG; see Table 3) and were asked to qualitatively analyze the DIF items. The two experts independently reviewed, examined, and characterized the items. The two experts then engaged in a joint discussion to compare their evaluations and resolve discrepancies until consensus was reached regarding the characterization of each item.

4. EXPERIMENTS AND RESULTS

This section applies the pipeline outlined in Section 3 and illustrated in Fig. 1. The computational part was applied to the psychometric and chemistry data. The qualitative part was applied to the chemistry dataset, and is described in Subsection 4.5. We then conclude with the resulted procedure for detecting human–chatbot DIF items.

4.1 Preprocessing

The following properties were computed *per item* j. Let $X_{ij} \in \{0, 1\}$ denote correctness for respondent i on item j.

(i) Ability proxy. We defined an *ability proxy* for respondent i as the *rest score* $R_{i,-j}$, the respondent's sum of correct answers across all items *except* item j. Using rest score for respondent i, as opposed to using the global score of respondent i, avoids contamination from the target item and provides a consistent matching variable (i.e., a covariate used to condition comparisons on respondents' overall proficiency).

(ii) Trimming non-overlapping ability ranges. Following DIF analysis guidelines to ensure that both groups were compared only within the overlapping strata of the ability proxy, we removed the human responses in the ability ranges not populated by the chatbots.

(iii) ITC. We computed the items' ITC values, which serve as indicators of item quality, and are later used as a diagnostic filter in the validation stage (see Subsection 4.4). ITC values were computed on the human-only, pre-trimmed responses, as restricting to human data ensures item quality is evaluated relative to the reference group, while using pre-trimmed data preserves the full range of human ability for a fair discrimination estimate.

4.2 RQ1: DIF Technique for Identifying Items on Which Humans and Chatbots Diverge

We applied MH-DIF and LR-DIF on the data, as defined in Subsection 2.2.1, with humans and chatbots as reference and focal groups, respectively.

1. MH-DIF. For each item, we: **(a)** Stratified the trimmed responses by rest score into up to four quantile bins (or fewer when necessary). **(b)** Applied `difMH` to compute the MH statistics retrieved below. **(c)** Retrieved the common odds ratio α_{MH}, the chi-square statistic, and its p-value. **(d)** Flagged items as DIF if $p < 0.05$ and the effect size category (defined in Section 2.2.1) was B or C. **(e)** Assigned direction: $\alpha_{\text{MH}} < 1 \Rightarrow$ POS (chatbot advantage), $\alpha_{\text{MH}} > 1 \Rightarrow$ NEG (chatbot disadvantage).

2. LR-DIF. For each item, we: **(a)** Fit three nested LR models: (i) Ability-only model. (ii) Uniform DIF model: add group membership as a predictor (iii) Non-uniform DIF model: add the ability×group interaction. **(b)** Computed likelihood-ratio test p-values for uniform (p_{uniform}) and non-uniform ($p_{\text{nonuniform}}$) DIF. **(c)** Calculated McFadden's ΔR^2 (fit improvement of the full model over the ability-only baseline). **(d)** Flagged items as DIF if

$$(p_{\text{uniform}} < 0.05 \text{ or } p_{\text{nonuniform}} < 0.05) \quad \text{and} \quad \Delta R^2 \geq 0.035.$$

(e) Summarized flagged items by: (i) Log-odds contrast at median ability. (ii) Odds ratio = exp{log-odds contrast}. (iii) Probability gap between humans and chatbots at median ability. **(f)** Classified direction as POS, NEG, or *sign-*

Table 1: Fraction correct per item for students and chatbots (chemistry).

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22
Students	0.836	0.531	0.603	0.699	0.921	0.918	0.968	0.968	0.924	0.944	0.740	0.563	0.820	0.827	0.783	0.429	0.484	0.487	0.374	0.552	0.672	0.658
Chatbots	0.979	0.792	0.917	0.781	0.802	0.479	0.667	0.990	1.000	0.979	0.510	0.865	0.979	0.458	0.365	0.917	0.979	0.688	0.802	1.000	0.604	0.240

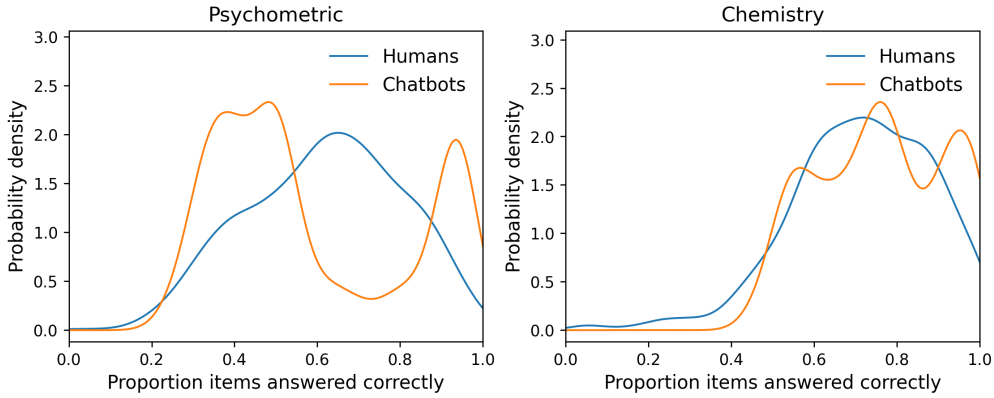


Figure 2: Human–Chatbot ability distribution.

Table 2: MH-DIF: Summary of DIF-flagged items.

Inst.	POS	NEG
Psych.	4-6, 9-10, 12, 16, 21-22, 25, 28, 30-33	11, 13-15, 17, 20, 23-24, 34-38, 40
Chem.	5-7, 11, 14-15, 21-22	1-3, 12-13, 16-19

Table 3: LR-DIF flagged items: $\bar{Q}$: uniform; $\tilde{Q}$: non-uniform.

Inst.	POS	NEG
Psych.	$\bar{1}$1, $\tilde{1}$3-$\tilde{1}$5, $\tilde{3}$4, $\bar{4}$0	$\tilde{1}$6, $\bar{2}$1, $\tilde{2}$5, $\tilde{3}$1
Chem.	$\bar{1}$2, $\bar{1}$6, $\bar{1}$7, $\bar{1}$9	$\bar{5}$-$\bar{7}$, $\bar{1}$4, $\bar{1}$5, $\tilde{2}$2

change if the group effect changed sign at different ability levels.

The items flagged as DIF by both methods are presented in Table 2 and Table 3. As can be seen, MH-DIF flagged a large set of items as DIF, and particularly, a large number of items as POS (chatbot advantages conditioned on ability), including items on which the students scored higher when compared only on raw averages (e.g., items 6-7 in the chem. instrument; see average group scores in Table 1). The LR-DIF was much more conservative, as expected, flagging less items overall. Its more sophisticated modeling enabled it also to identify items with non-uniform DIF (DIF magnitude changes across the ability ranges; e.g., chem. item 12).

These patterns indicate that chatbot–human differences are not only widespread but also vary by item, by direction, and by ability levels, highlighting the diagnostic power offered by DIF analysis. However, in light of the considerable disagreement between the methods (e.g., chemistry item 1, which was flagged as POS by the LR-DIF and NEG by the MH-DIF), we conducted Negative Control Analysis to discriminate true DIF from statistical artifacts.

4.3 Negative Control Analysis

For both datasets, we created $R = 50$ *null datasets* in which the chatbot group (n=60) was replaced by a random sample of human respondents (without replacement). We then reran the MH-DIF and LR-DIF pipelines on each null dataset. Table 4 reports the false-positive counts of items that were flagged at least once under the 50 placebo test runs. Two patterns stand out: **(1)** MH-DIF exhibits diffuse background noise: many items are flagged once or twice, and several reach 8–10% false-positive rates. This noise profile appeared on both instruments, indicating that, at least in our experiment, MH-DIF alone cannot yield reliable DIF signals without additional filtering. **(2)** LR-DIF is far more stable. On the psychometric instrument it produced no null flags across all 50 runs, demonstrating excellent specificity. On the chemistry instrument, LR-DIF was generally stable but showed a small cluster of unstable items (e.g., Item 5).

These findings support LR-DIF as the primary DIF detection method. Yet the presence of a small unstable chemistry items cluster, which showed elevated false-positive rates in the null runs, needs additional item-level analysis and filtering (see Section 4.4).

4.4 Psychometric Diagnostics and Results

We incorporated psychometric diagnostics to analyze and refine the candidate DIF set flagged by LR-DIF. As shown in Table 4, Negative Control Analysis confirmed LR-DIF's stability, though chemistry items 5,9, and 10 exhibited false-positive rates above 10%. To investigate these items further, we examined their ITC, and found that each had very low ITC values (Item 5 = 0.11, Item 9 = 0.08, Item 10 = 0.04). In contrast, chemistry items with ITC values above 0.20 were consistently stable across null runs. These results reinforced ITC as a reliable diagnostic for DIF analysis, using

Table 4: Negative Control Analysis: The items on each *false_positive_count* level (0 omitted).

Counts	Psych.(MH-DIF)	Chem.(MH-DIF)	Psych.(LR-DIF)	Chem. (LR-DIF)
1	6-7, 10, 12-15, 17, 19-21, 24, 27-28, 33, 38	2-3, 6, 9, 13-14, 17-19, 22	–	14
2	9, 18, 22, 29-32, 37, 39	7, 16	–	6-8
3	2, 8, 16, 26, 34	11, 20, 21	–	–
4	25, 35	–	–	–
5	23, 40	12	–	–
6-7	–	–	–	–
8-11	–	–	–	5, 9-10

the standard 0.20 as a cutoff for 'acceptable' [20].

4.5 RQ2: Domain Expert Analysis

As described in Subsection 3.3, the qualitative analysis of the DIF items was conducted by two subject-matter experts who received the instrument and a table that listed the properties of the LR-DIF items, taken from Table 3.

As shown in the Table 3, Items 12, 16, 17, and 19 had POS DIF, meaning that chatbots had *higher* likelihood of answering them correctly than humans at the same ability level. In 12, the system successfully circumvented a common alternative conception according to which sodium chloride is composed of atoms. Instead, it correctly distinguished between the ionic lattice structure of NaCl and the behavior of its constituent particles in solution, emphasizing that dissolution involves the separation of Na^+ and Cl^- ions rather than the breaking of atoms or the formation of atomic species. In 16, which involves a sequence of solution manipulations, the system effectively maintained consistency by monitoring the relationship between solute moles, solution volume, and concentration across several experimental stages. In 17 demonstrated a scientifically appropriate interpretation of the law of conservation of matter by situating the assessment of mass change at the level of the entire experimental system, rather than restricting the analysis just to the original metal. Item 19 further highlighted the system's capacity to apply algorithmic reasoning: the multiple-choice structure facilitated accurate assignment of oxidation states, identification of electron transfer, and correct designation of oxidizing and reducing agents. Collectively, these examples underscore the system's strength in tasks that prioritize rule-governed reasoning and conceptual clarity, where adherence to canonical chemical principles prevents susceptibility to intuitive but scientifically inaccurate interpretations.

In contrast, items 6, 7, 14, and 15, had NEG DIF, meaning that chatbots had *lower* likelihood of answering them correctly than humans at the same ability level. These items place greater demands on visual interpretation, linguistic nuance, and the execution of complex, multi-step procedures. Items 6-7 require learners to interpret partial microscopic models, such as diagrams omitting solvent molecules, or to map symbolic labels (A–C) onto microscopic and macroscopic representations of aqueous solutions. These forms of representation, which depend heavily on implicit visual conventions, posed substantial challenges for the system, whereas students were able to draw upon prior representational.

Questions 14 and 15 illustrate a further point of divergence. Beyond the computational demands involved, these questions also require sensitivity to linguistic nuance, namely the ability to interpret subtle wording, implied meanings, and context-dependent scientific language. For example, in Question 15, students needed to recognize that the statement "the volumes were measured under the same conditions" implicitly justifies the use of proportional relationships between gas volumes and moles according to Avogadro's law. Similarly, Question 14 requires interpreting the phrase "the volume of 1 mole of gas is 25 liters" as applying specifically to gaseous substances in the reaction, while distinguishing between gaseous and liquid species within the balanced equation. Such interpretive demands rely not only on factual chemical knowledge, but also on sensitivity to how scientific information is linguistically framed. Question 14 further involves a multi-stage stoichiometric calculation, proceeding from gas volume to moles, through mole ratios, and finally to mass determination, with each step contingent on the correctness of the one before it. Such extended procedures are particularly vulnerable to intermediate errors in algorithmic reasoning. Question 15 compounds this difficulty, as errors originating in the previous calculation can propagate forward, whereas students often display the metacognitive awareness required to detect and correct inconsistencies in their own work.

Taken together, these findings suggest that the chatbots excel in domains governed by formalized conceptual rules. By contrast, students retain a relative advantage in tasks that demand flexible interpretation of visual representations, sensitivity to linguistic subtleties, and careful monitoring of extended quantitative reasoning processes.

Resulting Method. The conclusion from the experimental results, which both validated the computational process statistically, and evaluated the meaningfulness of the LR-DIF results with subject matter experts on a sample dataset (chemistry), yielded that applying LR-DIF to items with ITC ≥ 0.2 is a reliable method for identifying items that, with high confidence, exhibit differential behavior across humans and chatbots.

5. DISCUSSION

The LR-based DIF approach developed and piloted in this research provides a method and conceptual framework for identifying assessment items that show differential behavior between humans and chatbots. In two assessment contexts and on leading chatbots, we demonstrated that the method provides reliable and stable results. A key observation that DIF, and this research, makes, is that differential behavior is a more nuanced property than merely looking at overall performance gaps between groups, since the DIF analysis controls for ability level, and can identify items as

positive/negative DIF (chatbots over/under-perform learners of the same skill) even if the difference between the mean group performance is in the opposite direction. Our LR-DIF based method offers a modeling that also enables the detection of non-uniform DIF patterns in which either the magnitude or direction of group differences vary across ability levels. Our experiments with both the MH- and LR-DIF highlighted that the MH-DIF, typically the default choice due to its simplicity, produces considerable noise and a high false positive rate. The LR-DIF excelled in this aspect as well. Its downside is its relative complexity and the fact that it typically requires larger sample sizes for stable estimates.

Using the DIF-method to deliver content analytics to the subject-matter experts (RQ2) drove an analysis that yielded interesting insights. As reported in Subsection 4.5, analyzing the POS DIF items, we found that the chatbots managed to circumvent distractors aimed at surfacing alternative conceptions (sometimes referred to as 'misconceptions') commonly held by students. Given that LLMs can be cognitively biased due to their training data [6], which likely included student data also representing wrong or incomplete knowledge [27], the fact that the chatbots dodged these designated 'traps' is interesting, and is in disagreement with previous work that found moderate alignment between LLMs and student misconceptions [12]. On the contrary, experts' analysis of the NEG DIF items revealed that the chatbots underperformed on items requiring visual interpretation and connecting visual and textual information, items that their wording required understanding linguistic nuances, and those requiring performing multi-step problem solving procedures. These findings reinforce previous work about GenAI problem solving in STEM domains [24, 26].

More generally, this research demonstrates a fruitful application of assessment and measurement theory to educational data mining, with the purpose of establishing theory-grounded approaches for understanding GenAI capabilities, and for developing assessment in the GenAI era.

Limitations. This research is the first, to our knowledge, to apply DIF for analyzing chatbot assessment data and, as such, naturally has several limitations. In terms of internal validity, the chatbots' bi-modal performance (fraction correct; see Figure 2) may impact the DIF stratification step, potentially reducing the effective size of the data and thus the method's robustness. The very high ability that the newer models exhibit may reflect data leakage or prior exposure during training, rather than genuine ability, and may falsely cause them to be identified as positive DIF (models outperform humans). A further limitation of our study is that human and chatbot ability estimates were based on a binary scoring model rather than a polytomous one. More nuanced scoring models may capture partial knowledge or intermediate reasoning patterns, and could therefore affect the ability estimates and the resulting DIF patterns. A related limitation is that DIF methods typically assume a relatively uni-dimensional construct and comparison along a common ability continuum [8, 28]. Given the heterogeneous task types included in the instruments, such as visual versus algorithmic reasoning, as well as the use of multiple LLM architectures, some of the observed DIF effects may reflect multidimensionality or within-group variability rather than item-level bias alone. In addition, the SME analysis focused only on LR-DIF flagged items, which were shown to favor either humans or chatbots, and did not compare them against non-DIF items. Such a comparison could provide a clearer interpretation of the features that distinguish DIF from non-DIF items and is left for future work.

The main limitation to external validity is that the results are based on a small number of instruments and specific GenAI tools. Additionally, the SME analysis was conducted only for the Chemistry instrument; extending this analysis to the quantitative reasoning instrument would be important for broadening observed patterns to additional domains.

Contribution and future work. The main contribution of this work is the introduction of a theory-driven and statistically sound method for detecting items that exhibit differential behavior between chatbots and human learners, and the presentation of evidence of its ability to yield meaningful analytics for subject-matter experts who seek to integrate GenAI considerations into assessment design. In future work, we plan to build on this foundation for (1) studying the task dimensions of Human–GenAI DIF items, to better understand the capabilities of this technology, (2) incorporating Human–GenAI DIF analysis into assessment design, and (3) Use DIF analysis to evaluate GenAI progress.

Ethics statement. The research was approved by the Institutional Review Board No. 3331-2.

Data and code availability. The materials needed to reproduce the analyses reported in this paper, including the code, sample data, and prompts used to generate the chatbot data, are available in the GitHub repository: (link)

6. ACKNOWLEDGMENTS

This work was supported by the Knell Family Institute for Artificial Intelligence, Israel. The authors thank the National Institute for Testing and Evaluation for providing access to psychometric exam data.

7. REFERENCES

[1] N. Akbari. The AI cheating crisis: Education needs its anti-doping movement, 2024. Retrieved from `https://www.edweek.org/technology/opinion-the-ai-cheating-crisis-education-needs-its-anti-doping-movement/2024/02`.

[2] T. Barnes. The q-matrix method: Mining student response data for knowledge. In *American association for artificial intelligence 2005 educational data mining workshop*, pages 1–8. AAAI Press, Pittsburgh, PA, USA, 2005.

[3] B. Borges, , et al. Could chatgpt get an engineering degree? evaluating higher education vulnerability to ai assistants. *Proceedings of the National Academy of Sciences*, 121(49), 2024.

[4] D. R. E. Cotton, P. A. Cotton, and J. R. S. and. Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. *Innovations in Education and Teaching International*, 61(2):228–239, 2024.

[5] R. J. De Ayala. *The theory and practice of item response theory*. Guilford Publications, 2013.

[6] J. M. Echterhoff, Y. Liu, A. Alessa, J. McAuley, and Z. He. Cognitive bias in decision-making with LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, 2024.

[7] A. C. Eggers, G. Tuñón, and A. Dafoe. Placebo tests for causal inference. *Am. J. Polit. Sci.*, 68(3):1106–1121, 2024.

[8] P. W. Holland and H. Wainer. *Differential item functioning*. Routledge, 2012.

[9] M. G. Jodoin and M. J. Gierl. Evaluating type i error and power rates using an effect size measure with the logistic regression procedure for DIF detection. *Applied Measurement in Education*, 14(4):329–349, 2001.

[10] M. Khalil and E. Er. Will ChatGPT get you caught? Rethinking of plagiarism detection. In *International Conference on Human-Computer Interaction*, pages 475–487. Springer, 2023.

[11] K. R. Koedinger, E. A. McLaughlin, and J. C. Stamper. Automated student model improvement. *International Educational Data Mining Society*, 2012.

[12] N. Liu, S. Sonkar, and R. Baraniuk. Do llms make mistakes like students? exploring natural alignments between language models and human error patterns. In *International Conference on Artificial Intelligence in Education*, pages 364–377. Springer, 2025.

[13] Y. Liu, S. Bhandari, and Z. A. Pardos. Leveraging LLM respondents for item evaluation: A psychometric analysis. *British Journal of Educational Technology*, 56(3):1028–1052, 2025.

[14] P. Lu et al. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

[15] D. Magis, S. Beland, F. Tuerlinckx, and P. De Boeck. A general framework and an r package for the detection of dichotomous differential item functioning. *Behavior research methods*, 42(3):847–862, 2010.

[16] N. Mantel and W. Haenszel. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the national cancer institute*, 22(4):719–748, 1959.

[17] P. Martinková, A. Drabinová, Y.-L. Liaw, E. A. Sanders, J. L. McFarland, and R. M. Price. Checking equity: Why differential item functioning analysis should be a routine part of developing conceptual assessments. *CBE Life Sci. Educ.*, 16(2), 2017.

[18] A. Prothero. New data reveal how many students are using AI to cheat, 2024.

[19] H. J. Rogers and H. Swaminathan. A comparison of logistic regression and mantel-haenszel procedures for detecting differential item functioning. *Applied psychological measurement*, 17(2):105–116, 1993.

[20] A. Shete. Item analysis: An evaluation of multiple choice questions in physiology examination. *J. of Contemporary Med. Educ.*, 2015.

[21] B. Sorenson and K. Hanson. Identifying generative artificial intelligence chatbot use on multiple-choice, general chemistry exams using Rasch analysis. *Journal of Chemical Education*, 101(8):3216–3223, 2024.

[22] A. Strugatski and G. Alexandron. Applying IRT to distinguish between human and generative AI responses to multiple-choice assessments. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference, LAK 2025, 2025*, pages 817–823. ACM, 2025.

[23] T. Susnjak and T. R. McIntosh. ChatGPT: The end of online exam integrity? *Education Sciences*, 14(6), 2024.

[24] K. D. Wang, E. Burkholder, C. Wieman, S. Salehi, and N. Haber. Examining the potential and pitfalls of ChatGPT in science and engineering problem-solving. In *Frontiers in Education*, volume 8, page 1330486. Frontiers Media SA, 2024.

[25] X. Wang, Z. Hu, P. Lu, Y. Zhu, J. Zhang, S. Subramaniam, A. R. Loomba, S. Zhang, Y. Sun, and W. Wang. Scibench: Evaluating College-Level Scientific Problem-Solving Abilities of Large Language Models. *arXiv preprint arXiv:2307.10635*, 2023.

[26] E. Yacobson, Y. Schleifer, Z. Bar-Dov, S. Rap, R. Blonder, and G. Alexandron. Benchmarking ai on standard chemistry exams: Llms still underperform compared to high school students. *Journal of Science Education and Technology*, pages 1–18, 2026.

[27] L. Yan, L. Sha, L. Zhao, Y. Li, R. Martinez-Maldonado, G. Chen, X. Li, Y. Jin, and D. Gašević. Practical and ethical challenges of large language models in education: A systematic scoping review. *Br. J. Educ. Technol.*, 55(1):90–112, 2024.

[28] B. Zumbo. A handbook on the theory and methods of differential item functioning (dif). 01 1999.

[29] R. Zwick. A review of ets differential item functioning assessment procedures: Flagging rules, minimum sample size requirements, and criterion refinement, 2012.

Improved Power in a Reanalysis of an ASSISTments Efficacy Trial by Leveraging Auxiliary Data

Adam C. Sales
Worcester Polytechnic Institute
asales@wpi.edu

Chun-Wei Huang
WestEd
chuang@wested.org

Mingyu Feng
WestEd
mfeng@wested.org

Johann A. Gagnon-Bartsch
University of Michigan
johanngb@umich.edu

ABSTRACT

Randomized controlled trials (RCTs) in social or biomedical science often rely on covariate and outcome data drawn from a larger administrative database. For instance, a recent paired, clustered RCT of an online homework platform used covariates (demographics and prior achievement measures) and outcomes (standardized test scores) from the state's educational database. To avoid confounding, typical analyses use only data from randomized subjects, discarding the remainder of the database.

In this study, we re-analyze data from the online homework study, substantially improving the precision of the effect estimates by using data from both randomized and non-randomized subjects. Our estimator does not introduce confounding and relies largely on standard modeling techniques. Specifically, we use observational data to train a random forest model predicting student test scores from student- and school-level covariates, and use it to generate predicted outcomes for students in the randomized schools. Then, we modify the hierarchical linear model from the original analysis by adding the predicted outcomes as an additional regressor, reducing the standard error by 15%—the improvement expected from increasing the sample size by over a third. Since it requires few additional modeling skills and no additional assumptions, this method can be easily adapted wherever auxiliary observational data are available.

Keywords

Causal Inference, Randomized Experiments, Supervised Learning, Hierarchical Linear Models

1. INTRODUCTION

In 2018, WestEd, an educational research agency, began a large-scale randomized field trial of the online mathematics application ASSISTments in 63 schools in North Carolina [3]. The researchers were granted access to the North Carolina student longitudinal data system, from which they drew most of the data used for the evaluation, including demographic and student achievement data from the students in the study—seventh-grade math students in the 63 randomized schools. However, the researchers had access to even more data: demographic, prior achievement, and outcome data from over 65 thousand additional 7th-grade students who attended North Carolina middle schools other than those 63.

The WestEd ASSISTments study is one example of a common phenomenon: covariate and outcome data for a randomized controlled trial (RCT) are drawn from a larger database that also includes data from similar subjects who were not randomized as part of the experiment. Other examples include A/B tests, which draw baseline covariates and outcomes from log data, and clinical trials, which draw baseline and outcome data from patients' medical records. In these examples, researchers have access to big auxiliary observational datasets that are typically discarded or left unused. Granted, pooling data from randomized and non-randomized subjects would introduce confounding—there is no guarantee that subjects in the auxiliary data are comparable to the randomized subjects—and undermine the main motivation for conducting an RCT in the first place.

However, there are other uses for auxiliary observational datasets that maintain the integrity of the RCT. Specifically, [4] suggested using big auxiliary data to train a model predicting experimental outcomes, then using the resulting predictions as additional super-covariates in the "leave-one-out potential outcomes" (LOOP) effect estimator [23]. In studies of educational A/B tests, [4] and [19] showed that this approach can substantially reduce standard errors and increase statistical power, relative to estimators that do not use the auxiliary data. Similarly, [10] found similar results in a secondary analysis of a subset of school-aggregated data from a large, school-randomized effectiveness study, using a different, publicly available outcome than the original study.

These results are promising, but may not be fully convincing to education researchers conducting randomized field trials. First, we are unaware of any published analysis of a large randomized educational effectiveness study like [3] that uses auxiliary data as [4] proposes that uses the study's primary

Adam Sales, Kevin Huang, Mingyu Feng, and Johann Gagnon-Bartsch. Improved Power in a Reanalysis of an ASSISTments Efficacy Trial by Leveraging Auxiliary Data. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 487–493. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039716

outcome measure and includes data from the full RCT. Second, the method from [4] has been applied only to single-level data (or to multilevel data aggregated to the school level). In contrast, most large-scale educational efficacy studies use multilevel data, analyzing students as nested within schools. Third, previous studies have focused on the use of auxiliary data within the LOOP effect estimator, a new and unfamiliar estimator with limited availability and applicability, rather than incorporating insights from auxiliary data into familiar models.

In this paper, we present a secondary analysis of the WestEd ASSISTments study incorporating predictions from a random forest model trained on the auxiliary observational dataset composed of North Carolina students who did not participate in the original RCT. Our analysis uses the entire dataset from the original study, along with the study's primary outcome measure, maintains and fully models the multilevel structure of the RCT dataset, and uses hierarchical linear modeling, the dominant approach for analyzing educational field trials, to estimate effects.

We find that, relative to the original estimates published in [3], our analysis reduces standard errors by 15%, roughly equivalent to increasing the number of schools in the study by 38%. Like previous auxiliary data methods, our approach does not introduce confounding and maintains the integrity of the randomization; unlike previous approaches, it is accessible to applied education researchers.

The following section describes the WestEd ASSISTments study of [3]. The next section reviews the relevant methodology, including the methods of [4] and hierarchical linear models. Section 4 describes our analysis, and Section 5 describes the results. Section 6 concludes.

2. THE ASSISTMENTS EFFICACY REPLICATION STUDY

ASSISTments [6] is an internet-based program for mathematics practice and learning, with a focus on middle school students. Teachers may draw from a large corpus of math problems—many drawn from popular curricula and textbooks—to create assignments for their classes. Students receive immediate feedback on the correctness of their work and often have access to hints or other tutoring materials. Teachers can monitor their students' progress and browse useful data summaries for insights into student performance. A large, federally-funded randomized study conducted in Maine [14] showed that randomization to the ASSISTments condition increased student standardized test scores by 0.22 ± 0.07 standard deviations with 95% confidence.

Starting in 2018, WestEd conducted a replication study based in North Carolina, with the goal of extending the previous results to a more diverse population [3]. The study team matched 65 schools into 31 pairs and one triplet based on school-level demographics and prior achievement measures, and then randomly selected one school from each pair and two schools from the triplet for the treatment arm of the study. In those schools, all 7th-grade math teachers used ASSISTments for two academic years, 2018–2019 and 2019–2020, with teachers in the remaining 31 schools continuing business as usual. The original analysis plan was to use the 2019–2020 cohort's end-of-grade mathematics state standardized test to measure efficacy. Unfortunately, due to the COVID-19 pandemic, the 2020 state standardized test was canceled. Fortunately, the study team was able to link the study data with student scores from a year after implementation ended, and estimate treatment effects using standardized test scores from the end of 8th grade. There were two different math standardized tests in 8th grade: the majority took the 8th-grade "end-of-grade" test (EOG08), while a subset of more advanced students took an Algebra I " end-of-course " exam. Following [3], our analysis focuses on the EOG08 exam, under the assumption that treatment assignment did not affect which exams students took. This assumption appears plausible: there was no evidence of differential attrition, and the analysis sample remained balanced on important covariates. All in all, the analysis sample includes 5,991 students, 3,030 randomized to the control condition, and 2,961 randomized to the treatment condition.

To estimate effects, the North Carolina Education Research Data Center granted the WestEd team access to demographic, achievement, and other educational data for all public school students in North Carolina. The study team selected covariates—including prior achievement measures at the student level (a math standardized test at the end of 6th grade, EOG6) and the school level (average 7th-grade scores from prior years), and demographics, including gender, race or ethnicity, and disability and economically disadvantaged status. They also calculated school-level demographics, including 7th-grade enrollment, whether the school was eligible for Title I funding, and the percentages of economically disadvantaged students and students from different ethnic groups. Following standard procedure, no use was made of covariate or achievement data from students in North Carolina schools that did not participate in the RCT.

The study's main overall effect estimate was that randomization to ASSISTments in 7th grade caused an average impact of 0.80 ± 0.64 points on the EOG08 exam (with 95% confidence), corresponding to an effect size (Hedges' g) of 0.10. Since the impact was measured a year after the intervention ended, this represents a persistent effect. See [3] for more details and estimated effects for subgroups.

3. METHODOLOGY BACKGROUND

Statistical analysis of RCTs requires data on the randomized condition, Z—which here we will take as binary, with $Z_i = 0$ if subject i, $i = 1, \ldots, N^{RCT}$ is randomized to the control condition and $Z_i = 1$ if i is randomized to treatment—the outcome of interest, Y, and, typically, a vector of covariates $\boldsymbol{x}$. Covariates $\boldsymbol{x}$, by definition, must be either fixed at baseline, i.e., at the time of randomization, or otherwise known to be independent of or invariant to randomization. Covariates are not always strictly necessary, but they can be invaluable for detecting, testing, and sometimes correcting irregularities in randomization, increasing the statistical precision of the effect estimate, and studying heterogeneity in causal impacts, and hence predicting treatment effects for future subjects or populations.

3.1 Observational Auxiliary Data

Data on Y and $\boldsymbol{x}$ may be available for a separate sample of units that did not participate in the RCT. For instance,

if researchers conducting an RCT drew data on $\boldsymbol{x}$ and Y from a larger database that included, say, historical data or data on current units that were not randomized as part of the study. Along these lines, [20] discussed A/B tests run on ASSISTments E-trials [22], a platform for educational experiments within an online tool for middle-school mathematics practice. In that case, data on the outcome Y, assignment completion, and covariates $\boldsymbol{x}$ (students' prior activities) were available for historical and concurrent users working on different parts of the platform. In this paper, we will re-analyze data from [3], a large educational field-trial testing the efficacy of ASSISTments in a set of North Carolina middle schools in which data on Y, student scores on state standardized tests, and $\boldsymbol{x}$, student demographics and prior standardized test scores, were drawn from the state department of education's longitudinal data system. WestEd, which conducted and analyzed the RCT, was given access to the entire database and thus had access to data from students attending schools that were not part of the RCT.

Auxiliary observational data may seem of little use. In the cases just mentioned, data on the treatment condition Z are not available for students in the observational samples—we presume their experience was akin to the control condition, but this cannot be known. Even if data on the treatment or exposure were available in the observational sample, there would be good reasons not to use them — unlike in an RCT, exposure to the treatment or control conditions may be confounded. Furthermore, there is no guarantee that expected potential outcomes in the auxiliary data represent those in the RCT or even that the conditional distribution of outcomes given treatment status and covariates, $p(Y|Z, \boldsymbol{x})$, is the same in the RCT and observational samples.

On the other hand, auxiliary observational data typically has a much larger sample size than an RCT, and could be helpful in modeling the conditional distribution of Y given $\boldsymbol{x}$, enabling more precise effect estimation. Covariate data in [20] consisted of measurements from the sequence of assignments that students had worked on prior to the onset of the experiment—a complex longitudinal structure that was best modeled by a recurrent neural network that would be impossible or ill-advised to train from a small or moderately-sized RCT sample. The dataset in [3] is high-dimensional, longitudinal—including measurements from multiple successive years—and multilevel, since student data are nested within schools. A model fit only to RCT data would be unable to capture those complexities fully. Moreover, even if RCT data were large enough to support modern ML models, incorporating such a model into a causal effect estimator poses its own challenges, especially when the ML model cannot be guaranteed to be unbiased, as is typically the case.

Recent literature [12, 17, 4, 20] suggests an approach to using big observational data to model the relationship between $\boldsymbol{x}$ and Y and then incorporating that relationship into an RCT analysis without requiring additional untestable assumptions regarding the auxiliary data or the ML model and without introducing any confounding bias. The idea is, essentially, to use the observational data to train a supervised learner predicting Y as a function of $\boldsymbol{x}$; to use that pre-trained model, along with $\boldsymbol{x}$ data from RCT subjects, to generate predicted outcomes $\hat{y}^{obs}$ for RCT subjects; and finally to statistically adjust for $\hat{y}^{obs}$ in an effect estimator. The predictions $\hat{y}^{obs}$ act as a one-dimensional summary of multidimensional covariates $\boldsymbol{x}$, where the summary was designed to extract as much relevant information about Y as possible. Because it is based on RCT covariates $\boldsymbol{x}$ and a model trained on an external dataset, $\hat{y}^{obs}$ are independent of Z—in other words, $\hat{y}^{obs}$ itself is a covariate.

Ideally, $\hat{y}^{obs}$ will be closely correlated with observed outcomes Y within each treatment group—in other words, the auxiliary-data model closely predicts RCT outcomes. A subject's treatment effect, by definition, is the difference between that subject's observed outcome Y_i and the counterfactual outcome, or what *would have* been observed if i had been randomized to the other treatment. Hence, accurate predictions of counterfactual outcomes can increase statistical precision and power when estimating average treatment effects. However, when $\hat{y}^{obs}$ are only weakly correlated with Y, or even anti-correlated, methods described in [4, 20] ensure that the effect estimates remain exactly unbiased and (approximately in small samples, exactly in large samples) no less precise than unbiased estimators that do not use auxiliary data.

Unfortunately, the specific methods recommended in those papers may not always be applicable or practical. [4, 20] recommend adjusting treatment effect estimation with a powerful but somewhat complex method [23, 24] that is currently only available for Bernoulli or paired individually-randomized trials and for paired group randomized trials in which covariates and outcomes are aggregated to the group level [9]. Moreover, these may be unfamiliar or inaccessible to most applied researchers (they are currently implemented only in an experimental `R` package [8]).

3.2 Group-Randomized Trials and Hierarchical Linear Models

Multilevel structures are a hallmark of K–12 education: students are nested within classrooms, which are nested within schools. It is often impractical or impossible to randomize classmates or schoolmates into different educational conditions, so a common technique in experimental education research is to randomize entire classrooms or schools between conditions [11, 15, 3]. In these cases, every student in a single classroom or school is randomized to the same condition, either treatment or control. Educational outcomes tend to be correlated among students within the same school [5], so it is crucial to account for the clustered-randomization design when analyzing clustered RCTs, for instance, by using school-average outcomes to estimate effects [9].

However, researchers often prefer to maintain their data's multilevel structure by using models that explicitly account for within-school correlations. In educational effectiveness research, the most common of these is the hierarchical linear model (HLM) [13]—or, more precisely, a random intercept model—such as:

$$Y_i = \alpha_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i + \tau Z_{s[i]} + \boldsymbol{\gamma}^T \boldsymbol{w}_{s[i]} + \eta_{s[i]} + \epsilon_i \qquad (1)$$

where the subscript $s[i]$ indicates student i's school, α_0 is an overall intercept, $\boldsymbol{w}_s$ are school-level covariates, and $\eta_s \sim N(0, \sigma^2_{scl})$ and $\epsilon \sim N(0, \sigma^2_{ind})$ are school- and student-level regression errors, with variances σ^2_{scl} and σ^2_{ind} esti-

mated from the data. When schools are paired prior to randomization—so that one school from each pair is randomized to treatment, and the other to control, as in [3]—pair indicators are included in $\boldsymbol{w}$.

The coefficient τ represents the estimated treatment effect—the extent to which student scores in treated schools vary, on average, from scores in control schools, conditional on covariates and other school-level factors.

In an HLM such as (1), student scores Y_i are modeled as varying based on their school, via η_s, $\boldsymbol{w}_s$ and Z_s, and individual characteristics $\boldsymbol{x}$. Since treatment is assigned at the school level, the treatment indicator Z has a subscript s and appears in the school-level model of (1). Student scores Y are assumed to be independent conditional on regressors $\boldsymbol{x}$, Z, and $\boldsymbol{w}$ and school errors $\boldsymbol{\eta}$; the school intercepts, in turn, are independent of each other conditional on Z and $\boldsymbol{w}$. Finally, conditional on $\boldsymbol{x}$, Z, and $\boldsymbol{w}$, student scores Y_i and Y_j are independent for students in different schools, $s[i] \neq s[j]$ but correlated with "intraclass correlation coefficient" $\rho = \sigma^2_{sch}/(\sigma^2_{sch} + \sigma^2_{ind})$ for two students in the same school. Hence, HLMs explicitly account for student correlation within schools and are therefore appropriate for school-randomized studies.

4. METHOD

The student-level data used in this study were confidential; hence, all analyses were conducted at WestEd, using Stata.

We followed a two-step procedure to estimate average effects:

1. Use the observational data to train a random forest [2] using baseline covariates to predict student test scores, and use that fitted model to predict test scores for students in the RCT.
2. Estimate average treatment effects by fitting a hierarchical linear model to RCT data, including the predictions from step 1 as a regressor.

The following sections will describe each of these steps in detail.

4.1 Predicting Student Test Scores

As described in Section 2, the research team had access to student-level data from across North Carolina. We used this database to compile an observational auxiliary dataset for training a model to predict students' EOG08 test scores.

First, we identified 65,658 North Carolina middle school students who were not part of the RCT and took the 2021 EOG08 exam. We used anonymized records from all of these students to compile an observational auxiliary dataset. Specifically, we first assembled relevant baseline covariates; since the experiment took place when students were in 7th grade, we restricted covariates to measurements recorded when students were in 6th grade or earlier. These included 6th-grade student background data—age; status as academically gifted, disabled, economically disadvantaged, and/or English language learner; ethnicity indicators; and whether a student was in foster care, homeless, a migrant, and/or connected to the military. We also included students' reading and math assessments from grades 3-6, and 5th-grade science assessments.

We suspected that school characteristics may predict student outcomes, even beyond individual student predictors. Hence, we calculated additional school-level features to include in the random forest. Specifically, we calculated the percentage of students at each school and grade level (grades 3–6) who belonged to each demographic category, as well as the school-grade-level average math, reading, and science assessment scores. All in all, this amounted to 94 student- and school-level features per student.

After compiling the covariates, we fit a random forest regression model to predict students' EOG08 scores. We fit the model using the `rforest` command in `Stata` [21], using the system default tuning parameters (i.e., investigate 9 variables per split and 100 iterations). The random forest predicted posttest scores in the auxiliary data with high accuracy: the out-of-bag prediction R^2 was roughly 0.92.

Finally, we used the fitted model to predict EOG08 math test scores for students in the RCT sample.

As argued above, these predictions, $\hat{y}^{obs}$, can be considered baseline covariates: they were constructed using RCT students' measurements from 6th grade or earlier, and a model fitted to an external sample. Hence, they were invariant to the randomized treatment assignment in the RCT.

4.2 Estimating Treatment Effects

To estimate average treatment effects, we simply re-ran the HLM from the original analysis with $\hat{y}^{obs}$ as an additional covariate, and reported the coefficient on the treatment randomization indicator, along with its standard error.

That is, we compared the original HLM of [3] with an HLM that included $\hat{y}^{obs}$ as an additional student-level predictor (i.e., an additional column of $\boldsymbol{x}$ in 1), and was otherwise identical.

In exploratory analyses, we compared our main estimator with an alternative in which $\hat{y}^{obs}$ *replaces* prior test scores EOG06, rather than supplementing it. That is, we fit an additional HLM that includes $\hat{y}^{obs}$ as a predictor but excludes EOG06. Since both $\hat{y}^{obs}$ and EOG06 are predictors of EOG08, including both in the model may lead to collinearity and inflate standard errors.

5. RESULTS

5.1 Predicting 8th Grade Scores from Observational Data

The random forest accurately predicted EOG08 scores: the out-of-bag root mean squared error was 2.1 points, corresponding to a pseudo-R^2 of roughly 93%. By comparison, an analogous linear regression model had a root-mean-square error of 2.6.

Figure 1 shows a variable importance measure for the top 25 most important predictors in the random forest. In this case, a predictor's importance is the average decrease in mean squared error across splits based on that predictor in the component trees of the random forest. The values in Figure

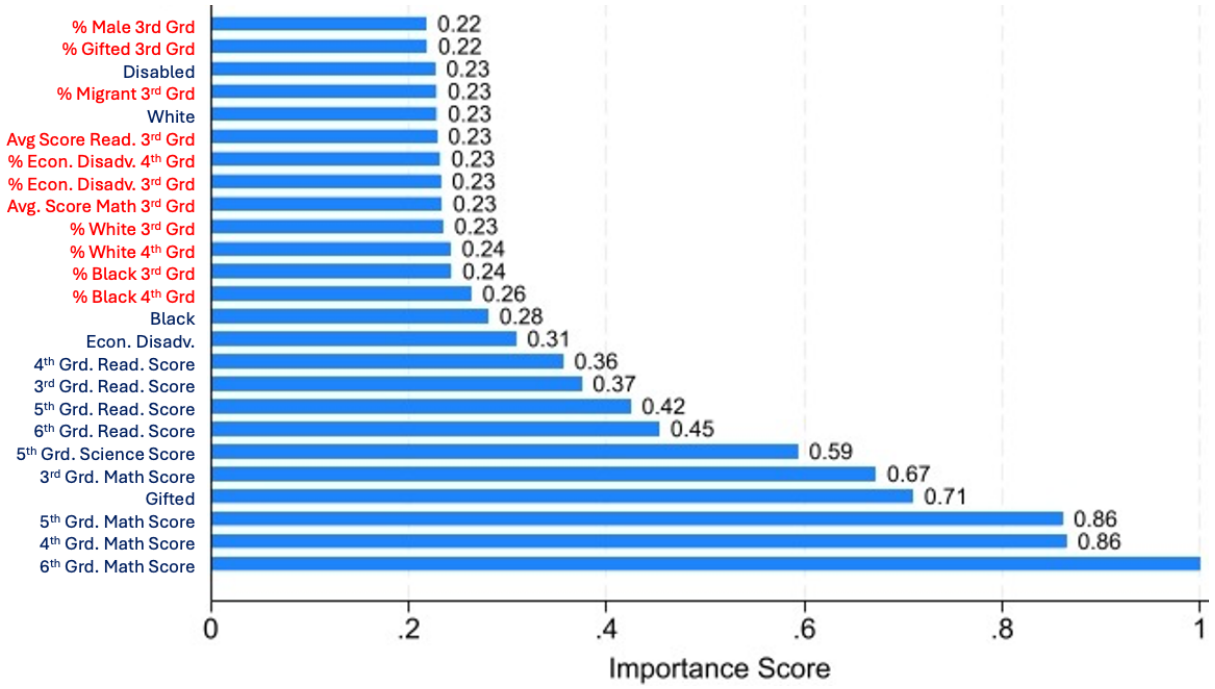


Figure 1: Variable importance for the random forest trained on observational data. Variables in red are school-level aggregates; those in dark blue are student-level. Variable importance is the mean reduction in mean squared error due to a split on that variable, rescaled to 0–1.

1 are divided by the importance measure for the most important predictor, 6th grade math score, so that they range from 0–1 [21].

The three most important predictors, by this measure, are the students' three prior math standardized test scores, followed by previous reading scores and other student-level variables. Most of the remaining top 25 variables are school-level aggregates, including school demographics and prior average exam scores. Notably, only two of the top ten and six of the top 25 predictors appeared in the original HLM (the 6th-grade math score and individual demographics), suggesting that the random forest identified important predictors previously underappreciated. Specifically, students' prior math scores from 4th and 5th grade and their prior reading scores seem to play an important role.

5.2 Estimating Effects

Table 1 and Figure 2 show estimated effects from the original model from [3] alongside two modified models: with $\hat{y}^{obs}$ as an additional predictor and with $\hat{y}^{obs}$ replacing students' prior scores, EOG06.

Including $\hat{y}^{obs}$ as an additional predictor decreased the standard error by 15%, from 0.317 to 0.270. Since standard error tends to be inversely proportional to the square root of the sample size, decreasing the standard error by this amount is roughly equivalent to increasing the sample size by 38%, a substantial improvement.

If $\hat{y}^{obs}$ replaces EOG06 in the HLM (i.e., if EOG06 is dropped from the model), the standard error decreases even more, to 0.265, a decrease of over 16%, corresponding to a 44% increase in effective sample size.

Table 1: Estimated effects, standard errors, effect sizes (Hedge's g), and p-values for three HLMs: the original as reported in [3], one with $\hat{y}^{obs}$ added as a predictor, and one with $\hat{y}^{obs}$ replacing EOG06.

Model	Eff.	SE	Hedges' g	p
Orig.	0.80	0.317	0.10	0.012
Orig.+$\hat{y}^{obs}$	0.68	0.270	0.08	0.011
Orig.+$\hat{y}^{obs}$-EOG06	0.68	0.265	0.08	0.011

6. DISCUSSION

In re-analyzing data from a recent large educational efficacy study, this paper described a method for incorporating data from across the state, including students from both randomized and non-randomized schools. The method combines insights from recent statistical literature [4] with more conventional hierarchical linear modeling, and captures many of the advantages of both: it uses observational data to increase statistical power and precision without introducing confounding bias, it uses familiar and flexible methods that are accessible to applied education researchers, and it fully models the hierarchical structure of the data. In the analysis presented here, incorporating observational data was equivalent to increasing the sample size by over a third.

The results presented here represent the beginning of our reanalysis of the North Carolina data. Future research will apply the method described here to estimate the effect of randomization on the more advanced "end of course" exam,

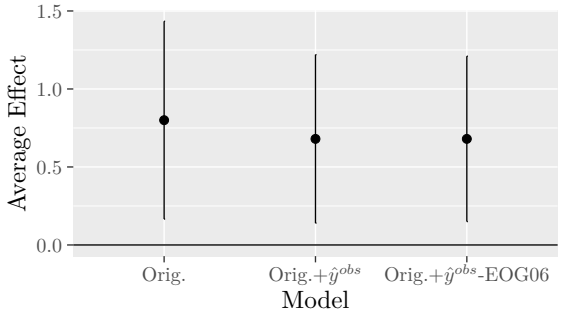


Figure 2: Point estimates and 95% confidence intervals from three HLMs for the average treatment effect in the North Carolina Assistments efficacy study.

as well as on students' choice to take one exam or the other—did assignment to the treatment condition cause some students to take the more (or less) advanced 8th grade class? We also hope to delve deeper into the random forest model discussed in Section 5.1 to better understand its superiority to more parsimonious linear models. Finally, we hope to use the method discussed here to estimate subgroup effects; hopefully, the additional power will enable the investigation of smaller subgroups.

In general, we hope that future research will apply this method to other datasets to better understand how often and by how much incorporating auxiliary data can improve statistical power. Ideally, the availability of auxiliary data can help researchers conduct smaller experiments—first, they will need a tool to prospectively estimate the power gains from auxiliary data, perhaps akin to [7].

Overall, the larger the (absolute) correlation between $\hat{y}^{obs}$ and the outcomes Y in the RCT sample, the greater the resulting gains in precision will be. In the worst case, when the correlation is low or zero, incorporating $\hat{y}^{obs}$ into a causal estimator may increase standard errors slightly—like adding any noise covariate to the model—and will not cause bias.

These considerations imply that choosing a better-performing predictive model can lead to more precision gains. In this study, we used the random forest, which has shown competitive performance in previous research [10], and is relatively easy to implement, but a better model would have led to even greater precision gains. Fortunately, auxiliary data can be used to both train and select the predictive model: since it is a completely separate dataset from the RCT, it can serve as a sandbox for attempting various predictive models and choosing between them. However, the model's performance in the RCT sample, where it counts, may be substantially lower than in the auxiliary data; we are actively researching and developing transfer learning models that may mitigate this issue.

Since precision gains are driven by the correlation between $\hat{y}^{obs}$ and Y, and not predictive accuracy, our approach, properly tweaked, may work even if the outcome in the auxiliary data is not the same as in the RCT, but is correlated. Our first attempts in this direction were not encouraging [18], but the work is ongoing.

The method described here is an example of data fusion, a growing interest among quantitative researchers in integrating observational and randomized data [1, 16]. These include methods for extrapolating RCT results to other populations, identifying important moderators or mediators, as well as methods, like the one discussed here, for increasing precision and power. Data fusion is an exciting new direction in statistical causal inference, and it is especially relevant to educational data mining. We hope that as more methods and datasets become available, incorporating randomized and non-randomized data into unified analyses will accelerate the science of learning.

7. ACKNOWLEDGMENTS

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305D210031. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education.

The authors are grateful for helpful comments from Charlotte Z. Mann, Jaylin Lowe, Yanping Pei, three anonymous reviewers, and a metareviewer. The authors are also grateful to Neil Heffernan for all sorts of things, including, but not limited to, ideas, encouragement, and string-pulling—and, by extension, Angela Kao, without whom none of this would have gotten done.

8. REFERENCES

[1] E. Bareinboim and J. Pearl. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113(27):7345–7352, 2016.

[2] L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[3] M. Feng, N. Brezack, C. Huang, and K. Collins. Long-term effects of an online math tool on us adolescents' achievement. *British Journal of Educational Technology*, 2025.

[4] J. A. Gagnon-Bartsch, A. C. Sales, E. Wu, A. F. Botelho, J. A. Erickson, L. W. Miratrix, and N. T. Heffernan. Precise unbiased estimation in randomized experiments using auxiliary observational data. *Journal of Causal Inference*, 11(1):20220011, Jan. 2023. Publisher: De Gruyter.

[5] L. V. Hedges and E. C. Hedberg. Intraclass correlation values for planning group-randomized trials in education. *Educational Evaluation and Policy Analysis*, 29(1):60–87, 2007.

[6] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[7] J. Lowe, C. Z. Mann, J. Wang, A. Sales, and J. A. Gagnon-Bartsch. Power calculations for randomized controlled trials with auxiliary observational data. In B. PaaÃŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 469–475, Atlanta, Georgia, USA, July

2024. International Educational Data Mining Society.

[8] C. Mann, E. Wu, A. Sales, and J. Gagnon-Bartsch. *dRCT: Design-Based Covariate Adjustment for the Average Treatment Effect*, 2025. R package version 1.0.0, commit d423d3de8f3befe78328d6bbd124916ecf7c7195.

[9] C. Z. Mann, A. C. Sales, and J. A. Gagnon-Bartsch. A general framework for design-based treatment effect estimation in paired cluster-randomized experiments. *arXiv preprint arXiv:2407.01765*, 2024.

[10] C. Z. Mann, J. Wang, A. Sales, and J. A. Gagnon-Bartsch. Using publicly available auxiliary data to improve precision of treatment effect estimation in a randomized efficacy trial. In B. PaaÃŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 518–525, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[11] J. F. Pane, B. A. Griffin, D. F. McCaffrey, and R. Karam. Effectiveness of Cognitive Tutor Algebra I at Scale. *Educational Evaluation and Policy Analysis*, 36(2):127–144, 2014. Publisher: [American Educational Research Association, Sage Publications, Inc.].

[12] T. Patikorn, D. Selent, N. T. Heffernan, J. E. Beck, and J. Zou. Using a single model trained across multiple experiments to improve the detection of treatment effects. *International Educational Data Mining Society*, 2017.

[13] S. Raudenbush and A. Bryk. *Hierarchical linear models: Applications and data analysis methods*. Sage, Chicago, 2002.

[14] J. Roschelle, M. Feng, R. F. Murphy, and C. A. Mason. Online mathematics homework increases student achievement. *AERA Open*, 2(4):2332858416673968, 2016.

[15] J. Roschelle, M. Feng, R. F. Murphy, and C. A. Mason. Online mathematics homework increases student achievement. *AERA open*, 2(4):2332858416673968, 2016.

[16] E. T. Rosenman. Methods for combining observational and experimental causal estimates: A review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 17(2):e70027, 2025.

[17] A. C. Sales, A. Botelho, T. M. Patikorn, and N. T. Heffernan. Using big data to sharpen design-based inference in a/b tests. In K. E. Boyer and M. Yudelson, editors, *Proceedings of the 11th International Conference on Educational Data Mining. (EDM 2018)*, pages 479–486. International Educational Data Mining Society, 2018.

[18] A. C. Sales, C. Mann, J. Gagnon-Bartsch, and N. Heffernan. Effect estimates using publicly available school- level data in a cluster-randomized educational experiment. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 578–581. International Educational Data Mining Society, July 2025.

[19] A. C. Sales, E. B. Prihar, J. A. Gagnon-Bartsch, and N. T. Heffernan. Using auxiliary data to boost precision in the analysis of a/b tests on an online educational platform: New data and new results. *Journal of Educational Data Mining*, 15(2):53–85, 2023.

[20] A. C. Sales, E. B. Prihar, J. A. Gagnon-Bartsch, N. T. Heffernan, et al. Using auxiliary data to boost precision in the analysis of a/b tests on an online educational platform: New data and new results. *Journal of Educational Data Mining*, 15(2):53–85, 2023.

[21] M. Schonlau and R. Y. Zou. The random forest algorithm for statistical learning. *The Stata Journal*, 20(1):3–29, 2020.

[22] A. Siedahmed, Y. Pei, A. C. Sales, N. T. Heffernan, J. Gagnon-Bartsch, D. Zhang, B. A. Schuetze, and A. Zengilowski. E-trials: Empowering data-driven decisions to enhance computer-based learning platforms. *arXiv preprint arXiv:2502.10545*, 2025.

[23] E. Wu and J. A. Gagnon-Bartsch. The loop estimator: Adjusting for covariates in randomized experiments. *Evaluation review*, 42(4):458–488, 2018.

[24] E. Wu and J. A. Gagnon-Bartsch. Design-based covariate adjustments in paired experiments. *Journal of Educational and Behavioral Statistics*, 46(1):109–132, 2021.

Improving Speech Recognition of Named Entities in Classroom Speech with LLM Revision and Phonetic-Semantic Context

Viet Anh Trinh
Worcester Polytechnic Institute
anhtv1@gmail.com

Xinlu He
Worcester Polytechnic Institute
xhe4@wpi.edu

Jacob Whitehill
Worcester Polytechnic Institute
jrwhitehill@wpi.edu

ABSTRACT

Classroom speech and lectures often contain named entities (NEs) such as names of people and special terminology. While automatic speech recognition (ASR) systems have achieved remarkable performance on general speech, the word error rate (WER) of state-of-the-art ASR remains high for named entities. Since NE are often the most critical keywords, misrecognizing them can affect all downstream applications, especially when the ASR functions as the front end of a complex system. In this paper, we introduce a large language model (LLM) revision pipeline to revise incorrect NEs in ASR predictions by leveraging not only the LLM's world knowledge and reasoning ability but also the available phonetic and semantic context. We also introduce the NER-MIT-OpenCourseWare dataset, containing 45 hours of data from MIT courses for development and testing. On this dataset, our proposed technique achieves up to 30% relative WER reduction for NEs.

Keywords

classroom speech analysis, contextual ASR, named entity transcription, LLM revision

1. INTRODUCTION

In the past several years, automatic speech recognition (ASR) performance has improved significantly by scaling data and model size. Whisper [24], which was trained on 765K hours of speech, has achieved remarkable performance, achieving a WER of 2.7% on the standard Librispeech clean corpus and an average of 12.8% across popular speech datasets [24]. However, Whisper (large-v3) still falls short in recognizing named entities (NEs). For instance, it has a WER of 28.9% for persons and 19.6% for organizations in the contextual ASR benchmarking dataset ConEC [12]. Since classroom and academic speech often contains NEs, the inability to recognize them correctly can cause downstream problems in various educational applications, such as topic modeling of curricula [25, 28, 2], interactions with real-time educational agents [9, 16], and more.

Humans, on the other hand, have the ability to reason and relate different NEs based on knowledge and local context. For example, students can use lecture slides to map the NEs in the speech from a lecture video to the content in the slides, using both semantic as well as phonetic information. Since NEs relate to events, persons' names, products, etc. which often depend on local context and change over time, the solution of retraining a model with new paired speech-text data is expensive, and hard to scale.

Contributions: (1) We propose a novel approach (see Fig. 1) that utilizes an LLM to revise incorrect NEs in ASR transcripts. Specifically, the LLM receives predictions from the ASR and filtered information from the local context, which contains similar-sounding NEs and sentences that include these words, and uses this information to revise the ASR predictions. In contrast to prior works on LLM revision of ASR, our method harnesses combines LLM-based reasoning and world knowledge with explicit phonetic and semantic context, without needing to be trained on a specific NE list. The method is interpretable because the LLM explains why it makes certain revisions, unlike other attention-based approaches that only output the revision without any explanation. (2) We demonstrate that our proposed method works effectively with different LLMs, ranging from closed-source to open-source models and across various sizes. (3) Finally, we introduce the MIT-OpenCourseWare NER contextual dataset, which includes both speech-text pairs and local contexts[1].

2. RELATED WORK

Fine-tuning Speech Foundation Models for Classroom Speech: The past few years have seen growing interest in fine-tuning speech foundation models (e.g., wav2vec2 [4] and WavLM [8]) for classroom-specific recording conditions [27, 3]. While this approach can increase the average transcription accuracy, it does not focus on named entities in particular. Even though NEs may occur more frequently in classroom speech than in everyday speech, they still make up only a small fraction of the overall audio recording; hence, the ability of fine-tuning to improve NE recognition performance is limited. In contrast, our proposed method is "zero-shot" in the sense that it uses an pre-trained LLM's ability to to fix errors, given enough phonetic and semantic context, without needing a large dataset for parameter tuning.

[1] https://huggingface.co/datasets/lucille0he/ocw

Viet Anh Trinh, Xinlu He, and Jacob Whitehill. Improving Speech Recognition of Named Entities in Classroom Speech with LLM Revision and Phonetic-Semantic Context. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 494–499. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039931

Biasing: A common way to improve NE recognition is by biasing the ASR toward a list of predefined words. This can be implemented with shallow fusion [13, 11, 30, 18] whereby, during inference, the ASR prediction is influenced by a contextual language model with a certain weight. In practice, such systems are sensitive to the weight parameter. More powerful are biasing approaches that combine the context and acoustic information more tightly, e.g., with a neural network that encodes contextual information and is trained jointly with the ASR system [23, 7, 26, 10]. One of the difficulties in biasing approaches is that performance often decreases as the number of approved NEs grows. Our proposed solution addresses this issue by selecting only the NEs in the context that sound similar to those in the ASR prediction.

LLM-based ASR revision: During the past two years, there has been significant interest in developing LLM-based methods to revise ASR hypotheses [20]. For instance, [22] use language model rescoring followed by an LLM in zero-shot fashion to revise the ASR hypothesis based on a ranked list of most likely candidates. [17] also use an LLM to revise the ASR hypotheses based on the most likely candidates but additionally impose constraints so that the revised prediction is similar to the original one in terms of Levenshtein distance. Neither of these methods harness semantic context or phonetic similarity. Most similar to ours is work by [29], which combines phonetic and contextual information to revise NEs in ASR transcriptions. However, in contrast to our method, which is training-free and generalizable to different vocabularies, their approach requires training a masked language model on a fixed set of NEs to detect NE mistranscriptions.

3. PROPOSED METHOD: PHONETIC & CONTEXTUAL LLM REVISION

We propose a novel method of using an LLM's reasoning abilities to revise ASR predictions by harnessing *contextual* as well as *phonetic* information on named entities in both the speech and the available context. The method consists of several steps. First, (1) speech is transcribed using an ASR model (e.g., Whisper). Next, (2) NEs are extracted from both the ASR predictions and the context documents using an off-the-shelf NE recognition tool. The context can be any text document related to the speech. For instance, when transcribing lectures, the context might include slides or reading lists. Due to the large number of words in the context, (3) the NEs are selected from the context only if they share the same entity type (e.g., person, organization) and sound similar to the NEs in the ASR predictions. Finally (4), an LLM analyzes the ASR predictions and their NEs, as well as the NEs in the context and the sentences containing them. It uses this information and a detailed prompt of instructions (see Fig. 3) to produce a revised prediction.

4. DATASET

As existing datasets for contextual ASR are very limited and often tailored to specific industry settings (e.g., earnings calls), we collected our own dataset on topics of wider scientific and educational interest. In particular, we constructed a new dataset, called NER-MIT-OpenCourseWare, based on content from MIT OpenCourseWare. We chose this educational repository as a basis for our dataset because of the availability of detailed context documents along with high-quality transcriptions of the speech content.

For the test set, we used the lectures from an undergraduate course on animal behavior, offered in Fall 2013 from MIT OpenCourseWare, containing 25 lecture audio recordings totaling 17 hours of audio. The test set labels contain approximately 121K words, while the context consists of 46K words. The ground-truth transcriptions contain 1.2K NE words, and about 66% of these words appear in the slides. Examples of lecture slides are shown in Figure 2.

For the development set, we chose an undergraduate course about brain structure, offered in Spring 2014. The course has 35 lectures, but we only use the first 3K utterances to enable faster execution. The labels contain approximately 32K words, while the context includes 14K words from six lecture slides. There are 0.2K NE words in the development set, and around 60% of these words appear in the slides.

We used BeautifulSoup[2] to download the audio, video, and lecture content; pdfplumber[3] to convert the ground-truth transcripts and the slides into text files; and spacy (specifically "en_core_web_lg") to remove unwanted characters and split the transcripts into sentences. The slides contain images, math equations, and other elements, making them noisy. We used pydub[4] to extract audio (flac) from video (mp4) and convert the audio to mono with a sampling rate of 16KHz. Since each audio file has a duration of 30-60 min, we used fairseq [19] to segment each audio into utterances corresponding to the sentences identified by the spacy model.

5. EXPERIMENTS

Using NER-MIT-OpenCourseWare (Sec. 4), we conducted experiments to assess the benefits of the proposed method (Fig. 1) to improve ASR transcription accuracy of NEs, while hopefully keeping the transcription accuracy of non-named entities (NNEs) unchanged. As our method focuses on filtering relevant information in the context using contextual and phonetic cues before feeding it to the LLM, we compare against three strong control conditions: (1) no LLM, just phonetic matching of the predicted NE with a randomly chosen similar sounding word of the same entity type from the context; (2) giving the revision LLM the entire context document corresponding to a given utterance (not just the filtered relevant entities); and (3) first using an LLM to summarize the context, and then providing the summary to the revision LLM.

5.1 Implementation Details

For the first-stage ASR, we use Whisper large-v3. We choose Flair [1] ("ner-english-ontonotes-large"), for named entity recognition (NER) due to its speed and high accuracy on standard NER benchmarks. To detect whether two words sound similar to each other, we use Double Metaphone [21], which is a popular rule-based algorithm that converts words into phonetic codes that is useful for detecting similar sounding words and name searches [5]. Specifically, Double Meta-

[2] `https://www.crummy.com/software/BeautifulSoup`
[3] `https://github.com/jsvine/pdfplumber`
[4] `https://github.com/jiaaro/pydub`

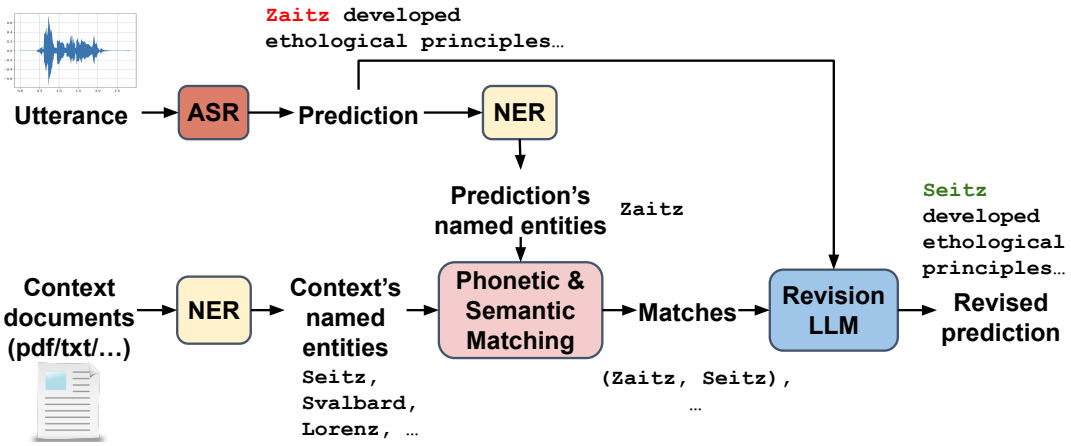


Figure 1: Proposed method of combining phonetic and contextual information to revise named entities in ASR predictions.

Table 1: Comparison of named entity (NE) word error rates (WER) (%) and non-named entity (NNE) WER (%) across methods using Whisper-large-v3 as the ASR.

Revision Method	LLM	NE	NNE
None	–	32.3	7.0
Phonetic matching	–	25.3	7.0
Full context	GPT-4o-mini	43.4	7.1
	GPT-4o	30.6	7.0
	Llama-3.1-70B	32.6	7.2
Context summary	GPT-4o-mini	38.6	7.1
	GPT-4o	54.2	7.1
	Llama-3.1-70B	47.4	8.2
Proposed method	GPT-4o-mini	22.7	7.0
	GPT-4o	22.9	7.0
	Llama-3.1-70B	24.8	7.2

phone outputs one primary and one alternative phonetic code (for names that may be pronounced in different ways). In our proposed method, we defined that two words sound "similar" to each other if either their primary phonetic codes or equivalent or their alternative phonetic codes are equivalent. We note that the use of Double Metaphone could conceivably be "swapped out" for another phonetic matching system (e.g., one based on multimodal LLMs with shared semantic and acoustic embeddings) to work for other languages. We leave this to future work.

Similar to prior contextual ASR research [12], we focus on eight specific NE types: PERSON, ORG, GPE, LOC, PRODUCT, EVENT, NORP, and FAC. We manually verify the NEs in 500 sentences from the test set and measure set-based evaluation recall, ignoring multiple occurrences of the same NE in a sentence, as the LLM replaces all instances of the same NE in the sentence. The precision of Flair is 96%, and the recall is 86%. Since manually labeling NEs in the dataset is expensive, and Flair's recall rate was reasonably high, we use Flair's detected NEs as a basis for computing WER.

Table 2: Comparison of named entity (NE) and non-named entity (NNE) WER (%) across speech recognition systems utilizing GPT-4o-mini in our proposed method.

ASR	Params	Before revision		After revision	
		NE	NNE	NE	NNE
Whisper-sm	244 M	39.3	9.4	29.5	9.4
Whisper-me	769 M	36.0	8.0	26.8	8.0
Whisper-la	1.54 B	32.3	7.0	22.7	7.0
Canary-1B	1B	36.6	8.3	26.8	8.3

For the revision LLM, we compare Llama3-70b-8192 (via Groq API), GPT-4o-mini API, GPT-4o API , and Meta-Llama-3.1-70B (using bitsandbytes quantization), running locally on a single A100 GPU (80GB) with vLLM [15]. We draw inspiration from [6] to design the prompt. Our prompt (see Fig. 3) is designed to ensure that the LLM revises NEs without altering other words and applies reasoning. The asr_confidence_threshold used in the prompt was optimized on the development set.

For the summarization task (in control condition 2), we use GPT-4o-mini with the prompt: "Summarize the following context concisely while ensuring all named entities are preserved. Include a sentence for each named entity, using only alphabets and digits. Context:{context}".

5.2 Results

As shown in Table 1, the proposed method demonstrates improved accuracy compared to the original Whisper predictions, regardless of which LLM was used. In particular, the proposed approach reduces the WER of NE to 22.7% (corresponding to a 30% relative WER reduction) while maintaining the NNE WER at 7%. Phonetic matching of NEs to the context (control condition 1) is also useful and decreases the WER to 25.3%. In contrast, when using an LLM with the full context (control condition 2) without any filtering, the WER actually increases from 32.3% to 43.4%. This could be due to the long prompt input, which causes the LLM to hallucinate: a follow-up analysis of the animal behavior course test set revealed that the full-context condition

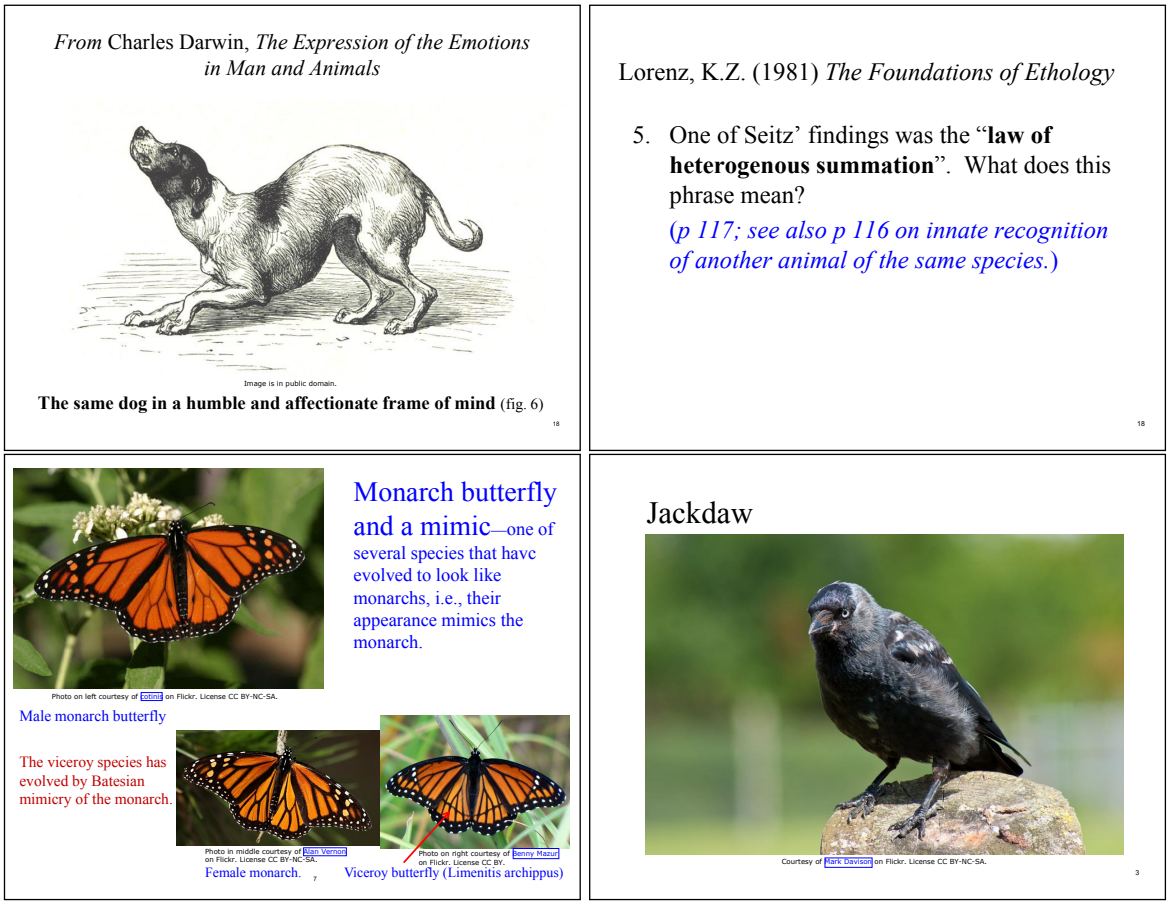


Figure 2: Examples of NER-MIT-OpenCourseWare slides.

using GPT-4o resulted in statistically significantly longer revisions, on average, compared to the proposed method ($t(9426) = 4.11, p < 0.0001$). Control condition 3, which uses a summary of the full context, is also ineffective, as the WER increases from 32.3% to 38.6%. Together these results indicate that, despite being trained on large amounts of text data, the LLM's world knowledge and linguistic reasoning are insufficient to improve the ASR accuracy of NEs. Instead, both a filtered context as well as explicit phonetic information are necessary to enable the LLM to improve NE transcription.

As a follow-up analysis, we also computed the WER of the proposed method (using GPT-4o-mini and Whisper) on named entities (NEs) not included in the context documents: it is 23.6%. This is only slightly higher than the overall WER on NEs (22.7%), and it is still substantially lower than the WER on NEs with just Whisper alone (32.3%). This suggests that the proposed method is unlikely to do harm, even if its benefits are more pronounced on NEs that are contained in the context.

Our proposed method also demonstrates its effectiveness across different LLM models, such as the large, state-of-the-art GPT-4o and the robust open-source Llama-3.1-70B, as shown in Table 1. For a powerful model like GPT-4o, utilizing the full context results in a slight WER reduction, from 32.3% to 30.6%, as indicated in Table 1. However, with our proposed method, the WER is significantly reduced to 22.9%.

To evaluate the effectiveness of the proposed approach over different ASR systems, we conducted experiments with Whisper (small, medium, or large-v3), and Nemo Canary-1B [14], another state-of-the-art ASR model. As shown in Table 2, our method generalizes well across different speech models: With Canary-1B, the NE WER is reduced from 36.6% to 26.8%. Similarly, the NE WER is reduced from 39.3% to 29.5% with Whisper small and from 36.0% to 26.8% with Whisper medium.

Finally, we also experiment with a simpler version of the prompt that omits the extra guidelines contained in the second, third, fourth, and fifth bullet points of Figure 3. This shortened prompt's results are reported in Table 3. As shown, both the full prompt and the short prompt perform well with GPT-4o-mini, while the open-source Llama-3.1-70B achieves better performance with the full prompt at 24.8% compared to the short prompt at 32.3%. This indicates that the paid API GPT-4o-mini may be more robust and have better reasoning capabilities than the open-source Llama-3.1-70B in this case. The open-source model appears to require more detailed instructions to achieve good results, whereas GPT-4o-mini can perform well without them.

Speech recognition may incorrectly capture named entities due to similar-sounding words or uncommon entities in training data. The named entities and their probabilities are provided below as "Named entities probability" in the format: entity : probability. Review the prediction's named entities. For each entity whose probability is below asr_confidence_threshold, or missing, please complete the following steps:

- Examine the context to determine what the similar-sounding entity refers to, and analyze the sentence to understand what the ASR-predicted entity refers to. Check if (1) both refer to the same thing, (2) the difference is likely due to an ASR transcription error, (3) the replacement improves accuracy, preserves the original sentence's meaning, and fits naturally within the sentence. If yes to those 3 criteria, then replace the ASR-predicted entity with the similar-sounding entity.
- Make sure to treat each named entity as a cohesive unit when evaluating or replacing. Do not modify individual words separately in a multi-word entity.
- If no suitable similar-sounding entity exists, leave the entity unchanged.
- Do not delete any words in the entity that are absent from the context.
- Keep all other words, punctuation, and formatting unchanged.

The context includes similar-sounding entities and sentences containing these named entities. The revised prediction should start with << @ and end with @ >>
Context: {context}
Speech recognition prediction: {preds}
Named entities probability: {ent_prob_dict }

Figure 3: ASR revision prompt given to the LLM. In an ablation analysis, we omit the second, third, fourth, and fifth bullets points.

5.3 Examples

The ground-truth transcript (converted to lower case) of one utterance is, "now this is a difficult concept but let me try to go through it anyway because seitz was very important in working out some of the principles of ethological investigation and the principles of ethology that lorenz was talking about". Canary 1B transcribed it as, "now this is a difficult concept ... zeitz ... lawrence ..." (where the ellipses indicate correctly transcribed words that we omit for brevity). Whisper transcribed it as, "now this is a difficult concept... zaitz ... lawrence ...". After LLM-based revision with the proposed method, both Canary and Whisper correctly transcribed it as, "now this is a difficult concept ... seitz ... lorenz ...".

In another utterance, the ground-truth transcript is, "one of those students became very well known margaret mead". With phonetic matching (control condition 1), the NE was replaced with a randomly chosen similar-sounding NE, resulting in "... margaret mit". With our proposed method (using GPT-4o-mini), however, the ASR transcript is correctly transcribed as, "... margaret mead".

5.4 Run-time Cost

Extracting NEs takes about 0.013sec per utterance (averaged over 9427 test utterances). Revising each ASR prediction using GPT-4o-mini takes about 1.2sec (averaged over 750 test utterances with NEs).

Table 3: Named entity WER (%) and non-named entity WER are evaluated using the full prompt or the short prompt.

Method	Revision LLM	NE (%)	NNE
Full prompt	GPT-4o-mini	22.7	7.0
Full prompt	Llama-3.1-70B	24.8	7.2
Short prompt	GPT-4o-mini	22.7	7.0
Short prompt	Llama-3.1-70B	32.3	7.0

6. CONCLUSIONS

We introduced a novel method to revise predictions from ASR using an LLM. Specifically, we employ a phonetic and contextual matching component to select NEs from the context that are most relevant to those in the ASR prediction. We compare our proposed method to strong LLM controls that use the full context or a context summary, as well as a phonetic matching baseline, and show that our method achieves superior performance – up to 30% relative WER reduction on a new and challenging test set based on MIT OpenCourseWare lecture series. We focused primarily on NEs since their definitions are standardized, and tools for NER are readily available. However, the method could be extended to revise rare words with the development of rare word detection models. Furthermore, the method could be expanded to integrate visual information, such as image captions or optical character recognition, so as to better harness available context.

Acknowledgement

This research was supported by the NSF National AI Institute for Student-AI Teaming (iSAT) under grants DRL #2019805 and DRL #2454151, as well as from an NSF CAREER grant #2046505. The opinions expressed are those of the authors and do not represent views of the NSF.

7. REFERENCES

[1] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *NAACL*, 2019.

[2] T. Amjad, Z. Shaheen, and A. Daud. Advanced learning analytics: aspect based course feedback analysis of mooc forums to facilitate instructors. *IEEE Transactions on Computational Social Systems*, 11(4):4698–4706, 2022.

[3] A. A. Attia, D. Demszky, T. Ogunremi, J. Liu, and C. Espy-Wilson. Continued pretraining for domain adaptation of wav2vec2. 0 in automatic speech recognition for elementary math classroom settings. *arXiv preprint arXiv:2405.13018*, 2024.

[4] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[5] P. Blair and K. Bar. Jrc-names-retrieval: A standardized benchmark for name search. In *LREC-COLING*, 2024.

[6] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam,

G. Sastry, A. Askell, et al. Language models are few-shot learners. *Neural information processing systems*, 2020.

[7] F.-J. Chang, J. Liu, M. Radfar, A. Mouchtaris, M. Omologo, A. Rastrow, and S. Kunzmann. Context-aware transformer transducer for speech recognition. In *ASRU*, 2021.

[8] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, 2022.

[9] S. K. D'Mello, Q. Biddy, T. Breideband, J. Bush, M. Chang, A. Cortez, J. Flanigan, P. W. Foltz, J. C. Gorman, L. Hirshfield, et al. From learning optimization to learner flourishing: Reimagining ai in education at the institute for student-ai teaming (isat). *AI Magazine*, 45(1):61–68, 2024.

[10] X. Gong, Y. Wu, J. Li, S. Liu, R. Zhao, X. Chen, and Y. Qian. Advanced long-content speech recognition with factorized neural transducer. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[11] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, et al. Streaming end-to-end speech recognition for mobile devices. In *ICASSP*, 2019.

[12] R. Huang, M. Yarmohammadi, J. Trmal, J. Liu, D. Raj, L. P. Garcia, A. V. Ivanov, P. Ehlen, M. Yu, D. Povey, et al. Conec: Earnings call dataset with real-world contexts for benchmarking contextual speech recognition. In *LREC-COLING*, 2024.

[13] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar. An analysis of incorporating an external language model into a sequence-to-sequence model. In *ICASSP*, 2018.

[14] O. Kuchaiev, J. Li, H. Nguyen, O. Hrinchuk, R. Leary, B. Ginsburg, S. Kriman, S. Beliaev, V. Lavrukhin, J. Cook, et al. Nemo: a toolkit for building ai applications using neural modules. *arXiv preprint arXiv:1909.09577*, 2019.

[15] W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica. Efficient memory management for large language model serving with pagedattention. In *Symposium on Operating Systems Principles*, 2023.

[16] G.-G. Lee, S. Mun, M.-K. Shin, and X. Zhai. Collaborative learning with artificial intelligence speakers: pre-service elementary science teachers' responses to the prototype. *Science & Education*, 34(2):847–875, 2025.

[17] R. Ma, M. Qian, M. Gales, and K. Knill. Asr error correction using large language models. *IEEE Transactions on Audio, Speech and Language Processing*, 33:1389–1401, 2025.

[18] M. Mohri, F. Pereira, and M. Riley. Weighted finite-state transducers in speech recognition. *Computer Speech & Language*, 2002.

[19] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *NAACL-HLT Demonstrations*, 2019.

[20] M. R. Peyghan, F. Rajabi, S. S. Roudi, S. Zouashkiani, S. Amini, and S. Ghaemmaghami. A survey on non-intrusive asr refinement: From output-level correction to full-model distillation. *arXiv preprint arXiv:2508.07285*, 2025.

[21] L. Philips. The double metaphone search algorithm. *C/C++ Users Journal*, 2000.

[22] J. Pu, T.-S. Nguyen, and S. Stüker. Multi-stage large language model correction for speech recognition. *arXiv preprint arXiv:2310.11532*, 2023.

[23] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao. Deep context: end-to-end contextual speech recognition. In *Spoken language technology workshop (SLT)*, 2018.

[24] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, 2023.

[25] J. M. Rouly, H. Rangwala, and A. Johri. What are we teaching? automated evaluation of cs curricula content using topic modeling. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pages 189–197, 2015.

[26] K. M. Sathyendra, T. Muniyappa, F.-J. Chang, J. Liu, J. Su, G. P. Strimel, A. Mouchtaris, and S. Kunzmann. Contextual adapters for personalized speech recognition in neural transducers. In *ICASSP*, 2022.

[27] R. Southwell, W. Ward, V. A. Trinh, C. Clevenger, C. Clevenger, E. Watts, J. Reitman, S. D'Mello, and J. Whitehill. Automatic speech recognition tuned for child speech in the classroom. In *Icassp 2024-2024 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 12291–12295. IEEE, 2024.

[28] P. A. Takizawa. Using a topic model to map and analyze a large curriculum. *Plos one*, 18(4):e0284513, 2023.

[29] Y.-C. Wang, H.-W. Wang, B.-C. Yan, C.-H. Lin, and B. Chen. DANCER: Entity description augmented named entity corrector for automatic speech recognition. In *LREC-COLING*, 2024.

[30] D. Zhao, T. N. Sainath, D. Rybach, P. Rondon, D. Bhatia, B. Li, and R. Pang. Shallow-fusion end-to-end contextual biasing. In *Interspeech*, 2019.

Mining how curiosity spreads in student-led online discussions

Farhan Ali
National Institute of Education,
Nanyang Technological University
farhan.ali@nie.edu.sg

ABSTRACT

Curiosity is central to learning and human development, yet it is typically studied as an individual psychological state. Using a social cognitive lens, this study examines whether and how curious behaviors are transmitted socially in student-led online discussions. We first constructed and validated a new large-scale dataset from a student-driven educational forum on Reddit (full dataset: ~55,000 users, ~39,000 posts; ~667,000 comments). Curiosity was operationalized as question-posing using a 2×2 question type taxonomy involving open/closed and possibility thinking dimensions. Generalized linear models revealed clear social transmission effects: question-posing in comments generally increased questioning in subsequent replies, regardless of question type. Parallel mediation analyses involving two categories of discussion features - cognitive and socio-affective - uncovered pathways of transmission of curiosity involving social endorsement, cognitive alignment, and selective mimicry. These findings advance the field by demonstrating that curiosity is socially transmitted, highlighting its contingent spread, and providing insights into how educators and platform developers can intervene and design features to steer discussion trajectories. Our research further contributes descriptively via a new taxonomy of question-posing and the dissemination of a new online discussion corpus useful for future research.

Keywords

adolescents, curiosity, online discussions, Singapore, Reddit, question-posing

1. INTRODUCTION & BACKGROUND

Curiosity is central to learning yet often treated as an internal state. There is a need to broaden our understanding in order to better foster curious learners. Drawing on social cognitive theory, we examine how curiosity, expressed through question-posing, spreads in peer online discussions where there are rich intersections of formal learning, informal learning, and community inquiry. Using large-scale data from an education-oriented Reddit community, data mining, and quantitative modeling, we investigate curiosity's transmission patterns and mechanisms in authentic student digital environments.

1.1 Social cognitive theory

Social learning theory conceptualizes learning as inherently social, shaped by observing others' behaviors and their consequences. Social cognitive theory further emphasizes cognitive and motivational processes that determine whether observed behaviors are adopted [3, 4]. A central process is the transmission, or diffusion, of thoughts and actions across individuals. The proposed mechanisms include imitation, modeling, and vicarious reinforcement, whereby behaviors perceived as rewarded are more likely to be replicated. Importantly, transmission is thought to be selective: individuals evaluate relevance and value before adopting the observed behaviors. In education, parallel theoretical traditions such as social constructivism highlight knowledge as co-constructed through dialogue and collaboration [20].

1.2 Curiosity

Curiosity - the desire for new information and knowledge [16] - is a foundational driver of learning and development and predicts important outcomes such as academic achievement, well-being, and longevity [22, 24, 27]. Yet contemporary schooling is often argued to dampen curiosity [7], with cross-cultural evidence indeed documenting systematic declines from elementary to secondary years [11]. Understanding how curiosity can be triggered and fostered is, therefore, an important educational challenge.

Most research conceptualizes curiosity as an intrapersonal state. However, curiosity is also expressed behaviorally in social contexts, most visibly through question-posing, a behavior widely studied in curiosity research in education [7, 13]. A question is not merely a private attempt to resolve an information gap; it signals to others what is worth asking, how to ask, and whether such behavior is socially valued. In this sense, curiosity becomes embedded within a social learning ecology, with potential to spread or dissipate depending on peer responses and contextual cues.

While question-posing is often used as an indicator of curiosity, questioning may not all reflect intrinsically motivated curiosity. Drawing on a goal-systems perspective, it is useful to distinguish between curiosity as a motive (ends) and curious behaviors that serve diverse goals (means) [23] such as performance-enhancement or tension-reduction. This distinction is especially relevant in large-scale online data, where inferring internal motivation is difficult without additional contextual or self-report measures. Accordingly, we follow framing by Szumowska & Kruglanski [23] to operationalize curious behaviors (as opposed to curious motivations) with question-posing. Importantly, in our context, questions are voluntary rather than teacher-prompted and non-evaluative with no immediate benefit, making them a meaningful indicator of momentary curious behaviors in authentic peer learning interactions.

Emerging evidence suggests that curiosity may indeed be socially transmissible from human to human in classroom settings [2], and even from robot to human in experimental settings [9]. However, the extent, mechanisms, and boundary conditions of curiosity transmission, particularly in authentic, peer-driven digital environments where many adolescents now regularly interact, remain

Farhan Ali. Mining how curiosity spreads in student-led online discussions. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 500–507. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040018

insufficiently understood. An important question is the nature of the transmission. Transmission may be broad, where exposure increases questioning generally, or narrow, where specific question types are preferentially replicated. Distinguishing these forms clarifies whether curious behaviors spread as generalized activation or subtype-specific processes, informing strategies for interventions or nudging on online platforms or by instructors.

1.3 Educational context: Reddit online discussions

Digital communities such as Reddit provide a valuable setting for examining curiosity in authentic peer interaction. Reddit hosts sustained, community-driven discussions spanning personal, academic, and social topics, leaving behind rich textual traces suitable for large-scale educational research. Prior studies have used Reddit data to illuminate important educational issues, including impact of COVID-19 [26] and the emergence of ChatGPT [18] . The platform's affordances also align well with a social learning perspective: for instance, upvotes function as social reinforcement, threaded replies make interaction paths explicit, and norms such as quoting or referencing can signal uptake and engagement.

Within this Reddit context, curiosity can be operationalized through question-posing, a visible indicator that researchers have emphasized as being important [7, 13]. Building on prior educational research and recent frameworks distinguishing open/closed and possibility-oriented questions[1], we propose crossing these dimensions to derive four nuanced question types, enabling automated analysis of how curiosity is expressed and potentially transmitted in adolescent online discussions.

To understand how curiosity spreads, discussion patterns must be analyzed. Prior work integrating social learning and community-of-inquiry perspectives identified three interdependent features [10]: cognitive presence (collective meaning-making through discourse), social presence (interpersonal and affective expressions), and teaching presence (directed facilitation toward learning goals). In adolescent-driven communities with minimal adult moderation, cognitive and social presence are especially central. This framework highlights how epistemic contributions (e.g., explanations, meaning) and socio-affective signals (e.g., endorsement, emotional tone) may shape peer transmission processes.

Guided by this perspective, we identified an online community using three criteria: predominantly student-driven participation with little to no adult or bot or spam activity; high activity and diverse threads to support robust linguistic analyses; and a focus on schooling-related experiences, not general lifestyle or pop culture chatter. We identified r/SGExams as meeting these criteria. SGExams is Singapore's largest youth-oriented education forum, where students discuss academic subjects, examinations, study strategies, transitions, and peer issues. Its shared national context enhances the likelihood that expressions of curiosity are recognized and taken up by peers, making it well-suited for examining social transmission in authentic digital learning environments.

1.4 Current study

The current study investigated the extent of social transmission of curiosity in online educational discussions. To the extent that social transmission exists, we further asked about the nature of its transmission in terms of specificity and features that mediate these transmission effects. To this end, we analyzed our newly curated, large-scale dataset from an education-related Reddit community, SGExams. Curiosity was operationalized as question-posing using a new 2×2 taxonomy, crossing open/closed and possibility/non-possibility question forms. We analyzed transmission using generalized linear models and quantified mediating pathways using Monte Carlo-based counterfactual contrasts with ten cognitive and socio-affective discussion features modeled.

2. METHOD

2.1 Dataset construction and validation

Institutional Review Board (IRB) approval for retrieving and analyzing publicly-available deidentified data with a waiver of consent was obtained prior to the start of research. We built a custom data collection pipeline using Python Reddit API (PRAW v7) to retrieve posts and comments from SGExams. The dataset was compiled through an iterative, query-based procedure. This iteration was necessary as there are constraints in returns from a single API query. An initial word set (e.g., "exam," "teacher," common function words) was used to generate automated queries and returns. Unique terms were then incorporated to expand subsequent queries and returns. Retrieval continued until saturation, defined as no new returned posts or comments from additional queries (as identified by post/comment ID). The final dataset comprised 38,920 posts and 666,978 comments from January 2019 to October 2023, including metadata (timestamps, identifiers, flairs, etc.). For this paper, we applied two-step preprocessing based on post flairs (topic labels assigned to a discussion thread). Flairs with fewer than 4,000 comments or activity concentrated within a few months were excluded, retaining 12 flairs grouped into five broader topic categories. We also reconstructed the discussion hierarchy by pairing parent posts/comments with their child replies. Note that "parent-child" here refers to the discussion hierarchy for ease of communication, and not familial relationship. Because users generally do not reply to themselves, these parent-child links largely reflect between-user interactions.

To assess completeness, we conducted two validation procedures. First, we restricted automated queries to a one-month window and compared retrieved counts with manual tallies of posts and comments directly on SGExams subreddit website. Second, we cross-validated our dataset with a third-party aggregator of Reddit statistics (https://subredditstats.com/). To verify that contributors were adolescent students, we manually reviewed 200 highly active users (≥10 posts/comments). For each, up to 50 posts and comments were examined for explicit age disclosures (e.g., "I'm 16") and contextual indicators (e.g., references to secondary school levels, examinations, or post-secondary pathways). These checks assessed the dataset's coverage and demographic relevance to the target population.

2.2 Curiosity measures

Curiosity was operationalized as question-posing, aligned with past work on indicators of curiosity [7, 13]. Because questions typically start with question words, we used a rule-based classification at the sentence level. Table 1 presents the full categorization of question types based on the rules and their implied intent and stance. Each sentence was assigned to one and only one category based on the rules. Individual comments were often composed of multiple sentences and thus could contain more than one type of question. We performed a validation of our automated classification by manually labelling 1,807 randomly selected sentences.

2.3 Discussion mediators

We identified and engineered 10 literature-informed mediators capturing shared meaning-making and affective exchange in Reddit discussions. Cognitive mediators indexed epistemic alignment

between parent and child comments (e.g., lexical overlap, conceptual similarity, content uptake), whereas socio-affective mediators reflected social valuation, social referencing, and emotional alignment. Together, they operationalized potential mechanisms for the transmission of curiosity. The mediators were engineered using natural language processing, unsupervised learning as well as a hybrid lexical-semantic mining process that we developed. First, we started with a small dictionary of lexicons (e.g., "you're right,") and embedded them along with the full corpus' n-grams (up to 5) using a pretrained sentence transformer (all-MiniLM-L6-v2). An iterative algorithm identified semantically close and frequent n-grams, followed by manual screening for a final lexical list. This hybrid lexical-semantic mining thus incorporates both lexical, semantic information, and contextual information. This approach contrasts with more standard methods, such as LIWC which typically lacks incorporation of context and is not grounded in actual use frequencies. This method was applied to two mediators of agreement and social referencing. This automated mining was then validated against 1,000 human-labeled sentences. See Table 2 for a detailed exposition of the mediators, measurements, and sources.

Table 1. Question types: description, rule, and examples

Type	Implied intent/stance	Sentence starters	Examples
O-NP: Open, Non-possibility questions	Factual clarification; descriptive or explanatory intent; assumption of a knowable answer	('who', 'when', 'what', 'where', 'how', 'why') AND (NOT ('can', 'may', 'might', 'could', 'would', 'if', 'will'))	"what is the DAE process like" "where is my monohybrid dihybrid epistasis"
O-P: Open, possibility questions	Exploratory reasoning; hypothetical thinking; acknowledgment of multiple possible outcomes	('who', 'when', 'what', 'where', 'how', 'why') AND ('can', 'may', 'might', 'could', 'would', 'if', 'will')	"how might we interpret the data here" "how can i explain and link this to evaluation"
C-NP: Closed, non-possibility questions	Verification, or confirmation; primary expectation of a yes/no answer; settled knowledge assumption	'must', 'should', 'do', 'is', 'are', 'does', 'did', 'have', 'has'	"Is the form for your course only?" "should i go back to sec 2 algebra"
C-P: Closed, possibility questions	Feasibility checking; bounded outcomes	'can', 'may', 'might', 'could', 'would', 'will'	"could it be such that the seats are all full and some are left out" "can it explain your action"

Note. 'if' was not included as sentence starter in the possibility-closed category as we found that sentences starting with 'if' were almost all non-questions.

Table 2. Cognitive and socio-affective discussion mediators to assess pathways for transmission effect

	Description	Measurement	Source
Cognitive			
Target lexicon mimicry	Extent of target lexicon similarity, capturing specific surface-level interactional alignment	Proportion of parent target lexicons in children's target lexicons. Target lexicons refer to the list of words used to classify question types (Table 1)	Combined from parent and child comments
General mimicry	Extent of overall lexicon similarity, indicating general surface-level interactional alignment	Jaccard similarity of content words (i.e., excluding stop words)	Combined from parent and child comments
Semantic similarity	Level of conceptual alignment beyond specific word use, indicating shared meaning and focus	Cosine distance between parent and child comments in the embedding space of a pre-trained sentence transformer model (all-MiniLM-L6-v2, 384 dimensions)	Combined from parent and child comments
Local referencing	Degree of uptake of specific content, reflecting focused engagement	Counts of '>' in children. '>' is the community markdown norm for replying to specific statements/sentences made by the parent comment	Child comments
Agreement	Extent of public concurrence or endorsement, signaling public alignment	Counts of any of the following lexicons as mined by our hybrid lexical-semantic approach: 'me too', 'sameeee','i agree','you're right', 'ur right', 'you right', 'you make sense', 'u make sense'	Child comments
Socio-affective			
Popularity	Level of community endorsement, reflecting a comment's perceived importance and value	Number of upvotes; similar to likes on other social media platforms	Parent comments
Social referencing	Extent of relational references, signaling interpersonal links	Counts of any of the following lexicons as mined by our hybrid lexical-semantic approach: 'you say that', 'u mentioned', 'u say', 'what you say', 'you mentioned', 'you explain', 'youve mentioned', 'saying you', 'you mention'	Child comments
Sentiment - Parent	Affective valence, capturing overall emotional tone	Sentiment analysis using a transformer model fine-tuned for social media sentiment classification into three classes [17]: positive, negative, neutral	Parent comments

	Description	Measurement	Source
Sentiment - Child	Affective valence, capturing overall emotional tone	Same as Sentiment - Parent but for child comments	Child comments
Sentiment mimicry	Degree of affective valence matching, indicating emotional alignment	Whether the parent and child sentiments match	Combined from parent and child comments

2.4 Quantitative modeling

We first fitted generalized linear models (GLMs), specifically negative binomial regressions. Alternatives such as linear models were not appropriate given the zero-bounded, count data with substantially larger variance than means. Each GLM model predicted one child question type count from all four parent question types. Thus, four separate GLM models were run as there were four question type outcomes. Wald tests compared regression coefficients to assess narrow versus broad transmission. To strengthen inference, we conducted 1,000 randomization tests within a discussion thread that preserved marginal count distributions while breaking parent-child links. This allowed us to evaluate whether observed effects exceeded chance while taking into account discussion threads and topics. We also fitted GLMs for non-examination topics to probe boundary conditions. A total of 40 Wald tests were performed. We used Šidák multiple-testing correction, a less conservative procedure compared to Bonferron correction. As a robustness check, we also implemented logistic regression (binarizing outcomes). GLMs were implemented in MATLAB (2025b).

To examine transmission mechanisms, we applied Monte Carlo g-computation, a flexible and rigorous approach to mediation originating from causal inference framework [12, 21]. It estimates how an outcome would change under hypothetical (counterfactual) manipulations of predictors and mediators. Intuitively, the method uses fitted models to simulate what would happen if the parent comment had different levels of a predictor, while allowing mediators to respond accordingly.

For each parent predictor-child outcome pair, we fitted a negative binomial model for the outcome and separate models for ten candidate mediators (Table 2), all included simultaneously in a parallel mediation framework. Let A denote the parent predictor, M the vector of mediators, and Y the child outcome. Using 1,000 Monte Carlo draws from the estimated parameter distributions, we simulated counterfactual expectations under contrasting predictor levels a_1 and a_0.

Effects were decomposed into indirect and direct components using the potential outcomes framework. The indirect effect (IE) is defined as:

$$IE = \mathbb{E}[Y_{a,M(a_1)}] - \mathbb{E}[Y_{a,M(a_0)}],$$

which captures the change in the outcome due to changes in mediators induced by the predictor, holding the predictor fixed. The direct effect (DE) is defined as:

$$DE = \mathbb{E}[Y_{a_1,M(a_0)}] - \mathbb{E}[Y_{a_0,M(a_0)}],$$

which captures the change in the outcome due to the predictor, holding mediators fixed at levels corresponding to a_0. The total effect satisfies:

$$TE = IE + DE = \mathbb{E}[Y_{a_1}] - \mathbb{E}[Y_{a_0}].$$

Mediator-specific indirect effects were evaluated using 95% confidence intervals, with the proportion mediated computed as $IE/(IE + DE)$. Given the observational nature of the data, these effects are interpreted descriptively rather than causally as key assumptions such as temporal ordering and sequential ignorability (e.g., no unmeasured confounders) are not fully satisfied. Analyses were conducted in R (MASS and related packages).

3. RESULTS

3.1 SGExams: Descriptive statistics and validation

Table 3 presents comment distributions across consolidated SGExams topics. Original post flairs were grouped based on shared themes, educational stages, and temporal trends into five categories: Exam Preparation, Exam Outcomes, Pre-university, University, and Others. Exam Preparation includes study strategies, content questions, and challenges surrounding high-stakes national examinations held annually. Exam Outcomes centers on exam results and admissions. Pre-university covers broader secondary and post-secondary experiences beyond exams (e.g., relationships, school activities, internships). University focuses on college admissions and courses. The heterogeneous "Others" category (e.g., jobs, scholarships, rants) was excluded from further analysis. Our primary analyses focused on Exam Preparation, which constituted the majority of activity and was theoretically most conducive to transmission effects due to its high-stakes context; results for the other topics are reported in Appendix Table 1.

We conducted extensive validation before analysis. Retrieval completeness was assessed in two ways: comparison with manual counts over a one-month window (99.8% retrieval) and cross-checking with third-party statistics from https://subredditstats.com/ (97.8%). Although third-party methodologies are not fully transparent, converging evidence suggests near-complete retrieval, lending confidence to complete conversational hierarchy.

Demographic validation of 200 randomly sampled users showed that 174 provided age-indicative statements; of these, 93.7% appeared to be adolescents broadly defined as students between secondary school and college. Finally, automated question classification was validated against 1,804 human-labeled sentences (κ = 0.97 for double-coded subset), yielding a weighted F1 of 0.90. Together, these extensive validation procedures support the dataset's completeness and validity. The code and dataset are publicly available at our institutional data repository (*https://doi.org/10.25340/R4/SYASFR*). We have removed Reddit usernames as the dataset is made up of discussions by minors in order to afford some level of privacy. Linked data is available upon request.

3.2 Social transmission of curiosity

Four GLMs were estimated, each predicting the count of a specific child question type from the four parent question types. As hypothesized, we uncovered strong and reliable transmission effects of curiosity as measured by question-posing. Results for Exam Preparation consolidated topic specifically are presented in Table 4 (N = 136,010 filtered parent-child pair). A parent question type strongly predicted the same question type in the child comments. For example, a one unit increase in parent open, non-possibility question type was associated with a 205% increase in expected count of child open, non-possibility question type. Evidence for narrow transmission where the same parent question type

Table 3. Consolidated discussion topics, original post flairs, and their distribution

Consolidated topics	Original Flair Title	Count (% of Total)
Exam Preparation (50.3%)	Exam Megathread	119608 (20.9%)
	O Levels	108271 (19.0%)
	A Levels	47347 (8.3%)
	N Levels	12070 (2.1%)
Exam Outcomes (4.4%)	Results Megathread	24966 (4.4%)
Pre-University (17.1%)	Junior Colleges	48726 (8.5%)
	JC vs Poly	3878 (0.7%)
	Polytechnic	45126 (7.9%)
University (17.0%)	University	97016 (17.0%)
Others (11.2%)	Rant	54005 (9.5%)
	Jobs	4493 (0.8%)
	Scholarships	5585 (1.0%)

Note. Data shown is post-filtering (see Methods). "O Levels" and "N levels" are Singapore's national exam for secondary school while "A levels" is the national exam for high school. "Junior Colleges" or "JC" are senior high schools, while "Polytechnic" or "Poly" post-secondary vocational institutions.

preferentially predicted its own type in child comments was tested using pairwise Wald test. Each GLM, superscript [a] in Table 4 indicates whether the bolded parent question type similar to child question type predicted the child question type significantly better than the other parent question types. For example, parent O-NP predicted child O-NP significantly more strongly than other parent question types, as indicated by superscripts [a] for O-P, C-NP, C-P. A narrow transmission effect will produce superscripts for all other parent types. Narrow transmission was observed for open type questions, both non-possibility and possibility (though with one non-significant result) types. However, closed question types still statistically significantly predicted open question types, though with smaller effect sizes. Our randomization procedure identified 5.2% of GLMs coefficients that were statistically significant, in line with a Type 1 error of 5%, suggesting that our results cannot be explained simply by chance or thread clustering. In the Appendix, we performed the same GLM analyses for the remaining three consolidated topics that do not directly relate to the Exam Preparation topic. We found transmission effects though not as strongly and statistically reliably as Exam Preparation. Interestingly, Exam Outcome discussions were more likely to exhibit suppression effects, the opposite of transmission (Appendix Table 1), indicating that the nature and goals of discussions play important roles in social transmission. In summary, curiosity as measured by question-posing in student discussions strongly exhibits social transmission in high-stakes exam-oriented student discussions, though with more limited evidence for specificity in its transmission.

Table 4. GLM results predicting the outcome variable of child question type from parent question type

	Beta	SE	z	Corrected p	Relative ratio
O-NP (outcome variable)					
Intercept	-2.51	0.01	-176.95	<0.01	0.08
O-NP	1.11	0.04	31.13	<0.01	3.05
O-P[a]	0.19	0.15	1.31	0.99	1.21
C-NP[a]	0.36	0.04	9.61	<0.01	1.43
C-P[a]	0.27	0.06	4.28	<0.01	1.32
O-P (outcome variable)					
Intercept	-5.43	0.05	-119.77	<0.01	0.00
O-NP[a]	0.57	0.09	6.23	<0.01	1.77
O-P	1.22	0.29	4.21	<0.01	3.39
C-NP[a]	0.16	0.11	1.42	0.99	1.17
C-P	0.46	0.17	2.65	0.26	1.58
C-NP (outcome variable)					
Intercept	-2.64	0.01	-188.12	<0.01	0.07
O-NP[a]	0.84	0.03	24.36	<0.01	2.31
O-P	0.46	0.14	3.33	<0.01	1.58
C-NP	0.45	0.04	12.82	<0.01	1.57
C-P	0.35	0.06	5.65	<0.01	1.41
C-P (outcome variable)					
Intercept	-3.36	0.02	-179.49	<0.01	0.03
O-NP	0.57	0.04	13.12	<0.01	1.77
O-P[a]	-0.12	0.22	-0.54	0.99	0.89
C-NP[a]	0.36	0.05	7.69	<0.01	1.43
C-P	0.67	0.08	8.90	<0.01	1.95

Note. Four GLMs were used to predict each dependent variable of child question type from the four parent question types (+ intercept). For each GLM, Beta: GLM regression coefficient estimate (log scale). SE: standard error of the coefficient. z: Wald test statistic (Beta/SE). p: p-value for the Wald test. Relative ratio: exponentiated coefficient, exp(Beta), indicating multiplicative change in expected count of outcome variable for a one-unit increase in the predictor. The number minus 1 times 100% indicates the percentage change. For example, a unit increase in the count of parent O-NP was linked to a 205% increase in the expected count of child O-NP.
For each GLM, superscript [a] indicates whether the bolded parent question type predicts the same question type in the child comment significantly better than the other parent question types (based on pairwise Wald test). For example, in the GLM to predict O-NP, parent O-NP predicted child O-NP significantly more strongly than other parent question types, as indicated by superscript [a] for O-P, C-NP, C-P. A narrow transmission effect will produce superscripts for all other parent types. Such an effect was observed only for parent O-NP and O-P question types.
O-NP: open, non-possibility questions; O-P: open, possibility questions; C-NP: closed, non-possibility questions; C-P: closed, possibility questions. * Corrected p-values: *p < 0.05; ** p < 0.01; *** p < 0.001

3.3 Mediating pathways

To examine pathways mediating transmission, we applied a Monte Carlo simulation-based g-computation approach. Ten discussion features, five each in cognitive and socio-affective categories were tested as potential parallel mediators. Table 5 presents the results. Numbers are the percentage of total effects of the predictor on the outcome mediated by the respective mediator, i.e., indirect effects. The vast majority of indirect effects were statistically significant, suggesting reliable mediations that were, however, generally small on the order of ~5-15%. Bolded estimates represent the top 3 mediators for each of the question types. Popularity, as measured by upvotes, was frequently amongst the top mediators. This result indicates that the popularity of parent comment functions as a marker of social endorsement, increasing the likelihood that subsequent child replies reproduce curiosity. Agreement, as measured by public concurrence and endorsement, also played an important mediating role for all question types, suggesting follow-up comments were more likely to display curiosity when students were more aligned in their views and opinions.

Interestingly, target lexicon mimicry also significantly mediated the social transmission of curiosity, indicating that adolescent students are prone to question word re-use in their curiosity expressions. However, this question word re-use contrasted with negative mediation by general mimicry, which measures overall lexical re-use. This pattern of results highlights the selective epistemic uptake, and not merely broad language imitation. We also observed that the percentage of indirect effects tended to be quite similar for the different question types. Indeed, formal quantification revealed very high correlation in indirect effects estimates for all question types (Spearman's r = 0.79). This provides a plausible explanation for the general finding of broad curiosity transmission:

similar mediating mechanisms appear to operate across different question types, suggesting shared cognitive and socio-affective pathways rather than type-specific processes.

Table 5. Proportion of total effects of the parent question type on child question type mediated by 10 conversational features.

	O-NP ↓ Media-tor ↓ O-NP	O-P ↓ Media-tor ↓ O-P	C-NP ↓ Media-tor ↓ C-NP	C-P ↓ Media-tor ↓ C-P
Mediator: Cognitive				
Target lexicon mimicry	**7.27***	2.54	**5.64***	**12.24***
General mimicry	**-7.44***	-2.34	-1.50	**-6.06***
Semantic similarity	0.19*	-0.27	1.06*	-0.21
Local referencing	3.07*	2.40*	4.04*	2.98*
Agreement	6.05*	**6.05***	**6.97***	**6.00***
Mediator: Socio-affective				
Popularity	**15.20***	**11.32***	**10.05***	3.28*
Social referencing	0.79*	-1.96	0.27	0.92*
Sentiment - Parent	0.35*	**-3.25***	0.21*	1.09*
Sentiment - Child	1.64*	2.12	-1.38*	1.06*
Sentiment mimicry	-0.26	0.86	0.51	0.26

Note. Numbers are estimated percentages of the total effects of predictor on outcome mediated by the respective mediator. Bolded percentages are the top 3 mediators for each predictor-outcome relationship. * indicates statistically significant indirect effects ($p < 0.05$).
O-NP: open, non-possibility questions; O-P: open, possibility questions; C-NP: closed, non-possibility questions; C-P: closed, possibility questions.

4. DISCUSSION

Our study analyzed a new dataset of online educational discussions led by adolescent students. Leveraging a newly created large-scale and topic-rich nature of the dataset, we mined for intricate signals of curiosity transmission and its potential pathways using advanced regression models followed by more flexible Monte Carlo g-computation to further quantify mediation. We found strong social transmission of curiosity: parent question-posing robustly predicted child question-posing, in agreement with past studies suggesting social transmission of curiosity [2, 9]. There was limited evidence of narrow transmission, though broader cross-type effects emerged as strongly. Social transmission was especially strongest in exam discussions. Monte Carlo g-computation identified statistically reliable though modest mediating pathways. We found that popularity (upvotes) and agreement consistently mediated social transmission.

The finding that comment popularity plays one of the most important roles in socially transmitting curiosity is supported by previous findings. Participants indicated greater curiosity about the answers to scientific questions when the questions were presented with experimentally manipulated high vs. a low number of up-votes [6], highlighting the importance of social cues in promoting curiosity. Our study further extends this finding by demonstrating that popularity not only heightens individual curiosity toward content, but also shapes subsequent behavioral expression of curiosity in peer interaction. Specifically, highly upvoted parent questions were more likely to elicit follow-up questions from others, indicating that social endorsement functions as a diffusion mechanism of question-posing. Cognitive alignment, as operationalized by our agreement conversational feature, is also an important mediator of social transmission. Students were more likely to be curious when they agreed with the stance or perspective expressed in the prior comment. One possible explanation is that cognitive alignment creates a psychologically safe and supportive space in which extending the discussion through further questions becomes more likely, thereby facilitating the social propagation of curiosity.

We also found a dissociation between question word mimicry and general lexical mimicry, in which the former increases, but the latter decreases, curiosity propagation. This pattern can be interpreted through a social cognitive lens. Social cognitive theory posits that observational learning is selective: individuals attend to, encode, and reproduce behaviors perceived as meaningful, functional, or socially valued, not merely any behavior. The positive mediation by target lexicon mimicry suggests that adolescents selectively model epistemically relevant features (e.g., interrogative structures signaling curiosity), rather than broadly imitating surface language features. An experimental study lends support to this idea: children and adults re-use or novelly re-combine previously exposed questions, especially when they are informative, likely in the service of efficient inquiry and effective learning [15]. Our work also links to prior work on linguistic style matching in discourse analysis [8, 19], though our results provide more nuances, such as what linguistic aspects are being matched by adolescent students.

Our study extends past online discussion analyses in educational data mining/learning analytics. These studies have fruitfully applied computational linguistics methods with a focus on better understanding and predicting course participation and performance [e.g., 25]. We extend these studies to focus more on social transmission, particularly its mediating mechanisms, applying it to curiosity expressions by way of questioning.

Our study makes further contributions to research by introducing a scalable, multidimensional taxonomy of question-posing and applying it to a large, naturalistic dataset. Existing taxonomies often dichotomize questions into lower- versus higher-order types or rely on domain-specific distinctions (e.g., causal vs. descriptive in science), which can be overly reductive or limited in generalizability or scalability. By crossing openness (open vs. closed) with possibility thinking based on a previous framework [1], our extension captures variation in information-seeking intent while remaining adaptable across domains. Moreover, its reliance on identifiable linguistic markers enables highly reliable automated classification at scale. Coupled with a large corpus of peer-driven discussions that we have curated and continue to build on, this approach advances methodological tools for studying many aspects of educational discussions, including curiosity, as they unfold dynamically in authentic digital environments. While discussions in the context of formal settings (group work in a classroom or online course discussions) are relatively well studied, the dynamics of informal, peer-driven learning communities remain underexplored.

Our study has implications for educators as discussion facilitators and for designers of social learning platforms, though with important caveats. For educators, the findings suggest that curiosity can be socially amplified through visible endorsement and constructive alignment. Publicly recognizing thoughtful questions and modeling how to extend peers' inquiries can help cultivate inquiry-oriented norms [2]. However, overemphasis on popularity could privilege confident or dominant students; highlighting alignment might limit epistemic diversity. Educators, therefore, need to balance public reinforcement with inclusive facilitation strategies that invite diverse voices and stances. For platform designers, features such as upvotes, threading, and content highlighting can shape epistemic behavior by signaling what is valued. Yet, simple popularity metrics may also distort attention toward socially appealing rather than substantively rich questioning. Designing

systems that surface high-quality questions, such as via rotating visibility or algorithmic weighting, may better support equitable and sustained discussions.

We acknowledge limitations in the study. First, while our dataset is grounded in general educational discussions (unlike other general adolescent-based Reddit communities which can be dominated by pop culture and general lifestyle), it is nonetheless limited to one particular national educational context. Second, our modeling of mediation based on g-computation which originates from a causal inference framework. However, failing to meet strict causal assumptions such as sequential ignobility (e.g., no unmeasured confounding) and temporal ordering limits strong causal statements. For instance, due to the limitations of the Reddit API, upvote counts were returned as a single total with no temporal information. Another limitation is that the data has a hierarchical and networked structure. As such, observations are not fully independent, and standard errors from our GLMs may be underestimated. While our analyses partially controlled for discussion context by analyzing consolidated topic categories separately and performing within-thread randomization tests, we did not explicitly model all forms of clustering such as users and higher-order sequences (e.g., subheads within threads) operating at different scales [5, 14].

Considering these limitations, future studies can be extended in several ways. One interesting question is whether social networks play a role. While Reddit discussions are largely anonymous, repeated interactions create recognizable usernames and emergent micro-communities. An important next step is to examine whether curiosity transmission is influenced by network position or prior interaction history between users. Another promising direction is to investigate temporal dynamics: e.g., does repeated engagement with curious peers increase an individual's baseline propensity to ask questions over time, suggesting longer-term socialization effects beyond immediate momentary exchanges? We believe addressing these questions can illuminate fine-grained processes of how students learn digitally socially beyond the classroom.

5. CONCLUSION

This study demonstrates that curiosity, operationalized as volitional question-posing, is not merely an individual disposition but a socially transmitted behavior in peer-driven online discussions. Using large-scale data and computational modeling, we show that questioning propagates across interactions, with both broad and selective patterns shaped by cognitive alignment and social endorsement. By framing curiosity at the behavioral level, our findings highlight how learning-related inquiry emerges and spreads in authentic digital collaborative environments, offering insights for designing educational contexts that cultivate sustained, self- and peer-driven questioning.

6. ACKNOWLEDGMENTS

None.

7. APPENDIX

Appendix Table 1. GLM results predicting the outcome variable of child question type from parent question type for each of the three remaining consolidated topics

	Exam outcome	Pre-University	University
O-NP (outcome variable)			
Intersect	0.07**	0.08**	0.09**
O-NP	2.01**	1.56**	1.50**
O-P	2.68	1.80*	1.66*
C-NP	0.66	1.29**	1.21**
C-P	0.49	1.31**	1.25**
O-P (outcome variable)			
Intercept	<0.01	<0.01**	<0.01**
O-NP	1.85	1.31*	1.47*
O-P	<0.01	6.67**	3.51*
C-NP	0.77	1.28*	1.19
C-P	<0.01	1.00	1.04
C-NP (outcome variable)			
Intercept	0.15**	0.10**	0.13**
O-NP	1.80*	1.25**	1.14**
O-P	3.38	1.34	1.65*
C-NP	0.48**	1.49**	1.35**
C-P	0.41**	1.39**	1.08
C-P (outcome variable)			
Intercept	0.06**	0.04**	0.07**
O-NP	1.98**	1.26**	1.15**
O-P	0.94	1.24	1.32
C-NP	0.64*	1.34**	1.13**
C-P	0.78	1.54**	1.36**
N (parent-child pairs)	11,134	50,053	53,356

Note. Statistics are relative ratios (regression coefficients exponentiated), which are multiplicative change in expected count of outcome variable for a one-unit increase in the predictor. The number minus 1 times 100% indicates the percentage change. Results can be compared to Table 4 in the main text.
O-NP: open, non-possibility questions; O-P: open, possibility questions; C-NP: closed, non-possibility questions; C-P: closed, possibility questions.
Corrected p-values: *p < 0.05; ** p < 0.01; *** p < 0.001

8. REFERENCES

[1] Acar, S., Berthiaume, K., and Johnson, R., 2023. What kind of questions do creative people ask? *Journal of Creativity 33*, 3, 100062. DOI= http://dx.doi.org/10.1016/j.yjoc.2023.100062.

[2] Ali, F., Wang, Y., Wang, S.J.-W., and Zhu, G., 2025. Triggers of curiosity in social constructivist classroom discourse. *npj Science of Learning 10*, 1, 33. DOI= http://dx.doi.org/10.1038/s41539-025-00330-5.

[3] Bandura, A., 1986. *Social foundations of thought and action: A social cognitive theory*.

[4] Bandura, A. and Walters, R.H., 1977. *Social learning theory*. Prentice Hall.

[5] Chen, B. and Poquet, O., 2023. Uncovering socio-temporal dynamics in online discussions: An event-based approach. *Australasian Journal of Educational Technology 39*, 6, 1-16. DOI= http://dx.doi.org/10.14742/ajet.8618.

[6] Dubey, R., Mehta, H., and Lombrozo, T., 2021. Curiosity is contagious: A social influence intervention to induce curiosity. *Cognitive Science 45*, 2, e12937. DOI= http://dx.doi.org/10.1111/cogs.12937.

[7] Engel, S., 2011. Children's need to know: Curiosity in schools. *Harvard Educational Review 81*, 4, 625-645. DOI= http://dx.doi.org/10.17763/haer.81.4.h054131316473115.

[8] Gonzales, A.L., Hancock, J.T., and Pennebaker, J.W., 2010. Language style matching as a predictor of social dynamics in small groups. *Communication Research 37*, 1, 3-19. DOI= http://dx.doi.org/10.1177/009365020935146.
[9] Gordon, G., Breazeal, C., and Engel, S., 2015. Can children catch curiosity from a social robot? In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, 91-98. DOI= http://dx.doi.org/https://doi.org/10.1145/2696454.269646.
[10] Haythornthwaite, C., Kumar, P., Gruzd, A., Gilbert, S., Esteve del Valle, M., and Paulin, D., 2018. Learning in the wild: coding for learning and practice on Reddit. *Learning, Media and Technology 43*, 3, 219-235. DOI= http://dx.doi.org/10.1080/17439884.2018.1498356.
[11] Huang, H., Tang, X., and Salmela-Aro, K., 2024. Facilitating youth's curiosity in learning: Needs-based ecological examinations. *Journal of Youth and Adolescence 53*, 3, 595-608. DOI= http://dx.doi.org/10.1007/s10964-023-01936-x.
[12] Imai, K., Keele, L., and Yamamoto, T., 2010. Identification, inference and sensitivity analysis for causal mediation effects. *Statistical Science 25*, 1, 51-71. DOI= http://dx.doi.org/10.1214/10-STS321.
[13] Jirout, J.J., Zumbrunn, S., Evans, N.S., and Vitiello, V.E., 2022. Development and testing of the curiosity in classrooms framework and coding protocol. *Frontiers in Psychology 13*, 875161. DOI= http://dx.doi.org/10.3389/fpsyg.2022.875161.
[14] Knight, S., Wise, A.F., and Chen, B., 2017. Time for change: Why learning analytics needs temporal analysis. *Journal of Learning Analytics 4*, 3, 7–17. DOI= http://dx.doi.org/10.18608/jla.2017.43.2.
[15] Liquin, E.G., Rhodes, M., and Gureckis, T.M., 2025. Seeking new information with old questions: Children and adults reuse and recombine concepts from prior questions. *Open Mind 9*, 885-925. DOI= http://dx.doi.org/10.1162/opmi.a.12.
[16] Litman, J., 2019. Curiosity: Nature, dimensionality, and determinants. In *The Cambridge handbook of motivation and learning*, K.A. Renninger and S.E. Hidi Eds. Cambridge University Press, 418-442.
[17] Loureiro, D., Barbieri, F., Neves, L., Anke, L.E., and Camacho-Collados, J., 2022. TimeLMs: diachronic language models from Twitter. *arXiv*, 2202.03829. DOI= http://dx.doi.org/10.48550/arXiv.2202.03829.
[18] Na, H., Staudt Willet, K.B., Shi, H., Hur, J., He, D., and Kim, C., 2024. Initial discussions of ChatGPT in education-related subreddits. *Journal of Research on Technology in Education 57*, 5, 953-971. DOI= http://dx.doi.org/10.1080/15391523.2024.2338091.
[19] Niederhoffer, K.G. and Pennebaker, J.W., 2002. Linguistic style matching in social interaction. *Journal of Language and Social Psychology 21*, 4, 337-360. DOI= http://dx.doi.org/10.1177/02619270223795.
[20] Palincsar, A.S., 2005. Social constructivist perspectives on teaching and learning. *An introduction to Vygotsky 2*, 285-314.
[21] Snowden, J.M., Rose, S., and Mortimer, K.M., 2011. Implementation of G-computation on a simulated data set: demonstration of a causal inference technique. *American Journal of Epidemiology 173*, 7, 731-738. DOI= http://dx.doi.org/10.1093/aje/kwq472.
[22] Swan, G.E. and Carmelli, D., 1996. Curiosity and mortality in aging adults: A 5-year follow-up of the Western Collaborative Group Study. *Psychology and Aging 11*, 3, 449. DOI= http://dx.doi.org/10.1037/0882-7974.11.3.449.
[23] Szumowska, E. and Kruglanski, A.W., 2020. Curiosity as end and means. *Current Opinion in Behavioral Sciences 35*, 35-39. DOI= http://dx.doi.org/10.1016/j.cobeha.2020.06.008.
[24] von Stumm, S., Hell, B., and Chamorro-Premuzic, T., 2011. The hungry mind: Intellectual curiosity is the third pillar of academic performance. *Perspectives on Psychological Science 6*, 6, 574-588. DOI= http://dx.doi.org/10.1177/17456916114212.
[25] Wen, M., Yang, D., and Rosé, C., 2014. Linguistic reflections of student engagement in massive open online courses. In *Proceedings of the International AAAI Conference on Web and Social Media*, 525-534.
[26] Yan, T. and Liu, F., 2022. COVID-19 sentiment analysis using college subreddit data. *PloS One 17*, 11, e0275862. DOI= http://dx.doi.org/10.1371/journal.pone.0275862.
[27] Zainal, N.H. and Newman, M.G., 2022. Curiosity helps: Growth in need for cognition bidirectionally predicts future reduction in anxiety and depression symptoms across 10 years. *Journal of Affective Disorders 296*, 642-652. DOI= http://dx.doi.org/10.1016/j.jad.2021.10.001.

Multimodal LLM-based approach for detecting Disengagement from Task Goal (DTG) in Math problem solving

Nidhi Nasiar[1], Ryan S. Baker[2], Jaclyn Ocumpaugh[3], Zeyneb S. Sarioglan[1], Caitlin Mills[4], and Samuel Rhodes[5]
[1]University of Pennsylvania
[2]Adelaide University
[3]University of Houston
[4]University of Minnesota
[5]Virginia Commonwealth University
{nasiar, zeysario}@upenn.edu, {ryanshaunbaker, jlocumpaugh}@gmail.com, cmills@umn.edu, rhodessr2@vcu.edu

ABSTRACT
Learner disengagement has been widely studied and modeled in math learning environments to enable timely supports and promote engagement. Recent advances in multimodal large language models (MLLMs) offer new opportunities for building such detectors. This study investigates the potential of an MLLM-based approach to detect disengagement from the task goal (DTG) in screen-recording videos of students solving math problems in CueThink. DTG is operationalized as actions unrelated to the intended learning task. We develop and validate a DTG detector using LLaVA-Video-7B-Qwen2(Large Language and Vision Assistant) using various prompting strategies, and analyze themes across the few model errors. We also examine associations between DTG behavior and learning gains, as well as students' math beliefs, anxiety, and metacognition. Results indicate that the model, using role assignment and chain-of-thought prompting, identifies DTG behavior with good performance ($\kappa = 0.71$, precision = 77.6%, recall = 76.4%). Performance limitations that emerged related to low-fidelity data, limited contextual knowledge, and following instructions. DTG behavior was marginally negatively correlated with normalized learning gains later in the academic year (winter to spring), but not during the earlier fall-to-winter period. There were no significant correlations to beliefs or anxiety.

Keywords
Multimodal LLM, LLaVA, disengagement, math, problem-solving, detector, video-analysis.

1. INTRODUCTION
The field of Educational Data Mining (EDM) has long-standing interest in various types of disengagement within and across learning platforms [34, 21]. Student disengagement is a multifaceted meta-construct [43] that includes behavioral (e.g. gaming the system), emotional (e.g. boredom), and cognitive components (e.g. mind wandering) [19]. Identifying and understanding student disengagement is important to efforts to reduce it [1], as engagement is a necessary condition for learning and a key indicator of long-term achievement and academic success [36, 39]. Students can only benefit from the virtual learning environments if they are engaged with the system [10]. Engagement also plays a role in students' academic resilience and the development of resources for coping adaptively with stressors, which in turn may affect the development of long-term academic mindsets [37].

Within online learning contexts, disengagement has been modeled using various Machine Learning (ML) models. Specific examples of disengaged behavior include DTG (Disengaged from Task-Goal) [22], gaming the system [5], off-task behavior [4], and mind wandering [32]. There has been increased interest in identifying the various ways students can deviate from expected patterns while using educational software, including not using features of the platform and failure to engage with content [21]. Detection of these behaviors has helped researchers better understand them, allowing researchers and teachers to develop interventions that can remediate them and reduce their negative impacts on user outcomes [2].

For this paper, we focus specifically on Disengagement from Task Goal (DTG). This behavior has been observed in online learning but given a variety of names in the published literature [22, 35, 8, 42]. Examples include students who, instead of plotting points from a mathematical function in a learning platform for high school mathematics, plotted a smiley face [42]. In other learning systems, learners have avoided diagnosing virtual patients and instead placed virtual unrelated objects on them [35], tinkered with science simulations in ways unrelated to the stated learning goals [8], or otherwise engaged in actions that appear to have no relationship to the intended learning task [42]. To date, the majority of studies of DTG have been in science learning environments [22, 35, 42], with very limited work in math contexts.

In the context of math problem solving, DTG may take several forms, including using math tools and text to draw silly drawings, making smiley faces when plotting points, creating repetitive designs and patterns, typing curse words instead of answers, or writing funny terms for friends. It is worth noting that DTG differs from off-task behavior as that is typically understood: off-task behavior typically involves disengaging completely from the learning task, whereas in DTG, the student is engaging with the task, but in a fashion unrelated to the learning task's design goals or incentive structure.

Disengagement detectors have often been built using classical machine learning (ML). However, ML-based models are time and resource-intensive, requiring extensive feature engineering and labeled datasets. With the advancements in generative AI and large

Nidhi Nasiar, Ryan S. Baker, Jaclyn Ocumpaugh, Zeyneb S. Sarioglan, Caitlin Mills, and Samuel Rhodes. Multimodal LLM-based approach for detecting Disengagement from Task Goal (DTG) in Math problem solving. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 508–515. International Educational Data Mining Society (2026).
© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039855

language and multimodal models, new opportunities have emerged for building detectors. Large language models like ChatGPT have shown potential in facilitating education research through supporting qualitative data analysis tasks such as coding transcripts and analyzing open-ended responses [45, 48, 30]. Recent advances in multimodal large language models (MLLMs) have extended these capabilities to process both visual and textual information [47]. MLLMs have demonstrated promising results in tasks such as image captioning, video understanding, and visual question answering [44, 48].

One potential application of MLLMs in educational research is facilitating qualitative video analysis, which has yielded important results about students' learning processes, but previously required time-intensive coding of behavioral patterns and interactions [16, 33]. Emerging educational research has explored Machine Learning classification to detect teaching and learning moves [18], but these efforts also relied on manually coded data for training. Early work with MLLMs in online learning has been promising [51], demonstrating their potential for streamlining this process by supporting video selection and behavior coding—tasks that traditionally require significant manual effort from researchers. To date, however, their use for modeling DTG has not yet been studied.

Accordingly, the focus of this paper is to build and validate a DTG detector based on the multimodal LLM, LLaVA (Large Language and Vision Assistant) [27], and to systematically analyze its errors. Additionally, we examine how DTG behavior correlates with learning gains and measures of math beliefs, anxiety, and metacognition in math problem-solving—constructs that research has shown are increasingly important for math learning [3, 13, 15].

2. METHODS

2.1 Learning Platform and Data Collected

CueThink (2022) is a digital learning application designed to enhance middle-school students' mathematics problem-solving skills by encouraging self-regulated learning (SRL) and supporting students in sharing their problem-solving processes. The platform asks students both to solve a mathematics problem and to create a shareable screencast video that presents their solution and demonstrates their reasoning. CueThink structures each problem into a Thinklet, which consists of four phases—Explore, Plan, Solve, and Review—that closely align with Winne and Hadwin's [41] SRL model. A full description of the CueThink phases is provided in Zhang et al. [49].

The study included 180 students in Grades 6–8 from suburban schools in the Southwestern United States who used CueThink during the 2021–2022 school year. Self-reported gender demographics were collected (66 boys, 98 girls, 1 non-binary, 3 chose not to report, and 12 had missing values). Across the year, students completed 746 Thinklets (4.09 Thinklets per student, on average). Pre- and post-test surveys were administered at the beginning and end of the school year; the survey scales are described in detail below (section 2.4).

This study primarily makes use of data from the Solve phase. During this phase, CueThink collects fine-grained interaction log and trace data capturing students' whiteboard actions, including position coordinates for each pen stroke (e.g., initiating a path and rendering it point-by-point). Students can use a variety of mathematical tools and objects on the whiteboard—such as number lines, protractors, tables, highlighters, and text/equation entry (Figure 1). These interaction traces can be rendered into a screen-recording that visualizes students' step-by-step work as they solve the problem. These videos were used both as inputs to the model and to establish ground truth labels (see section 2.2).

All data were collected and analyzed under a protocol approved by the University of Pennsylvania Institutional Review Board (IRB protocol number 856358). The LLaVA model was run locally on our lab server for analysis, so no data was transmitted to any external server, minimizing data privacy risks.

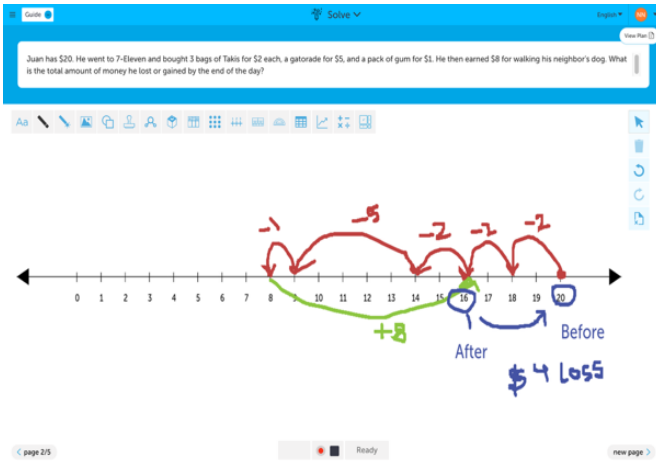


Figure 1. Solve Phase in CueThink

2.2 Operationalizing DTG

DTG (Disengaged from Task Goal) was operationalized as actions that have no relation to the intended learning task. The intended learning task here is to use system-internal tools to write and solve an equation on the digital whiteboard, and tools included a number line, table, protractor, ruler, or mathematical symbols. Each video was binary coded for DTG behavior by 2 researchers to establish the ground truth. Each researcher watched the full video, which showed all the students' actions in chronological order as they solved the math problem in the context of the learning environment, to then make a decision about whether DTG behavior was ever observed or not.

The operationalization was developed through an iterative process to ensure that the behaviors classified as DTG aligned with previous conceptualizations [22] and were salient in the dataset. Through this process, indicators of engagement (non-DGT behavior) were readily agreed upon, including using a number line to add or subtract or organizing given values in the problem statement into a table to solve the problem, or any written text or images drawn for finding a solution. Meanwhile, several specific criteria for DGT were identified, including: (1) text unrelated to given math problems, (2) usage of given math tools for purposes other than solving the math problem, and (3) other task-unrelated actions (silly drawings, smileys).

Some clarifications were made to the DGT criteria. For example, behaviors that might have indicated DTG were ignored if they occurred after the problem was completed. As such, if the video included a student who finished solving the problem on the whiteboard with a solution before creating an unrelated drawing, the video was not labeled as DTG. If the video is blank and no actions take place on the whiteboard, it is also not treated as DTG because there was no action.

Using this operationalization, two coders first independently coded 120 videos from a separate dataset from another year on the same learning system to establish interrater reliability ($\kappa = 0.79$). Once consensus was reached, the coders proceeded to code a total of 746 videos from 180 students. These coded videos constituted our "ground truth" that was used to compare against model labels to validate the model's performance. Of these 746 videos, 546 videos from 129 students were designated as the development set and used for iterative prompt refinement and evaluating strategies, while the remaining 200 videos from 51 students (none of whom appeared in the development set) served as a held-out test set for evaluating the final model's performance and further analysis.

2.3 LLaVA as a DTG detector

2.3.1 LLaVA Model Details

We used LLaVA-Video-7B-Qwen2 [50], a state-of-the-art video MLLM was selected because it reflects recent technological advances in video MLLMs, is open source, and has demonstrated suitable performance on video understanding benchmarks [20,50] and in educational applications [51]. We used the pre-trained model with temperature set to 0 to ensure reproducibility. We did not fine-tune the model because our aim was to assess the potential of existing pre-trained models for detecting disengagement behaviors, and fine-tuning requires substantial computational resources and large annotated datasets.

2.3.2 Video Analysis and Script Setup

To process the video data, we selected sampling and segmentation settings that balance temporal coverage and processing capacity. Each video was first decomposed into individual frames. These frames, together with the text prompts—were tokenized as inputs to the model. Because the MLLM has a fixed context window that limits the total number of tokens it can process at once, analyzing longer videos involves a trade-off between the number of tokens allocated per frame (increasing visual detail) and the total number of frames included (increasing temporal coverage).

Each video was segmented into 10-second clips and sampled at three frames per clip. This conservative choice was derived empirically by testing multiple segmentation lengths (i.e., 10, 5, and 1-second clips) and sampling values (3 vs. 5 frames per clip), in order to capture sufficient student actions and visual context to support DTG coding while remaining within token limits. Different segment lengths and sampling rates showed no meaningful differences in output quality, and therefore, we adopted a 10-second segmentation with three frames per clip (approximately one frame every 3.3 seconds, or ~0.3 fps) to maintain computational efficiency. All clips were then processed through LLaVA using the refined prompt. To obtain a single video-level prediction per student, clip-level predictions were aggregated using an inclusive decision rule, such that, if DTG behavior was identified in one or more clips within a video, the student was labeled as exhibiting DTG behavior for that problem.

2.3.3 Prompting strategies

We employed an iterative approach to refine the text prompts given to LLaVA using the development set (546 videos), focusing on maximizing the accuracy of labeling for DTG behavior. Once prompt development was complete, we evaluated the final prompt configuration on a held-out test set of 200 videos to assess the model's performance on unseen data. Given that LLaVA—like other generative models—is known to be sensitive to prompt framing and task specification [40, 29], we tested several prompting strategies, including (1) a baseline prompt, (2) a role-assignment strategy (which assigns a specific role or perspective to the model in order to facilitate a more contextualized interpretation e.g., Liu et al. [29]), (3) a chain-of-thought strategy (which facilitates step-by-step reasoning processes), (4) a combination of both approaches. These prompting strategies were selected based on prior work showing that prompting choice can affect multimodal LLM output. Chain-of-thought prompting has been shown to improve visual reasoning in multimodal LLMs, including LLaVA [48], while role-assignment has produced mixed results across settings [29]. We tested both techniques and their combination to identify the configuration best suited to DTG behavior labeling for our data. We also tested variations by changing key instructional verbs (e.g., "analyze" vs. "examine") while maintaining the role assignment component. Additionally, we provided contextual knowledge about math problem-solving in the prompt to help the model identify regular math-solving actions (task-related actions), such as strikethroughs to numbers and text that indicate problem-solving processes.

Baseline Prompt. The baseline prompt was as follows:

"Assign a binary code for DTG (disengagement from task goal) behavior defined as 'actions that have no relation to the math problem."

Role assignment and chain-of-thought strategy. The baseline combined-approach prompt was as follows:

"You are a [role assignment: qualitative coding expert] and your task is to [analyze/examine] and assign a binary coding for the construct of DTG (disengagement from task goal), which is defined as 'actions that have no relation to the math problem'. The videos show the whiteboard with student actions. The intended learning task for students was to solve math problems using the given tools (like a number line, table, protractor, or ruler, and other mathematical symbols). [chain-of-thought reasoning: insert step-by-step rules for operationalization criteria]."

Throughout this iterative process, we maintained concise prompts to optimize token allocation, as longer prompts can lead to attention dilution [23, 28] and degrade model performance even before reaching technical token limits [26].

2.3.4 Validation of LLaVA's DTG labels

During the prompt-refinement process, prompts given to LLaVA used a development set (546 videos), focusing on maximizing the performance of the model for correctly labeling DTG behavior. Once prompt development was complete, we evaluated the final prompt configuration on a held-out test set of 200 videos to assess the model's performance on unseen data across three metrics—kappa, precision, and recall.

2.3.5 Error Analysis

After finalizing the prompt, the test dataset was sent to the model for binary coding the videos for DTG behavior. For any clips where the MLLM label contradicted the human ground-truth label, further analyses were conducted. Namely, the 1st and 3rd authors re-watched the associated videos and analyzed these clips to identify potential causes of disagreement and possible instances of model hallucinations. We then conducted a thematic analysis [7] to distill recurring patterns across these errors, iteratively grouping these observations into broader categories.

2.4 External Survey Measures

Validated external measures were used to assess students' content knowledge [14], math anxiety [9], metacognitive awareness [38], and beliefs on problem-solving [25]. Each of these measures was

used first as a pre-test and then as a post-test. Once the pre-test was complete, students were provided with access to the learning platform. Their interaction with the platform was integrated into their regular classroom instruction, with their teacher assigning problems for them to complete.

Content Knowledge. Partnering districts administered i-Ready diagnostic assessments [14] as a proxy for mathematics content knowledge. The i-Ready [14] instrument is an adaptive assessment tool used to identify math topics students are struggling with; it examines students' understanding of mathematical sub-domains, including numbers and operations, algebra, geometry, and measurement. This assessment was administered three times, once each at the beginning (September), middle (December to January), and end (May) of the academic year.

Normalized learning gains [31] were calculated for those periods for the i-Ready tests from Fall to Winter; and Winter to Spring as NormLG_F-W and Norm_LG_W-S, respectively:

$$\text{NormLG} = \begin{cases} (\text{post} - \text{pre})/(\text{max} - \text{pre}) & \text{if post} \geq \text{pre} \\ (\text{post} - \text{pre})/\text{pre} & \text{if post} < \text{pre} \end{cases}$$

The normalized gains for two semesters were used for correlation analysis, and not the raw i-Ready scores.

The Junior Metacognitive Awareness Inventory (JrMAI) is a measure of metacognitive and cognitive strategies applied by learners [38] that was developed for students in grades 6 through 12. The current study uses a 9-item abbreviated version of the JrMAI that was validated with students in grades 6 -8 [24]. The measure includes 5 items identified as regulation of cognition and 4 items identified as knowledge of cognition.

The modified Abbreviated Math Anxiety Scale (mAMAS; [9]) uses a two-factor structure, which results in two subscales learning math anxiety (Learning subscale), and math evaluation anxiety (Evaluation subscale) [9]. The scale has shown good internal consistency, with an overall Cronbach's α of 0.85, a Cronbach's α of 0.77 for the Learning subscale and a Cronbach's α of 0.79 for the Evaluation subscale [9, 11]. The scale was developed for math learners between 8 and 13 years old (i.e., overlapping our research sample) and includes a 9-item self-report of math anxiety that are averaged to produce one final scale for analysis. The mAMAS was slightly modified to change adapted words to American English (e.g., "maths" to "math").

The Indiana Mathematics Belief Scales (IMBS; [25]) prompts students regarding their beliefs about mathematics and mathematical problem-solving over a 36-item survey. The measure is divided into six subscales. In this work, we used 2 of the 6 subscales (1 and 6). Specifically, we administered shortened versions of three subscales, which measure (a) student beliefs that they can solve time-consuming math problems (5 items), and (b) student beliefs on useful of math in their real lives, respectively (3 items). The shortened versions were validated using middle school students.

2.4.1 Analysis

In addition to the analysis of LLaVA's performance to accurately label DTG behaviors, we also examined the association between DTG labels and the external measures in this study. For analysis, first we aggregated DTG behavior at the student level by averaging DTG across each student's Thinklets. Then, for calculating correlations with survey measures, we used Spearman rank correlation because the variables were non-normally distributed. Afterwards, we applied Benjamini–Hochberg post-hoc corrections to control the false discovery rate for multiple comparisons [6].

3. RESULTS

3.1 Performance across prompts

Several metrics were used for evaluation including Cohen's kappa (κ), which was calculated for how well specific prompts produced outputs in agreement with human labels. Precision indicated the proportion of positive predictions that were correct. Recall measured the proportion of actual positive instances that the model successfully identified.

Table 1 reports the 3 metrics across the baseline prompt and prompt variations that introduced role assignment, chain-of-thought–inspired prompting, and their combination, evaluated under two phrasings ("analyze and assign" vs. "examine and assign").

Table 1. Kappa (κ), precision, and recall across prompts

Prompt Strategy	Analyze and Assign			Examine and Assign		
	κ	Prec.	Recall	κ	Prec.	Recall
Baseline	0.40	50.4%	59.0%	0.34	48.7%	59.2%
Role-assignment only	0.68	75.0%	78.3%	0.57	62.7%	75.5%
CoT only	0.60	66.7%	74.5%	0.53	59.2%	71.0%
Role-assignment + CoT	0.71	77.6%	76.4%	0.65	71.4%	75.3%

Across different prompting strategies, the 'role-assignment and chain-of-thought (CoT)' prompt with 'analyze and assign' achieved the highest kappa (κ = 0.71) and precision (77.6%), though role-assignment only had marginally higher recall (78.3% vs. 76.4%). This was followed by 'role-assignment only' with 'analyze and assign' (κ = 0.68, precision = 75%, and recall = 78.3%). Overall, 'analyze and assign' had higher values across all metrics and prompting strategies in comparison to 'examine and assign'. The baseline prompt, without any context or role, had the lowest scores for both 'analyze and assign' (κ = 0.40, precision = 50.4%, recall = 59%) and 'examine and assign' (κ = 0.34, precision = 48.7%, recall = 59.2%). We selected 'role-assignment and CoT' with 'analyze and assign' as the final prompting strategy due to higher metrics overall, and also balanced precision and recall, This prompt was used by the model to code the test set that was used for further analyses.

3.2 DTG Rates

The descriptive statistics of Disengaged-from-Task-Goal (DTG) behavior coded by the LLaVA model show that, across all Thinklets (N = 746), 79 were coded as DTG (10.59%), consistent with prior reports of DTG rates (e.g., Gobert et al., 2015; Sabourin et al., 2013). At the student level (N = 180), 67 students (37.22%) exhibited DTG behavior at least once across Thinklets. For each student, the DTG average was computed as the proportion of that student's Thinklets coded as DTG (DTG-coded Thinklets ÷ total Thinklets), with a score ranging from 0 (no DTG behavior) to 1 (DTG behavior on all Thinklets completed by that student). By self-identified gender, 32 of 66 boys (48.5%) and 33 of 98 girls (33.7%) exhibited DTG behavior. One non-binary student showed no DTG; among the three students who chose not to report gender, one showed DTG. A small number of students had missing gender values and were excluded from gender-specific percentages.

3.3 Error Analysis

The model showed good performance, comparable to or better than previous ML-based approaches for DTG behavior [22, 35].

However, to understand the few cases where it failed and enable future improvements, we conducted an analysis of cases incorrectly labeled as DTG behavior by the LLaVA model. We identified distinctive themes that can be categorized as follows:

3.3.1 Visual processing difficulties (low-fidelity data)

The LLaVA model exhibited systematic mislabeling stemming from low-fidelity visual interpretation challenges. Small-scale features posed particular difficulties: miniature smiley faces were incorrectly flagged as DTG when they were constructed from text characters like semicolons and parentheses (e.g., ";)" or ":)"), and small font sizes substantially increased OCR (optical character recognition) error rates. When handwritten objects appeared visually similar to numeric values, the model struggled to distinguish between task-relevant content and off-task elements—a differentiation that proved especially challenging given the handwritten nature of student work.

The model demonstrated poor recognition of cursive script and unclear handwriting. Pen scratches used by students to correct or cross out content were systematically mislabeled as DTG behaviors, as were unclear handwriting, numerical scribbles, and ambiguous pen-drawn elements that human annotators identified as legitimate task-related work. Additionally, when small objects overlaid background elements, human raters easily recognized the highlighted foreground from its color contrast, but the model often did not, indicating limitations in using color and spatial reasoning for labeling by the model. LLaVA connects a vision encoder (e.g., CLIP ViT) to a language model via visual instruction tuning [27], and not a conventional OCR pipeline, which might explain its sensitivity to tiny/cursive text and overlaps.

3.3.2 Lack of contextual knowledge

The model demonstrated insufficient understanding of mathematical tools, their appropriate usage contexts, and math problem-solving conventions. It mislabeled legitimate mathematical notation and work-in-progress indicators as DTG. For instance, tally marks, which human annotators recognized as counting tools based on their implicit contextual knowledge, were incorrectly flagged as disengagement. Similarly, incomplete solutions where students had drawn circles or shown intermediate steps without final answers were coded as DTG, whereas humans interpreted these as unfinished work rather than disengagement.

The model also struggled with distinguishing purposeful tool use from disjointed, unrelated behavior. Cases involving overlays of multiple mathematical objects (e.g., protractors, circles of different radii, rows of circles, colorful blocks, empty tables) were sometimes coded as 'not DTG' by the model when they clearly represented disengagement rather than actual problem-solving. Similarly, the rapid use of multiple mathematical tools to create unrelated imagery (e.g., multiple consecutive circles and lines forming an eye shape) was sometimes also difficult for LLaVA to recognize as DTG. Furthermore, the model did not interpret the written text correctly in the math problem-solving context. When students wrote text clearly unrelated to the problem, human annotators could read and infer the DTG behavior of the content, but LLaVA lacked this interpretive capability.

3.3.3 Instruction-following limitations

Consistent with previous findings [51], LLaVA model sometimes was unable to follow all of the instructions it was given, particularly in following conditional rules specified in the prompt. For example, it was unsuccessful at following instructions stating that activities that occurred after the student had completed the problem should not be classified as DTG.. Such errors are instructive about the limitations of LLaVA, which nonetheless achieved an overall model performance high enough to justify its use in aggregate correlational analyses.

3.4 Correlating DTG and Survey Measures

After applying the Benjamini–Hochberg post-hoc correction [6], none of the results of the Spearman correlations between students' disengagement from task goal (DTG) behavior and various outcome measures. However, marginally significant results (defined as $p < 2 \times$ adjusted α) were observed for one measure: normalized learning gains from winter to spring ($\rho = -0.278$, $p = 0.02$, adjusted $\alpha = 0.013$). In other words, students who more frequently engaged in DTG behavior demonstrated lower learning gains during this period. However, the relationship did not hold for normalized learning gains as calculated from fall to winter ($\rho = -0.155$, $p = 0.118$, adjusted $\alpha = 0.02$), suggesting that the negative effects of DTG may become more pronounced in the later part of the academic year.

The positive correlation between DTG behavior and post-test scores on the Indiana Mathematics Belief Scale (IMBS) subscale measuring beliefs about solving time-consuming math problem was not significant after post-hoc correction ($\rho = 0.157$, $p = 0.009$, bh-adjusted $\alpha = 0.004$), as the p-value was not below the critical (adjusted α) value. This correlation was also absent at pre-test ($\rho = -0.031$, $p = 0.687$, bh-adjusted $\alpha = 0.038$). No significant correlations were observed for the belief subscale on the usefulness of mathematics at either of the tests.

The negative correlation between DTG behavior and pre-test scores on the Math Anxiety Scale learning subscale was not significant after the Benjamini–Hochberg post-hoc correction ($\rho = -0.146$, $p = 0.018$, bh-adjusted $\alpha = 0.008$), as it did not meet the critical value criterion. This relationship was not significant at post-test ($\rho = -0.026$, $p = 0.736$, bh-adjusted $\alpha = 0.042$). The evaluation subscale of mathematics anxiety showed no significant correlations at either time point, which may reflect the fact that students were engaged in regular mathematics instruction rather than high-stakes testing contexts.

No other significant correlations were observed between DTG behavior and the metacognitive awareness scale.

4. DISCUSSION & CONCLUSION

4.1 Main Findings

This study demonstrates the potential of using LLaVA (Large Language and Vision Assistant), a multimodal large language model (MLLM) for automated detection of disengagement from task goal (DTG) behavior in screen recordings of virtual educational software. The pre-trained LLaVA model achieved acceptable performance ($\kappa = 0.71$, precision of 77.6%, and recall of 76.4%) in detecting DTG behavior during math problem-solving tasks. This shows that the model had high agreement with human raters while effectively minimizing both false positives and false negatives, and maintained balanced performance across the two metrics, suggesting robust and reliable detection capability across different types of DTG instances.

Rates of DTG labels by LLaVA model were consistent with prior reports of rates of this behavior across learning platform [22, 35]. Different prompting strategies were evaluated to improve the model performance, since the models are known to be sensitive to prompts. The results showed that framing the task to "analyze and

assign" and including assigning a role and chain-of-thought reasoning produced the highest kappa scores for coding DTG.

Analyzing the few error cases, the visual processing issues align with known weaknesses in vision-language models that rely on visual instruction tuning rather than dedicated OCR pipelines, particularly when processing low-resolution or visually ambiguous inputs [27]. Such challenges can be harder in educational settings where student-generated content is inherently messy, handwritten, and varies in quality. A lack of in-depth contextual knowledge of math, specifically understanding the tools and problem-solving processes required for particular tasks, represented another category with instances of incorrect DTG labels. A similar effect has been documented previously by Huang et al. [23], where they show that when multimodal LLMs lack sufficient contextual grounding, they can move toward speculative interpretation rather than evidence-based analysis. Lastly, the instruction-following limitations, including failures to apply conditional logic, are consistent with past findings [51], suggesting that sometimes gaps in sequential reasoning exist for the model.

While no correlations were statistically significant after the B&H correction, the marginally significant negative correlation between DTG behavior and normalized learning gains from winter to spring reflects an interesting relationship between DTG behavior and learning outcomes that warrants further investigation with larger samples. Notably, this association was absent in the fall-to-winter period, which may indicate that the cumulative effects of DTG behavior can become more harmful as the academic year progresses and mathematical content increases in complexity, consistent with prior research linking off-task behavior to reduced learning over time [12].

4.2 Future Work

This work has demonstrated the ability of MLLMs to considerably speed the labeling of DTG behaviors. Future research should investigate the individual differences and situational factors that lead students to engage in DTG behavior. For example, it is possible that students who engage in DTG are exhibiting a work avoidance goal orientation [17]. If researchers could distinguish between students who engage in DTG because they lack the necessary knowledge or skills and students who engage in DTG because the task is too easy, for example, we might be able to develop stronger scaffolding and adaptivity.

One approach to exploring this would be to examine DTG behavior using a broader range of survey metrics, including those that might be used for studies of students across learning domains (i.e., to determine if the same kinds of behaviors that emerge in a students' math classes also show up in science). However, additional qualitative data might also be informative. For example, now that there are functional DTG detectors for CueThink, it would be possible to use them to trigger classroom interviews, which allow researchers to approach, observe, and interview students at the moment a construct of interest occurs. In the context of DGTs, this could lead to a deeper understanding of this behavior and the needs of the students who exhibit it.

In addition to deeper investigations of DTG behaviors, future work should also explore ways to improve model performance and to validate it across different learning domains and student populations. These approaches should likely include:

Improving data size and quality: Researchers should provide high-fidelity data when using MLLMs for automated coding. Since large language models cannot reliably predict uncertainty or provide confidence intervals for their labeling [46], they cannot effectively mark boundary cases where human judgment would naturally acknowledge ambiguity.

Domain knowledge integration: The model would benefit from a lightweight domain ontology for common math tools and their typical usage patterns (e.g., protractor → angle labels; tally marks → counting). Additionally, implementing consistency checks to ensure tool presence co-occurs with task-relevant structure (labels, measurements, aligned marks), enhancing text understanding for semantic relevance to flag unrelated content as DTG, and training with negative examples showing decorative or purposeless tool use would improve contextual understanding.

Experimenting with fine-tuning: OCR capabilities can be enhanced through fine-tuning via visual instruction tuning specifically for handwritten mathematical content [27]. This targeted fine-tuning could address the model's current sensitivity to cursive text, small-scale features, and overlapping elements. However, this would require a much larger dataset, and is computationally expensive.

4.3 Conclusion

This study developed and validated a multimodal large language model (MLLM)–based detector for Disengagement from Task Goal (DTG) behavior using LLaVA (Large Language and Vision Assistant) on screen recordings of middle school students solving math problems on the CueThink platform. Among the prompting strategies evaluated, the combination of role assignment and chain-of-thought reasoning with an "analyze and assign" framing achieved the highest performance ($\kappa = 0.71$, precision = 77.6%, recall = 76.4%). These results suggest that an off-the-shelf, pretrained MLLM can detect DTG behavior in visual data with acceptable performance, without the extensive feature engineering and large annotated datasets traditionally required by classical machine learning approaches.

Analysis of the misclassified instances helped us identify key model limitations and inform areas for future improvement. DTG was negatively correlated with learning gains from winter to spring, with a marginally significant association, suggesting that the negative effects of DTG may become more apparent later in the academic year.

Overall, the ability to detect DTG in screen recordings of learners' interactions with a math platform opens new possibilities for analyzing behavior and delivering timely support in digital learning environments. This could help ensure that students engage meaningfully with assigned tasks and ultimately achieve better learning outcomes. As MLLMs continue to improve, they hold considerable potential to make large-scale educational video analysis more reliable and scalable.

5. ACKNOWLEDGMENTS

The authors would like to thank the CueThink team for their partnership and collaboration on this work. The authors also acknowledge Yiqiu Zhou and Maciej Pankiewicz for their support in setting up the model.

This research was supported by NSF Award #DRL-2300829. Any opinions, findings, and conclusions expressed in this paper are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] Appleton, J. J., Christenson, S. L., & Furlong, M. J. (2008). Student engagement with school: Critical conceptual and methodological issues of the construct. Psychology in the Schools, 45(5), 369-386.

[2] Arroyo, I., Ferguson, K., Johns, J., Dragon, T., Meheranian, H., Fisher, D., Barto, A., Mahadevan, S., & Woolf, B. P. (2007). Repairing disengagement with non-invasive interventions. Proceedings of the 13th International Conference on Artificial Intelligence in Education (pp. 195–202)

[3] Ashcraft, M. H., & Kirk, E. P. (2001). The relationships among working memory, math anxiety, and performance. Journal of Experimental Psychology: General, 130(2), 224–237.

[4] Baker, R. S. (2007). Modeling and understanding students' off-task behavior in intelligent tutoring systems. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 1059-1068).

[5] Baker, R.S., Corbett, A.T., Koedinger, K.R., Wagner, A.Z.: Off-Task Behavior in the Cognitive tutor classroom: when students "Game The System". In: Proceedings of ACM CHI 2004: Computer-Human Interaction, pp. 383–390. Vienna, Austria (2004)

[6] Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal statistical society: series B (Methodological), 57(1), 289-300.

[7] Braun, V., & Clarke, V. (2022). Conceptual and design thinking for thematic analysis. Qualitative psychology, 9(1), 3.

[8] Buckley, B., Gobert, J., Horwitz, P., & O'Dwyer, I. (2010) Looking inside the black box:assessments and decision-making in BioLogica. International Journal of Learning Technology, 5 (2), 166190.

[9] Carey, E., Hill, F., Devine, A., & Szűcs, D. (2017). The modified abbreviated math anxiety scale: A valid and reliable instrument for use with children. Frontiers in Psychology, 8, 11.

[10] Christenson, S. L., Reschly, A. L., & Wylie, C. (2012). Handbook of research on student engagement. Springer.

[11] Cipora, K., Szczygieł, M., Willmes, K., & Nuerk, H. C. (2015). Math anxiety assessment with the AMAS: Applicability and usefulness: Insights from the Polish adaptation. Frontiers in Psychology, 6, 1833

[12] Cocea, M., Hershkovitz, A., Baker, R.S.J.d.: The Impact of Off-task and Gaming Behaviors on Learning: Immediate or Aggregate? In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, 507—514 (2009).

[13] Craig, S. D., Xie, J., Huang, X., Graesser, C., & Hu, X. (2014). The impact of epistemological beliefs on student interactions with an intelligent tutoring system. Proceedings of the International Conference on Intelligent Tutoring Systems (pp. 660–661). Springer.

[14] Curriculum Associates. (2017). i-Ready diagnostic & instruction: Reading and mathematics. North Billerica, MA: Author

[15] Dent, A. L., & Koenka, A. C. (2016). The relation between self-regulated learning and academic achievement across childhood and adolescence: A meta-analysis. Educational Psychology Review, 28, 425–474.

[16] Derry, S. J., Pea, R. D., Barron, B., Engle, R. A., Erickson, F., Goldman, R., Hall, R., Koschmann, T., Lemke, J. L., Sherin, M. G., & Sherin, B. L. (2010). Conducting Video Research in the Learning Sciences:Guidance on Selection, Analysis, Technology, and Ethics. Journal of the Learning Sciences, 19(1), 3–53.

[17] Duda, J. L., & Nicholls, J. G. (1992). Dimensions of achievement motivation in schoolwork and sport. Journal of educational psychology, 84(3), 290.

[18] Foster, J. K., Korban, M., Youngs, P., Watson, G. S., & Acton, S. T. (2024). Automatic classification ofactivities in classroom videos.Computers and Education: Artificial Intelligence,6, 1-13, Article 100207.https://doi.org/10.1016/j.caeai.2024.100207

[19] Fredricks, J. A., Blumenfeld, P. C., & Paris, A. H. (2004). School engagement: Potential of the concept, state of the evidence. Review of educational research, 74(1), 59-109.

[20] Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., ... & Sun, X. (2025). Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In Proceedings of the Computer Vision and Pattern Recognition Conference (pp. 24108-24118).

[21] Glazewski, K. D., Hmelo-Silver, C. E., & Lester[1], J. C. (2020, July). Detecting Off-Task Behavior from Student. In Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part I (Vol. 12163, p. 55). Springer Nature.

[22] Gobert, J. D., Baker, R. S., & Wixon, M. B. (2015). Operationalizing and detecting disengagement within online science microworlds. Educational Psychologist, 50(1), 43-57.)

[23] Huang, Q., Dong, X., Zhang, P., Wang, B., He, C., Wang, J., ... & Yu, N. (2024). Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 13418-13427).

[24] Kim, B., Zyromski, B., Mariani, M., Lee, S. M., & Carey, J. C. (2017). Establishing the factor structure of the 18-item version of the junior metacognitive awareness inventory. Measurement and Evaluation in Counseling and Development, 50(1-2), 48-57.

[25] Kloosterman, P., & Stage, F. K. (1992). Measuring beliefs about mathematical problem solving. School Science and Mathematics, 92(3), 109–115. https://doi.org/10.1111/j.1949-8594.1992.tb12154.x

[26] Levy, M., Jacoby, A., & Goldberg, Y. (2024). Same task, more tokens: the impact of input length on the reasoning performance of large language models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 15339-15353).

[27] Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2023). Visual instruction tuning. Advances in neural information processing systems, 36, 34892-34916.

[28] Liu, F., Lin, K., Li, L., Wang, J., Yacoob, Y., & Wang, L. (2024). Mitigating Hallucination in Large Multi-Modal Models via Robust Instruction Tuning (arXiv:2306.14565).

[29] Liu, X, Zhou, Y., Wei, Z., Baker, R.S., Barany, A., Ocumpaugh, J. (2026). Supporting Qualitative Video Analysis with Multimodal Large Language Models. In Journal of Computer Assisted Learning (in press).

[30] Liu, X., Wei, Z., Baker, R. S., Metcalf, S. J., Zhang, J., Barany, A., Slater, S., Swanson, L. & Gagnon, D. J. (2025). Integrating large language models and machine learning to detect struggle in educational games. In International Conference on Artificial Intelligence in Education (pp. 398-405). Cham: Springer Nature Switzerland.

[31] Marx, J. D., & Cummings, K. (2007). Normalized change. American Journal of Physics, 75(1), 87-91.

[32] Mills, C., D'Mello, S., Bosch, N., & Olney, A. M. (2015, June). Mind wandering during learning with an intelligent tutoring system. In International conference on artificial intelligence in education (pp. 267-276). Cham: Springer International Publishing.

[33] Ramey, K. E., Champion, D. N., Dyer, E. B., Keifert, D. T., Krist, C., Meyerhoff, P., Villanosa, K., & Hilppö, J. (2016). Qualitative Analysis of Video Data: Standards and Heuristics.

[34] Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey. Wiley interdisciplinary reviews: Data mining and knowledge discovery, 10(3), e1355.

[35] Sabourin, J. L., Rowe, J. P., Mott, B. W., & Lester, J. C. (2013). Considering Alternate Futures to Classify Off-Task Behavior as Emotion Self-Regulation: A Supervised Learning Approach. Journal of Educational Data Mining, 5(1), 9-38.

[36] San Pedro, M. O., Baker, R., Bowers, A., & Heffernan, N. (2013). Predicting college enrollment from student interaction with an intelligent tutoring system in middle school. In Educational Data Mining 2013.

[37] Skinner, E. A., & Pitzer, J. R. (2012). Developmental dynamics of student engagement, coping, and everyday resilience. In Handbook of research on student engagement, 21-44. Boston, MA: Springer US.

[38] Sperling, R. A., Howard, B. C., Miller, L. A., & Murphy, C. (2002). Measures of children's knowledge and regulation of cognition. Contemporary Educational Psychology, 27, 51–79.

[39] Tobin, T. J., & Sugai, G. M. (1999). Using sixth-grade school records to predict school violence, chronic discipline problems, and high school outcomes. Journal of emotional and Behavioral Disorders, 7(1), 40-53.

[40] Trad, F., & Chehab, A. (2025). Evaluating the efficacy of prompt-engineered large multimodal models versus fine-tuned vision transformers in image-based security applications. ACM Transactions on Intelligent Systems and Technology, 16(4), 1-22.

[41] Winne PH and Hadwin AF. Hacker DJ, Dunlosky J, and Graesser A. (1998). Studying as self-regulated learning. Metacognition in educational theory and practice. Hillsdale, NJ Lawrence Erlbaum, 277-304.

[42] Wixon, M., Baker, R. S. D., Gobert, J. D., Ocumpaugh, J., & Bachmann, M. (2012). WTF? detecting students who are conducting inquiry without thinking fastidiously. In International Conference on User Modeling, Adaptation, and Personalization (pp. 286-296). Berlin, Heidelberg: Springer Berlin Heidelberg.

[43] Wong, Z. Y., & Liem, G. A. D. (2022). Student engagement: Current state of the construct, conceptual refinement, and future research directions. Educational Psychology Review, 34(1), 107-138.

[44] Wu, J., Gan, W., Chen, Z., Wan, S., & Yu, P. S. (2023). Multimodal Large Language Models: A Survey. 2023 IEEE International Conference on Big Data (BigData), 2247–2256.

[45] Xiao, Z., Yuan, X., Liao, Q. V., Abdelghani, R., & Oudeyer, P. Y. (2023). Supporting qualitative analysis with large language models: Combining codebook with GPT-3 for deductive coding. In Companion proceedings of the 28th international conference on intelligent user interfaces (pp. 75-78).

[46] Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., & Hooi, B. (2023). Can llms express their uncertainty? An empirical evaluation of confidence elicitation in llms. arXiv preprint arXiv:2306.13063.

[47] Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., & Chen, E. (2024). A Survey on Multimodal Large Language Models (arXiv:2306.13549). arXiv. http://arxiv.org/abs/2306.13549

[48] Zhang, D., Yu, Y., Dong, J., Li, C., Su, D., Chu, C., & Yu, D. (2024). MM-LLMs: Recent Advances in MultiModa Large Language Models (arXiv:2401.13601). arXiv. http://arxiv.org/abs/2401.13601

[49] Zhang, J., Andres, J. M. A. L., Hutt, S., Baker, R. S., Ocumpaugh, J., Nasiar, N., ... & Young, T. (2022). Using machine learning to detect SMART model cognitive operations in mathematical problem-solving process. Journal of Educational Data Mining, 14(3), 76-108.

[50] Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., & Li, C. (2024a). Video instruction tuning with synthetic data (arXiv Preprint No. arXiv:2410.02713).

[51] Zhou, Y., Kang, J., & Nguyen, H. (2025). Can We Trust Large Language Models for Video Analysis: An Exploration of Hallucination in Multimodal LLMs. In Proceedings of the 19th International Conference of the Learning Sciences-ICLS 2025, pp. 110-118. International Society of the Learning Sciences.

Columns on Last Page Should Be Made as Close As Possible to Equal Length

Multimodal Learning Analytics for Programming: Cognitive Difficulties, Attention Dynamics, and Physiological Regulation

Huiyong Li
Kyushu University
li.huiyong.194@m.kyushu-u.ac.jp

Boxuan Ma
Kyushu University
boxuan@artsci.kyushu-u.ac.jp

ABSTRACT

AAT-based programming practice often relies on log outcomes that obscure learners' underlying cognitive difficulties, behavioral workflows, and regulation dynamics. This study synchronizes submission logs, eye tracking, and wearable physiology to examine how these processes unfold during authentic AAT-based programming and how they differ by prior knowledge. Twenty-two university students completed a 35-minute session with four beginner Python questions while wearing a Pupil Core eye tracker and an EmbracePlus wristband. We coded attempt-level syntactic, conceptual, and strategic difficulties, annotated seven task-relevant behavioral states from screen–gaze recordings, and analyzed physiological signals across preparation and programming stages. Results showed that low prior-knowledge learners produced more syntactic and conceptual difficulty attempts, cycled more often between learning materials, feedback, and code revision, and showed larger reductions in physiological regulation during programming. These findings demonstrate how temporally aligned multimodal evidence can move AAT analytics beyond outcome tracking toward process-sensitive learner models that adapt support to cognitive difficulty, behavioral transition patterns, and regulation states.

Keywords

Introductory programming, automated assessment tools, learning analytics, eye-tracking, physiological regulation

1. INTRODUCTION

Learning to program involves progressing from mastering language features, to forming domain concepts, and ultimately to solving novel problems [34, 28, 21]. In large programming courses, scaling such formative practice increasingly relies on automated assessment tools (AATs) embedded in learning management systems. AATs execute submissions against predefined test cases and return error messages, failing test indicators, and scores [22], that making practice frequent, measurable, and comparable across students.

Huiyong Li, and Boxuan Ma. Multimodal Learning Analytics for Programming: Cognitive Difficulties, Attention Dynamics, and Physiological Regulation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 516–523. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039831

This fast feedback has enabled a growing body of computing education and educational data mining research on engagement, persistence, and performance in AAT-based settings [8]. However, the same observable trace can be produced by fundamentally different cognitive states: a student may repeatedly fail due to a minor syntax slip, a fragile misconception, or an unproductive strategy (e.g., trial-and-error), yet these processes can appear similar in logs. In this sense, log traces primarily capture what learners did and whether it worked, offering limited evidence about how and why those behaviors unfold. Complementary evidence streams point to this missing process layer. Prior work suggests that programming success and failure are associated with distinct visual attention strategies [35], and physiological sensing can index variations in cognitive and affective effort during code comprehension [7]. However, in AAT-based problem solving, explaining why superficially similar log traces arise often requires linking observable actions to learners' attention allocation and regulation dynamics at a fine temporal granularity. Multimodal learning analytics (MMLA) provides a methodological basis for synchronizing and triangulating these heterogeneous streams [27]. Yet such synchronized, fine-grained integration remains uncommon in authentic AAT task solving, where behavioral traces, attention dynamics, and regulation signals must be aligned to reveal process mechanisms rather than outcomes alone.

Despite the data richness of AAT-based programming learning, three gaps remain. First, prior work typically analyzes a single modality in isolation, limiting explanatory power because each stream is ambiguous on its own [36, 18]. Second, process-level differentiation of learners remains underspecified: we still lack sequence-sensitive characterizations of behavioral states, which cognitive difficulties emerge and how they co-occur with strategy-relevant behaviors across problem-solving episodes [16, 14]. Third, physiological regulation dynamics are rarely examined alongside fine-grained programming behaviors in authentic AAT task solving, leaving unclear whether and how physiological regulation signals difficulty, supports recovery, or differentiates effective versus ineffective strategy shifts.

To address these gaps, we integrate log-based cognitive difficulty coding, eye-tracking attention dynamics, and physiological regulation measures to characterize programming processes and contrast high and low prior-knowledge learners. We address the following research questions:

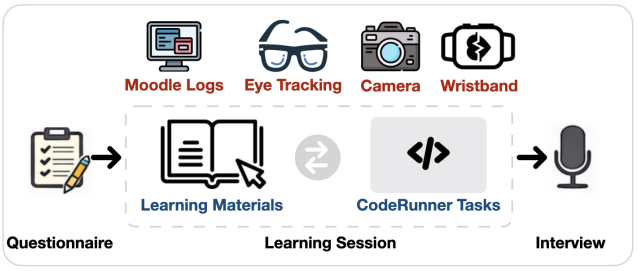


Figure 1: Overview of the study procedure.

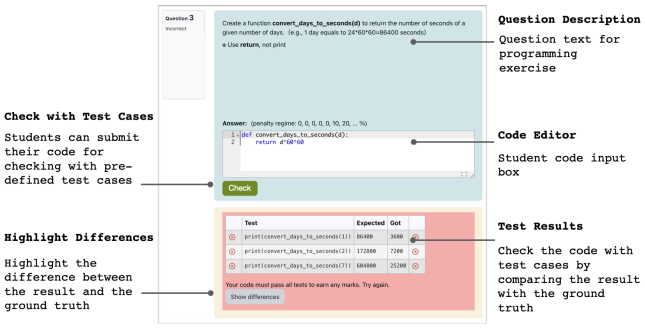


Figure 2: CodeRunner interface in Moodle.

RQ1: What cognitive difficulties (syntactic, conceptual, and strategic) emerge in programming tasks, and how do they differ between high and low prior-knowledge learners?

RQ2: What visual attention and behavioral transitions across task-relevant states emerge, and how do they differ between high and low prior-knowledge learners?

RQ3: What physiological regulation patterns/indices emerge, and how do they differ between high and low prior-knowledge learners?

2. RELATED WORK

Automated assessment tools are widely used in introductory programming because executable submissions enable scalable, consistent evaluation and rapid feedback [2, 13]. In the EDM and LA field, AAT logs (e.g., attempts, compiler outcomes, timestamps, and pass or fail test results) have been mined to model performance, persistence, and iterative behaviors in problem-level analyses [20, 1, 16, 15]. However, similar log traces can reflect different underlying processes, motivating process-oriented modeling beyond outcome indicators.

To interpret these ambiguous traces, computing education research commonly adopts a multi-layer knowledge framework with syntactic knowledge of language features, conceptual understanding of execution and constructs, and strategic knowledge for problem solving and debugging [21, 28, 34]. Large-scale analyses of submission corpora further show that student errors span additional categories (e.g., misinterpretation or carelessness), cautioning that code-only evidence may under-specify underlying causes [3]. Recent work operationalizes closely related difficulty categories (syntactic, conceptual, strategic) with explicit definitions and expert labeling [12], while semi-automated approaches such as code-embedding-based clustering have been proposed to discover problem-specific misconceptions from incorrect submissions at scale [31]. These lines of work support interpretable difficulty characterization, but typically do not connect difficulties to concurrent attention allocation and regulation dynamics.

To bridge this gap, complementary modalities provide process evidence that logs alone cannot. Eye-tracking studies in programming report systematic differences in attention strategies across skill levels and performance, motivating transition-based analyses over task-relevant regions and representations [25, 11, 5]. Physiological sensing has also been used to estimate cognitive effort and affect during programming and code comprehension (e.g., via EDA and heart-rate-related measures), with validation work supporting wearable instrumentation for more naturalistic settings [7, 23, 10]. Methodologically, multimodal learning analytics provides frameworks for synchronizing and fusing heterogeneous streams, highlighting the role of fusion design choices for validity and interpretability and showing benefits over single-modality models [26, 4, 24, 9, 33, 27].

While prior multimodal work demonstrates feasibility in programming education [19], relatively few studies jointly link AAT log-derived cognitive difficulties, attention transitions, and physiological regulation in authentic AAT-based programming task solving. Building on these strands, our study synchronizes AAT logs, eye movements, and wearable physiology to examine how these process indicators co-occur and differ by prior knowledge.

3. METHOD

3.1 Participants

Twenty-two university students voluntarily participated in this study (14 males and 8 females, with an average age of 24). Participants were recruited through campus announcements at a national university in Japan and included both undergraduate and graduate students from diverse academic disciplines. Programming experience ranged from no prior experience to nine years, covering both novice and experienced learners. The study protocol was reviewed and approved by the authors' university Research Ethics Committee and conducted in accordance with institutional guidelines for human-participant research. Participants provided written informed consent and could withdraw at any time.

3.2 Study Design and Learning Tasks

The overview of the study procedure is shown in Figure 1. This study was conducted in a controlled laboratory environment. Firstly, the students completed a pre-test and pre-questionnaire after a brief experimental introduction. Then the students were asked to access an introductory programming course in the Moodle learning management system (LMS). They were asked to read the lecture's introductory materials, which covered fundamental concepts in Python programming (e.g., basic functions) and solve four beginner-level programming questions using CodeRunner, an open-source Moodle plugin automated assessment tool [17]. Students were given the flexibility to complete the task at their

Table 1: Coding scheme for student cognitive difficulties, descriptions, and examples.

Difficulty	Description	Example
Syntactic	Misunderstanding of syntactic facts related to a particular programming language or failure on applying rules of syntax.	(1) Missing colon in `def` function header: `def print_morning print("")` (2) Non-standard symbols (fullwidth parentheses in function) (3) Wrong newline escape (uses /n instead of \n)

Conceptual	Misunderstanding of programming constructs or execution semantics.	(1) `return` vs. `print` confusion (prints but should return):
		`def convert_days_to_seconds(d): print(d*86400)`
		(2) Identifier misuse (uses undeclared name):
		`def repeat_five_times(s): print(sssss)`
		(3) Misunderstanding "repeat 5 times" (single call / one line):
		`def repeat_five_times(s): print(s,s,s,s,s)`

Strategic	Difficulties applying programming to solve novel problems by design, coding, and test.	(1) Task confusion (wrong function name than specified) `def print_morning(): print("...") ...` (2) Output-format planning issue (unwanted newlines): `for i in range(5): print(s)` (*missing* `end=""`)

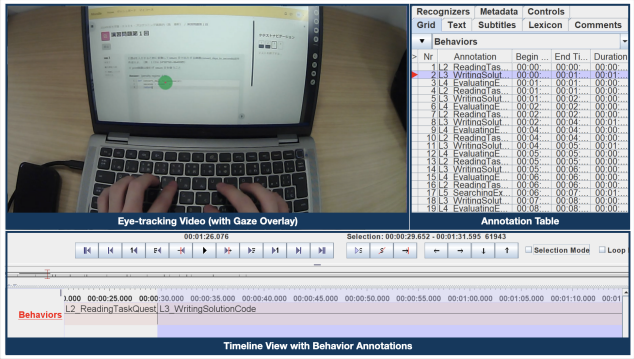


Figure 3: ELAN software used to annotate learning behaviors from eye-tracking videos.

own pace within a 35-minute learning session. An example CodeRunner question is shown in Figure 2. They could read the problem statement, input their code, and submit their solutions to be automatically checked against pre-defined test cases.

Students wore a 200-Hz Pupil Core eye tracker while solving programming tasks. Pupil Core (Pupil Labs GmbH, Berlin, Germany) is a pair of eye-tracking glasses endowed with two infrared real-time eye cameras and an environmental camera focused straight ahead of the screen. The eye-tracking screen data was recorded from the cameras and processed via the Pupil Player software later. Students also wore an EmbracePlus wristband before and during the learning session. They spent approximately 20 minutes in the preparation stage before they started 35-minute programming tasks. EmbracePlus wristband is a medical-grade wearable device that offers real-time physiological data collection [6]. The wristband embeds 3-axis accelerometer, electrodermal activity, temperature, and gyroscope sensors. It is a non-invasive device worn like a wristwatch, minimizing any discomfort during usage.

After completing the learning session, all participants took part in semi-structured, face-to-face interviews. The interviews probed students' learning experiences, focusing on perceived successes as well as aspects of the learning process that required improvement.

3.3 Data Collection and Analysis

Students' prior knowledge of Python programming was assessed using a 10-item pre-test covering core concepts, including data types, control flow, functions, and basic syntax in Python. Based on their pre-test scores, students were classified into two prior-knowledge groups: high prior-knowledge who scored 8 or higher out of 10 ($n = 10$), and low prior-knowledge who scored 6 or lower ($n = 12$). The cut-off leverages a natural gap in the distribution (no student scored 7), producing clearly separated groups for subsequent multimodal comparisons.

A pre-questionnaire was administered to collect participants' demographic information, including gender, age, and academic major.

All student interactions with the Moodle platform were logged automatically. CodeRunner problem-solving behavior was captured in learning logs, including assessment logs, submitted codes, and system feedback. The raw logs contained question-level information, including question contents, question access timestamps, and question start times, as well as attempt-level step histories for each learner and question. Each step record included the step number, timestamp, action type, submitted answer, correctness state, score, and CodeRunner feedback such as error messages. These learning logs were classified into cognitive difficulties including syntactic, conceptual, and strategic types, adopted a prior conceptual framework [28]. The coding scheme is summarized in Table 1.

Students' behaviors during programming tasks were captured using eye-tracking video recordings. The raw eye-tracking data consisted of time-stamped gaze recordings collected with the Pupil Core eye tracker and synchronized screen recordings of students' interactions with the Moodle-based CodeRunner environment. Fixations were automati-

Table 2: Coding scheme of student learning behaviors during programming tasks.

Code	Behavior	Description
Lecture	Watching Lecture Material	Students viewed the instructional materials provided in the Moodle course.
Question	Reading Task Question	Students read and examined the programming task statement.
Code	Writing Solution Code	Students wrote program code in the coding box to solve the task.
Fail	Execution Results Fail	Students submitted their code for automated assessment, but it failed one or more test cases or returned an execution error.
Pass	Execution Results Pass	Students submitted their code for automated assessment, and it successfully passed all test cases.
Search	Searching External Resources	Students searched the internet for task-related information, programming concepts, or error explanations. Use of generative AI tools like ChatGPT was not allowed.
Note	Taking Notes	Students wrote notes or sketched ideas on paper during the learning session.

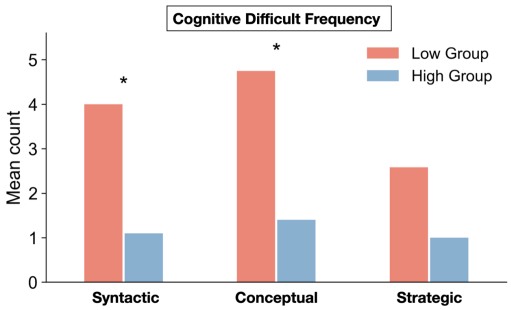


Figure 4: The distribution of cognitive difficulties faced by learners with high and low prior-knowledge.

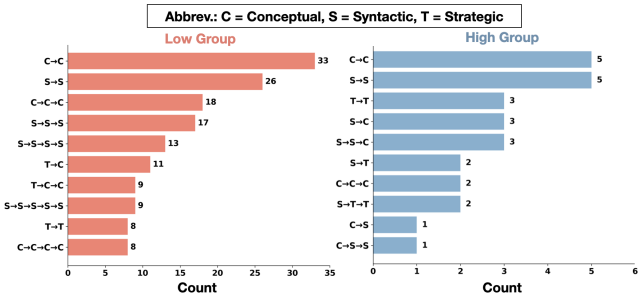


Figure 5: Top 10 most frequent cognitive transition sequences by group.

cally extracted in Pupil Player using the dispersion-threshold identification (I-DT) algorithm, with a dispersion threshold of 1.5° visual angle and a minimum fixation duration of 100 ms [29]. The synchronized screen and gaze videos were manually annotated to extract observable programming behaviors as shown in Figure 3. Following prior research on SRL in programming [32], seven programming behaviors were coded as summarized in Table 2. These behaviors captured how students engaged with lecture materials, understood problem requirements, created and evaluated solutions, sought external information, and noted their thinking processes. The annotated behaviors were used to compute behavioral frequency, duration, and transitions, enabling comparison of behavioral dynamics across different task contexts. Specifically, temporal behavioral sequences were extracted at the programming question level, with each programming question treated as an independent learning episode.

To compare programming behaviors between groups, we analyzed both overall activity and temporal dynamics. For frequency and duration measures, we used independent-samples t-tests or Wilcoxon rank-sum tests after a normality check. To model behavioral dynamics, we applied Transition Network Analysis (TNA) [30], representing each learner's question-level sequence as a weighted directed network (nodes = behavioral states; edges = transition probabilities). Transition probabilities were aggregated within each group. Between-group differences in transition probabilities were tested using Fisher's exact test.

Students' physiological signals were collected using the EmbracePlus wristband during the learning session. The raw data consisted of time-stamped electrodermal activity (EDA) and blood volume pulse (BVP) streams synchronized with the task timeline. Because wrist-based physiological data are sensitive to movement artifacts and physiological responses can appear with a delay after cognitive or affective events, we used a window-based analysis. Specifically, signals were segmented into non-overlapping 1-minute windows and aligned with two task stages: preparation and programming. We derived four features: (1) tonic EDA as log-transformed skin conductance level (log(SCL)); (2) phasic activity as log1p-transformed skin conductance response rate (log1p(SCR/min)); (3) heart rate as mean beats per minute (BPM); and (4) heart-rate variability as log-transformed RMSSD (log(RMSSD)). To reduce between-person baseline differences, we computed within-person z-scored composites: (5) an *Arousal Index* as the mean of z-scored log(SCL) and log1p(SCR/min), and (6) a *Regulation Index* as z-scored log(RMSSD). We tested main effects and interactions of *Group* (low vs. high prior knowledge) and *Stage* (preparation vs. programming) on each feature and index using mixed-effects models.

4. RESULTS

4.1 Log-based Cognitive Difficulties (RQ1)

Figure 4 shows the distribution of cognitive difficulties in the syntactic, conceptual and strategic types faced by learners with high and low prior-knowledge. Low prior-knowledge students produced significantly more *Syntactic* (U = 97.0, p = .037, r = 0.62) and *Conceptual* (U = 96.5, p = .037, r = 0.61) difficulty attempts than high prior-knowledge stu-

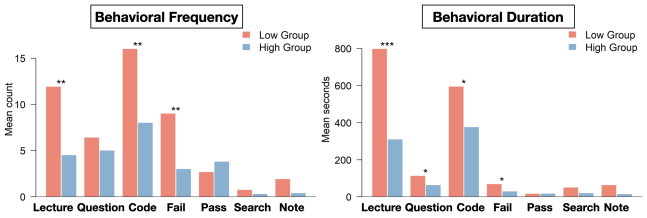


Figure 6: Programming behavioral differences in frequency and duration between high and low prior-knowledge learners.

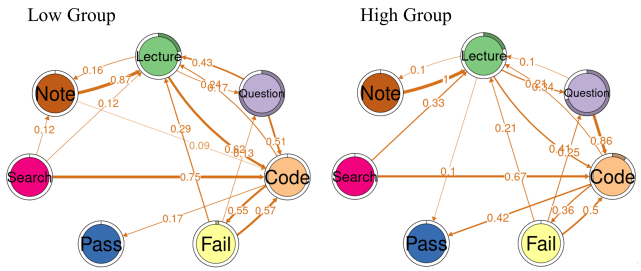


Figure 7: Transition network of programming behaviors between high and low prior-knowledge learners. Nodes represent each coded behavior. Edges represent transition probabilities between nodes.

dents. *Strategic* differences trend higher for low prior-knowledge students but were not significant (U = 83.5, p = .119).

We further analyzed sequence patterns by extracting learner's attempt-level contiguous subsequences. Mann–Whitney U tests show no subsequence differences that remained significant ($p_{\text{Holm}} = 1.0$ for all). Low prior-knowledge learners exhibited more persistence of conceptual difficulty, more longer repeated syntactic failures, and more strategic-to-conceptual shifts, whereas high prior-knowledge learners show fewer failure episodes overall and fewer transitions. Figure 5 shows top 10 most frequent cognitive transition sequences for low (left) and high (right) prior-knowledge groups.

4.2 Eye-tracking Attention Dynamics (RQ2)

Figure 6 shows the programming behavioral differences in frequency and duration between high and low prior-knowledge learners. Results show that low prior-knowledge learners spent significantly more time on Lecture viewing (M = 797s vs. 310s, p < .001), Question reading (M = 113s vs. 63s, p = .017), Code writing (M = 594s vs. 376s, p = .033), and examining Fail results (M = 69s vs. 29s, p = .012). Notably, Lecture duration showed the largest effect size (d = 1.81), indicating that low-group learners spent more than double the time revisiting lecture materials.

Figure 7 shows the transition network of programming behaviors between high and low prior-knowledge learners. Seven significant transition differences were found between low and high prior-knowledge learners using Fisher's exact test. The high prior-knowledge group followed a more execution-oriented path, with higher *Question*→*Code* (86.0% vs. 50.6%; $p = 5.1 \times 10^{-5}$) and *Code*→*Pass* (42.5% vs. 16.8%; $p = 1.9 \times 10^{-5}$), and fewer *Code*→*Fail* transitions (36.2% vs. 55.5%; $p = 5.0 \times 10^{-3}$). In contrast, the low prior-knowledge group relied more on materials, showing higher *Question*→*Lecture* (42.9% vs. 10.0%; $p = 5.9 \times 10^{-5}$) and *Lecture*→*Code* (62.2% vs. 41.5%; $p = 3.0 \times 10^{-2}$). Lecture use also differed in integration: high prior-knowledge learners more often returned to refine task understanding (*Lecture*→*Question*: 34.1% vs. 17.0%; $p = 2.8 \times 10^{-2}$) and were more likely to reach success after consulting materials (*Lecture*→*Pass*: 9.8% vs. 0.0%; $p = 2.6 \times 10^{-3}$). Overall, high prior-knowledge learners progressed more directly from problem understanding to coding and success, whereas low prior-knowledge learners showed a lecture-mediated and more failure-prone workflow.

4.3 Physiological Regulation Signals (RQ3)

Figure 8 presents minute-level physiological trajectories (panels a–d) and Group × Stage comparisons (panels e–h) for low and high prior-knowledge learners. Panels (a)–(d) show GAM-smoothed group trajectories with ±1 SEM bands for tonic EDA [log(SCL)], phasic SCR rate [log1p(SCR/min)], heart rate (BPM), and RMSSD [log(RMSSD)], respectively. Mixed-effects models controlling for MET indicated a robust main effect of stage for three features: tonic EDA increased from Preparation to Programming ($b = 0.33$, $p < .001$), phasic SCR rate increased ($b = 0.24$, $p < .001$), and RMSSD decreased ($b = -0.36$, $p < .001$), whereas heart rate showed no stage shift ($b = -0.12$, $p = .859$). A significant Group x Stage interaction was observed for phasic SCR rate (b = -0.21, p = .006; panel f), indicating a larger Programming-related increase for low prior-knowledge learners than for high prior-knowledge learners. No Group × Stage interactions were found for tonic EDA ($p = .680$), heart rate ($p = .260$), or RMSSD ($p = .095$).

Figure 9 summarizes the physiological results using within-person standardized, minute-level composite indices. Panels (a–b) show GAM-smoothed trajectories (±1 SEM) for the Arousal Index, combining tonic SCL and phasic SCR activity, and the Regulation Index, based on RMSSD. These smoothed trajectories indicate relative changes in physiological activation and regulation across the preparation and programming stages. Panels (c–d) show the mixed-effects model estimates for Group × Stage effects. Arousal increased from Preparation to Programming ($b = 0.59$, $p < .001$), with no Group × Stage interaction ($b < 0.01$, $p = .992$), suggesting comparable increases in sympathetic activation for both prior-knowledge groups. Regulation decreased from Preparation to Programming ($b = -0.52$, $p < .001$), indicating lower RMSSD-based regulation during active programming. This decrease was qualified by a significant Group × Stage interaction ($b = 0.40$, $p < .001$): low prior-knowledge learners showed a larger programming-related reduction in the Regulation Index than high prior-knowledge learners.

5. DISCUSSION

In this study, we conduct a multimodal study that synchronizes submission logs, eye tracking, and wearable physiology to characterize how cognitive difficulties, visual attention transitions, and physiological regulation unfold during an AAT-based programming learning environment and how they differ by prior knowledge. Across learning log, attention-based behavior, and physiological signal, the re-

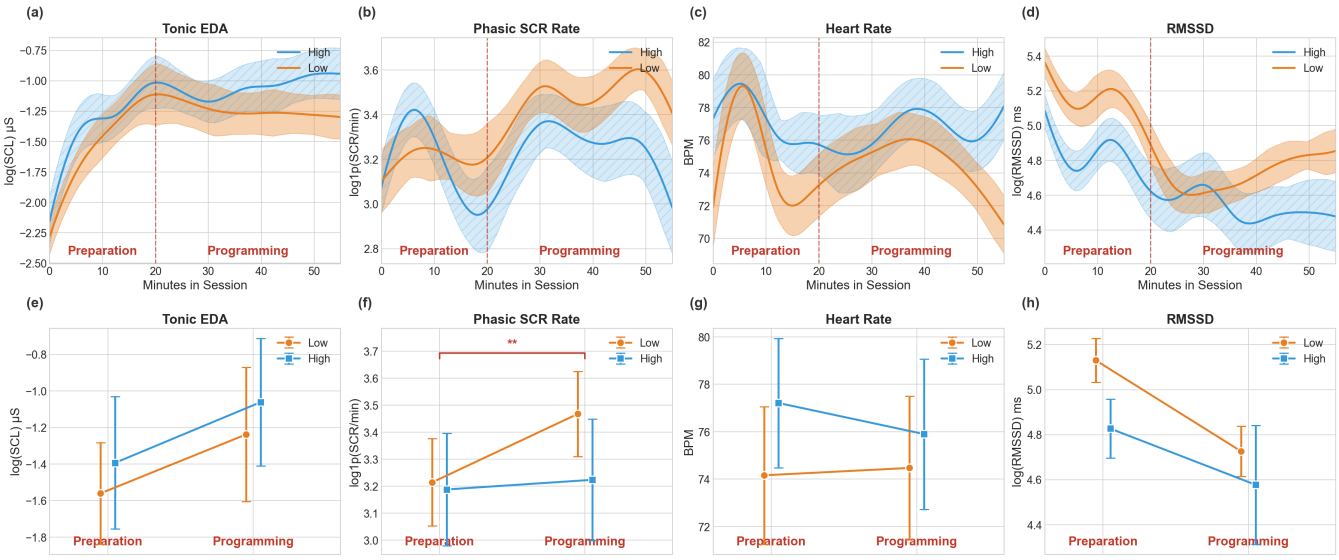


Figure 8: Physiological trajectories and Group × Stage effects for low and high prior-knowledge learners.

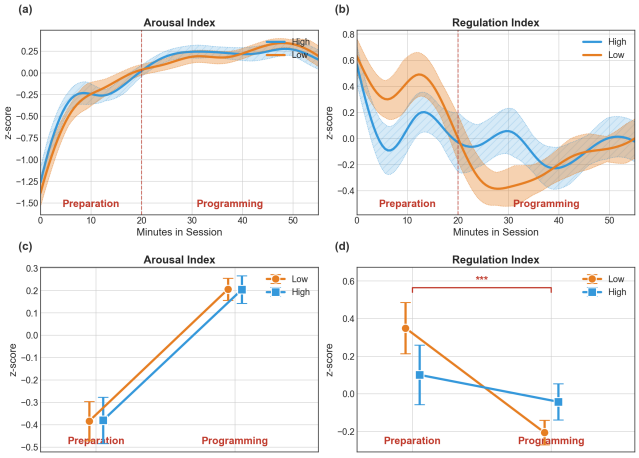


Figure 9: Higher-order physiological indices (within-person z-scored) and Group × Stage effects for low and high prior-knowledge learners.

sults consistently indicate that prior knowledge shapes how learners allocate resources and regulate effort during AAT-based programming.

Syntactic and conceptual difficulties. Low prior-knowledge learners exhibited more frequent difficulty episodes, particularly in *syntactic* and *conceptual* categories, whereas strategic differences were weaker. Sequence-level analyses suggested that low prior-knowledge learners experienced more extended failure episodes (e.g., repeated syntactic failures) and sustained conceptual difficulty, while high prior-knowledge learners showed fewer difficulty episodes and less complex difficulty transitions overall.

Lecture-mediated cycle vs. execution-oriented workflow. Eye-tracking measures revealed that low prior-knowledge learners spent substantially more time consulting learning materials and feedback, including longer lecture viewing, more question rereading, longer code-writing time, and more time inspecting failing outcomes. Transition Network Analysis further showed distinct workflows: high prior-knowledge learners followed a more execution-oriented pathway. Lecture use also differed qualitatively: high prior-knowledge learners more often returned from lecture to the question and were more likely to succeed after lecture consultation.

Increased arousal and reduced regulation during programming. The shift from Preparation to Programming elicited a clear physiological change in both groups: sympathetic arousal increased (higher tonic and phasic EDA) and parasympathetic regulation decreased (lower RMSSD), while heart rate remained comparatively stable. Importantly, group differences emerged in *dynamic regulation* rather than average arousal: low prior-knowledge learners showed a larger programming-related increase in phasic SCR and a larger decrease in the Regulation Index, indicating stronger moment-to-moment arousal reactivity and weaker regulation during programming.

Multimodal analytics based learner model. Taken together, the multimodal results suggest a coherent pattern: low prior-knowledge learners encountered more basic (syntax/concept) breakdowns, engaged in more material- and feedback-driven cycles, and showed heightened physiological reactivity with larger reductions in regulation during programming. In contrast, high prior-knowledge learners progressed more directly from problem understanding to code execution and success, and their lecture use appeared more selective and integrative. These findings motivate learner models and supports that adapt not only to error types but also to workflow dynamics and regulation states during feedback-intensive programming.

Implications for adaptive support and learning design. Our findings may have implications for adaptive support and learning design in programming education. For low prior-

knowledge learners, supports should prioritize (a) syntax-concept remediation (common error patterns, execution semantics checks), and (b) workflow guidance that helps convert lecture consultation into progress (e.g., prompts that connect a specific lecture snippet to a concrete code revision, or structured question rereading before coding). For high prior-knowledge learners, lighter supports that preserve autonomy may suffice (e.g., optional checks for edge cases or output formatting). Physiological findings suggest that regulation signals can complement logs/gaze to flag moments where learners may benefit from brief metacognitive prompts (e.g., pause–plan–test routines) or reduced cognitive load (e.g., simplified feedback, stepwise hints) rather than repeated automatical feedback.

Limitations. Several limitations should be noted. First, the sample was relatively small and drawn from a single university, which may limit the generalizability of the findings. Thus, our results should be interpreted as exploratory, process-level evidence from multimodal triangulation in AAT-based programming. Second, categorizing learners into low and high prior knowledge enabled interpretable contrasts but may obscure graded relationships between prior knowledge and multimodal processes. Third, wrist-worn physiological signals can be noisy, highly individualized, and may lag behind task events and therefore our minute-level aggregation and stage-level comparisons may miss event-locked regulation changes.

6. CONCLUSION

This work contributes temporally aligned evidence linking difficulty types, attention transitions, and regulation dynamics in an authentic AAT setting. It demonstrates the value of synchronizing AAT logs, eye tracking, and wearable physiology to model programming task solving beyond outcome-based indicators. Our findings motivate process- and regulation-aware learner models and targeted supports (e.g., difficulty-specific scaffolds and workflow guidance) to better assist learners who struggle in AAT-based programming environments.

7. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Numbers JP24K20903 and JP25K17078.

8. REFERENCES

[1] A. Ahadi. Early identification of novice programmers' challenges in coding using machine learning techniques. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, pages 263–264, 2016.

[2] K. M. Ala-Mutka. A survey of automated assessment approaches for programming assignments. *Computer science education*, 15(2):83–102, 2005.

[3] E. Albrecht and J. Grabowski. Sometimes it's just sloppiness-studying students' programming errors and misconceptions. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 340–345, 2020.

[4] P. Blikstein and M. Worsley. Multimodal learning analytics and education data mining: Using computational technologies to measure complex learning tasks. *Journal of learning analytics*, 3(2):220–238, 2016.

[5] T. Busjahn, R. Bednarik, A. Begel, M. Crosby, J. H. Paterson, C. Schulte, B. Sharif, and S. Tamm. Eye movements in code reading: Relaxing the linear order. In *2015 ieee 23rd international conference on program comprehension*, pages 255–265. IEEE, 2015.

[6] Empatica. EmbracePlus. `https://www.empatica.com/embraceplus/`, 2026. Accessed: 2026-02-07.

[7] D. Fucci, D. Girardi, N. Novielli, L. Quaranta, and F. Lanubile. A replication study on code comprehension and expertise using lightweight biometric sensors. In *2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC)*, pages 311–322. IEEE, 2019.

[8] H. Gabbay and A. Cohen. Investigating the effect of automated feedback on learning behavior in moocs for programming. *International Educational Data Mining Society*, 2022.

[9] M. N. Giannakos, K. Sharma, I. O. Pappas, V. Kostakos, and E. Velloso. Multimodal data as a means to understand the learning experience. *International Journal of Information Management*, 48:108–119, 2019.

[10] J. Gorson, K. Cunningham, M. Worsley, and E. O'Rourke. Using electrodermal activity measurements to understand student emotions while programming. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1*, pages 105–119, 2022.

[11] P. Hejmady and N. H. Narayanan. Visual attention patterns during program debugging with an ide. In *proceedings of the symposium on eye tracking research and applications*, pages 197–200, 2012.

[12] M. Hoq, J. Vandenberg, B. Mott, J. Lester, N. Norouzi, and B. Akram. Towards attention-based automatic misconception identification in introductory programming courses. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2*, pages 1680–1681, 2024.

[13] P. Ihantola, T. Ahoniemi, V. Karavirta, and O. Seppälä. Review of recent systems for automatic assessment of programming assignments. In *Proceedings of the 10th Koli calling international conference on computing education research*, pages 86–93, 2010.

[14] T. Iwanaga, H. Li, B. Ma, and C. Yin. Visual attention transitions and self-regulated help seeking in programming comprehension. In *Proceedings of the International Conference on Artificial Intelligence in Education 2026 (AIED 2026)*, 2026. In press.

[15] H. Li and B. Ma. Coderunner agent: Integrating ai feedback and self-regulated learning to support programming education. In *Proceedings of the 33rd International Conference on Computers in Education*. Asia-Pacific Society for Computers in Education, 2025.

[16] H. Li, B. Ma, and C. Yin. Examining metacognitive difficulties in learning programming: Analysis of student behavior and strategy. In *Proceedings of the 1st International Conference on Learning Evidence and Analytics*. Asia-Pacific Society for Computers in

Education, 2025.

[17] R. Lobb and J. Harlow. Coderunner: A tool for assessing computer programming skills. *ACM Inroads*, 7(1):47–51, 2016.

[18] S. López-Pernas and M. Saqr. Bringing synchrony and clarity to complex multi-channel data: A learning analytics study in programming education. *IEEE Access*, 9:166531–166541, 2021.

[19] K. Mangaroska, K. Sharma, D. Gaševic, and M. Giannakos. Multimodal learning analytics to inform learning design: Lessons learned from computing education. *Journal of Learning Analytics*, 7(3):79–97, 2020.

[20] J. McBroom, B. Jeffries, I. Koprinska, and K. Yacef. Mining behaviours of students in autograding submission system logs. *International Educational Data Mining Society*, 2016.

[21] T. J. McGill and S. E. Volet. A conceptual framework for analyzing students' knowledge of programming. *Journal of research on Computing in Education*, 29(3):276–297, 1997.

[22] M. Messer, N. C. Brown, M. Kölling, and M. Shi. Automated grading and feedback tools for programming education: A systematic review. *ACM Transactions on Computing Education*, 24(1):1–43, 2024.

[23] N. Milstein and I. Gordon. Validating measures of electrodermal activity and heart rate variability derived from the empatica e4 utilized in research settings that involve interactive dyadic states. *Frontiers in Behavioral Neuroscience*, 14:148, 2020.

[24] S. Mu, M. Cui, and X. Huang. Multimodal data fusion in learning analytics: A systematic review. *Sensors*, 20(23):6856, 2020.

[25] U. Obaidellah, M. Al Haek, and P. C.-H. Cheng. A survey on the usage of eye-tracking in computer programming. *ACM Computing Surveys (CSUR)*, 51(1):1–58, 2018.

[26] X. Ochoa and M. Worsley. Augmenting learning analytics with multimodal sensory data. *Journal of Learning Analytics*, 3(2):213–219, 2016.

[27] H. Ouhaichi, D. Spikol, and B. Vogel. Research trends in multimodal learning analytics: A systematic mapping study. *Computers and Education: Artificial Intelligence*, 4:100136, 2023.

[28] Y. Qian and J. Lehman. Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education (TOCE)*, 18(1):1–24, 2017.

[29] D. D. Salvucci and J. H. Goldberg. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78, 2000.

[30] M. Saqr, S. López-Pernas, T. Törmänen, R. Kaliisa, K. Misiejuk, and S. Tikka. Transition network analysis: A novel framework for modeling, visualizing, and identifying the temporal patterns of learners and learning processes. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 351–361, 2025.

[31] Y. Shi, K. Shah, W. Wang, S. Marwan, P. Penmetsa, and T. Price. Toward semi-automatic misconception discovery using code embeddings. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, pages 606–612, 2021.

[32] L. Silva, A. Mendes, A. Gomes, and G. Fortes. What learning strategies are used by programming students? a qualitative study grounded on the self-regulation of learning theory. *ACM Transactions on Computing Education*, 24(1):1–26, 2024.

[33] M. Worsley, R. Martinez-Maldonado, and C. D'Angelo. A new era in multimodal learning analytics: Twelve core commitments to ground and grow mmla. *Journal of Learning Analytics*, 8(3):10–27, 2021.

[34] S. Xinogalos. Designing and deploying programming courses: Strategies, tools, difficulties and pedagogy. *Education and Information Technologies*, 21(3):559–588, 2016.

[35] W. Yang and U. Obaidellah. Attention dynamics in programming: Eye gaze patterns of high-vs. low-ability novice coders. In *Proceedings of the 2024 Symposium on Eye Tracking Research and Applications*, pages 1–6, 2024.

[36] A. F. Zambrano, J. Zhang, M. Pankiewicz, and R. S. Baker. Practice as the key to success: Understanding the role of prior knowledge, affective states and learning resources in computer science education. In *Proceedings of the LAK26: 16th International Learning Analytics and Knowledge Conference*, pages 85–95, 2026.

Simulating Learners' Task-Selection Strategies and System Constraints in Mastery Learning

Haley Noh
New York University
hhn2020@nyu.edu

Aarna Chowdhary
Carnegie Mellon University
achowdha@andrew.cmu.edu

Jeroen Ooge
Utrecht University
j.ooge@uu.nl

Vincent Aleven
Carnegie Mellon University
aleven@cs.cmu.edu

Conrad Borchers
Carnegie Mellon University
cborcher@cs.cmu.edu

ABSTRACT

Intelligent Tutoring Systems often grant learners shared control over skill and problem selection. This choice brings motivational and metacognitive benefits. At the same time, past literature suggests that learners exhibit diverse preferences and strategies in selecting tasks, for instance, by avoiding challenge. Although underexplored, differences in learner task-selection strategies may interact with mastery learning systems that optimize task-selection based on estimated knowledge, potentially leading to undesirable student-level differences in learning outcomes. Algorithmic constraints on problem selection may help mitigate this issue. However, this possibility has not been comprehensively explored in prior work, in part because testing such constraints in real-world classrooms is costly. We propose a simulation-based framework to observe how varying learner task-selection strategies combined with system constraints shape mastery learning efficiency. Using interaction data from 261 students across two mathematical domains with different problem structures (equation solving, graph interpretation), we simulate common task-selection strategies such as Weakness Targeting and Interleaving, grounded in prior literature. We then evaluate how these strategies affect overpractice as a common measure of mastery learning efficiency. Results show substantial variability in efficiency across strategies, with risk-averse strategies producing higher levels of overpractice, especially for more complex multi-step problems. Targeted system constraints significantly reduce these inefficiencies for maladaptive strategies while having minimal impact on already efficient strategies. Together, these findings demonstrate how simulation grounded in real student data can support data-driven redesign of shared-control tutoring systems by identifying when and where constraints are most beneficial prior to classroom deployment.

Keywords

Simulation, Intelligent Tutoring Systems, Mastery Learning, Learner Control, Task Selection

Haley Noh, Aarna Chowdhary, Jeroen Ooge, Vincent Aleven, and Conrad Borchers. Simulating Learners' Task-Selection Strategies and System Constraints in Mastery Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 524–531. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039887

1. INTRODUCTION

Mastery-based learning is widely adopted in Intelligent Tutoring Systems (ITS) to personalize instruction and ensure learners demonstrate proficiency in prerequisite skills before advancing [14, 19]. Although ITS typically do not involve learners in the adaptive task-selection process, prior work has shown that involving learners by allowing them to express or act on task-selection preferences can yield motivational and metacognitive benefits, including increased engagement, perceived autonomy, and reflection on learning progress [5, 8, 17, 20]. However, differences in learner task-selection strategies can introduce substantial inefficiencies in mastery-based learning, particularly when learners choose problems that involve skills they have already mastered or avoid challenging ones [19, 23]. Prior research has shown that learners adopt diverse task-selection strategies such as Strength Targeting, Weakness Targeting, Interleaving, or Blocking [6, 10, 17]. While some strategies align well with mastery-based learning objectives, others can result in excessive overpractice and delayed progression, particularly in domains with multi-step problem structures where mastered skills frequently reappear.

Studying how different learner strategies paired with task-selection constraints through classroom experiments is costly, time-consuming, and may expose learners to suboptimal learning experiences. Alternatively, simulation studies with real learner interaction data enable systematic and efficient comparison of learner strategies and instructional design choices before deployment [16, 19, 23]. By modeling how different forms of learner control and system guidance shape learning trajectories, simulation offers a scalable method for identifying inefficiencies and testing corrective mechanisms before they affect real students.

We present a simulation-based framework for analyzing how learner task-selection strategies interact with system constraints to shape mastery-based learning efficiency in ITS. Using interaction data from two mathematical domains (equation solving and graph interpretation), we simulate common learner strategies and evaluate their effects using measures of overpractice. This approach enables controlled comparison of learning control and system-control configurations, providing insights into when targeted scaffolding is necessary

to mitigate inefficiencies while preserving learner choice. In sum, our research question is as follows:

RQ. How do different learner task-selection strategies, combined with system constraints, affect the efficiency of mastery-based learning in ITS?

2. RELATED WORK

2.1 Task-Selection Strategies in ITS

Mastery learning frameworks advance students only after demonstrating mastery of prerequisite skills, using criteria based on performance [19]. This ensures competence but can create inefficiencies, as some skills become overpracticed while others are neglected [23]. In many modern ITS, learners are increasingly involved in selecting which skills or problems to practice, often supported by learning analytics and open learner models (OLMs) that make progress and mastery visible [5, 17]. Allowing learners to make such choices is motivated by self-determination theory, which emphasizes autonomy as a key driver of intrinsic motivation, engagement, and persistence in learning [22]. Prior work in educational data mining and ITS shows that learners adopt diverse task-selection strategies [6]. Studies show that when learners freely select skills or problems, they pursue strategies of uneven effectiveness, indicating that control alone rarely improves outcomes and can even hinder them if left unsupported [20]. Some choose familiar tasks, others target weaknesses [6]; some interleave across skills, while others block practice within a single skill, often shaped by confidence and goals [10]. These strategies can nurture persistence and reflection, but also cause inefficiencies, such as overpractice of mastered skills or avoidance of challenges. Understanding how learners naturally manage practice is therefore essential, both to amplify productive strategies and to counter maladaptive ones. Despite this, little research has systematically examined how such strategies interact with system-level supports like constraints, guidance, or explanations.

2.2 System Constraints and Simulation

To address variability in learner behavior, ITS increasingly combines learner choice with system-level constraints that guide task-selection while preserving agency, a paradigm often described as shared student-system control over task-selection. Prior work shows that constrained problem-selection and guided choice can reduce overpractice and improve efficiency without fully removing learner control [17, 23, 24]. However, evaluating how different learner strategies interact with such constraints through classroom studies is costly and risks exposing students to suboptimal learning conditions.

Simulation studies have therefore become increasingly valuable in education research, especially where in vivo experimentation is infeasible or cost-prohibitive [18]. In addition, deploying poor algorithms can be harmful to participants. Simulation enables systematic exploration of both learner strategies and shared-control mechanisms, such as weightings, recommendations, and explanation styles, in a controlled environment. Prior research demonstrates that simulated learners can reproduce realistic learning trajectories and support the development of novel optimization techniques for tutoring systems [23]. Rather than aiming to predict individual learning outcomes, simulation enables comparative analysis of alternative design configurations under consistent assumptions [16, 19]. In this work, we build on this line of research by using simulation to examine when shared-control constraints are necessary to mitigate inefficiencies arising from specific learner task-selection strategies.

3. METHODS AND MATERIALS

3.1 Datasets

We used two datasets to specify model parameters and enhance validity. Following [23], we defined the most common solution pathways in the data and based the simulations on them [13]. The first dataset is on equation-solving and used an open-source dataset available on PSLC DataShop (datasets #5549 and #5604), collected from IRB-approved middle school studies using the APTA ITS for linear equation solving [1, 15]. The dataset includes students in grades 6–8 attending two public schools in the eastern United States and contains over 10,000 step-level interactions with multi-step equation problems [14]. To model authentic first-attempt problem solving, only initial responses were retained, and steps involving hints were treated as incorrect [14]. The step-level structure and repeated use of skills enable reconstruction of typical solution paths, which form the basis for simulating learning trajectories and task-selection strategies. This dataset contains 58 problems and 10 skills. The average number of skills that appeared in a problem was 5.67 ± 2.47.

We additionally used another DataShop dataset based on graph interpretation (#5360) [15]. This dataset consisted of classroom transaction data from the Mathtutor ITS, collected through IRB-approved studies with 97 ninth-grade students in a U.S. high school. Students alternated between paper practice and Mathtutor across three units on linear graphs, with the problem decomposed into step-level interactions (though we only used data from the ITS condition). The dataset represents another domain of multi-step problem solving where students may engage in overpractice to improve generalizability beyond the equation-solving data. This dataset contains 31 problems and 13 skills. The average number of skills that appeared in a problem was 8.57 ± 8.41.

The datasets differ in problem length and problem-skill composition: graph interpretation involves more skills and steps per problem than equation solving. Because these factors affect overpractice [23], this enables evaluation of skill and problem-selection strategies across varied problem types.

3.2 Learner Simulation Methods

We use a simulation-based approach to model student learning under different task-selection strategies in a mastery-based ITS. We specifically use two widely used student modeling approaches in educational data mining research. We use the Additive Factors Model (AFM; [7]) to simulate step-level performance and Bayesian Knowledge Tracing (BKT; [9]) to model knowledge evolution over time. The simulation framework is built on real student interaction data using open-source learner simulation libraries described in Xia et al. [23].

To estimate skill difficulty and learning parameters for our two datasets, we fit both AFM and BKT to the PSLC

DataShop and MathTutor ITS data collections and use the learned parameters to drive the student simulations. Each problem is associated with a typical solution path derived from the most frequent skill-application sequences in the data [21]. During simulation, student responses are sampled at each step according to the AFM-predicted probability of correctness, and both AFM and BKT states are updated after each interaction. The resulting BKT mastery estimates are then used to guide subsequent problem-selection. Our simulation and analysis code is open-source.[1].

AFM models the probability of a correct response as a logistic function of a global intercept, a student ability term, and additive skill-specific difficulty and practice effects. Specifically, calculations are made using information such as the set of skills associated with a problem step, the difficulty of these skills, and the attempts the student has previously made for these skills. BKT models student learning as a hidden Markov process in which each skill has a binary latent state: either learned or unlearned. The model updates mastery using observed responses and four parameters: Prior knowledge estimates (p_{init}), learning rate (p_{learn}), guess rate (p_{guess}), and slip rate (p_{slip}).

Upon receiving a student response, BKT updates the estimated mastery level using Bayes' rule. A correct answer increases the mastery probability, which adjusts the potential for guessing, while an incorrect answer decreases the probability, accounting for possible slips. A skill is considered mastered once the posterior probability surpasses a predefined mastery threshold, in our case 95%, consistent with prior ITS research [2]. The initial values of the BKT parameters are based on the following default values in TutorShop: $p_{\text{init}} = 0.25$, $p_{\text{learn}} = 0.22$, $p_{\text{guess}} = 0.2$, and $p_{\text{slip}} = 0.1$.

3.3 Simulation Overview

Each simulation allows 1,000 learners to progress through a set of skills until mastery is achieved. The simulation is designed to support controlled comparison of learner strategies and system constraints under standardized conditions. The simulation models mastery learning as a repeated problem-level decision cycle. The iterative cycle of task-selection, problem-solving, and mastery updating defines a learning path for each simulated learner. Specifically, a learning path is represented as the sequence of skills selected and problems attempted over time in combination with the changing mastery estimates. As task-selection decisions are made based on the learner's current knowledge state and updated after each simulated problem attempt, the resulting trajectories reflect the learner's emergent path through an adaptive task sequence. Between problems, a simulated learner selects a skill to practice based on a predefined task-selection strategy, optionally shaped by system-imposed constraints on the available skill choices. Once a skill is selected, the system chooses a problem that uses that skill, potentially applying additional problem-selection constraints. This modeling choice reflects a common shared-control setup in mastery-based ITS, where learners select skills, and the system selects corresponding problems, but does not capture all possible forms of task-selection [2, 12].

[1] `https://github.com/conradborchers/sim-skill-selection`

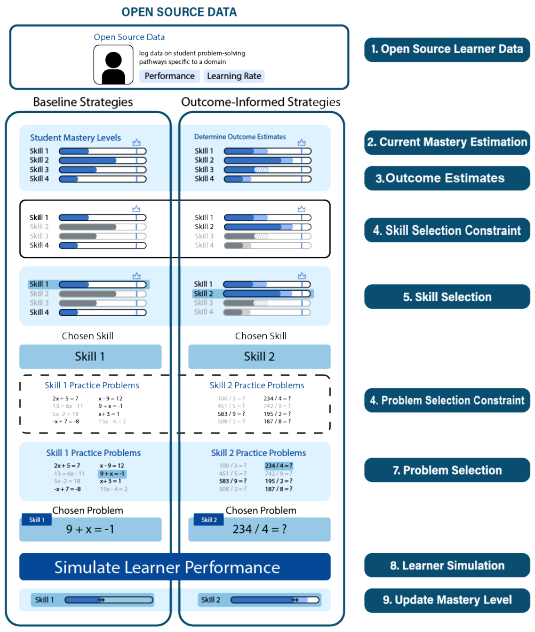


Figure 1: Simulation Methodology Summary.

The simulated learner then attempts the selected problem, which may consist of multiple steps, each mapping to at least one skill. The steps are simulated sequentially, with performance generated probabilistically based on the learner's current knowledge state and step difficulty. Mastery estimates are updated after each step, but the learner does not make additional choices until the entire problem has been completed. After the problem attempt concludes and mastery estimates are updated, the learner again selects a skill for the next problem based on their updated skill mastery estimates, and the cycle repeats.

This structure mirrors the way that learners can select tasks when interacting with shared-control mastery-based ITS, where strategic decisions are made between problems rather than within them [2, 17]. By grounding step-level performance and mastery updates in parameters estimated from real student data, the simulation produces realistic learning trajectories while enabling controlled comparison of how different learner task-selection strategies and system constraints affect mastery learning efficiency. Figure 1 summarizes this iterative problem-selection-practice loop.

Below, we elaborate on the individual components of the simulation framework:

1. **Open Source Learner Data** We begin by using open-source log data from both middle school and high school math ITS, which contains detailed records of student interactions and performance across a range of equation-solving and graph-interpretation skills, respectively (see Section 1).

2. **Current Mastery Estimation** The estimated mastery levels for each skill present in the dataset are calculated after every practice opportunity.

3. **Outcome Estimates** For strategies that specifically rely on outcome-estimations, we compute the projected mastery outcomes under three scenarios: best case, usual case, and worst case.

4. **Task-Selection Guidance** Front-end task-selection guidance simulates UI cues by restricting the pool of selectable skills to those closer or further from the mastery threshold, shaping learner decisions without altering algorithms.

5. **Task-Selection** A skill is selected using either a general strategy (e.g., Strength Targeting, Interleaving) or an Outcome-informed strategy (e.g., Maximize Usual Case, Minimize Worst Case Loss), simulating different learner decision-making behaviors.

6. **Problem-Selection Constraint** Once a skill is selected, a system control constraint introduces bias in the subsequent problem-selection process based on problem difficulty, modeling how backend design influences the learning path.

7. **Problem-Selection** A specific problem is randomly drawn from the available pool associated with the selected skill, either randomly or following any imposed system control constraints.

8. **Learner Simulation** The simulated student attempts the selected problem step, and performance is determined probabilistically using AFM, based on current skill mastery and step difficulty.

9. **Update Mastery Level** Following the simulated attempt, mastery estimates are updated using BKT, and the simulation loop repeats until all skills reach mastery or a stopping condition is met.

3.4 Task-Selection Strategies

We model learner task-selection using predefined decision rules that determine which skill a simulated learner chooses to practice at each step. The eight selected strategies are designed to reflect commonly observed patterns of learner behavior in mastery-based tutoring systems and are applied consistently across all simulation conditions.

3.4.1 Baseline Task-Selection Strategies

We implement four baseline strategies drawn from prior work on learner practice behavior. **Strength Targeting** prioritizes skills that are close to mastery, while **Weakness Targeting** prioritizes skills with the lowest estimated mastery. **Interleaving** alternates practice across multiple skills, whereas **Blocking** concentrates practice on a single skill until mastery is achieved. These strategies capture a range of common learner approaches to managing practice and have been widely studied in educational psychology and ITS research. Finally, a **Random** strategy selects skills at random and serves as a control condition.

3.4.2 Outcome-Informed Strategies

In addition to baseline strategies, we model decision rules that select skills based on projected mastery outcomes under different assumptions about performance. **Maximize Usual Case Improvement** selects the skill with the greatest expected mastery gain, while **Maximize Usual Case Outcome** selects the skill with the highest projected post-practice mastery. **Minimize Worst Case Loss** represents a risk-averse strategy that selects the skill with the smallest potential decrease in mastery under unfavorable performance. This strategy is risk-averse because it prioritizes minimizing potential negative mastery outcomes over maximizing expected gains.

Together, these strategies enable systematic comparison of how different decision-making behaviors interact with system-level constraints to influence learning efficiency in mastery-based tutoring systems.

3.5 System Constraints

To examine how system design interacts with task-selection strategies, we introduce two system constraints that shape learning while preserving learner control over skill prioritization.

3.5.1 Task-Selection Constraints

Task-selection constraints limit the set of skills available for selection at each decision point based on current mastery estimates. We implement two variants. The *closer-to-mastery* constraint restricts selection to skills with proficiency levels nearest to, but below, the mastery threshold. The *further-from-mastery* constraint restricts selection to skills with lower proficiency levels, representing a weaker understanding. These constraints reduce learners' choice set without prescribing a specific selection strategy, allowing learners to retain control within a bounded decision space. We also simulate learners' interactions under unconstrained task-selection to provide a baseline for comparison.

3.5.2 Problem-Selection Constraints

Problem-selection constraints influence which problem is delivered after learners select a skill. Rather than selecting problems uniformly at random, the system applies biased sampling within the chosen skill. Problem difficulties are determined using a weighted average of the learners' past performance with skills present in problem steps. Under the *prefer-easier constraint*, problems associated with lower difficulty skills are assigned higher weight, while under the *prefer-harder constraint*, problems associated with higher difficulty skills are given higher weight. Once all the problem difficulty scores are assigned, they are normalized to form a multinomial distribution from which weighted random sampling is used to select a problem. These constraints operate independently of task-selection and affect only the execution of practice within a chosen skill.

3.6 Metric: Overpractice

We evaluate the impact of system constraints in ITS using **overpractice** as the primary outcome. Overpractice refers to continued practice of skills that have already reached mastery, a phenomenon that commonly arises in mastery-based tutoring systems [19, 23]. This particularly occurs when problems involve multiple skills or steps, as problems primarily based on already proficient skills may be chosen for including an unmastered component [14]. Overpractice has been studied as an inefficiency in mastery-based ITS, as it increases time on task without proportional learning benefits and can slow progression through the learning process

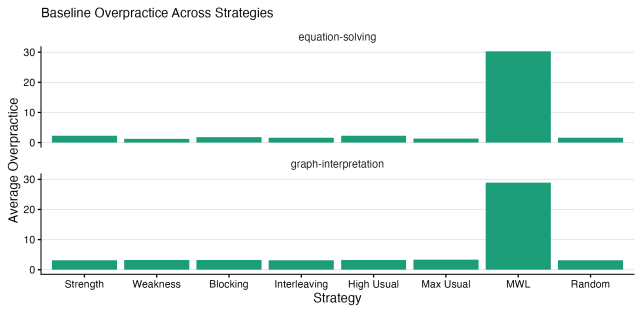


Figure 2: Baseline Average Overpractice Across Strategies for Both Datasets. MWL denotes the Minimize Worst Case Loss strategy.

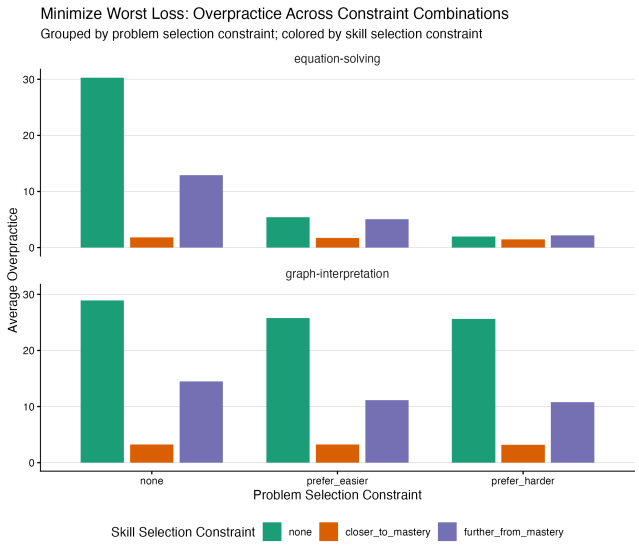


Figure 3: Average Overpractice for the Minimize Worst Case Loss Strategy with Task-Selection and Problem-Selection Constraints.

[16, 19, 23]. For each strategy-constraint condition, simulation runs produced data on the overpractice aggregated across all skills present in the dataset, yielding descriptive statistics, including mean and standard deviation. In addition to comparing means and effect sizes using Cohen's d, the analysis focused on determining which strategies and constraints resulted in the most efficient mastery learning, indicated by reduced overpractice. The findings helped identify optimal combinations of strategies and constraints that minimized unnecessary practice while maintaining mastery.

4. RESULTS

The results presented in Figures 2 and 3 summarize the average overpractice observed under different task-selection strategies and system constraint conditions. These values are computed by aggregating overpractice across all simulated learners and skills. This allows for direct comparison of learning efficiency across strategies and datasets.

4.1 Performance Across Strategies

Figure 2 compares baseline overpractice levels across all task-selection strategies without system constraints. Across both datasets, minimizing worst case loss was a clear outlier in terms of its average amount of overpractice. Without system constraints, using the Minimize Worst Case Loss strategy produced overpractice levels of 30.28 times in equation-solving and 28.93 times on average in graph-interpretation (Figure 2). In contrast, the average of the other seven strategies' overpractice amounts were 1.733 ± 0.383 and 3.223 ± 0.050 for the equation-solving and the graph-interpretation data sets, respectively. Additionally, the second-worst performing strategy for the equation-solving dataset was the Strength Targeting strategy. This indicates that informed risk-averse strategies are substantially less efficient than other strategies in self-guided systems.

On the other hand, in the equation-solving dataset, Weakness Targeting and Maximize Usual Case Improvement performed slightly better than other strategies, yielding the lowest overpractice of 1.26 and 1.29 average practice opportunities, respectively (Figure 2). This suggests that prioritizing less-mastered skills is a comparatively efficient approach.

4.2 Incorporating System Constraints

Figure 3 illustrates how task-selection and problem-selection constraints affect overpractice for the Minimize Worst Case Loss strategy. Introducing task-selection constraints significantly moderated outcomes for otherwise inefficient strategies. Most notably, when the risk-averse strategy (Minimize Worst Case Loss) was guided toward skills closer to mastery, the excess overpractice was sharply reduced from 30.28 to 1.83 times in equation-solving and 28.93 to 3.22 times in graph-interpretation relative to the unconstrained baseline, with large respective Cohen's d effect sizes of 1.87 and 1.86 (Figure 3). Task-selection constraints that prioritized skills further from mastery also improved efficiency, reducing overpractice to 12.88 and 14.46, respectively. By contrast, most other strategies were relatively unaffected by these constraints. Blocking, Interleaving, Strength Targeting, and Weakness Targeting all maintained stable levels of overpractice across conditions. Overall, the results indicate that specific forms of system control exert their strongest effects on risk-averse learners, while already efficient strategies show minimal change.

Problem-selection constraints also had a considerable effect on reducing inefficiencies in simulated learning. When paired with the Prefer Harder constraint, Minimize Worst Case Loss showed a greatly reduced overpractice value from 30.28 to 1.98 times in the equation-solving dataset, a large Cohen's d effect size value of 1.86, leveling the other strategies without the constraint (Figure 3). This suggests that system control constraints can help counterbalance risk-averse tendencies by increasing exposure to more challenging problems. The equation-solving dataset, where multi-step structures magnified redundancy, produced a stronger corrective effect when paired with the Prefer Harder constraint compared to graph-interpretation.

5. DISCUSSION

As learner control becomes increasingly common in ITS and other adaptive learning environments [4, 8, 20], understanding how differences in task-selection behavior affect learn-

ing efficiency is critical. Rather than assuming that learner choice uniformly benefits all students, our results demonstrate substantial variability in outcomes across task-selection strategies. Using a simulation derived from real student interaction data, this study shows how certain learner decisions can systematically hinder mastery learning, while others align well with learning objectives.

5.1 Strategy Variability and Validation

Across both datasets, weakness-targeting strategies consistently produced efficient learning paths with minimal overpractice. In contrast, the risk-averse strategy that minimizes worst case loss, which tends to prioritize skills with already high estimated mastery, emerged as a clear outlier and resulted in substantially higher levels of overpractice. This pattern may be explained by the fact that stronger skills are closer to mastery and involve lower expected errors and mastery loss, leading the risk-averse strategy to favor already well-practiced skills and thereby induce redundant practice. Notably, this form of risk aversion has been documented in real learners and is therefore likely to generate meaningful differences in learning efficacy across students, depending on whether they exhibit this behavior [2]. Next we discuss how constraints on task-selection can mitigate these differences.

5.2 System Constraints Mitigate Inefficiencies

Our findings show that system constraints on task-selection are more effective when applied selectively. Task-selection and problem-selection constraints substantially reduced inefficiencies for the risk-averse strategy, bringing its performance to levels similar to those of other, more efficient strategies overall. At the same time, these constraints had little to no effect on strategies that were already efficient, such as Weakness Targeting or Interleaving. This asymmetry suggests that applying the same constraints to all learner types is unnecessary (and that student preferences for specific skills can be honored for some students more than others without sacrificing mastery efficiency). Instead, simulation enables identification of which learner behaviors warrant targeted system constraints, allowing system designers to intervene only where inefficiencies are likely to arise. Such selective constraint application aligns with common practice in ITS [12, 19], where system constraint is routinely adapted based on learner state rather than applied uniformly. An open question for future work is whether students would be able to recognize different levels of system constraints on task-selection (some being invisible to them but potentially felt during problem completion). Overly prominent system constraints may undermine student autonomy perceptions and thereby lower motivation, ultimately hindering learning [3]. However, because our constraints are designed to preserve shared control by bounding rather than removing learner choice, they are intended to mitigate inefficiencies while maintaining the motivational benefits associated with autonomy.

5.3 Equity and Shared Control

Prior work based on large-scale evidence has shown that students exhibit large differences in their level of engagement and persistence in various adaptive learning systems [11, 16]. Here, we turn to another, underinvestigated aspect of student outcomes in mastery learning: the impact of decision-making strategies when choosing tasks. While our findings suggest that constraining control may lead to more equitable benefits of mastery learning, it is also worth highlighting that there are benefits to student control, particularly with regards to the aforementioned persistence differences highlighted in prior work [11, 16]. In particular, past research notes motivational benefits of control related to engagement, motivation, and metacognitive development. By contrast, targeted system constraints offer a middle ground where preserving learner choice while mitigating extreme inefficiencies that disproportionately affect certain learners [8, 17, 20].

5.4 Simulation to Close the Loop

More broadly, this work contributes a novel framework and evidence of how simulations can support close-the-loop experimentation in educational data mining by serving as a bridge between data analysis and learning system redesign [12]. Grounded in real student interaction data and using standard EDM models such as AFM and BKT, our simulation enables systematic evaluation of how alternative design choices, specifically different levels of learner control and system constraints, shape learning efficiency prior to classroom deployment. This aligns with prior work on design-loop adaptivity, which emphasizes using data-driven analyses to inform and prioritize redesign decisions that can later be validated through in vivo experimentation [12]. We treat simulation as a complementary pre-deployment tool that reduces risk and helps identify which learner behaviors and system interventions warrant targeted classroom evaluation.

Additionally, the simulation was reproducible across datasets of student interaction data spanning different problem domains. Common anticipated trends—such as inefficiencies arising from risk-averse strategies—were consistently observed across both datasets. Notably, overpractice outcomes for equation-solving tasks exhibited greater variability than those observed in other domains. This finding aligns with the notion that equation-solving problems often admit multiple solution paths involving distinct underlying skills, in contrast to graph-interpretation tasks. Moreover, equation-solving problems frequently exhibit highly recursive structures with greater skill repetition, which may further contribute to the observed variability [23].

5.5 Limitation and Future Work

This study models a limited set of learner task-selection strategies and system constraints grounded in prior empirical work, and does not capture the full range of motivational, metacognitive, or affective factors that may influence learner behavior. Our simulation focuses on a task-selection-first setup and does not model alternative task-selection paradigms (e.g., direct problem-selection, multi-task-selection, or fully system-driven sequencing), though the framework could be extended to support these regimes. Learner strategies are modeled as static, whereas real learners adapt their strategies over time in response to feedback or outcomes. We leave the investigation of dynamic combinations of strategies to future work. Future work could also extend our framework by incorporating models of motivation and self-regulated learning to examine how different forms of shared control affect engagement and metacognitive processes. Finally, the simulation-based approach can inform iterative, close-the-loop system redesign by identifying promising constraint strategies prior to classroom de-

ployment and refining models using additional learner data across domains [12].

6. CONCLUSION

This study contributes a novel simulation-based educational data mining framework for examining how learner task-selection strategies and system constraints jointly shape mastery learning efficiency in ITS. Grounded in real student data and implemented using standard EDM models, our simulations reveal substantial variability in the efficiency of different learner task-selection strategies. In particular, while some strategies align with mastery learning objectives, others can lead to extreme inefficiencies in practice. Our results show that targeted system constraints can mitigate these inefficiencies without removing learner control, while having minimal impact on strategies that are already effective. This highlights the value of selectively deploying scaffolding mechanisms rather than adopting uniform control policies, and underscores the importance of accounting for learner strategy diversity in the design of adaptive learning systems. Broadly speaking, this study demonstrates how simulation can support close-the-loop data-driven redesign in educational data mining. By enabling systematic evaluation of alternative design choices prior to classroom deployment, simulation helps reduce risk, accelerate iteration, and prioritize which interventions merit further classroom testing. As learner-controlled systems become more widespread, these methods will be increasingly important for ensuring that the benefits of educational technology are distributed equitably without amplifying disparities in learning outcomes. Practically, this work suggests that ITS designers seeking to implement shared control should move toward selectively or adaptively applying constraints that target inefficient learner task-selection strategies. Our findings highlight the value of simulation as a low-cost and scalable tool for evaluating ITS design choices, enabling researchers to test and refine potentially inequitable or suboptimal interventions before deploying them in real learning environments.

Acknowledgments

This research was funded by the Institute of Education Sciences (IES) of the U.S. Department of Education (Award #R305A220386). H. Noh was supported by the National Science Foundation (NSF) Research Experiences for Undergraduates (REU) program. A. Chowdhary was supported by the Summer Undergraduate Research Apprenticeship (SURA) program at Carnegie Mellon University.

References

[1] Conrad Borchers, Kexin Yang, Jionghao Lin, Nikol Rummel, Kenneth R. Koedinger, and Vincent Aleven. Combining dialog acts and skill modeling: What chat interactions enhance learning rates during ai-supported peer tutoring? In Benjamin Paaßen and Carrie Demmans Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 117–130, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society. doi: 10.5281/zenodo.12729784.

[2] Conrad Borchers, Jeroen Ooge, Cindy Peng, and Vincent Aleven. How learner control and explainable learning analytics about skill mastery shape student desires to finish and avoid loss in tutored practice. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 810–816, 2025. doi: 10.1145/3706468.3706484.

[3] Conrad Borchers, Cindy Peng, Qianru Lyu, Paulo F Carvalho, Kenneth R Koedinger, and Vincent Aleven. Student perceptions of adaptive goal setting recommendations: a design prototyping study. In *International Conference on Artificial Intelligence in Education*, pages 244–251. Springer, 2025. doi: 10.1007/978-3-031-98462-4_31.

[4] Peter Brusilovsky. Ai in education, learner control, and human-ai collaboration. *International Journal of Artificial Intelligence in Education*, 34(1):122–135, 2024. doi: 10.1007/s40593-023-00356-z.

[5] Susan Bull. There are open learner models about! *IEEE Transactions on Learning Technologies*, 13(2): 425–448, 2020. doi: 10.1109/TLT.2020.2978473.

[6] Deborah L Butler, Beverly Beckingham, and Helen J Novak Lauscher. Promoting strategic learning by eighth-grade students struggling in mathematics: A report of three case studies. *Learning disabilities research & practice*, 20(3):156–174, 2005. doi: 10.1111/j.1540-5826.2005.00130.x.

[7] Hao Cen, Kenneth Koedinger, and Brian Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*, pages 164–175. Springer, 2006. doi: 10.1007/11774303_17.

[8] Gemma Corbalan, Liesbeth Kester, and Jeroen JG Van Merriënboer. Selecting learning tasks: Effects of adaptation and shared control on learning efficiency and task involvement. *Contemporary educational psychology*, 33(4):733–756, 2008. doi: 10.1016/j.cedpsych.2008.02.003.

[9] Albert T Corbett and John R Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4): 253–278, 1994. doi: 10.1007/BF01099821.

[10] Nathaniel L Foster, Michael L Mueller, Christopher Was, Katherine A Rawson, and John Dunlosky. Why does interleaving improve math learning? the contributions of discriminative contrast and distributed practice. *Memory & Cognition*, 47(6):1088–1101, 2019. doi: 10.3758/s13421-019-00918-4.

[11] Laurence Holt. The 5 percent problem. *Education Next*, 24(4):26–31, 2024.

[12] Yun Huang, Vincent Aleven, Elizabeth McLaughlin, and Kenneth Koedinger. A general multi-method approach to design-loop adaptivity in intelligent tutoring systems. pages 124–129, 2020. doi: 10.1007/978-3-030-52240-7_23.

[13] Tanja Käser and Giora Alexandron. Simulated learners in educational technology: A systematic literature review and a turing-like test. *International Journal of Artificial Intelligence in Education*, 34(2):545–585, 2024. doi: 10.1007/s40593-023-00352-1.

[14] K. Koedinger, Sidney K. D'Mello, Elizabeth Mclaughlin, Zachary A. Pardos, and Carolyn Penstein Rosé. Data mining and education. *Wiley interdisciplinary reviews. Cognitive science*, 6 4:333–353, 2015. doi: 10.1002/wcs.1350.

[15] Kenneth R Koedinger, Ryan SJd Baker, Kyle Cunningham, Alida Skogsholm, Brett Leber, and John Stamper. A data repository for the edm community: The pslc datashop. *Handbook of educational data mining*, 43:43–56, 2010.

[16] Kenneth R Koedinger, Paulo F Carvalho, Ran Liu, and Elizabeth A McLaughlin. An astonishing regularity in student learning rate. *Proceedings of the National Academy of Sciences*, 120(13):e2221311120, 2023. doi: 10.1073/pnas.2221311120.

[17] Yanjin Long and Vincent Aleven. Enhancing learning outcomes through self-regulated learning support with an open learner model. *User Modeling and User-Adapted Interaction*, 27:55–88, 2017. doi: 10.1007/s11257-016-9186-6.

[18] Christopher J MacLellan. Model human learners: Computational models to guide instructional design. *arXiv preprint arXiv:2502.02456*, 2025. doi: 10.48550/arXiv.2502.02456.

[19] Radek Pelánek. Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User modeling and user-adapted interaction*, 27(3):313–350, 2017. doi: 10.1007/s11257-017-9193-2.

[20] Katharina Scheiter and Peter Gerjets. Learner control in hypermedia environments. *Educational Psychology Review*, 19(3):285–307, 2007. doi: 10.1007/s10648-007-9046-3.

[21] John Stamper, Michael Eagle, Tiffany Barnes, and Marvin Croy. Experimental evaluation of automatic hint generation for a logic tutor. *International Journal of Artificial Intelligence in Education*, 22(1-2):3–17, 2013. doi: 10.3233/JAI-130029.

[22] Detlef Urhahne and Lisette Wijnia. Theories of motivation in education: An integrative framework. *Educational Psychology Review*, 35(2):45, 2023. doi: 10.1007/s10648-023-09767-9.

[23] Meng Xia, Robin Schmucker, Conrad Borchers, and Vincent Aleven. Optimizing mastery learning by fast-forwarding over-practice steps. In *European Conference on Technology Enhanced Learning*, pages 549–563. Springer, 2025. doi: 10.1007/978-3-032-03870-8_37.

[24] Hongxin Yan, Fuhua Lin, and Kinshuk. An ai-learner shared control model design for adaptive practicing. In *International Conference on Intelligent Tutoring Systems*, pages 272–280. Springer, 2024. doi: 10.1007/978-3-031-63028-6_21.

Simulating Students' Java Programming Errors with Large Language Models

Ali Keramati*
University of California, Irvine
a.kera@uci.edu

Jie Cao*
The University of North Carolina at Chapel Hill
jiecao@unc.edu

Iman Mohammadi
University of California, Irvine
imohamm1@uci.edu

Mark Warschauer
University of California, Irvine
markw@uci.edu

Yang Shi
Utah State University
yang.shi@usu.edu

ABSTRACT

Understanding student errors in the programming is a cornerstone of programming education, yet obtaining a representative set of student errors for any newly designed task remains slow and costly, since authentic submissions only accumulate after extensive classroom deployment. This paper explores whether large language models (LLMs) can serve as scalable proxies for students by simulating realistic logical errors in code submissions. Using the CodeWorkout dataset of 74,000+ unique student Java submissions across 37 problems, we evaluate five LLMs under three mainstream prompting strategies: Input-Output (IO), Chain-of-Thought (CoT), and iterative Self-Refine. We assess performance along two key dimensions: diversity (the range of distinct error patterns) and alignment (alignment with authentic student mistakes), and examine how these vary by struggling level of programming tasks. Our quantitative findings reveal that while all models generate diverse errors, their alignment to human submissions diverges: *Claude Sonnet 4* achieves the most balanced performance. In addition, we conducted a blinded expert annotation study ($N = 401$) comparing synthetic and authentic errors. This qualitative analysis confirms that the generated errors are functionally indistinguishable from authentic student errors. Moreover, higher-struggling-level problems elicit more diverse but less student-like errors. These results highlight trade-offs in using LLMs to simulate human learners and suggest design considerations for integrating synthetic errors into teachable agents, intelligent tutoring systems, and large-scale learning analytics.

Keywords

Student Error Simulation, Logical Error Generation, Programming Misconceptions, Large Language Models (LLMs)

[1]*These authors contributed equally to this work.

Ali Keramati, Jie Cao, Iman Mohammadi, Mark Warschauer, and Yang Shi. Simulating Students' Java Programming Errors with Large Language Models. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 532–540. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040050

1. INTRODUCTION

Programming education plays a foundational role in modern STEM curricula. Throughout this learning process, students inevitably make errors. Such errors not only expose students' weaknesses but also reveal their underlying cognitive processes—especially logical errors, which trace how learners reason about a problem. Unlike syntactic mistakes, logical errors are more subtle: they reflect learners' incomplete understandings, misconceptions, and flawed problem-solving strategies. For decades, capturing such errors has been a cornerstone of learner modeling, personalized feedback, and adaptive instructional support in computer science education [4, 25]. When teachers can foresee likely student errors, they can proactively design targeted scaffolds and address common pitfalls before they hinder student progress. Furthermore, integrating realistic error patterns into teachable agents can prompt learners to recognize and learn from these errors [19]. Yet, anticipating the full spectrum of plausible student solutions and their corresponding failure modes for different tasks remains difficult, especially for novice educators or system designer.

Recent advances in Large Language Models (LLMs) suggest a promising path forward. LLMs can serve as scalable proxies for learners[8, 32, 5, 9], generating realistic error datasets that can power personalized feedback systems, intelligent tutoring environments, and deeper investigations into novice cognitive processes [18]. More broadly, LLMs are increasingly positioned as versatile tools for simulating students in learning analytics. This includes modeling learning behaviors [23], enriching classroom analytics with simulated reasoning [32], and reflecting the collaborative and motivational dynamics of online learning environments [13].

Despite these advancements, the specific challenge of simulating *logical* errors in programming education remains largely underexplored [16], and current approaches sit at two unsatisfying extremes. On one end, general prompting of LLMs typically yields correct or near-perfect solutions, entirely missing the characteristic flaws inherent to novice code [29, 26]. On the other, random corruption strategies (e.g., inserting arbitrary bugs) produce artificial mistakes that lack the pedagogical value of errors emerging from genuine, flawed reasoning. The challenge is thus one of indistinguishability a generated error must be wrong in ways a real student

would be wrong—reflecting partial understandings, misconceptions, and incomplete problem-solving strategies rather than surface-level noise [21, 27]. Yet existing evaluation infrastructure offers little traction here. Programming benchmarks such as HumanEval, MBPP, and CodeContests are designed around absolute correctness, and recent work exploring synthetic buggy code has treated such code primarily as a vehicle for data augmentation [14] or focused on overall error distribution [16] rather than logical errors

To evaluate whether LLMs can simulate students' logical errors, two properties must hold simultaneously: the generated errors need (i) span a sufficiently diverse space of failure modes, and (ii) exhibit high alignment by aligning with the actual errors students produce. Moreover, both properties may depend significantly on the intrinsic difficulty of the programming task. Problems that elicit numerous repeated attempts likely surface qualitatively different, and more varied, patterns of novice reasoning than those solved on the first try. We therefore operationalize the *struggling level* of a problem using the total volume of student submissions as a behavioral proxy for empirical learning difficulty. Building on this perspective, we investigate how different combinations of models and prompting strategies influence the ability of LLMs to simulate student-like logical errors in programming. Specifically, we aim to answer the following research questions:

- **RQ1 (Diversity):** How do different LLMs and prompting techniques vary in generating a diverse and expansive set of erroneous code?
- **RQ2 (Alignment):** To what extent does LLM-generated erroneous code align with real student logical errors?
- **RQ3 (Struggling Level as a Moderator):** How does the struggling level of problems affect LLMs' performance in simulating student-like logical errors?

2. RELATED WORK

2.1 Student Errors in Programming

Errors are the most common byproduct of the students' learning process and serve as valuable learning resources. Teachers can analyze student errors to provide feedback and adjust their instructional plans, while students can engage in self-regulated learning through analyzing their own mistakes [34] or vicarious learning through erroneous examples [20]. In programming education, student errors have different types. Albrecht et al [2], examined 12,371 submissions from 280 students and identified six error categories: syntactic, conceptual, strategic, sloppiness, misinterpretation, and domain knowledge errors. According to other researchers, logic errors represent the primary challenge that programming education must address [25, 15]. Logic errors also encompass various subcategories. For instance, [7] analyzed 15,000 code fragments containing logic errors and classified them into algorithmic errors, misinterpretations of the problem, and fundamental misconceptions, finding that misconceptions are the most frequent source of logic errors. Other researchers have identified additional logic error types, such as loop condition errors and logical operator misuse [12]. By summarizing and analyzing these typical errors, valuable guidance can be provided for teachers' instruction, or as the error proxy for students when they interact with teachable agent.

2.2 LLM-Based Student Simulation

Researchers have begun designing specific prompts to enable LLMs to assume particular roles or generate content that aligns with role-specific characteristics, leading to the emergence of student simulation research [32, 24, 31, 5]. These studies encompass three main approaches: (1) direct agent simulation where LLMs embody virtual personas for flexible interactions, such as [33] trained models to simulate students' cognitive abilities using transaction data from tutoring systems, and [24] introduced TutorUp, a GPT-4-based platform enabling novice educators to practice engagement strategies through scenario-based interactions with simulated students; (2) teacher-student dialogue simulation, exemplified by [11] utilized GPT-4 and Claude-3.5 to enhance teaching plan quality by simulating teacher-student interactions and generating teaching reflections; (3) mimicking students' responses and behaviors, such as [31] build the students agent to respond to responsive teaching in teacher training. In the programming education domain, [16] conducted a comparative study examining bug distributions generated by GPT-4 versus those produced by students, finding that while unguided LLMs do not generate plausible error distributions, they can be guided to produce realistic error patterns when provided with descriptions of common errors and typical frequencies. However, despite this preliminary exploration of LLM-based student programming error simulation, several gaps remain: the study lacks comparison across different LLMs, does not focus specifically on logical errors, and fails to examine the diversity of generated errors.

3. METHODOLOGY

To investigate to what extent LLMs can simulate logical errors in programming, we designed and executed a multi-stage methodology, leveraging the CodeWorkout dataset and a suite of prompting strategies across several LLMs, which is summarized in Figure 1. The process begins with (1) Data Preparation phase, where we scrape, merge, and filter the CodeWorkout dataset to create a clean set of Java programming problems. Next, in the (2) LLM Error Simulation phase, each problem is presented to a suite of LLMs using three distinct, parallel prompting strategies: an IO Prompting, a CoT Prompting, and an iterative Self-Refine loop.

3.1 Dataset

We constructed our dataset from *CodeWorkout*, which provides detailed logs of students' programming submissions for introductory Java exercises. These data were collected from 410 undergraduate students in an introductory Java course at Virginia Tech during the Spring and Fall 2019 semesters [6]. The raw data includes problem descriptions, code snapshots, and error-labeled submissions et al. To prepare it for analysis, we developed a preprocessing pipeline (Figure 1) that transformed the raw logs into a clean, Java-focused corpus. Because large-scale CodeWorkout logs can contain abnormal or non-representative submissions (e.g., empty, duplicated, or otherwise anomalous traces), we follow prior work on detecting abnormal CodeWorkout submissions when considering data quality and filtering decisions

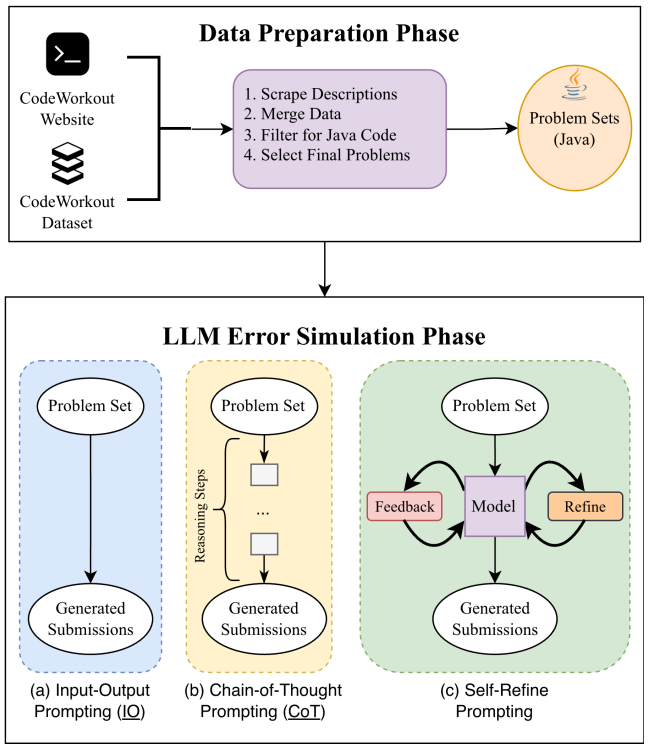


Figure 1: An overview of the experimental pipeline.

[10]. In addition, since the logs contained multiple programming languages, we applied a heuristic classifier based on regular expressions to identify Java submissions. Finally, to focus specifically on logical errors, we filtered out submissions that contained syntax errors (which fail to compile) and those that contained completely correct code. Consequently, the retained submissions consist exclusively of compilable yet incorrect code, ensuring the presence of logical errors. From the filtered corpus, we extracted the unique function names students were required to implement and aligned them with the scraped descriptions, ensuring all tasks used in our study were authentic CodeWorkout problems. The final dataset comprised 37 distinct Java problems. To capture variation in student struggle, we categorized them into three levels: Low (avg. 1,187 submissions), Medium (avg. 2,049) and High (avg. 4,125) based on the total number of student submissions per problem. Problems were sorted by submission count and evenly divided across levels (See detail in the Digital Appendix[1] D, Table 5).

3.2 Prompting Strategies

We compared three prompting strategies: Input-Output (IO), Chain-of-Thought (CoT), and iterative Self-Refine. *Input–Output prompting* is the standard direct-instruction baseline in which the model is given the task description and produces the final output in a single step, without articulating intermediate reasoning; it serves as the reference against which more elaborate strategies are compared [28]. *Chain-of-Thought prompting* [28] instructs the model to articulate intermediate reasoning before emitting the final answer; in our setting, the model first writes a one- to three-sentence rationale describing the intended logical slip and then produces the buggy code. *Self-Refine* [17] is an iterative framework in which the model generates an initial output, a critic instance produces structured feedback against a rubric, and a refiner instance rewrites items flagged for revision; this loop repeats for up to four iterations or terminates early when the critic judges the entire set compliant. Each strategy was implemented as a distinct prompting pathway in our pipeline (Figure 1), but all share the same experimental goal: generate multiple compilable Java submissions per problem, each containing exactly one non-trivial logical error and reflecting a plausible student attempt. The exact strategy instantiations, related prompts, and examples are documented in Digital Appendix A.

4. EXPERIMENTS

4.1 Experimental Setup

Our experiment was designed to compare the performance of different LLMs and prompting strategies in generating plausible, student-like programming errors for the 37 problems in our finalized dataset. As noted in the "LLM Error Simulation" phase of Figure 1, we employed a diverse set of LLMs, including `google/gemini-2.5-pro`, `openai/gpt-5`, `openai/gpt-4o`, `anthropic/claude-sonnet-4`, and `x-ai/grok-code-fast-1`. These models were selected based on their performance on programming and code generation tasks at the time of this study, as indicated by leading industry benchmarks including the Vellum AI LLM Leaderboard [2], LiveBench [3], and OpenRouter [4]. All code generation tasks were performed with a sampling temperature of 0.0 to minimize output variance. An exception to this temperature setting was the feedback step in the Self-Refine strategy, which used a temperature of 0.3 to encourage more descriptive critiques.

4.2 Data Analysis

4.2.1 Automated Metrics

For RQ1, we constructed abstract syntax trees (ASTs) for the generated erroneous code and computed pairwise edit distances using the Zhang–Shasha (ZSS) algorithm for each experimental condition [30]. Specifically, for each problem, we calculated the mean edit distance across the erroneous code samples, and then averaged these values across all problems. Larger distances indicate greater diversity in the generated erroneous code, meaning that within a given programming task, the more dissimilar the erroneous code samples are, the less consistent their error patterns become, thereby reflecting higher diversity.

For RQ2, to evaluate the alignment between LLM-generated and human erroneous code (fidelity), we first deduplicated student submissions, yielding 74,080 unique codes across 37 programming problems. Each LLM-generated erroneous code was then compared against human submissions for the

[1] `https://github.com/ErrorSimulEDM/Digital-Appendix/blob/main/DigitalAppendix_ErrorSimul.pdf`

[2] `https://www.vellum.ai/llm-leaderboard`

[3] `https://livebench.ai/`

[4] `https://openrouter.ai/rankings`

corresponding problems. Since exhaustive all-pairs comparison was computationally infeasible (on average 2,002 human codes per task), we employed a two-stage matching strategy. In Stage 1, we represented each code as a bag of tokens and used Jaccard similarity to retrieve the top-K most similar human codes for each LLM-generated submission ($K = 100$, i.e., Recall@100 retrieval). In Stage 2, we applied the Zhang–Shasha tree edit distance within this candidate set to identify the nearest human match for each LLM-generated code. We use the ZSS distance to the nearest human match as our structural alignment metric (lower values indicate closer alignment with authentic student errors).

For RQ3, to investigate the effect of the struggling level of a problem, we compared edit distances across three struggling levels to analyze how the struggling level of a problem moderates both the diversity of LLM-generated erroneous code and their alignment with authentic student erroneous code.

4.2.2 Qualitative Evaluation

While automated metrics provide a scalable measure of structural similarity, they do not fully capture the pedagogical utility of the generated errors. To address this limitation and to support construct validity, we conducted a qualitative study designed to assess whether the errors generated by the LLM are indistinguishable from authentic student errors to an expert instructor.

Study Design and Protocol. We employed a stratified sampling strategy to construct a balanced dataset of $N = 444$ code submissions across all 37 problems [3, 1]. For each problem we sampled 12 submissions (6 authentic from the CodeWorkout logs and 6 synthetic generated by our best-performing model, *Claude Sonnet 4* with Self-Refine), yielding $37 \times 12 = 444$ items. All submissions were stripped of source-identifying metadata before annotation.

Pilot Study and Codebook Development. Two authors first conducted a pilot study on a random subset of $N = 43$ submissions, double-coding each item independently. Through an iterative collaborative process, the annotators reconciled disagreements and established a consensus codebook [22] covering (i) eight logic-error categories with operational definitions and disambiguation rules (e.g., distinguishing Boundary errors from Condition Logic errors), and (ii) anchor descriptions for each level of the five-point plausibility scale, and (iii) whether the code was written by a Human Student or an LLM. The eight categories used in the final analysis (Table 4) were: *Condition Logic* (faulty boolean expressions in `if`/`while` conditions); *Boundary* (off-by-one and edge-case handling); *Concept* (misunderstanding of the problem requirements or intended algorithm); *Method* (incorrect use of library or built-in methods); *Infinite Loop* (non-terminating control flow); *Scope* (variable-scope or lifetime errors); *Type* (type-related faults, e.g., `==` vs `.equals`); and *Other* (a residual bucket for logical errors that did not map cleanly to a single named category, including strategy- or algorithm-level faults where the chosen approach itself is wrong, and, for authentic submissions, multi-fault or structurally awkward but compilable attempts).

Annotation Phase. The remaining $N = 401$ submissions (205 authentic, 196 synthetic) were divided between the two annotators, who each single-coded their assigned items under the finalized codebook with submissions blinded to source. Cases that the assigned annotator found ambiguous were brought to joint review and resolved by discussion. We did not compute a formal inter-rater reliability statistic on the main set because items were single-coded; the pilot phase served as the calibration step, after which the codebook was held fixed. The annotation criteria were:

(1). Source Attribution (Turing Test): A binary judgment on whether the code was written by a Human Student or an LLM. **(2). Plausibility Score:** A five-point Likert rating (1 = Highly Artificial, 5 = Highly Authentic) assessing whether the error reflects a genuine novice misconception. **(3). Logic Error Taxonomy:** Categorization of the primary functional fault into one of the eight categories above.

5. RESULTS

5.1 RQ1: Diversity

LLMs can generate diverse erroneous code, but vary on different models and prompt strategies. Overall, the results in Table 1 indicate that LLMs (across five models) are capable of producing diverse erroneous code, as reflected in the non-trivial mean edit distances observed across all models and prompting techniques. This suggests that, when prompted for multiple outputs on the same programming task, the generated code samples are not identical but vary in structure and content, thereby enabling the study of model-driven error diversity. However, some differences emerged both across models and across prompting strategies. At the model level, Gemini 2.5 Pro and Claude Sonnet 4 exhibited the largest average edit distance diversity, consistently exceeding 60 under most prompting techniques, indicating that they tend to generate more heterogeneous erroneous solutions. In contrast, GPT-4o demonstrated the smallest diversity, with mean edit distances around 25–40 depending on the prompt, suggesting more conservative or repetitive error patterns. GPT-5 and Grok Code Fast 1 occupied an intermediate position, with mean edit distances in the 40–60 range. Prompting strategies also modulated the diversity of erroneous outputs, though the effect was less uniform across models. For instance, in Claude Sonnet 4, the Self-Refine prompt reduced the mean edit distance compared to IO and CoT, suggesting more constrained revisions. By contrast, for Gemini 2.5 Pro, the CoT prompting increased variability, yielding the highest distance (72.55) observed in the dataset. Meanwhile, in models such as GPT-4o and Grok Code Fast 1, IO and Self-Refine performed comparably, with CoT prompting producing slightly higher diversity in some cases.

5.2 RQ2: Alignment

LLMs can simulate student erroneous code, but vary on different models and prompt strategies. We evaluated the alignment between LLM-generated erroneous code and human submissions by computing the mean of the nearest edit distance between each simulated code and its most similar human counterpart. Smaller distances indicate greater similarity to authentic student errors.

5.2.1 Structural Alignment (AST Analysis)

Table 1: Mean Edit Distance by Prompting Method and Model (values are mean ± SE).

	Models				
Method	*Claude Sonnet 4*	*GPT-4o*	*GPT-5*	*Gemini 2.5 Pro*	*Grok Code Fast 1*
IO	**63.33**±1.56	25.26±1.17	52.54±1.37	63.03±2.16	43.42±1.47
CoT	61.34±1.92	**39.93**±1.96	**60.28**±1.59	**72.55**±2.61	**54.94**±2.41
Self-Refine	46.89±2.02	25.97±1.33	59.97±2.10	59.28±1.83	44.43±2.18

As shown in Table 2, the results show that LLMs are capable of generating erroneous code that resembles student submissions (the nearest edit distance is 16.35), but the alignment of this simulation depends on the choice of model and prompting strategy. The distances are never negligible, meaning the simulation is not perfect, yet some models produce errors strikingly close to those made by students. Claude Sonnet 4 provides the most consistent alignment: across all prompting strategies, its distances remain relatively low, with the best case (16.35 under Self-refine) being the closest match to real student code among all conditions. Grok Code Fast 1 also demonstrates strong potential, achieving the absolute minimum distance (16.22) with Self-Refine. However, its performance fluctuates more across prompts (rising above 30 under IO and CoT), showing that while Grok can mimic student errors well, this depends more on the prompting technique. By contrast, GPT-5 and Gemini 2.5 Pro exhibit much larger distances, often exceeding 80, suggesting that although they generate erroneous code, the errors diverge considerably from authentic student patterns. GPT-4o lies in between: sometimes close to students (20.41 with CoT) but inconsistent, particularly when Self-Refine is used. Prompting strategies influence these outcomes. CoT generally improves alignment, especially for Claude and GPT-4o. Self-Refine has mixed effects: it delivers Grok's best result but worsens GPT-4o's alignment.

5.2.2 Human Evaluation

(1). Source attribution. For source attribution, we report the deception rate: the proportion of LLM-generated submission that annotators misclassified as human-written. By construction, this metric is defined only on the 196 synthetic items. Annotators misclassified 164/196 (**83.7**%) of LLM-generated submissions as human-written, indicating a low level of distinguishability.

(2). Plausibility score: Contrary to the expectation that synthetic errors might be noisier or less coherent, the LLM-generated submissions received a significantly higher mean plausibility score ($M = 4.27, SD = 1.07$) than authentic student errors ($M = 3.78, SD = 1.11$). A Mann-Whitney U test confirmed this difference is statistically significant ($p < 0.001$).

This counterintuitive pattern is explained by the differences in shape between the two error distributions (Table 4). LLM-generated submissions concentrate on clean, single-purpose logical bugs, Condition Logic (46.4%) and Boundary (21.9%) faults dominate, which annotators read as canonical, textbook-style misconceptions. Authentic student code, although pre-filtered for compilability, exhibits a broader and more diffuse distribution: a residual 28.3% of submissions fell into the *Other* category, which captured logical errors that did not map cleanly to a single named type, including strategy- or algorithm-level faults and multi-fault attempts where several slips co-occurred in the same submission. Because the plausibility scale rewarded errors that read as a single, identifiable misconception, the LLM's narrower, well-formed bug distribution scored higher than the more diffuse one in authentic submissions. In practice, the LLM behaves like an idealized novice, valuable for isolating individual misconceptions when training intelligent tutoring systems, but a less faithful proxy for the full messiness of authentic student work.

5.3 RQ3: Struggling Level as a Moderator

LLMs simulate better in low-level questions but have higher diversity in high-level questions. As shown in Figure 2, mean edit distances among LLM submissions increased substantially from low- to medium-struggling problems, and remained high at high-struggling problems for most models. This indicates that harder problems elicited more heterogeneous erroneous outputs, consistent with the expectation that higher-struggling problems open up more possible failure paths. For example, Gemini 2.5 Pro (CoT) rose from 62.34 (low) to 82.89 (medium) and stayed high at 74.59 (high). To probe the diversity of simulated errors, we conducted a case-level analysis using the best-performing setting (Gemini 2.5 Pro with CoT). For each coding problem, we measured the mean pairwise edit distance among all generated erroneous submissions. The results reveal strong problem-specific variation. For example, for problem "sortasum" (mean distance = 8.00), the model repeatedly produced nearly identical error patterns, with minimal structural variation. By contrast, in "fix45" (mean distance = 211.17), the generated errors were far more heterogeneous, with edit distances ranging from near zero to above 250. Heatmaps (Figure 3) illustrate these two representative cases. Additional code examples for both problems are provided in Digital Appendix H.

Alignment of LLM-generated code (simulated–human). Figure 4 illustrates how the struggling level of coding problems affects the similarity between LLM-generated erroneous code and authentic student submissions. Overall, we observe a clear trend: as the level of struggle increases, the mean nearest edit distance also increases, indicating that LLMs find it harder to reproduce student-like logical errors on higher-struggle problems. Across models, Claude Sonnet 4 demonstrates the most stable performance, maintaining relatively low distances across all levels, but still has a slightly longer distance as the struggling level increases. Moreover, other models, such as GPT-5 and Gemini 2.5 Pro, show substantial increases in distance at medium and high levels, indicating that their simulated errors diverge further from authentic student behavior as the struggling level rises.

Table 2: Mean Nearest Edit Distance by Method and Model (values are mean ± SE).

Method	Models				
	Claude Sonnet 4	*GPT-4o*	*GPT-5*	*Gemini 2.5 Pro*	*Grok Code Fast 1*
IO	22.48±1.80	34.30±2.70	93.75±2.46	**56.66**±3.85	39.66±3.06
CoT	24.94±2.22	**20.41**±2.15	100.56±2.47	87.44±3.48	33.54±3.07
Self-Refine	**16.35**±1.80	49.85±3.08	**80.59**±3.47	82.01±3.16	**16.22**±1.78

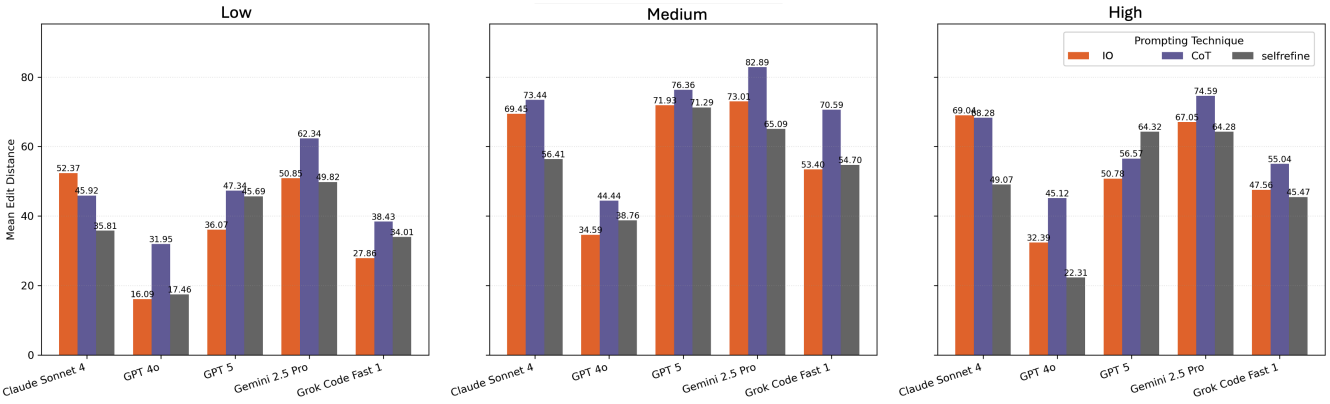


Figure 2: Mean Nearest Edit Distance Between LLM-Generated Errors Across Similar Tasks of Different Struggling Levels

Table 3: Qualitative Evaluation Results: Plausibility and Deception ($N = 401$)

Source	N	Mean Plaus.	Std. Dev	Deception Rate
Human	205	3.78	1.11	-
LLM	196	**4.27**	1.07	**83.7%**

*Mann-Whitney U: $p < 0.001$

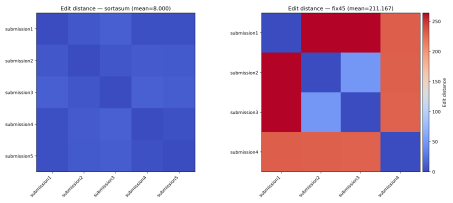


Figure 3: Pairwise Edit Distance Heatmaps Across LLM Code for Two Problems (Gemini 2.5 Pro, CoT Prompting).

6. DISCUSSION AND CONCLUSION

This study investigated whether LLMs can simulate student-like logical errors in programming tasks and under what conditions. We identified two key dimensions, diversity and alignment, as central to their usefulness as proxies for authentic errors. On diversity, all models produced varied erroneous code, though to different extents: Gemini 2.5 Pro and Claude Sonnet 4 showed the broadest diversity (especially with CoT prompting), whereas GPT-4o generated more homogeneous patterns. On alignment (fidelity), i.e., resemblance to authentic erroneous code, Claude Sonnet 4, GPT-4o, and Grok Code Fast 1 aligned most closely with real submissions (notably with Self-Refine prompting), while GPT-5 and Gemini 2.5 Pro diverged more substantially. These trade-offs are illustrated in Figure 5: models strong in diversity (e.g., Gemini) often exhibit weaker alignment, and vice versa, with Claude Sonnet 4 standing out as the most balanced option. Yet even Claude's performance varied across tasks and prompts, underscoring that no single LLM is universally optimal. These findings suggest that model choice should be guided by analytic goals: diversity-oriented models can help explore the breadth of potential misconceptions, while alignment-oriented models are better suited for approximating authentic error distributions.

In addition, our results suggest that the coding problem-struggling level plays a dual role (also see Figure 5): while a higher struggling level increases diversity of simulated errors, expanding the coverage of potential failure modes, it simultaneously reduces alignment between simulated and authentic code, suggesting that while models generate a wide variety of errors, these may diverge from the authentic patterns observed in learners. This duality highlights the importance of considering both diversity and fidelity when utilizing LLMs as proxies for student error data generation and analysis: models may excel at producing diverse erroneous code for higher-struggling tasks, but not all of those errors are truly representative of human misconceptions. Beyond the immediate evaluation of LLMs as error simulators, our findings intersect with ongoing discussions in learning analytics. In practical deployments, these diversity–fidelity trade-offs can be used to tune synthetic error generation toward either broad misconception coverage or closer approximation of observed classroom error distributions. The capacity of LLMs to generate diverse yet realistic errors offers a potential avenue for augmenting student data, particularly in contexts where authentic data is scarce, sensitive, or difficult to collect.

Finally, simulating students' programming errors using LLMs holds significant pedagogical value, particularly in address-

Table 4: Distribution of Error Types: Human students vs. LLM-generated.

Source	Condition	Boundary	Concept	Method	Infinite	Scope	Type	Other
Human students	**24.9%**	12.2%	14.6%	9.3%	1.5%	5.4%	3.9%	28.3%
LLM-generated	**46.4%**	**21.9%**	16.8%	5.6%	0.5%	0.0%	0.5%	7.1%

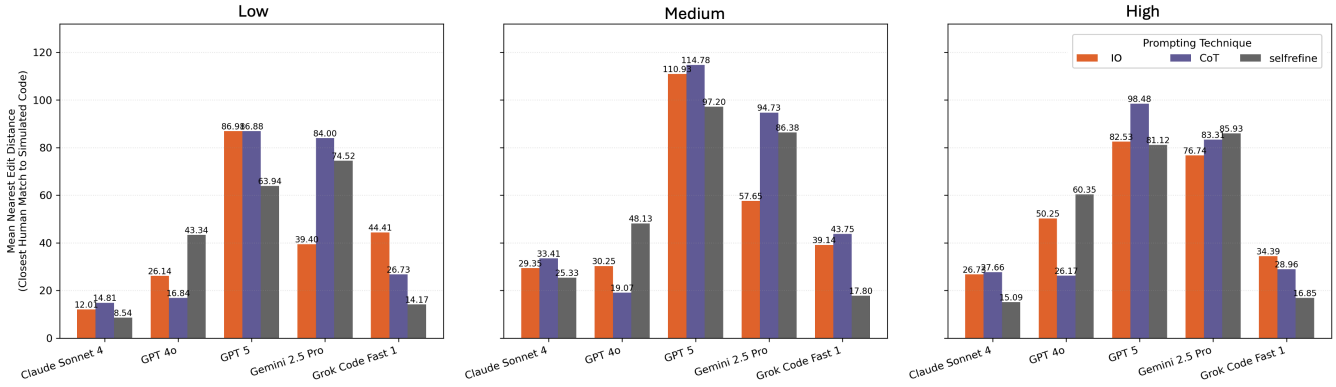


Figure 4: Mean Nearest Edit Distance Between LLM-Generated and Human Code Errors Across Tasks of Different Struggling Levels

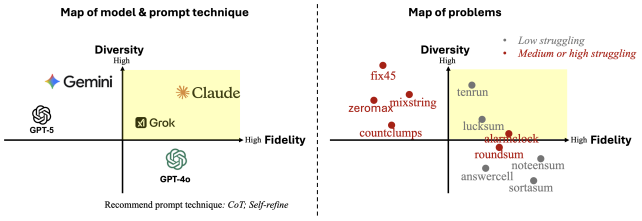


Figure 5: Trade-offs Between Alignment (fidelity) and Diversity in LLM-Simulated Student Errors

ing the challenge of data scarcity in educational technology. First and foremost, large-scale synthetic error datasets can be leveraged to train and evaluate automated tutoring systems (AI Tutors). This allows intelligent systems to better diagnose and provide feedback on diverse, realistic student misconceptions without being constrained by the limited availability of human-generated data. Second, from a student-centric perspective, these generated errors can be integrated into "teachable agent" environments. By tasking learners with diagnosing and correcting mistakes made by an AI proxy, the system fosters vicarious learning and deepens students' metacognitive debugging skills. Lastly, while it is impractical for educators to manually review thousands of synthetic code snippets, curated subsets of these LLM-generated errors can still serve as valuable resources for teacher professional development [31]. Exposure to a concentrated variety of anticipated misconceptions allows teachers to practice formulating effective pedagogical responses and design more targeted instructional strategies prior to actual instruction [11].

Despite these promising results, several limitations should be considered when interpreting our findings. Our experiments are grounded in CodeWorkout and a curated set of 37 introductory Java problems, so the observed diversity-alignment = trade-offs may not generalize to other institutions, curricula, assessment formats, or programming languages. We operationalized struggling level as the total number of submissions per problem, which is a scalable proxy but may conflate difficulty with other factors (e.g., assignment placement, popularity, or course policies); alternative measures, such as error rates, time-to-correct, or concept difficulty models, could yield different patterns. Furthermore, our generation constraints (compilable code with exactly one non-trivial logical error) intentionally reduce syntactic noise and multi-fault behaviors common in authentic novice submissions, potentially producing an "idealized learner" distribution that is useful for isolating misconceptions but less representative of messy real-world student work; future studies should extend evaluation across broader datasets, richer difficulty measures, and semantic metrics. Finally, while the AST edit distance serves as a robust quantitative proxy for measuring structural variation, we acknowledge its inherent limitations. First, raw numerical distances can lack direct interpretability context across different programming tasks. Second, as a purely structural metric, it fails to fully capture semantic diversity; functionally equivalent codes may exhibit distinct AST structures, and vice versa. Future research should consider employing normalized metrics alongside execution-based semantic evaluations (e.g., test-case execution traces) to provide a more holistic assessment of error diversity.

Acknowledgments

This research was originally conducted during the LearnLab Summer School in Carnegie Mellon University, with registration supported by CSSplice (NSF: #2213789, #2213790, #2213791, #2213792). This material is based in part upon work supported by the National Science Foundation under Grant No. 2315294. The authors gratefully acknowledge Utah State University for funding the API tokens. [5] [6]

7. REFERENCES

[1] S. K. Ahmed, R. A. Mohammed, A. J. Nashwan, R. H. Ibrahim, A. Q. Abdalla, B. M. M. Ameen, and R. M. Khdhir. Using thematic analysis in qualitative research. *Journal of Medicine, Surgery, and Public Health*, 6:100198, 2025.

[2] E. Albrecht and J. Grabowski. Sometimes it's just sloppiness-studying students' programming errors and misconceptions. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 340–345, 2020.

[3] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, 2006.

[4] N. C. Brown and A. Altadmri. Investigating novice programming mistakes: educator beliefs vs. student data. In *Proceedings of the Tenth Annual Conference on International Computing Education Research*, ICER '14, page 43–50, New York, NY, USA, 2014. Association for Computing Machinery.

[5] J. Cao, H. Nguyen, S. Yavuz, B. Yu, S. Wang, P. K. Bharaj, and D. C. Francis. Developing authentic simulated learners for mathematics teacher learning: Insights from three approaches with large language models. *arXiv preprint arXiv:2604.04361*, 2026.

[6] S. H. Edwards and K. P. Murali. Codeworkout: Short programming exercises with built-in data collection. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '17, page 188–193, New York, NY, USA, 2017. Association for Computing Machinery.

[7] A. Ettles, A. Luxton-Reilly, and P. Denny. Common logic errors made by novice programmers. In *Proceedings of the 20th Australasian Computing Education Conference*, pages 83–89, 2018.

[8] A. Fallah, A. Keramati, M. A. Nazari, and F. S. Mirfazeli. Automating theory of mind assessment with a llama-3-powered chatbot: Enhancing faux pas detection in autism. In *2024 14th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 365–372, 2024.

[9] Z. Hao, J. Cao, R. Li, J. Yu, Z. Liu, and Y. Zhang. Mapping student-ai interaction dynamics in multi-agent learning environments: Supporting personalized learning and reducing performance gaps. *Computers & Education*, 241:105472, 2026.

[10] A. Hicks, Y. Shi, A.-B. Lekshmi-Narayanan, W. Yan, and S. Marwan. An approach to detect abnormal submissions for codeworkout dataset, 2024.

[11] B. Hu, J. Zhu, Y. Pei, and X. Gu. Exploring the potential of llm to enhance teaching plans through teaching simulation. *npj Science of Learning*, 10(1):7, 2025.

[12] J. Jackson, M. Cobb, and C. Carver. Identifying top java errors for novice programmers. In *Proceedings frontiers in education 35th annual conference*, pages T4C–T4C. IEEE, 2005.

[13] A. Keramati, M. R. Keramati, and M. H. Arefian. Students' reflection on the effect of collaborative learning on the learning environment and academic achievement in online reflective platforms. *Reflective Practice*, 25(2):117–131, 2024.

[14] J. Leinonen, P. Denny, O. Kiljunen, S. MacNeil, S. Sarsa, and A. Hellas. Llm-itation is the sincerest form of data: Generating synthetic buggy code submissions for computing education. In *Proceedings of the 27th Australasian Computing Education Conference*, ACE '25, page 56–63, New York, NY, USA, 2025. Association for Computing Machinery.

[15] S. MacNeil, P. Denny, A. Tran, J. Leinonen, S. Bernstein, A. Hellas, S. Sarsa, and J. Kim. Decoding logic errors: a comparative study on bug detection by students and large language models. In *Proceedings of the 26th Australasian Computing Education Conference*, pages 11–18, 2024.

[16] S. MacNeil, M. Rogalska, J. Leinonen, P. Denny, A. Hellas, and X. Crosland. Synthetic students: A comparative study of bug distribution between large language models and computing students. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 1*, SIGCSE Virtual 2024, pages 137–143, New York, NY, USA, 2024. Association for Computing Machinery.

[17] A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, N. Dziri, S. Prabhumoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, and P. Clark. Self-refine: iterative refinement with self-feedback. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[18] D. Martynova, J. Macina, N. Daheim, N. Yalcin, X. Zhang, and M. Sachan. Can LLMs effectively simulate human learners? teachers' insights from tutoring LLM students. In E. Kochmar, B. Alhafni, M. Bexte, J. Burstein, A. Horbach, R. Laarmann-Quante, A. Tack, V. Yaneva, and Z. Yuan, editors, *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 100–117, Vienna, Austria, July 2025. Association for Computational Linguistics.

[19] N. Matsuda. Teachable agent as an interactive tool for cognitive task analysis: A case study for authoring an expert model. *International Journal of Artificial Intelligence in Education*, 32(1):48–75, mar 2022.

[20] J. Mayes. Still to learn from vicarious learning. *E-Learning and Digital Media*, 12(3-4):361–371, 2015.

[21] E. Mazzullo, O. Bulut, T. Wongvorachan, and B. Tan. Learning analytics in the era of large language models. *Analytics*, 2(4):877–898, 2023.

[5] The source code associated with this study is available at: `https://github.com/ErrorSimulEDM/Code`

[6] Digital appendix is available at: `https://github.com/ErrorSimulEDM/Digital-Appendix`

[22] M. Naeem, W. Ozuem, K. Howell, and S. Ranfagni. A step-by-step process of thematic analysis to develop a conceptual model in qualitative research. *International Journal of Qualitative Methods*, 22:16094069231205789, 2023.

[23] M. H. Nguyen, S. Tschiatschek, and A. Singla. Large language models for in-context student modeling: Synthesizing student's behavior in visual programming. In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 341–348, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[24] S. Pan, R. Schmucker, B. Garcia Bulle Bueno, S. A. Llanes, F. Albo Alarcón, H. Zhu, A. Teo, and M. Xia. Tutorup: What if your students were simulated? training tutors to address engagement challenges in online learning. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA, 2025. Association for Computing Machinery.

[25] Y. Qian and J. Lehman. Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Trans. Comput. Educ.*, 18(1), Oct. 2017.

[26] F. Tambon, A. Moradi-Dakhel, A. Nikanjam, F. Khomh, M. C. Desmarais, and G. Antoniol. Bugs in large language models generated code: an empirical study. *Empirical Softw. Engg.*, 30(3), Feb. 2025.

[27] Z. Wang, Z. Zhou, D. Song, Y. Huang, S. Chen, L. Ma, and T. Zhang. Towards understanding the characteristics of code generation errors made by large language models. In *Proceedings of the IEEE/ACM 47th International Conference on Software Engineering*, ICSE '25, page 2587–2599. IEEE Press, 2025.

[28] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

[29] T. Wu, J. Chen, W. Lin, M. Li, Y. Zhu, A. Li, K. Kuang, and F. Wu. Embracing imperfection: Simulating students with diverse cognitive levels using LLM-based agents. In W. Che, J. Nabende, E. Shutova, and M. T. Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9887–9908, Vienna, Austria, July 2025. Association for Computational Linguistics.

[30] K. Zhang and D. Shasha. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal on Computing*, 18(6):1245–1262, 1989.

[31] N. Zhang, F. Ke, C.-P. Dai, S. A. Southerland, and X. Yuan. Seeking to support preservice teachers' responsive teaching: Leveraging artificial intelligence-supported virtual simulation. *British Journal of Educational Technology*, 56(3):1148–1169, 2025.

[32] Z. Zhang, D. Zhang-Li, J. Yu, L. Gong, J. Zhou, Z. Hao, J. Jiang, J. Cao, H. Liu, Z. Liu, L. Hou, and J. Li. Simulating classroom education with LLM-empowered agents. In L. Chiruzzo, A. Ritter, and L. Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10364–10379, Albuquerque, New Mexico, Apr. 2025. Association for Computational Linguistics.

[33] L. Zheng, A. He, C. Qi, H. Zhang, and X. Gu. Cognitive echo: Enhancing think-aloud protocols with llm-based simulated students. *British Journal of Educational Technology*, 56(5):2019–2042, 2025.

[34] B. J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory into practice*, 41(2):64–70, 2002.

Spatiotemporal Link Formation Prediction in Social Learning Networks Using Graph Neural Networks

Ali Mohammadiasl
UC San Diego
amohammadiasl@ucsd.edu

Bita Akram
NC State University
bakram@ncsu.edu

Seyyedali Hosseinalipour
University at Buffalo–SUNY
alipour@buffalo.edu

Rajeev Sahay
UC San Diego
r2sahay@ucsd.edu

ABSTRACT

Social learning networks (SLNs) are graphical representations that capture student interactions within educational settings (e.g., a classroom), with nodes representing students and edges denoting interactions. Accurately predicting future interactions in these networks (i.e., link prediction) is crucial for enabling effective collaborative learning, supporting timely instructional interventions, and informing the design of effective group-based learning activities. However, traditional link prediction approaches are typically tuned to general online social networks (OSNs), often overlooking the complex, non-Euclidean, and dynamically evolving structure of SLNs, thus limiting their effectiveness in educational settings. In this work, we propose a graph neural network (GNN) framework that jointly considers the temporal evolution within classrooms and spatial aggregation across classrooms to perform link prediction in SLNs. Specifically, we analyze link prediction performance of GNNs over the SLNs of four distinct classrooms across their (i) temporal evolutions (varying time instances), (ii) spatial aggregations (joint SLN analysis), and (iii) varying spatial aggregations at varying temporal evolutions throughout the course. Our results indicate statistically significant performance improvements in the prediction of future links as the courses progress temporally. Aggregating SLNs from multiple classrooms generally enhances model performance as well, especially in sparser datasets. Moreover, we find that jointly leveraging both the temporal evolution and spatial aggregation of SLNs significantly outperforms conventional baseline approaches that analyze classrooms in isolation. Our findings demonstrate the efficacy of educationally meaningful link predictions, with direct implications for early-course decision-making and scalable learning analytics in and across classroom settings.

Keywords

Graph Neural Networks, Link Prediction, Social Learning Networks, Social Network Analysis.

Ali Mohammadiasl, Bita Akram, Seyyedali Hosseinalipour, and Rajeev Sahay. Spatiotemporal Link Formation Prediction in Social Learning Networks Using Graph Neural Networks. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 541–548. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040072

1. INTRODUCTION

Collaborative learning is a major pillar of modern education, shaping student engagement, persistence, and academic success. Consequently, instructors increasingly incorporate structured peer interactions, such as peer instruction and discussion forums, to enhance learning outcomes [1, 3]. Yet, in practice, collaborative behaviors do not emerge uniformly across students or over time: some students rapidly become socially central, while others remain weakly connected or isolated, particularly in the early stages of a course. These uneven and evolving interaction patterns complicate instructors' ability to anticipate collaboration dynamics and intervene effectively when support is most needed.

In order to model and analyze such interaction dynamics, social learning networks (SLNs) have emerged as a powerful abstraction. SLNs represent classroom interactions as graphs, where *nodes* correspond to students and *edges* denote interactions arising from discussions, question–answer exchanges, or collaborative activities [12, 22]. This abstraction enables a principled investigation of network inference problems in educational graphs, with *link prediction* (i.e., forecasting which student interactions are likely to form in the future) serving as one of the fundamental problems in this setting.

1.1 Challenges and Motivations

Within SLNs, *link prediction* has emerged as a valuable tool for forecasting future collaborations, identifying at-risk students, and informing instructional interventions. Furthermore, accurate link prediction can reveal relationships between students' behavioral characteristics and their tendencies toward class involvement, which is often linked to their learning performance [29]. However, prior work on link prediction in educational settings has largely focused on analyzing each classroom independently, treating the SLN associated with a single course offering as an isolated graph [20, 30]. While this approach is intuitive, it implicitly assumes that meaningful structure must be learned anew for every classroom, ignoring common patterns of social and pedagogical dynamics that recur across courses, instructors, and academic terms. In addition, this approach fails to capture the temporal processes through which collaborations form early in the course when few observed links have been formed (i.e., the cold-start problem), thus preventing accurate link prediction precisely when early intervention would be most beneficial. Moreover, the inherent complexity of SLNs, including their evolving topology and irregular non-Euclidean

structure, renders traditional data-driven models [19] [27], such as convolutional neural networks (CNNs), ill-equipped for the task of link prediction.

Graph Neural Networks (GNNs), on the other hand, offer a powerful alternative to traditional CNNs when it comes to predicting links in SLNs. Unlike CNNs, which are primarily designed for grid-like data such as images, GNNs are tailored for irregular graph structures. This makes them particularly well-suited for SLNs, where nodes (students) and edges (interactions) form complex non-Euclidean structures [5]. Despite their potential, the application of GNNs in SLNs remains in its early stages, with only a handful of studies exploring their potential and limitations [20].

To address this gap in the literature, *our work leverages GNNs to unlock two critical domains in link prediction for SLNs: the 'temporal' domain and the 'spatial' domain.* Specifically, in the temporal domain, we examine the evolution of SLN graphs over time, while in the spatial domain, we investigate the effects of combining SLN graphs from different classroom contexts to improve link prediction performance equitably across classrooms.

1.1.1 Exploring the Temporal Domain

SLNs are inherently dynamic: the topology of these networks evolves as student interactions change throughout the course. This temporal evolution can significantly affect GNNs' future link prediction performance in SLNs. By analyzing how SLN structures shift at different stages of a course, we can uncover deeper insights into student interactions and the conditions under which link prediction performs best. We subsequently present and investigate, for the first time in the literature, the following hypothesis:

Hypothesis 1: *Do SLNs' temporal structures have a significant impact on the prediction capability of GNNs trained on them? If yes/no, how do link prediction performance metrics vary during the temporal evolution of SLNs?*

1.1.2 Investigating the Spatial Domain

Our study on the spatial domain of SLNs introduces the combination of SLN graphs across different classrooms. At first glance, merging SLNs might seem to improve model performance by providing more data. However, our intuition suggests that the unique topological structures inherent in each SLN, stemming from varied interaction patterns among peers in different classrooms, can lead to unexpected outcomes. Subsequently, our formally proposed hypothesis that we investigate for the first time in the literature is as follows:

Hypothesis 2: *Does combining SLNs together lead to better performance of the GNN models trained on them? If yes/no, how do link prediction performance metrics vary upon combining SLNs?*

1.1.3 Spatiotemporal Analysis

A careful observation of the interrelations between the temporal and spatial domains reveals non-trivial interactions. For instance, our intuition may suggest that combining SLNs at earlier stages of a course might be more beneficial, as networks are still forming and less defined. However, as SLNs develop their distinct patterns later in the course, merging them might not yield the same performance improvements. To encapsulate this nuanced view, we propose and investigate the following hypothesis:

Hypothesis 3: *Does combining SLNs together at different time stamps during a course lead to varying performance of the GNN models trained on them? If yes/no, how do link prediction performance metrics vary during the combination of SLNs at different time stamps of a course?*

1.2 Overview and Summary of Contributions

Formally, our contributions can be summarized as follows:

- We pioneer an investigation into the temporal evolution and spatial aggregation of SLNs by analyzing how link formation and interaction patterns change throughout a class and how classrooms can benefit one another through sharing their data. Our time-evolving analysis over multiple classroom SLNs provides novel insights into these dynamics.
- We develop a GNN-based framework designed to tackle the unique challenges presented by sparse, imbalanced, and temporally evolving SLN datasets. This framework is optimized to capture the complex topological features inherent in SLNs. We then use this framework to validate our ***Hypothesis 1***.
- We demonstrate that combining data from multiple SLNs, spanning different classroom contexts, can enhance link prediction performance under certain conditions. This contribution is focused on validating our ***Hypothesis 2***.
- By examining the interplay between the temporal evolution and spatial data aggregation of SLNs, we uncover non-trivial interactions that influence overall link prediction accuracy. This integrated analysis addresses our ***Hypothesis 3***, where our findings offer a refined understanding of how temporal and spatial factors collectively inform link prediction strategies.

2. RELATED WORK

Link prediction in general computational graphs (i.e., non-SLN graphs such as Online Social Networks (OSNs)) has evolved through various methodologies over time. Early methods relied on deriving features from the graph topology [15, 17] to infer future trends [13, 31]. While effective in some cases, these heuristic-based methods often fail to capture the inherent complexities of dynamic, rapidly-evolving graphs. To address these limitations, more rigorous link prediction approaches were developed, such as matrix factorization and probabilistic models, which aim to learn low-dimensional representations of nodes [11], [26], [18], [33]. Although these models yielded improved prediction accuracy, later methods further advanced performance by offering greater scalability and adaptability to capture the complex, evolving nature of OSNs. In particular, the advent of feature-based deep learning introduced models based on artificial neural networks for link prediction, which leverage node characteristics and graph adjacency information to improve link prediction in both general OSNs [8, 16, 24, 32] as well as SLNs [7, 20, 30]. While these methods achieved notable success, they typically treat graphs as static structures and lack the capability to effectively capture the temporal dynamics inherent in SLNs. Moreover, they fail to account for

the intricate topological structures and co-temporal evolution of interactions across multiple classrooms, limiting their applicability in educational settings.

More recently, GNNs have emerged as powerful tools for link prediction in general graph structures due to their ability to learn both the local structure and the global topology of a graph [2,4,9,14,21,23,28]. GNN-based models, such as Graph Convolutional Networks (GCNs) [10], GraphSAGE [6], and Graph Attention Networks (GATs) [25], have demonstrated superior performance compared to feature-based deep learning models by aggregating neighborhood information through learned message-passing schemes among nodes. Despite these advancements, no prior work has specifically examined GNN-based link prediction approaches in dynamically evolving SLNs, thus overlooking the temporal evolution and spatial aggregation properties that are unique to SLNs. As a result, the impact of SLN dynamics over time, the integration of information across classroom contexts, and the interplay between these two dimensions remain largely unexplored. Our work aims to address this critical gap in the literature.

3. METHODOLOGY

In this section, we begin by describing the mathematical representation of SLNs as graphs (Sec. 3.1). Next, we outline the task of link prediction in SLNs, detailing how GNNs are used to infer future links (Sec. 3.2). We then introduce our joint classroom approach, detailing the integration of SLNs to improve predictive performance in individual classrooms (Sec. 3.3). Lastly, we describe the evaluation metrics used to assess the link prediction performance (Sec. 3.4).

3.1 Representation of SLNs as Graphs

We represent each SLN as a graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of *nodes* (i.e., students) and $\mathcal{E}$ represents the set of *edges* (i.e., interactions between students). Each edge $e_{ij} \in \mathcal{E}$ signifies an interaction between two nodes/students v_i and v_j, where $v_i, v_j \in \mathcal{V}$, such as collaborative projects, discussions, or information exchanges. Subsequently, we define the adjacency matrix of the SLN graph as $\mathbf{A} = [A_{ij}]_{v_i, v_j \in \mathcal{V}}$, where

$$A_{ij} = \begin{cases} 1, & \text{if } (v_i, v_j) \in \mathcal{E} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

In words, A_{ij} represents the presence ($A_{ij} = 1$) or absence ($A_{ij} = 0$) of a connection/edge between nodes v_i and v_j.

3.2 Link Prediction in SLNs

The temporal evolution of an SLN is illustrated in Fig. 1, showcasing a simplified example of an SLN across three sequential stages of a course. Subsequently, given the SLN graph described in Sec. 3.1 at some point t during the duration of a course/class (e.g., week 4 of the semester), our objective is to predict which links will form at a future time $t+t_0$ (e.g., week 6) between students by using GNNs. To this end, we adopt GraphSAGE [6], which learns node embeddings/features by iteratively aggregating information from local neighborhoods. As new nodes are not introduced in the SLN, this inductive learning approach enables the model to dynamically adapt node embeddings as local structures (i.e., student interaction patterns) evolve over time, making it well-suited for the temporal changes of student interactions throughout the course.

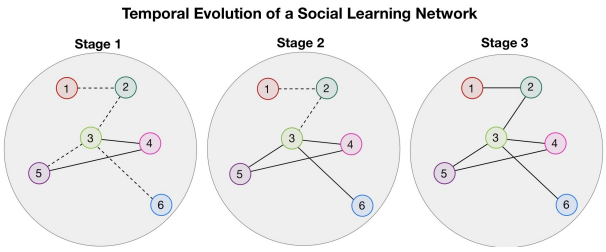


Figure 1: Visualization of SLN temporal evolution. Left to right shows the progress of the classroom from the beginning to the end. A solid line denotes an existing edge at that stage, and a dotted line denotes a non-existing edge at that stage that will become an existing one in a future stage.

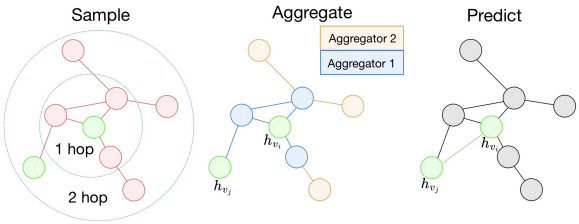


Figure 2: An overview of the GraphSAGE pipeline for link prediction. Left: Sampling stage, where the 1-hop and 2-hop neighborhoods of the target nodes (highlighted in green) are collected. Middle: Aggregation stage, where each node's embedding is updated by aggregating the features of its neighbors using multiple layers of aggregation (in our case, mean aggregator). Right: Prediction stage, where a dot product is computed between the final embeddings of the two target nodes to predict the existence of a link.

3.2.1 *Link Predictions based on GNNs in a Nutshell*

In a nutshell, to predict whether an interaction (i.e., a link) will form between two students v_i and v_j, we compute a numerical representation (embedding) for each student using a GraphSAGE model as we will later discuss (see Sec. 3.2.2). These embeddings capture the structural context of each student within the SLN. The likelihood of a future interaction is then estimated by taking the dot product of the respective node embeddings (potentially after propagating through one or more hidden layer) as follows:

$$\hat{y}_{v_i, v_j} = \sigma(h_{v_i}^\top h_{v_j}), \quad (2)$$

where $\sigma(\cdot)$ is the sigmoid function, h_{v_i} and h_{v_j} are the learned embeddings for students v_i and v_j, respectively ($h_{v_i}, h_{v_j} \in \mathbb{R}^d$), and $\hat{y}_{v_i, v_j} \in \mathbb{R}$.

3.2.2 *Obtaining Node Embeddings via GNN*

To obtain the node embeddings, we use a *two-layer GraphSAGE model*, where each layer aggregates information from a student's neighbors in the SLN. This model, depicted in Fig. 2, replicates a convolutional filter that is capable of operating in a non-Euclidean space (i.e., graph topology). Intuitively, at each SAGEConv layer, GraphSAGE updates the representation of a node by combining its own features with the aggregated features of its neighbors. By the second layer, the embedding of a node reflects not only its own character-

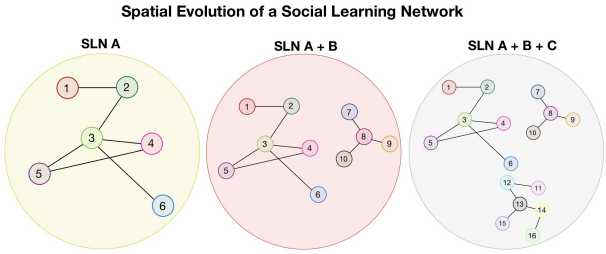


Figure 3: Visualization of spatial evolution. The left displays a single SLN A (representing a single classroom), the middle is SLN A combined with SLN B (i.e., two classrooms combined together), and the right is SLN A and B combined with SLN C (i.e., three classrooms combined together).

istics but also the structure and features of its broader local neighborhood, enabling the model to better infer potential future connections among nodes. In particular, the model begins with an input feature matrix $\mathbf{X} = [x_{v_i}]_{v_i \in \mathcal{V}} \in \mathbb{R}^{|\mathcal{V}| \times F}$, where the i^{th} row (i.e., $x_{v_i} \in \mathbb{R}^F$) corresponds to the feature vector of student v_i consisting of F features. Since our SLNs do not contain explicit node features (e.g., profile or grade data), each student is initialized with a unique trainable embedding vector. These initial embeddings are then refined through neighbor aggregation as follows.[1]

In the first layer, the model updates each student's embedding by averaging its own feature vector with those of its directly connected peers (1-hop neighbors). This step captures local interaction patterns through the following embedding:

$$h_{v_i}^{(1)} = \zeta\Big(W_1 \cdot \text{MEAN}\left(\{x_{v_j} | v_j \in \mathcal{N}(v_i) \cup \{v_i\}\}\right) + b_1\Big), \quad (3)$$

where $\zeta(\cdot)$ is the activation function, $h_{v_i}^{(1)} \in \mathbb{R}^{d_1}$ is the updated embedding for student v_i after one round of aggregation, $\mathcal{N}(v_i)$ is the set of students directly connected to v_i in the SLN (i.e., students who have interacted with v_i), $x_{v_j} \in \mathbb{R}^F$ is the feature vector of student v_j, W_1 and b_1 are trainable weights and bias parameters, and MEAN$(\cdot)$ takes the element-wise average across the set.

In the second layer, the model does the same aggregation, except this time, it uses the embedding from the first layer, allowing to implicitly capture information from 2-hop neighbors. In particular, the second layer of the model captures the following embedding:

$$h_{v_i}^{(2)} = \zeta\left(W_2 \cdot \text{MEAN}\left(\{h_{v_j}^{(1)} | v_j \in \mathcal{N}(v_i) \cup \{v_i\}\}\right) + b_2\right), \quad (4)$$

where $h_{v_i}^{(2)}$ is the final embedding for student v_i (i.e., used as h_{v_i} in (2)), and W_2 and b_2 are trainable parameters.

3.3 Classroom Merging and GNN Training

3.3.1 Merging Classrooms

To assess the impact of training a GNN for link prediction, we conduct experiments on individual and joint SLNs, capturing their spatial evolution. In particular, initially, we train separate GNN models for each classroom, using the same time instances of the SLN graph across the classrooms to ensure consistency across experiments. We then construct combined SLN graphs by merging classroom SLNs, systematically evaluating the combination of multiple classroom SLNs into a single joint graph. Subsequently, we train GNNs on these combined SLNs and evaluate their performance on individual classrooms to verify the impact of classroom combination on the link prediction performance of the trained models. We visualize the spatial aggregation of SLNs in Fig. 3.

3.3.2 GNN Training

To train GNNs over our SLNs (either on isolated or merged classrooms), we first note that there is an extreme class imbalance inherent in SLNs. This imbalance arises because *SLNs are sparse graphs*: the vast majority of possible node pairs in the graph do not share an edge, meaning that most students do not interact with each other.

To mitigate this imbalance during training, we use a weighted binary cross-entropy loss: for a predicted link score s of a given node pair and the respective ground truth label $y \in \{0, 1\}$ (1 for edge, 0 for non-edge), the loss is defined as:

$$\ell(s, y) = \begin{cases} \text{PCW} \cdot \log\left(1 + e^{-s^+}\right), & \text{if } y = 1, \\ \log\left(1 + e^{s^-}\right), & \text{if } y = 0. \end{cases}$$

The total loss across a positive and negative sample pair is:

$$\mathcal{L} = \text{PCW} \cdot \log\left(1 + e^{-s^+}\right) + \log\left(1 + e^{s^-}\right),$$

where s^+ and s^- denote the scores for a positive (i.e., connected) and negative (i.e., not connected) student pair, respectively. We found through a grid search, optimizing for the minimization of false negatives, that fixing *positive class weight* PCW $= 20$ provides a stable and effective training regime across all SLNs, balancing the impact of class imbalance without over-amplifying the minority class loss.

3.4 Evaluation Metrics

Although accuracy is a conventional metric for classification tasks, it is not appropriate in link prediction due to the large class-imbalance, where a model with relatively high accuracy may fail to predict any edges. Instead, we use the *Area Under the receiver operating characteristic Curve (AUC)*, as it provides a robust measure of predictive effectiveness in class-imbalanced settings since it evaluates a model's ability to rank positive samples (i.e., student pairs with edges) higher than negative ones, making it a more reliable indicator of performance in sparse graphs. The AUC is computed by

$$\text{AUC} = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{(s^+, s^-) \in (\mathcal{P}, \mathcal{N})} \mathbb{I}(s^+ > s^-), \quad (5)$$

where $\mathcal{P}$ and $\mathcal{N}$ denote the sets of predicted scores for positive and negative samples, respectively, and $\mathbb{I}(\cdot)$ is an indicator function that returns 1 if the positive score s^+ is greater than the negative score s^-, and 0 otherwise. In our context, the model assigns a score to each link (i.e., a student pair), where a higher score indicates a higher likelihood that the link should exist and a threshold score of 0.5 is used to determine the prediction of a link.

[1]Fig. 2 illustrates GraphSAGE assuming aggregation over all neighbors for clarity. However, in practice, GraphSAGE samples a fixed-size subset of neighbors at each layer to reduce computational overhead and memory usage.

Table 1: Metrics for each considered dataset including the number of nodes, edges, duration (weeks), and posts.

Dataset	# Nodes	# Edges	Duration	# Posts
VS	677	4702	5	7484
ML	3290	30610	12	25481
AL	1165	6773	13	16276
CP	900	3418	8	8255

4. RESULTS AND DISCUSSION

Henceforth, we discuss our empirical setup (Sec. 4.1) followed by analyzing our results in temporal (Sec. 4.2), spatial (Sec. 4.3), and spatiotemporal settings (Sec. 4.4). The code used in this work is available at `https://github.com/AliMohammadiasl/GNNs-for-SLNs`.

4.1 Empirical Setup

For our empirical analysis, we employ four SLNs collected from four distinct Massive Open Online Course (MOOC) datasets: Virtual Shakespeare (VS), Machine Learning (ML), Algorithms (AL), and English Composition (CP). The SLN data we used was introduced in [30] and captures structured interactions that occurred between students on discussion forums. Specifically, if student v_i and v_j interact on a discussion forum by posting an answer or a follow-up question to an initial post, we consider $A_{ij} = 1$. Our considered datasets intentionally span both STEM and humanities courses so that the efficacy of our approach can be evaluated across various disciplines. Tab. 1 summarizes these datasets.

Our GraphSAGE model consists of an $F = 16$-dimensional embedding for each node at the input followed by two convolutional layers. We use Rectified Linear Unit (ReLU) as the activation function following both hidden layers. For training, we perform full-graph updates for a total of 1500 epochs, optimizing the model with the Adam optimizer at a learning rate of $5e-4$ using the above-described weighted binary-cross entropy loss function.

To ensure robustness (in terms of the time that the model is used during the term/semester), we evaluate our models in multiple temporal snapshots of the SLNs. We define progress as the percentage of the term that has elapsed, with 0% representing the start and 100% the end. We consider 25%, 50%, 75%, and 90% progress, and we perform temporal k-fold cross-validation (with $k = 10$) and report the average AUC across all partitions for statistical validation.

4.2 Temporal Graph Analysis

We begin by examining how link prediction in SLNs evolve temporally throughout the duration of a single course (addressing **Hypothesis 1** from Sec. 1). Inspecting Tab. 2 and focusing on isolated GNN training (i.e., 'GNN iso'), which represents GNNs trained on a particular course in isolation, we observe consistent AUC improvements as more temporal data become available in each course. As expected, the AUC improves steadily as the course progresses, and the model benefits from richer structural information in later stages, particularly for smaller courses (e.g., VS, AL, and CP), which suffer from link formation data early on, than larger ones (ML). This indicates that the model becomes increasingly capable of distinguishing between likely and unlikely links between students as it observes more interactions. These trends indicate that temporal progress has a measurable impact on model performance, affirming **Hypothesis 1** from Sec. 1. Overall, later snapshots provide more robustness for learning, while early-course graphs pose prediction challenges due to the limited structural context.

Table 2: AUC performance of each considered model across four stages of course progress (25%, 50%, 75%, 90%). Temporal trends can be observed across progress stages of each course, while spatial effects are seen by comparing 'GNN iso' and 'GNN all' at each progress stage.

Dataset	Progress	GNN all	GNN iso	GCN [10]	GAT [25]	CNN [20]
VS	25%	**0.88**	0.86	0.82	0.87	0.64
VS	50%	**0.92**	0.89	0.85	0.89	0.62
VS	75%	**0.94**	0.88	0.86	0.90	0.60
VS	90%	**0.94**	0.88	0.85	0.89	0.67
ML	25%	**0.85**	0.84	0.76	0.82	0.72
ML	50%	**0.86**	0.85	0.70	0.83	0.83
ML	75%	0.86	0.84	0.69	0.83	**0.90**
ML	90%	**0.88**	0.83	0.87	0.84	0.81
AL	25%	**0.92**	0.92	0.83	0.89	0.82
AL	50%	**0.94**	0.93	0.86	0.90	0.91
AL	75%	**0.94**	0.93	0.86	0.92	0.93
AL	90%	**0.95**	0.94	0.86	0.92	0.93
CP	25%	**0.86**	0.84	0.82	0.83	0.66
CP	50%	**0.90**	0.84	0.85	0.85	0.71
CP	75%	**0.89**	0.85	0.83	0.85	0.80
CP	90%	**0.91**	0.88	0.85	0.85	0.60

4.3 Spatial Graph Analysis

We next analyze the spatial evolution of link prediction in SLNs by exploring how combining SLNs across different classrooms affects model performance (addressing **Hypothesis 2** from Sec. 1). To evaluate this, in Tab. 2, we compare the performance of the models when trained on individual SLNs versus on combined SLNs (i.e., 'GNN iso' and 'GNN all'). The results reveal how merging SLNs positively impacts model performance across different stages of the course. For example, the AUC for VS improves from an average of 0.88 to an average of 0.94 when combined with others over the course progression. Moreover, our results show that, although combining SLNs generally improves AUC performance, particularly in smaller or sparser datasets like VS and CP, the gains are less pronounced in larger classrooms such as AL and ML, where the SLN topological structure is already rich and adding more data to it may not affect its topological significance in a meaningful way. This suggests that the benefits of spatial aggregation are not solely due to having more data, but rather depend on the compatibility of interaction patterns across classrooms. These results support **Hypothesis 2** from Sec. 1, affirming and revealing the importance and timing of spatial aggregation.

4.4 Spatiotemporal Graph Analysis

Table 3: Hypothesis testing (one-sided t-test, $\alpha = 0.10$) comparing GNN isolated (IS) vs. GNN combined (CO) vs. CNN baseline (CNN) at 25% and 75% course progress. Each row compares spatial performance, while reading down the table reveals temporal trends across progress stages.

Models	Datase	Progress	p-value	Decision
IS vs CO	AL	25%	4.18e−1	Fail to Reject H_0
IS vs CO	AL	75%	8.03e−2	Reject H_0
CNN vs CO	AL	25%	6.02e−3	Reject H_0
CNN vs CO	AL	75%	1.25e−1	Fail to Reject H_0
IS vs CO	CP	25%	2.90e−2	Reject H_0
IS vs CO	CP	75%	1.42e−2	Reject H_0
CNN vs CO	CP	25%	5.75e−8	Reject H_0
CNN vs CO	CP	75%	1.23e−8	Reject H_0
IS vs CO	VS	25%	6.86e−4	Reject H_0
IS vs CO	VS	75%	8.00e−6	Reject H_0
CNN vs CO	VS	25%	8.13e−7	Reject H_0
CNN vs CO	VS	75%	3.7e−5	Reject H_0
IS vs CO	ML	25%	3.06e−1	Fail to Reject H_0
IS vs CO	ML	75%	6.14e−2	Reject H_0
CNN vs CO	ML	25%	5.77e−2	Reject H_0
CNN vs CO	ML	75%	2.15e−1	Fail to Reject H_0

In Secs. 4.2 and 4.3, we saw that (i) SLNs become structurally richer as a course progresses (temporal evolution) and (ii) combining multiple classroom graphs (spatial evolution) generally boosts AUC by exposing the model to a broader distribution of link patterns. However, these two dimensions do not operate independently. In particular, the benefit of combining SLNs (i.e., using spatial features of the graph) appears to depend on when in the course (temporally) the graphs are merged. Thus, in this section, we address **Hypothesis 3** from Sec. 1, which investigates whether merging SLNs at different stages of course progress leads to measurable gains in model performance.

As shown in Tab. 2, we see that combining SLNs earlier in the course leads to higher performance improvements compared to merging them at later stages. This is consistent with the intuition that early in the term, when individual SLNs are sparse, combining information across classrooms helps stabilize learning and improves performance. In contrast, in the later stages of a course, each SLN develops distinct structural characteristics, potentially reflecting subject matter, classroom dynamics, or group behavior and, as a result, attain lower performance gains upon being combined with other SLNs towards the end of the course.

Further, to highlight the superiority of our methodology, we compare our GNN-based framework to the current state-of-the-art framework for link prediction in SLNs, which consists of manually engineering fixed graph features and using them as input into a convolutional neural network (CNN) for link prediction [20], along with state-of-the-art approaches for link prediction in general OSNs, which consist of a sum pooling graph convolutional network (GCN) [10] and a Graph Attention Network (GAT) [25]. As can be seen from Tab. 2, each baseline struggles to attain consistently high performance across the spatiotemporal topology of the graph. The CNN in particular struggles due to its fixed feature engineering, whereas the GNN is able to offer high link prediction performance due to its dynamic adaptation to the SLN graph topology (via its node embeddings) and its evolution over time. This is highlighted in Tab. 2, where the GNN trained on a single SLN (GNN iso) generally outperforms the CNN baseline across all datasets and time instances. Furthermore, the GNN trained on combined SLNs (GNN all) not only outperforms the CNN but also significantly surpasses the isolated GNN as well as the other baselines in many cases, particularly in earlier stages of small courses. This demonstrates the dual benefit of our framework: first, to outperform traditional feature-based methods, and second, to exploit shared link formation patterns across SLNs to enhance predictive performance.

To further support the above observations, we perform formal hypothesis testing using a one-sided t test ($\alpha = 0.10$) comparing the AUC performance of our GNN model trained in combination (CO) with other courses against both the GNN trained in isolation (IS) on a single course and the CNN baseline (CNN). Here, we focus on the CNN baseline to demonstrate the statistical significance of using our framework's learned features via node embeddings in comparison to fixed feature engineering as currently done in state of the art link prediction frameworks [20]. The null hypothesis in each test states that CO does not outperform the baseline, while the alternative asserts that CO achieves a higher AUC. Tab. 3 summarizes the results in 25% and 75% course progress for the four SLNs. We show results at these two progress points to highlight the superiority of our method in an early and a late stage of the course. Here, we see that the improvement of our framework is statistically significant, particularly for small courses (e.g., CP and VS). Larger courses (e.g., AL and ML), on the other hand, contain sufficient training data that can be captured in varying time instances. This is largely due to the bias of the larger courses as their SLNs account for a larger partition of the spatially aggregated graph structure. Yet, we still observe statistically significant gains in AUC when comparing CO against both CNN and IS. In particular, the p-values are especially low for smaller and sparser datasets such as VS and CP, even at early progress stages. This further supports **Hypothesis 3**: the spatiotemporal consideration of SLNs generally yields stronger link prediction performance compared to the current state of the art, particularly in smaller courses (CP and VS) that suffer from an abundance of links early in the course.

5. CONCLUSION

In this work, we developed a link prediction framework in SLNs through the lens of two critical dimensions: temporal evolution and spatial aggregation. Using multiple SLNs at varying progress stages, we analyzed how SLNs develop over time and how combining multiple SLNs affects model performance. Our results show that model performance improves as the SLNs progress through the course timeline: richer interaction structures at later stages offer stronger signals for link prediction. We also found that merging SLNs from multiple classrooms can enhance model performance as spatial aggregation helped overcome data sparsity in smaller courses with lower numbers of links throughout the course progression. Finally, our analysis of the spatiotemporal

aspects revealed that the benefits of merging graphs are most pronounced early in the course, and particularly in smaller courses, which require more structure early on.

6. ACKNOWLEDGMENT

This work was supported in part by the U.S. National Science Foundation (NSF) under Grant No. SaTC-2513164, ECCS-2512911, ECCS-2543754, and DRL-2418658.

7. REFERENCES

[1] T. Babić, L. Kolar, and M. Miličević. Individual, cooperative and collaborative learning and students' perceptions of their impact on their own study performance. In *Proceedings of 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, pages 864–869, 2021.

[2] X. Chen, P. I. Lei, Y. Sheng, Y. Liu, and Z. Gong. Social influence learning for recommendation systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 312–322, 2024.

[3] F. P. A. de Medeiros and A. S. Gomes. An approach based on social network analysis to enhance social presence in a collaborative learning environment. *IEEE Transactions on Education*, 65(4):608–616, 2022.

[4] M. Dileo, M. Zignani, and S. Gaito. Temporal graph learning for dynamic link prediction with text in online social networks. *Machine learning*, 113(4):2207–2226, 2024.

[5] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin. Graph neural networks for social recommendation. In *Proceedings of The World Wide Web Conference*, pages 417–426, 2019.

[6] W. L. Hamilton, R. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1025–1035, 2017.

[7] A. P. Hridi, M. Hoq, Z. Gao, C. Lynch, R. Sahay, S. Hosseinalipour, and B. Akram. Privacy-preserving distributed link predictions among peers in online classrooms using federated learning. In *Conference on Educational Data Mining (EDM)*, 2025.

[8] M. M. Keikha, M. Rahgozar, and M. Asadpour. Deeplink: A novel link prediction framework based on deep learning. *Journal of Information Science*, 47(5):642–657, 2021.

[9] S. Khoshraftar and A. An. A survey on graph representation learning methods. *ACM Transactions on Intelligent Systems and Technology*, 15(1):1–55, 2024.

[10] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations*, 2017.

[11] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[12] A. Krouska, C. Troussas, and M. Virvou. Social networks as a learning environment: Developed applications and comparative analysis. In *Proceedings of 8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–6, 2017.

[13] M. Kumar, S. Mishra, S. S. Singh, and B. Biswas. Community-enhanced link prediction in dynamic networks. *ACM Transactions on the Web*, 18(2):1–32, 2024.

[14] J. Li, H. Shomer, H. Mao, S. Zeng, Y. Ma, N. Shah, J. Tang, and D. Yin. Evaluating graph neural networks for link prediction: Current pitfalls and new benchmarking. *Advances in Neural Information Processing Systems*, 36:3853–3866, 2023.

[15] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, pages 556–559, 2003.

[16] F. Mohamady and S. Dami. Link prediction in dynamic social networks using deep learning. In *2024 20th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP)*, pages 1–5. IEEE, 2024.

[17] H. Nassar, A. R. Benson, and D. F. Gleich. Pairwise link prediction. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 386–393, 2020.

[18] A. Ninagawa and K. Eguchi. Link prediction using probabilistic group models of network structure. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1115–1116, 2010.

[19] V. Raghavan, G. Ver Steeg, A. Galstyan, and A. G. Tartakovsky. Modeling temporal activity patterns in dynamic social networks. *IEEE Transactions on Computational Social Systems*, 1(1):89–107, 2014.

[20] R. Sahay, S. Nicoll, M. Zhang, T.-Y. Yang, C. Joe-Wong, K. A. Douglas, and C. G. Brinton. Predicting learning interactions in social learning networks: A deep learning enabled approach. *IEEE/ACM Transactions on Networking*, 31(5):2086–2100, 2023.

[21] A. Sankar, Y. Liu, J. Yu, and N. Shah. Graph neural networks for friend ranking in large-scale social platforms. In *Proceedings of the Web Conference 2021*, pages 2535–2546, 2021.

[22] M. Saqr and C. S. Montero. Learning and social networks - similarities, differences and impact. In *Proceedings of IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, pages 135–139, 2020.

[23] K. Sharma, Y.-C. Lee, S. Nambi, A. Salian, S. Shah, S.-W. Kim, and S. Kumar. A survey of graph neural networks for social recommender systems. *ACM Computing Surveys*, 56(10):1–34, 2024.

[24] J. Sosa, D. Martínez, and N. Guerrero. An unified approach to link prediction in collaboration networks. *arXiv preprint arXiv:2411.01066*, 2024.

[25] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph attention networks. In *Proceedings of the International Conference on Learning Representations*, 2018.

[26] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 322–331, 2007.

[27] W. Wei and K. M. Carley. Measuring temporal patterns

in dynamic social networks. *ACM Transactions on Knowledge Discovery from Data*, 10(1):1–27, 2015.

[28] S. Xu, C. Yang, C. Shi, Y. Fang, Y. Guo, T. Yang, L. Zhang, and M. Hu. Topic-aware heterogeneous graph neural network for link prediction. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 2261–2270, 2021.

[29] Z. Xu, H. Yuan, and Q. Liu. Student performance prediction based on blended learning. *IEEE Transactions on Education*, 64(1):66–73, 2021.

[30] T.-Y. Yang, C. G. Brinton, and C. Joe-Wong. Predicting learner interactions in social learning networks. In *Proceedings of IEEE Conference on Computer Communications*, pages 1322–1330, 2018.

[31] M. Zhang and Y. Chen. Weisfeiler-lehman neural machine for link prediction. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 575–583, 2017.

[32] M. Zhang and Y. Chen. Link prediction based on graph neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 5171–5181, 2018.

[33] J. Zhu. Max-margin nonparametric latent feature models for link prediction. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1179–1186, 2012.

TS-Interact: A Multimodal Dataset for Characterizing Classroom Teacher- and Student-Led Instructional Behaviors

Weigang Lu
Department of Education
Ocean University of China
luweigang@ouc.edu.cn

Yaoxuan Peng
Department of Education
Ocean University of China
pengyx@stu.ouc.edu.cn

Saisai Ye
Department of Education
Ocean University of China
yesaisai@stu.ouc.edu.cn

Cunling Bian*
Department of Education
Ocean University of China
clbian@ouc.edu.cn

ABSTRACT

Classroom instructional behavior analysis seeks to model teacher- and student-led behaviors to support instructional quality assessment, teaching feedback, and AI-assisted classroom analytics. With the rapid progress of Large Language Models (LLMs) and multimodal learning, reliable analysis increasingly depends on well-structured multimodal datasets with pedagogically grounded behavior definitions. However, existing classroom datasets are often limited to single modalities or coarse, task-agnostic behavior categories, failing to capture fine-grained teacher–student instructional dynamics and limiting model interpretability and generalizability. To address these issues, we introduce TS-Interact (Teacher–Student Interaction Dataset), a multimodal dataset for characterizing classroom teacher- and student-led instructional behaviors. TS-Interact integrates synchronized video, audio, and text from real classroom scenarios, comprising approximately 1,000 instructional segments across multiple core subjects. Built upon an S–T interaction analysis framework, instructional behaviors are systematically decomposed into teacher-led and student-led categories, each further organized into four pedagogically motivated behavior types. All segments are manually annotated, with annotation reliability validated using the Kappa coefficient. Benchmark experiments with representative multimodal behavior recognition models demonstrate that TS-Interact achieves higher annotation consistency and improved recognition accuracy than commonly used generic classroom datasets. Overall, TS-Interact provides a high-quality multimodal benchmark with theoretically grounded behavior definitions, enabling precise and interpretable LLM-based classroom behavior analysis and supporting future research in intelligent teaching evaluation.

Keywords

Classroom Instructional Behavior Analysis, Multimodal Educational Dataset, Teacher–Student Interaction, Pedagogically Grounded Annotation, LLM-based Classroom Analytics

1. INTRODUCTION

Classroom teaching behavior analysis, which involves the quantitative modeling and qualitative characterization of teacher-initiated and student-initiated behaviors during instructional activities, serves as a cornerstone for advancing modern educational practices[11]. Its application value spans three core domains: in teaching evaluation, it provides objective, data-driven evidence to complement subjective observation scales; in teaching feedback, it enables targeted improvement suggestions for teachers by identi-fying interaction bottlenecks in classroom processes; in intelligent education systems, it underpins the development of adaptive teaching modules and automated classroom monitoring tools[12]. With the rapid advancement of large language models (LLMs) and multimodal learning technologies, the integration of cross-modal information (video, audio, text) has revolutionized the way class-room behaviors are analyzed—LLMs, in particular, empower the mining of implicit semantic relationships behind behavioral sig-nals, while multimodal fusion enhances the comprehensiveness of behavior recognition[10]. This technological evolution has put forward new demands for classroom analysis research: the need for high-quality multimodal datasets with pedagogically grounded behavior definitions, which can bridge the gap between technical model training and practical educational scenarios.

However, a critical review of existing classroom datasets reveals three prominent limitations that hinder the advancement of LLM-driven multimodal classroom behavior analysis. First, in terms of modality coverage, most datasets prioritize single-modal signals (e.g., learning management system logs or video-only recordings), ignoring the comple-

*Corresponding author

Weigang Lu, Yaoxuan Peng, Saisai Ye, and Cunling Bian. TS-Interact: A Multimodal Dataset for Characterizing Classroom Teacher- and Student-Led Instructional Behaviors. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 549–555. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039742

mentary value of synchronized audio and text information in capturing complex interaction dynamics such as teacher-student dialogue and experimental operation sounds[7]. Second, behavior definition frameworks are often coarse-grained and task-irrelevant. Many datasets categorize classroom behaviors into overly general types (e.g., "interaction" and "non-interaction"), failing to distinguish subject-specific behavioral characteristics such as science experiment operations and art creation practices[13]. Third, there is a lack of solid pedagogical theoretical support for behavior classification systems. Existing datasets rarely integrate classic educational theories such as student-teacher (S-T) interaction analysis, resulting in behavior taxonomies that are disconnected from teaching practice and reducing the interpretability of downstream analysis models[9]. These limitations collectively restrict the performance of multimodal and LLM-based analysis models in real classroom scenarios, leading to a mismatch between technical solutions and educational application needs.

To address the aforementioned gaps, this study introduces TS-Interact, a multimodal teacher-student interaction dataset dedicated to classroom teaching behavior analysis. The core contributions of this study are threefold. First, we construct TS-Interact, a high-quality multimodal dataset comprising approximately 1,000 teaching segments across seven core middle school subjects, which integrates synchronized video, audio, and text data collected from authentic classroom environments. Second, based on the classic S-T interaction analysis framework, we design a dual-dominant teaching behavior classification system, which divides classroom behaviors into teacher-dominant and student-dominant categories, with each category further refined into four pedagogically moti-vated subtypes, ensuring both fine-grained modeling and semantic interpretability. Third, we conduct comprehensive experiments to verify the dataset's effectiveness: we quantify annotation consistency using Kappa coefficients to ensure data quality, and per-form benchmark tests with representative multimodal behavior recognition models to demonstrate that TS-Interact outperforms general-purpose datasets in behavior recognition accuracy.

2. RELATED WORK

Classroom Instructional Behavior Analysis. Classroom instructional behavior analysis is a central topic in educational technology, aiming to model and interpret the dynamic interactions between teachers and students to support instructional evaluation and intelligent educational systems. Existing studies largely rely on single-modal data sources—video, audio, or text—enabled by advances in computer vision, speech processing, and natural language processing. Video-based approaches leverage visual cues such as posture, gestures, and movement patterns to recognize classroom behaviors[6]. While automatic recognition models (e.g., CNNs and pose-based methods) have reduced manual annotation costs, most studies adopt coarse-grained behavior categories and often fail to distinguish teacher and student roles, limiting interpretability of interaction dynamics. Audio-based methods focus on speech features and dialogue structures to analyze verbal interaction and emotional states[4]. However, these approaches are highly sensitive to noise and typically rely on shallow behavior definitions (e.g., speaking vs. silence), overlooking pedagogically meaningful interaction types. Text-based studies exploit transcribed classroom discourse and learning logs to analyze teaching content and participation quality [3]. Although semantic modeling offers interpretability advantages, such approaches are constrained by limited data scale and the lack of integration with visual and auditory cues. Overall, prior work exhibits three common limitations: coarse behavioral granularity, ambiguous subject distinction between teachers and students, and insufficient multimodal integration. These issues restrict the applicability of existing models for fine-grained, theory-aligned instructional behavior analysis.

Classroom Instructional Behavior Datasets. Several datasets have been proposed to support classroom behavior analysis; however, notable limitations remain in modality coverage, annotation granularity, and theoretical grounding. Existing datasets either focus on engagement estimation or general instructional actions, with limited representation of authentic classroom interaction dynamics[5]. Most available datasets employ single or weakly integrated modalities[8], preventing effective modeling of cross-modal behavioral cues. In addition, behavior labels are typically broad (e.g., "engaged" or "active"), lacking alignment with pedagogical constructs and obscuring distinctions between different instructional intents. More critically, many datasets are not grounded in established educational theories, resulting in annotations that reflect surface-level observations rather than interpretable instructional behaviors[1]. These limitations highlight the need for a multimodal classroom dataset that integrates synchronized video, audio, and text, adopts fine-grained and role-aware behavior definitions, and is explicitly informed by instructional theory. Addressing these gaps forms the motivation for the TS-Interact dataset proposed in this work.

3. METHODOLOGY

To provide a clear and systematic understanding of how TS-Interact is designed, constructed, and validated, this section presents the methodological framework of the proposed dataset from theory to practice. As illustrated in Fig.1, our method follows a structured pipeline that starts from a pedagogically grounded instructional behavior modeling framework, proceeds through multimodal data collection, preprocessing, and rigorous annotation, and culminates in dataset validation via representative multimodal baseline models. This end-to-end design ensures that the dataset is not only empirically reliable but also theoretically interpretable, enabling fine-grained characterization of teacher–student instructional behaviors. In the following subsections, we first introduce the theoretical foundation underlying the instructional behavior taxonomy, then describe the dataset construction and annotation process in detail, and finally report validation experiments that demonstrate the effectiveness and adaptability of TS-Interact for multimodal classroom behavior analysis

3.1 Theoretical Foundation of Instructional Behavior Modeling

TS-Interact is built based on the Teacher-Student (S-T) interaction framework, which originates from the classic classroom interaction theory [2] and has been modernized in recent multimodal educational research. The framework

Table 1: Classroom Teaching Behavior Categories

Major Category	Subcategory	Category
Teacher-led Teaching Behavior	Knowledge Presentation	Explain and state; Summarize the lecture; Evaluation summary; Demonstrate and explain
	Guiding Questions	Raise a question; Assign tasks; Follow up with a counter-question
	Resource Display	Physical display; Experimental demonstration; Text display; Picture display; Data display; Blackboard writing display
	Process Management	Activity guidance; Communication guidance; Order management; Emotional motivation; Performance evaluation
Student-led Teaching Behavior	Independent Exploration	Independent thinking; Information query
	Oral Expression	Student read aloud; Speak and answer together; Student's direct answers
	Operational Practice	Start writing by hand; Hands-on operation
	Cooperation and Exchange	Group discussion; Report conversation; Work display

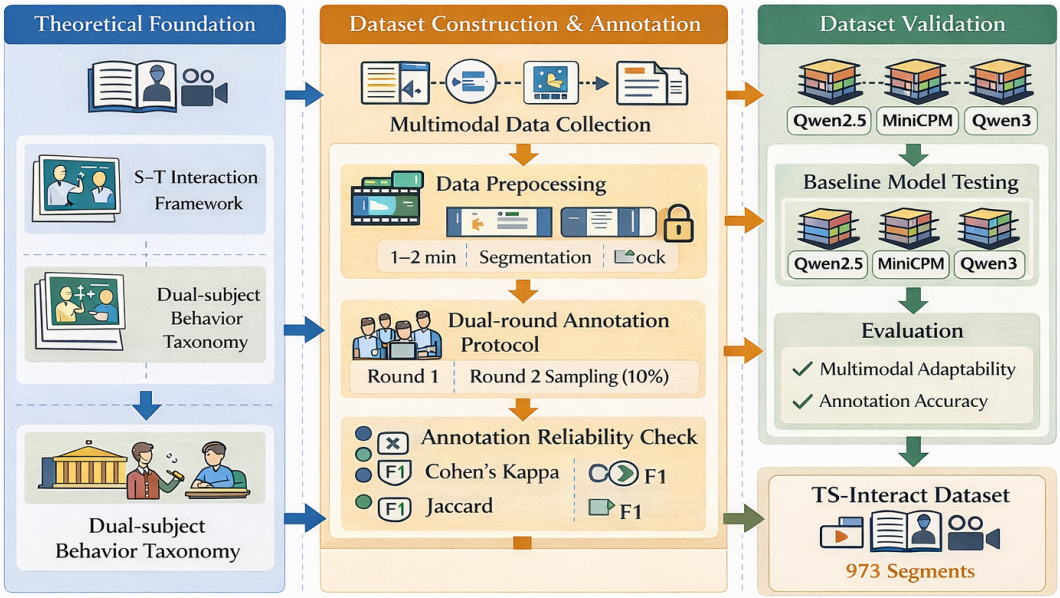


Figure 1: The workflow for constructing the dataset of T-S interaction

clearly defines teacher-led and student-led teaching behaviors, providing a rigorous theoretical foundation for multimodal observation and teaching semantic alignment. Both the behavior classification system and the three-layer annotation scheme in this paper are strictly designed based on this theory. The instructional behavior annotation system constitutes the theoretical foundation of the TS-Interact dataset, enabling structured and interpretable modeling of classroom teaching and learning processes. Existing classroom behavior annotation schemes often adopt generalized, subject-agnostic categories, which fail to capture the core logic of teacher–student interaction and limit their applicability to multimodal analysis. To address this issue, TS-Interact grounds its annotation design in an interaction-oriented pedagogical framework, ensuring both theoretical validity and annotation feasibility.

Student–Teacher Interaction Framework. TS-Interact adopts the Student–Teacher (S–T) interaction analysis framework as its theoretical basis, viewing classroom instruction as a dynamic process formed through continuous interaction between teachers and students. In this framework, teachers primarily guide and regulate instructional processes, while students actively participate in learning activities. Since instructional effectiveness emerges from this bidirectional interaction, instructional behaviors must be explicitly associated with their initiating subjects and pedagogical functions. This subject-aware design provides a clear theoretical foundation for aligning multimodal observations with instructional semantics.

Dual-subject Instructional Behavior Taxonomy. Based on the S–T interaction framework, TS-Interact employs a dual-subject instructional behavior taxonomy that distinguishes teacher-led and student-led behaviors, as shown in Table 1. Teacher-led behaviors describe instructional actions such as

content presentation, task guidance, resource support, and process management, while student-led behaviors capture learning activities including exploration, expression, operation, and collaboration. This hierarchical taxonomy ensures clear behavioral boundaries, reduces annotation ambiguity, and aligns each behavior category with observable multimodal signals from video, audio, and text. As a result, the proposed taxonomy supports reliable annotation, interpretable modeling, and effective multimodal behavior recognition.

3.2 Dataset Construction and Annotation

Multimodal Classroom Data Collection. To ensure the authenticity, representativeness, and disciplinary diversity of the dataset, all classroom videos used in this study were collected from the National Smart Education Platform for Primary and Secondary Schools and several provincial-level smart education platforms. All recorded lessons available on these platforms had been officially reviewed and approved by educational administrative authorities, ensuring high instructional quality and authoritative reliability. The collected videos represent real-world daily teaching scenarios rather than artificially designed experimental classrooms, with clear and stable audio–visual signals. A total of 41 complete classroom recordings were initially selected, covering seven academic subjects: physics, art, mathematics, chemistry, Chinese, history, and biology. This subject diversity enables the dataset to reflect varied instructional styles and interaction patterns across science, humanities, and arts disciplines, supporting broad applicability in multimodal classroom behavior analysis.

Data Preprocessing and Segmentation. To accurately capture meaningful instructional behavior units and meet the requirements of fine-grained behavior recognition, all collected classroom videos were segmented by trained personnel with an educational technology background. Using professional video editing tools, each full-length recording was divided into short clips with an average duration of 1–2 minutes. During segmentation, special care was taken to avoid splitting a coherent teacher–student interaction across multiple segments, ensuring that each clip contains a complete behavioral unit or a continuous interaction sequence. Segments with blurred visuals, severe background noise, missing instructional content, or disrupted audio–visual signals were excluded. For science experiment lessons, clips with unclear experimental procedures or missing key operational steps were further removed. All original videos had already undergone privacy protection procedures on the hosting platforms, and additional audio de-sensitization was applied to strictly comply with educational data ethics and privacy regulations. Classroom automated monitoring is now widely used in school education and has been generally accepted by teachers and students. Therefore, the potential negative impact of automated monitoring on teacher-student teaching interaction is negligible.

Dual-round Annotation Protocol. To enhance annotation reliability, a dual-round independent annotation protocol was adopted. The annotation team consisted of four annotators holding master's degrees in educational technology, all of whom had solid theoretical training and experience in multimodal data analysis. Prior to formal annotation, the annotators participated in a one-day centralized training session, which included theoretical instruction on the annotation taxonomy, detailed explanation of category definitions and decision rules, and hands-on practice with annotated and unannotated samples. In the first round, all annotators independently labeled the full dataset in isolated environments, with no communication permitted. One month later, to mitigate memory bias, a second-round annotation was conducted on a randomly sampled subset of 10% of the data (100 segments). The sampled segments preserved the original subject distribution and were randomly assigned to annotators such that no annotator re-labeled their own previous samples. All annotations in the second round were completed strictly according to the same guidelines, ensuring independence between rounds.

Annotation Reliability and Quality Control. To verify the consistency and reliability of the multi-label annotations, a two-dimensional evaluation framework was employed, addressing both label agreement and label set similarity. At the label level, Cohen's Kappa coefficient was used to assess whether individual behavior labels were consistently assigned. Following standard practice, values above 0.8 indicate almost perfect agreement, while values between 0.6 and 0.8 indicate high agreement. At the set level, the Jaccard coefficient and set-level F1 score were adopted to evaluate the overlap and balance of the complete label sets assigned to each segment, accommodating variations in the number of labels per sample. All annotation results were organized into a structured dataset, converted into binary label representations, and evaluated using Python-based scripts. The results demonstrate high consistency across all metrics, confirming the reliability of the annotation process.

Dataset Statistics and Content. As shown in Table 2, TS-Interact consists of 973 multimodal classroom segments, each with an average duration of 1–2 minutes, enabling fine-grained quantitative analysis of instructional behaviors at the interaction unit level. The dataset covers seven academic subjects, with physics (25.4%) and art (21.3%) forming the largest portions, followed by mathematics and chemistry (both 14.3%), ensuring balanced cross-disciplinary representation across science, humanities, and art domains. From a behavioral perspective, teacher-led behaviors dominate the dataset, with guiding question behaviors occurring most frequently (1,155 instances), followed by process management (539), knowledge presentation (538), and resource display (428). Student-led behaviors exhibit a complementary but less frequent distribution, where oral expression (690) constitutes the primary form of learner participation, while independent exploration (158), practical operation (187), and collaborative communication (16) reflect varying levels of learner autonomy and interaction intensity. Annotation quality is quantitatively supported by high inter-annotator agreement. Label-level consistency achieves an average Cohen's κ of 0.779 and a micro-average κ of 0.808, while set-level evaluation yields a Jaccard coefficient of 0.787 and an average set-level F1 score of 0.834. These quantitative indicators demonstrate that TS-Interact provides statistically reliable, behaviorally diverse, and well-balanced data support for multimodal instructional behavior analysis.

Fig.2 provides representative visual samples from TS-Interact,

Table 2: Core Dataset Statistics of TS-Interact

Dimension	Attribute	Description / Statistics
Data Scale	Total teaching segments	973 multimodal classroom segments
	Average segment duration	1–2 minutes per segment
	Data source	National and provincial smart education platforms
	Classroom setting	Authentic middle school classrooms (non-experimental)
	Privacy protection	Platform anonymization and audio de-sensitization
Subject Coverage	Subject distribution	Physics (247), Art (207), Mathematics (139), Chemistry (139), Chinese (91), History (87), Biology (63)
Behavior Distribution	Teacher-led behaviors	Knowledge presentation (538), guiding questions (1155), resource display (428), process management (539)
	Student-led behaviors	Independent exploration (158), oral expression (690), practical operation (187), collaborative communication (16)
Annotation Quality	Annotators	4 trained annotators with educational technology background
	Annotation protocol	Dual-round independent annotation with a one-month interval
	Label-level consistency	Average Cohen's Kappa Coefficient:0.779, Micro-average Cohen's Kappa Coefficient:0.808
	Set-level consistency	Average Jaccard Coefficient:0.787, Average Set-level F1 Score:0.834, Average Difference in Number of Labels:0.660

directly demonstrating the authenticity and high quality of the dataset. All samples are extracted from real, non-experimental middle school classrooms, preserving natural instructional rhythms, teacher–student interactions, and classroom layouts without staged or scripted behaviors. The visual examples show clear correspondence between annotated behavior labels and observable multimodal cues, such as teacher gestures, student postures, gaze directions, and interactions with instructional resources. This alignment indicates that the annotations are grounded in concrete visual and contextual evidence rather than abstract or inferred descriptions. Moreover, the diversity of scenes, subjects, and interaction patterns visible in the samples reflects the broad subject coverage and behavioral variability reported in Table 2. Together, these image-based samples substantiate the quantitative statistics by providing intuitive evidence of data realism, annotation reliability, and semantic clarity, reinforcing TS-Interact as a high-quality benchmark for multimodal instructional behavior analysis. To facilitate transparency and reproducibility, samples from the TS-Interact dataset has been made publicly available. Interested researchers can access these examples via the following link: https://osf.io/5q7kj/overview?view_only=6847956cd787410aa72f9cd407a0da91

3.3 Dataset Validation

Baseline Models for Multimodal Behavior Recognition. To evaluate the adaptability and effectiveness of the TS-Interact dataset for fine-grained classroom instructional behavior recognition, three representative multimodal baseline models were selected. All chosen models are lightweight, open-source, and easy to reproduce, with strong multimodal fusion capabilities, making them suitable as general benchmarks for assessing dataset quality rather than pursuing model-level state-of-the-art performance. The Qwen-2.5-Omni-3B (Qwen-2.5) model was selected due to its efficient multimodal fusion design and moderate parameter scale, which allows effective processing of video, audio, and textual inputs. It adopts a three-stage architecture consisting of single-modality feature extraction, cross-modal fusion via attention mechanisms, and behavior classification through fully connected layers, making it well aligned with the multimodal characteristics of TS-Interact. MiniCPM-V-4.5 (MiniCPM-V) emphasizes enhanced visual–language alignment and logical reasoning capabilities, enabling it to capture fine-grained instructional interaction patterns such as teacher–student question–answer dynamics. Its strong generalization ability makes it suitable for evaluating the discriminability and pedagogical rationality of the proposed annotation system. Qwen3-VL-4B-Thinking (Qwen3) adopts a dual-branch fusion architecture, where the visual branch extracts spatiotemporal video features and the textual branch models semantic information from transcribed classroom discourse, with audio features mapped into the textual space. Adaptive fusion between the two branches enables robust behavior recognition under limited data conditions, making this model particularly appropriate for validating dataset representativeness. Together, these three models cover diverse multimodal fusion strategies and provide a comprehensive evaluation of the dataset's adaptability, annotation quality, and practical usability.

Experimental Setup. The TS-Interact dataset was split into training and test sets using subject-stratified sampling with an 80%/20% ratio to maintain disciplinary balance. All splits were performed at the segment level to prevent data leakage. Classification performance was evaluated using Accuracy, Precision, Recall, and Macro-F1 to reflect both overall recognition and robustness to imbalanced behavior categories. Generation quality was assessed with ROUGE-1/2/L and BLEU-4, measuring semantic overlap and content completeness between generated descriptions and gold annotations. All baseline models were evaluated under identical settings within the LlamaFactory framework (max input length 2048, batch size 4, max generation length 384, temperature 0.01, nucleus sampling 1.0), with a fixed random seed (42) to ensure reproducibility.

Evaluation Results and Analysis. Table 3 reports the quanti-

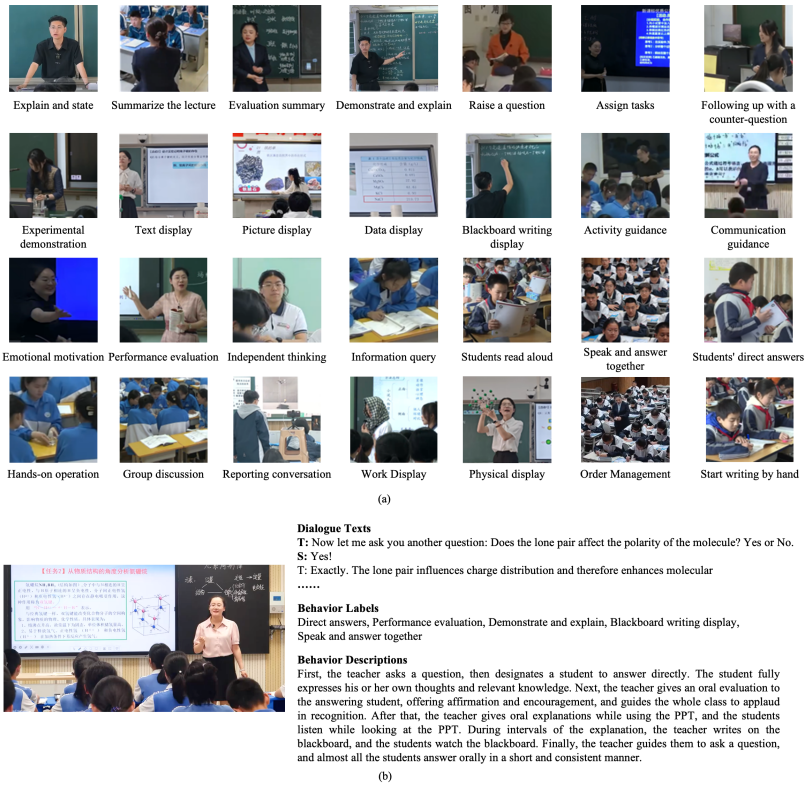


Figure 2: (a) Examples of Teacher-and-Student-led instructional behaviors in TS-interact (single-frame screenshot from a multimodal segment) (b) Annotated segment example in TS-Interact.

Table 3: Average Performance of Baseline Models on TS-Interact Across Seven Subjects

Model	F1	Accuracy	Precision	Recall	ROUGE-1	ROUGE-2	ROUGE-L	BLEU-4
Qwen3	**0.6582**	**0.1482**	**0.6525**	**0.6625**	**0.3572**	**0.1641**	**0.3330**	**0.0913**
Qwen-2.5	0.6275	0.1005	0.6208	0.6345	0.3315	0.1455	0.3068	0.0781
MiniCPM-V	0.5788	0.0615	0.5704	0.5860	0.2974	0.1239	0.2753	0.0647

tative results averaged over seven subjects. Qwen3 achieves the strongest overall performance, obtaining a Macro-F1 of 0.6582 and consistently outperforming other baselines across all generation metrics (e.g., ROUGE-L 0.3330, BLEU-4 0.0913). Compared with MiniCPM-V, Qwen3 yields a relative improvement of 13.7% in Macro-F1 and 21.8% in ROUGE-1, indicating superior alignment between multimodal inputs and fine-grained instructional behavior labels. Qwen-2.5 shows competitive results with reduced model size, suggesting a favorable trade-off between efficiency and accuracy. The consistent ranking of models across both classification and generation metrics demonstrates strong metric agreement and confirms the internal consistency of the evaluation protocol. Overall, the clear performance gaps among baselines highlight the discriminative power of TS-Interact and validate its suitability as a high-quality benchmark for multimodal classroom behavior recognition and instructional modeling.

4. DISCUSSION

This study presents TS-Interact as a multimodal classroom interaction dataset grounded in a structured and pedagogically informed teaching behavior taxonomy. From the perspective of the overall research design, several important insights emerge.

First, the results demonstrate that incorporating multimodal information is essential for fine-grained instructional behavior understanding. Across all baseline models, performance gains are consistently observed when visual, auditory, and textual cues are jointly leveraged, indicating that classroom teaching behaviors cannot be reliably inferred from a single modality alone. This finding aligns with educational theories that emphasize the embodied and situational nature of teaching, where gestures, visual materials, and classroom dynamics play a critical role alongside spoken language.

Second, the proposed teaching behavior annotation system contributes to more scientifically grounded behavior modeling. By explicitly distinguishing teacher-led and student-led behaviors and further decomposing them into pedagogically meaningful subcategories, TS-Interact avoids overly coarse

or purely action-level labels. This hierarchical structure enables models to capture both instructional intent and interaction patterns, which is reflected in improved classification stability and more coherent generation outputs. Compared with prior classroom datasets that rely on flat or loosely defined labels, the TS-Interact taxonomy provides clearer semantic boundaries and stronger interpretability.

Third, the performance gap among different multimodal large language models highlights both the promise and current limitations of general-purpose MLLMs in educational scenarios. While advanced models achieve reasonable results on behavior recognition and description, their accuracy remains relatively low, suggesting that classroom interaction understanding poses challenges beyond conventional vision-language tasks. This indicates opportunities for future work on education-specific adaptation, multimodal alignment, and reasoning mechanisms tailored to instructional contexts.

Finally, this study also has limitations. The dataset focuses on structured classroom settings and predefined academic subjects, which may limit direct generalization to informal or online learning environments. In addition, although the annotation framework is designed to be extensible, further validation across diverse educational cultures and instructional styles is still needed. Addressing these limitations will be an important direction for future research.

5. CONCLUSION

In this paper, we introduced TS-Interact, a multimodal classroom interaction dataset designed to support fine-grained analysis of teaching and learning behaviors. By integrating video, audio, and textual information with a pedagogically grounded annotation system, TS-Interact provides a more comprehensive and interpretable representation of classroom instructional dynamics. We conducted extensive experiments using state-of-the-art multimodal large language models, evaluating both classification and generation tasks. The results demonstrate that multimodal information is critical for understanding instructional behaviors and that the proposed teaching behavior taxonomy enables more structured and meaningful modeling of classroom interactions. Overall, TS-Interact offers a valuable benchmark for advancing multimodal educational AI research, particularly in areas such as classroom behavior understanding, instructional analysis, and intelligent tutoring systems. We hope this dataset will facilitate future studies on interpretable, scalable, and pedagogically informed multimodal learning models, and contribute to the development of AI systems that better support teaching and learning in real-world classrooms.

In future work, we plan to expand TS-Interact to cover a broader range of disciplines, extending from seven to nine subjects while maintaining a balanced distribution across categories. In addition, to complement the current validation based on lightweight multimodal models, we will incorporate larger-scale multimodal models to further examine the dataset's scalability and robustness. Beyond dataset expansion and validation, we aim to explore interdisciplinary instructional behavior analysis, dataset-driven model optimization, and the development of ethically grounded AI applications for educational contexts.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (No. 62407042, 62277 045), Humanity and Social Science Foundation of Ministry of Education (No. 24YJC880004), and Natural Science Foundation of Shandong Province (No. ZR2024QF075)

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

6. REFERENCES

[1] B. Cunling, L. Weigang, and F. Wei. Skeleton-based human action recognition: History, status and prospects. *Journal of Computer Engineering & Applications*, 60(20):1–29, 2024.

[2] N. A. Flanders. Analyzing teaching behavior. 1970.

[3] D. K. Geeganage, Y. Xu, and Y. Li. A semantics-enhanced topic modelling technique: semantic-lda. *ACM Transactions on Knowledge Discovery from Data*, 18(4):1–27, 2024.

[4] A. Geetha, T. Mala, D. Priyanka, and E. Uma. Multimodal emotion recognition with deep learning: advancements, challenges, and future directions. *Information Fusion*, 105:102218, 2024.

[5] C. Huang, J. Zhu, Y. Ji, W. Shi, M. Yang, H. Guo, J. Ling, P. De Meo, Z. Li, and Z. Chen. A multi-modal dataset for teacher behavior analysis in offline classrooms. *Scientific Data*, 12(1):1115, 2025.

[6] J. Huang, H. Hashim, H. Norman, M. H. Zaini, and X. Zhang. Automatic detection of teacher behavior in classroom videos using alphapose and faster r-cnn algorithms. *PeerJ Computer Science*, 11:e2933, 2025.

[7] A. Kapoor and R. W. Picard. Multimodal affect recognition in learning environments. In *Proceedings of the annual ACM international conference on Multimedia*, pages 677–682, 2005.

[8] Q. Liu, X. Jiang, and R. Jiang. Classroom behavior recognition using computer vision: A systematic review. *Sensors*, 25(2):373, 2025.

[9] X. Ma, Y. Xie, X. Yang, H. Wang, Z. Li, and J. Lu. Teacher-student interaction modes in smart classroom based on lag sequential analysis. *Education and Information Technologies*, 29(12):15087–15111, 2024.

[10] K. Mangaroska, R. Martinez-Maldonado, B. Vesin, and D. Gašević. Challenges and opportunities of multimodal data in human learning: The computer science students' perspective. *Journal of Computer Assisted Learning*, 37(4):1030–1047, 2021.

[11] C. Romero and S. Ventura. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics*, 40(6):601–618, 2010.

[12] C. Romero and S. Ventura. Educational data mining and learning analytics: An updated survey. *Wiley interdisciplinary reviews: Data mining and knowledge discovery*, 10(3):e1355, 2020.

[13] J. Xiao, M. Chen, Y. Yang, and M. Liu. An exploratory multimodal study of the roles of teacher-student interaction and emotion in academic performance in online classrooms. *Education and Information Technologies*, 30(11):15507–15527, 2025.

Teaching Language Models How to Code Like Learners: Conversational Serialization for Student Simulation

Charles Koutcheme
Aalto University
Espoo, Finland
charles.koutcheme@aalto.fi

Juho Leinonen
Aalto University
Espoo, Finland
juho.2.leinonen@aalto.fi

Arto Hellas
Aalto University
Espoo, Finland
arto.hellas@aalto.fi

ABSTRACT

Artificial students—models that simulate how learners act and respond within educational systems—are a promising tool for evaluating tutoring strategies and feedback mechanisms at scale. However, most existing approaches rely on prompting large, proprietary language models, limiting adaptability to specific courses and raising concerns around privacy, cost, and dependence. In this work, we propose a framework for training open-weight artificial programming learners directly from authentic student process data. Our approach serializes temporal log traces into a conversational format, representing each student's problem-solving process as a dialogue between the learner and their automated assessment system. Student code submissions and environment feedback, such as test outcomes, grades, and error traces, form alternating conversational turns, enabling models to learn from the iterative debugging process. We additionally introduce a training pipeline combining supervised fine-tuning with preference optimization to align models with authentic student debugging behavior. We evaluate our framework by training Qwen models at 4B and 8B scales on a large-scale dataset of real student submissions to Python programming assignments. Our results show that incorporating environment feedback strengthens models' ability to replicate student debugging behavior, improving over both prior code-only approaches and prompted large language models baselines in functional alignment and code similarity. We release our code to support reproducibility.

Keywords

generative AI, language models, simulated students, process data, preference optimization

1. INTRODUCTION

To better support learners at scale, computing education research has long relied on models of students built from the rich log data collected as they solve programming assignments [17, 46]. Much of this work focuses on capturing what students know, such as estimating mastery over concepts through knowledge tracing [7, 18, 44], or identifying misconceptions that explain incorrect outcomes [40]. These models provide valuable insights into learning progress and have informed the design of successful interventions [3]. However, while such approaches are effective for analyzing student cognition and predicting outcomes, operationalizing interventions based on these models remains challenging. In many settings, what is needed is not only an estimate of what a student knows, but a model of how a student behaves: how they act, react, and progress when interacting with learning systems. This has motivated growing interest in *artificial students*—models that act as stand-ins for real learners by generating plausible sequences of actions [43, 26, 5]. When realistic, artificial students could enable the simulation of student–system interactions and support controlled experimentation, such as comparing tutoring strategies, evaluating feedback policies, or assessing the difficulty of programming tasks, as already done in prior works [32, 12, 29], without requiring repeated classroom deployments. Despite this promise, existing artificial student models remain limited, particularly in programming education. Recent work using large language models has shown that it is possible to generate student-like code [23, 25], but these approaches often rely on prompting alone and tend to reproduce surface-level error patterns rather than the dynamic nature of real student debugging. In particular, existing methods remain limited in their ability to realistically simulate how learners tackle programming assignments.

In this work, we leverage the rich process data already collected by programming learning environments to train small and medium size open-weight models for *within-assignment student simulation* [27, 35]. Since student data is sensitive and institutional, this offers a more practical path than prompting proprietary systems [35], keeping data local while learning the iterative debugging patterns that prompting alone cannot reliably elicit. Our core innovation is to represent each problem-solving trajectory as a dialogue between a student and their learning environment. Student code submissions are encoded as assistant turns, while outputs from automated assessment systems (e.g., unit-test outcomes, or grader messages) form the corresponding user turns. By adopting the conversational format commonly used to interact with language models, we turn raw log traces into structured sequences that capture how a student's code evolves in response to feedback. This representation enables chat-capable language models to learn from iterative debugging sequences without requiring architectural modifications.

Charles Koutcheme, Juho Leinonen, and Arto Hellas. Teaching Language Models How to Code Like Learners: Conversational Serialization for Student Simulation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 556–563. International Educational Data Mining Society (2026).
© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039778

To train artificial students from this conversational representation, we propose a pipeline combining supervised fine-tuning with offline preference optimization via Direct Preference Optimization [33]. We additionally explore online preference optimization methods based on Group Relative Preference Optimization [50]. We evaluate our framework on FalconCode [9], a large-scale dataset of student submissions to Python programming assignment, training Qwen models [41] at 4B and 8B scales. Our results show that incorporating environment feedback strengthens models' ability to replicate student debugging behavior, improving over both prior code-only baselines and prompted large language models in functional alignment and code similarity.

Summary contributions. Our main contributions are:

- **Student-environment serialization.** We introduce a conversational formulation of student–environment interactions that enables language models to learn students iterative debugging processes.
- **A training methodology for artificial students.** We propose a pipeline combining supervised fine-tuning and preference optimization to align language models with authentic student coding trajectories.
- **Empirical evaluation on real student data.** We validate our framework on a large-scale dataset of student programming submissions, showing that environment feedback and preference optimization yield complementary gains in simulation realism. We also release our code: `KoutchemeCharles/edm-conv-ser`

2. RELATED WORK

A complementary line of work focuses on *knowledge tracing* (KT), whose primary goal is to estimate a student's mastery of knowledge components across exercises. While early approaches focused exclusively on predicting success on future assignments, more recent methods leverage language-model–based architectures to predict students' first submission to following assignments and the associated outcomes [24, 13, 15]. While such methods may predict student programs, this prediction serves mainly as an intermediate signal for updating knowledge estimates across programming tasks, not as a goal in itself. In contrast, we aim to obtain models that can be unrolled over multiple steps within a single assignment to generate and simulate problem-solving trajectories. Closer to our work in the educational domain is Miroyan et al. [27], who train open-weight language models using supervised fine-tuning to predict a student's next submission to Python programming assignments given a limited context of prior attempts. Their work shows that trained language models perform better than prompted LLMs. Ross et al. [35] go further by fine-tuning pre-trained language models on 3.8 million traces from a block-based educational programming platform, showing that models trained on real edit sequences yield richer representations of student behavior than those trained on final programs alone. However, neither approach incorporates the learning environment's responses to student submissions. Yet, serializing structured data into conversational format for language model training is well-established in NLP, with extensive work on synthetic dialogue generation [28, 11, 49] and math-informed dialogues [1].

Closest to our approach is OpenCodeInterpreter [51], which fine-tunes open-weight language models on multi-turn code-execution dialogues to improve code generation and refinement abilities. While their approach also serializes execution feedback into multi-turn dialogs, our work differs in two key aspects. First, our goal is to reproduce student problem-solving behavior, not to improve code generation; we train on authentic student trajectories rather than synthetic expert-generated interactions refined through execution outputs. Second, we develop preference optimization methods specifically adapted to student programming trajectories, leveraging infrastructure tied to learning environments, in particular the autograder, to construct preference pairs for training.

3. FROM LOGS TO DIALOGS

In this section, we introduce our approach for transforming student log data into suitable data for student simulation. Our work assumes that students interact with an automated assessment system returning summative feedback [6].

3.1 Assumptions

We assume access to a deterministic grading function which, for a submitted program a to a programming assignment d, can return two observable components: $v_r^{(d)}(a) = r$ and $v_f^{(d)}(a) = o$. The numerical score $r \in [0, 1]$ measures the correctness of the executed program (e.g., proportion of passed test cases). The textual feedback o contains the textual outcome of executing that submission: summaries of passed tests, numerical grades, and/or even runtime error traces, depending on what each dataset provides. We do not assume any internal structure of the (summative) feedback, only that it is consistently produced and directly observable in the logs. We also assume access to a dataset of institutional log data containing sequences of graded submissions and their outcomes: $\mathcal{D} = \{(d^i, s^i)\}_{i=1}^N$, where d^i denotes an assignment textual description and $s^i = (u_1^i, u_2^i, \ldots, u_{T_i}^i)$ denotes the sequence of graded interactions for a single student on that assignment. Each entry $u_t^i = (a_t^i, o_t^i)$ contains the submitted program a_t^i and its outcome $o_t^i = v_f^{(d^i)}(a_t^i)$. The trajectory length T_i equals the number of graded submissions recorded for assignment d^i.

3.2 Serialization

We represent student programming logs using the standard conversational format used to train and interact with chat-capable language models [31]. We structure each trajectory as a dialogue where the `user` role represents the automated assessment system and the `assistant` role represents a student learning how to program. The `system` prompt establishes the context: the model is a novice student learning to program, solving assignments while interacting with a learning environment that provides summative feedback. The initial `user` message provides the assignment description d_i and any relevant context. Each logged interaction $u_t^i = (a_t^i, o_t^i)$ then maps to two conversational turns: the student's code submission a_t^i appears as an `assistant` turn, followed by the grading environment's feedback o_t^i as the next `user` turn. This alternating structure naturally captures the iterative debugging process: students write code, receive feedback, then revise based on that feedback. Below is an illustration using a simple Python assignment:

System

```
You are a first-year novice student learning
programming in Python. Solve the given programming
assignment(s). You will be interacting with a learning
environment which will provide you with summative feedback.
```

User

```
Write a Python function called "compute_average".
The function should take as argument a (non-empty) list of
integers and returns the mean over those elements.
```

Assistant

```
def compute_average(nums):
    total = nums[0]
    for i in nums:
        total += i
    return average / len(nums)
```

User

```
Runtime error: undefined variable "average".
```

Assistant

```
def compute_average(nums):
    total = nums[0]
    for i in nums:
        total += i
    return total / len(nums)
```

User

```
Tests passed: 1/8
```

Assistant

```
def compute_average(nums):
    total = 0
    for i in nums:
        total += i
    return total / len(nums)
```

User

```
Tests passed: 8/8
```

4. TRAINING ARTIFICIAL LEARNERS

In this section, we present our training pipeline for training a language model π_θ to simulate how programming students solve assignments. Our core pipeline combines supervised fine-tuning with offline preference optimization on a serialized dataset $\mathcal{D}$. We additionally explore online preference optimization as an alternative to the offline stage.

4.1 Supervised Fine-tuning

The first method is to simply supervised finetune our model π_θ using the negative log likelihood (i.e., SFT) objective:

$$\mathcal{L}_{\text{SFT}}(\theta, \mathcal{D}) = -\sum_{s^i \in \mathcal{D}} \sum_{t=1}^{T_i} \log \pi_\theta\left(a_t^i \mid u_{<t}^i\right).$$

Following prior work, we only backpropagate the loss on assistant turns [10] containing student code submissions. The supervised fine-tuning step enables models to quickly learn patterns of student problem-solving.

4.2 Offline Preference Optimization

We use Direct Preference Optimization [33] (DPO), an offline preference optimization algorithm that aligns language models using pairwise preference datasets. This algorithm has shown much success in applied AI in education [21, 48, 38]. In this work, we construct preference datasets by forming contrastive pairs of candidate continuations from the same student trajectory. Let $u_{\leq t}^i$ denote a sampled partial trajectory of a student up to step t. The preferred continuation is the student's immediate next submission a_{t+1}^i, while the dispreferred continuation is the first following submission with a different grade. Formally, if we define:

$$k^\star(i,t) = \min\left\{k \in [2, T_i - t] : v_r^{(d)}(a_{t+k}^i) \neq v_r^{(d)}(a_{t+1}^i)\right\}.$$

then the resulting preference dataset is :

$$\mathcal{D}_{\text{DPO}} = \left\{\left(u_{\leq t}^i, a_{t+1}^i, a_{t+k^\star(i,t)}^i\right)\right\}_{i=1,\ldots,N;\ t \in \mathcal{T}_i}.$$

We believe $a_{t+k^\star(i,t)}^i$, the nearest semantically distinct code the student actually wrote, provides a high-quality contrast that teaches the model *why* students submit specific solutions at each step, preventing models from jumping prematurely to fully correct solutions, a pattern observed in prior work [27]. Using the resulting dataset, we optimize our model using the DPO loss [33]:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(u_{\leq t}^i, a_{t+1}^i, a_{t+k}^i) \sim \mathcal{D}_{\text{DPO}}} \left[\log \sigma\big(r_\theta(u_{\leq t}^i, a_{t+1}^i) - r_\theta(u_{\leq t}^i, a_{t+k^\star(i,t)}^i)\big)\right] \quad (1)$$

where σ is the sigmoid function and

$$r_\theta(u, a) \triangleq \beta \log \frac{\pi_\theta(a \mid u)}{\pi_{\text{ref}}(a \mid u)}. \quad (2)$$

with π_θ being the model being optimized and π_{ref} being the reference policy (the model before the start of the optimization (often π_{SFT})). In essence, Equation (1) penalizes the model based on how much more it prefers the dispreferred generation over the preferred one using an implicit reward (Equation (2)), defined by the model probabilities, with β controlling how much the trained model weights θ can deviate from their initialization point (e.g., the SFT weights).

4.3 Online Preference Optimization

We additionally explore whether online preference optimization can exceed offline methods for student simulation. Specifically, we use a variant of Group Relative Preference Optimization (GRPO) [16], an online algorithm that iteratively (1) samples multiple candidate next-step programs from the current policy, (2) evaluates them through a reward function, and (3) updates the model according to their relative quality. At each iteration, given a trajectory prefix $u_{\leq t}^i$, we sample $G = 4$ candidate next-step programs using top_p sampling. Each candidate is scored using a discrete reward that captures two complementary aspects [19] of next-step program prediction: functional alignment and syntactic similarity. Candidates whose abstract syntax tree exactly matches the student's actual next submission receive the highest reward (+2.0). Candidates that do not match exactly but achieve the same grade as the student's next attempt receive a smaller reward (+1.0). Candidates that fail to compile are penalized (-1.0), while all remaining candidates receive no reward. This tiered design prioritizes reproducing the student's specific solution while still rewarding functionally equivalent behavior, and aligns with outcome-based reward practices in code generation [22]. We update model parameters using DAPO [50], a more stable version of the original GRPO algorithm. Due to space constraints, we refer the reader to the original paper and our code base for the full loss formulation.

5. EXPERIMENTS

In this section, we detail the components of our experiments aimed at evaluating the utility of our framework.

5.1 Dataset

We evaluate our framework using FalconCode [9], a large-scale CS1 Python programming dataset from the United States Air Force Academy. The dataset includes student submissions, the associated grades, and the course autograder, which we used to regenerate the same detailed summative feedback students received during the course, showing pass/fail and expected outputs per test case. FalconCode contains three assignment difficulty levels: "skill", "lab", and "project". We excluded project assignments (lacking automated tests) and skill assignments (1–10 lines of code). Lab assignments represent medium-sized programs (10–50 lines). To manage the costs of running our experiments, we arbitrarily selected the assignment with the highest number of submissions from each lesson regrouping assignments targetting similar concepts. We retained trajectories where students passed at least one test, as all-failing trajectories often reflect non-serious attempts. We further removed non-compiling submissions but retained runtime errors and traces as environment feedback. We leave the handling of syntax errors to future work. FalconCode spans three semesters. To mimic realistic deployment, we use data from Spring 2021 for training, and evaluate on a randomly sampled susbet of 1000 trajectories from Spring 2022.

5.2 Models

We fine-tune two open-weight language models from the Qwen family [41]: `Qwen3-4B-Instruct` and `Qwen3-8B`. We selected these models for their strong performance on coding benchmarks and to study model size effects on student simulation quality.

Model variants. We evaluate variants to isolate the contribution of our framework components. As a baseline to prior work [27], we train models *without* environment feedback (**PARA**). Within our framework, we compare **SFT** (supervised fine-tuning), **DPO** (offline preference optimization), and **DAPO** (online preference optimization). For both preference optimization, we start training from the SFT weights.

Training setup. We use `unsloth` [8] to enable long-context fine-tuning on NVIDIA V100 GPUs with 32 GB of VRAM, using `triton`, our institution research cluster. All models are fine-tuned with QLoRA [10], following established hyperparameter recommendations [39, 42]: rank $r = 8$, $\alpha = 16$. Models are trained with a maximum context length of 4096 tokens. When trajectories exceed this limit, we remove earlier assistant turns until the sequence fits, creating a dynamic sliding window over recent submissions. For PARA models, unlike [27] who use a fixed window of $k = 4$ prior submissions, we dynamically limit prior submissions based on the context window size such that all trained models operate under the same computational constraints. We use a cosine learning rate of 1×10^{-4} for SFT variants (including PARA) (one epoch) and 1×10^{-5} for preference optimization, with $\beta = 0.5$ for DPO following [38]. For SFT variants and DPO, we select the best checkpoint based on validation loss from 20% of training data. For DPO, we sample all positions from all available trajectory prefixes from the training set. For GRPO, we sample two positions from each trajectory due to computational costs, generating $G = 4$ possible attempts per trajectory.

In-context learning baselines. To assess the value of fine-tuning, we include in-context learning (ICL) baselines using **BASE** model variants without training. We also evaluate the proprietary `GPT-5-mini` [30] model under the *minimal* reasoning configuration (see OpenAI model documentation).

5.3 Evaluation

For each test trajectory, we evaluate model predictions at all possible starting positions $t > 2$. Given a partial history $u^i_{\leq t}$, we autoregressively roll out the model for up to 5 steps using greedy-decoding. At each rollout step $j \in \{1, \ldots, 5\}$, the model generates a submission $\hat{a}^i_{t+j}$ conditioned on the original history and all previously generated rollout steps. Each generated submission is evaluated by the grading environment to produce the outcome $\hat{o}^i_{t+j} = v_f^{(d^i)}(\hat{a}^i_{t+j})$, which is then appended to the context for subsequent predictions. Rollout terminates early if there is no corresponding ground truth student generation or if the model generates a fully correct solution.

Metrics. We report three complementary metrics averaged across all predictions. First, **coverage** measures the proportion of ground-truth student submissions with model predictions. Coverage is less than 1.0 when models prematurely submit a correct solution (at step k where $v_r^{(d^i)}(\hat{a}^i_{t+k}) = 1$) while the student has not yet finished ($v_r^{(d^i)}(a^i_{t+k}) < 1$), a known challenge in student simulation [27]. For matched pairs where models did generate, we evaluate generation quality using two metrics: (1) **CodeBLEU** [34], measuring token-level syntactic and semantic similarity [15, 13, 24]; and (2) **grade proximity**, computed as $1 - |v_r^{(d)}(\hat{a}^i_{t+j}) - v_r^{(d)}(a^i_{t+j})|$, where higher values indicate closer functional alignment. These quality metrics are computed only on matched pairs where models did generate, to avoid biasing results against models that correctly stop at perfect grades. Conversely, this means models with different coverage levels have their quality metrics computed on different susbet of predictions, which should be kept in mind when comparing across methods. To assess robustness over rollouts, we also report **average degradation** Δ for each metric, which we compute as $\Delta m = \frac{1}{K-1} \sum_{k=2}^{K} (m_k - m_1)$, where m_k denotes metric performance at rollout step k and $K = 5$. More negative values indicate faster degradation from the first step.

6. RESULTS

Table 1 summarizes our training and test split statistics. Table 2 reports coverage and generation quality metrics averaged across all rollout steps. Figure 1 details performance at each rollout step, and Figure 2 illustrates how model-generated grades evolve compared to students ground truth grades. We highlight several key findings below.

Table 1: FalconCode dataset statistics. Summary of training and test splits. #Traj: total trajectories; #Stud: unique students; #Succ/#Fail: trajectories with final grade 100%/below 100%; #Asg: unique programming assignments; Avg. Len: mean submissions per trajectory;

Split	#Traj	#Stud	#Succ	#Fail	#Asg	Avg. Len
Train	1762	448	1712	50	17	22.4
Test	1000	384	981	19	15	20.1

Table 2: Rollout performance metrics. Legend: Cov (Coverage), GP (Grade Proximity), CB (CodeBLEU). Metrics averaged across k=1 to k=5 steps ahead, with degradation (Δ) showing the average drop from k=1 to k=2..5. Negative Δ values indicate performance worsens over longer rollouts. Bold: best performance within each model.

		Average			Degradation		
Model	Method	Cov	GP	CB	ΔCov	ΔGP	ΔCB
Qwen3-4B	BASE	0.528	0.690	0.491	-0.640	**0.278**	**-0.063**
	PARA	0.918	0.763	0.695	-0.118	-0.071	-0.137
	SFT	0.922	0.782	**0.704**	-0.107	-0.060	-0.138
	DPO	**0.982**	**0.797**	0.700	**-0.024**	-0.079	-0.137
	DAPO	0.859	0.781	0.700	-0.203	-0.068	-0.138
Qwen3-8B	BASE	0.570	0.733	0.581	-0.581	**0.287**	**-0.046**
	PARA	0.921	0.785	0.714	-0.112	-0.059	-0.121
	SFT	0.921	0.793	**0.718**	-0.110	-0.062	-0.126
	DPO	**0.957**	**0.796**	0.714	**-0.061**	-0.074	-0.129
	DAPO	0.919	**0.796**	0.709	-0.116	-0.078	-0.133
GPT-5-mini	BASE	**0.340**	**0.549**	**0.433**	**-0.910**	**0.184**	**-0.078**

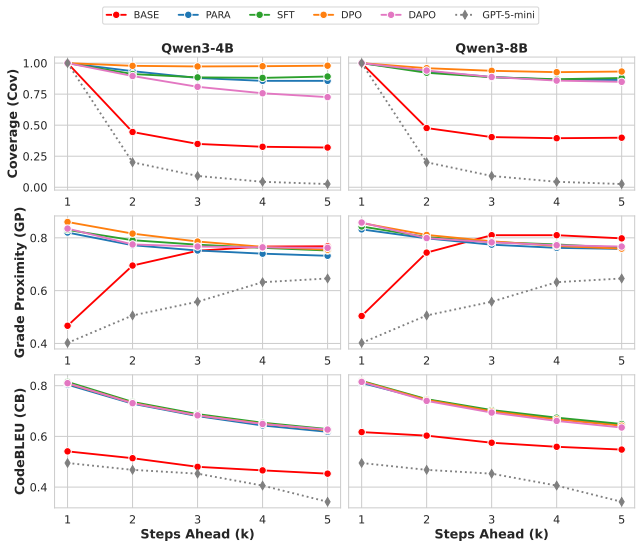


Figure 1: Performance across rollout steps. Coverage (top), Grade Proximity (middle) and CodeBLEU (bottom) as a function of autoregressive steps ahead (k=1 to k=5).

Prompted baselines cannot simulate realistic student behavior. BASE Qwen models achieve substantially lower coverage (0.528 for 4B, 0.570 for 8B), grade proximity (0.690, 0.733), and CodeBLEU (0.491, 0.581) than any fine-tuned model. GPT-5-mini performs worst overall, with a coverage of only 0.340 and grade proximity of 0.549. Its low coverage indicates that it frequently generates fully correct solutions that terminate trajectories after one or two steps, while its low CodeBLEU (0.433) confirms that the code it does produce has little resemblance to student submissions. Interestingly, BASE Qwen models outperform GPT-5-mini, possibly because their smaller capacity leads to genuine mistakes that incidentally resemble student errors, whereas GPT-5-mini's stronger coding ability defaults to expert-like solutions despite the explicit personification prompt.

Environment feedback improves student simulation. SFT performs as well as or better than PARA on all three metrics for both models (Table 2). The margins are modest but the pattern is consistent. This is notable because PARA has a structural advantage: without environment feedback in the context, it can fit more prior student submissions within the same token budget. Despite this, environment feedback proves more informative than additional code history. We hypothesize that earlier submissions, particularly from already-passed test stages, carry diminishing signal about the student's current debugging strategy, whereas grader feedback directly conditions the next revision.

Offline preference optimization yields the strongest results. DPO achieves the highest coverage across both model sizes (0.982 for 4B, 0.957 for 8B), producing valid code at more rollout steps than any other method. It also achieves the highest grade proximity for 4B (0.797 vs. 0.782 for SFT). SFT achieves slightly higher CodeBLEU (0.704 vs. 0.700 for 4B; 0.718 vs. 0.714), although this gap may partly reflect DPO's higher coverage: by generating at more rollout steps, DPO is evaluated on a broader and likely harder subset of predictions. DAPO shows mixed results: it matches DPO on grade proximity for 8B but achieves the lowest coverage among fine-tuned models on 4B (0.859), and does not consistently improve over SFT.

Performance over steps and trajectory position. Looking at Figure 1, BASE model coverage drops sharply ($\Delta = -0.640$ for 4B, -0.581 for 8B), as they frequently generate solutions that terminate the trajectory prematurely. Trained models degrade less, with DPO maintaining the highest coverage throughout (-0.024 for 4B, -0.061 for 8B), while DAPO coverage reduces more substantially over time (-0.203 for 4B). All fine-tuned models exhibit similar grade proximity degradation across rollout steps, with drops of approximately 0.06–0.08 from k=1 to k=5. The pattern over size is nuanced: for the 4B model, PARA degrades faster than SFT on grade proximity (-0.071 vs. -0.060), but for 8B, PARA degrades the least (-0.059) while preference-optimized models degrade the most (DPO: -0.074, DAPO: -0.078). Despite this, DPO maintains a higher absolute grade proximity compared to PARA at every rollout depth. Figure 2 shows that DPO-trained models most closely follow real student grade progression throughout normalized trajectories. DAPO models in contrast struggle to track students progressions. BASE models exhibit the opposite pattern: their grade proximity *improves* over rollout steps (+0.278 for 4B, +0.287 for 8B), and GPT-5-mini shows the same trend (+0.184). Figure 2 reveals an interesting pattern: untrained models generate solutions that achieve similar grades independently of trajectory position. We hypothesize that these models attempt to generate correct solutions, rather than adapting to the student's current progress. The difference is that GPT-5-mini largely succeeds, producing near-perfect grades throughout, while BASE Qwen models fail enough test cases that the corresponding grades happen to align with students grades at later trajectory positions. Neither reproduces the progressive improvement characteristic of real students. CodeBLEU shows more uniform downward trends across all trained models (≈ -0.13), suggesting that syntactic divergence accumulates at a similar rate regardless of training method. BASE model CodeBLEU remains comparatively stable (-0.063 for 4B, -0.046 for 8B).

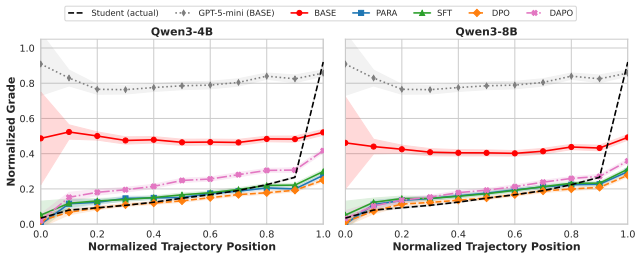


Figure 2: Grade progression across normalized trajectory position. Comparison of predicted grades between artificial student models and real students (black dashed, the average). The x-axis represents normalized position (by length) within a student's problem-solving trajectory (0 = start, 1 = end). Shaded regions indicate 95% confidence intervals. The ground truth line does not reach 1.0 as some students failed. The abrupt jump reflects the proportion of students who resolve multiple failing test cases in a single revision.

7. CONCLUDING DISCUSSION

Our experiments show that conversational serialization of student–environment interactions, combined with preference optimization, produces artificial students that more closely track real learners' debugging behavior than prompted baselines and models trained without feedback. Performance improvements compared to baselines are consistent in direction across both model sizes and all rollout steps. We believe coverage and grade proximity are particularly important dimensions for student simulation: models that generate code at every trajectory step, passing and failing the same tests at the same stages as real students, capture the incremental nature of real problem-solving more faithfully than models optimizing for surface-level code similarity alone. Among our findings, offline preference optimization via DPO provides the largest gains on these dimensions, achieving the highest coverage and grade proximity across both model sizes. Online preference optimization (DAPO) on the other hand does not consistently improve over SFT. We hypothesize that DAPO's weaker results may stem from (a) our computational constraint of sampling only 2 prefixes per trajectory compared to DPO's exhaustive coverage, and (b) the absence of chain-of-thought reasoning [47] prior to code generation, which online methods typically rely on for effective optimization. Our results thus suggest that without explicit reasoning traces, offline methods may remain more practical than recent online alternatives for student simulation.

While our experiments focus on within-assignment simulation, we hope the proposed framework will support several existing research directions.

Bridge with knowledge tracing and tutoring. Knowledge tracing (KT) models estimate a learner's latent proficiency across assignments, often predicting whether a student will solve a future task on the first attempt. Our framework operates at a complementary granularity: modeling the fine-grained debugging steps and intermediate attempts that precede success within a single task. We envision connecting these two levels by using cross-problem KC estimates as conditioning inputs to our simulation prompt [14], allowing our artificial students to exhibit more realistic learner behavior.

Bridge with tutoring. Moreover, our conversational serialization treats the automated assessment system as a dialogue partner that provides summative feedback. This is structurally identical to how a tutor agent would interact with a student. By representing student–environment interactions in this format, we could seamlessly extend sequences to include student–tutor exchanges, combining debugging traces with tutoring interactions [2] in a single sequence. This opens several directions: our models could be adapted to predict student success conditioned jointly on debugging attempts and tutor exchanges [37], or integrated into conversational knowledge tracing frameworks [36]. More broadly, artificial students trained with our approach could serve as scalable simulators for reinforcement-learning–based tutor training [12], where student success provides a training signal of tutoring strategy effectiveness.

Limitations. Our work is not free of limitations. We experimented with a single Python programming dataset and one model family across two sizes. Validation on additional datasets from different institutional contexts (with different autograder feedback styles) and programming languages is needed. Our evaluation data is heavily skewed toward successful students: 95% of trajectories end with a perfect score, meaning our results primarily reflect simulation quality for learners who ultimately succeed. This skew is partly inherent to programming courses, where most students eventually pass [45] and those who do not often abandon the course entirely [4, 20]. However, struggling students are precisely those most in need of intervention; modeling their behavior, including trajectory abandonment, remains important future work. We also report results using greedy decoding on a single semester of test data without statistical testing; evaluating under varied sampling strategies and data splits would strengthen confidence in our findings. Finally, while our metrics demonstrate that models replicate student grade trajectories and code similarity, one needs to validate whether these new artificial students will be effectively useful for the downstream applications we motivate, such as tutor training or intervention testing. Establishing this educational validity is important future work.

Future work. Beyond addressing the highlighted limitations, there are three direct next steps for student simulation. First, we will explore agent-based artificial students equipped with execution, submission, and editing tools, mirroring the interactive environments real students use when programming. Second, we will improve our online preference optimization method. GRPO-style methods are known to improve model performance when elicited to produce chain-of-thought reasoning. By prompting models to generate explicit student-like thought traces in between tool operations, we hope to improve such models' ability to follow students' processes while potentially yielding post-analysis insights into why students code the way they do. Lastly, we will explore methods to address the challenge of efficiently integrating students' prior history (i.e., their attempts at prior assignments) to enable better personalization.

8. REFERENCES

[1] S. N. Akter, S. Prabhumoye, J. Kamalu, S. Satheesh, E. Nyberg, M. Patwary, et al. MIND: Math informed synthetic dialogues for pretraining LLMs. In *The*

Thirteenth International Conference on Learning Representations, 2025.

[2] N. Ashok Kumar and A. Lan. Improving socratic question generation using data augmentation and preference optimization. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 108–118, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[3] D. Azcona and A. F. Smeaton. Targeting at-risk students using engagement and effort predictors in an introductory computer programming course. In *European Conference on Technology Enhanced Learning*, pages 361–366. Springer, 2017.

[4] J. Bennedsen and M. E. Caspersen. Failure rates in introductory programming: 12 years later. *ACM Inroads*, 10(2):30–36, Apr. 2019.

[5] A. H. Brown. Simulated classrooms and artificial students: The potential effects of new technologies on teacher education. *Journal of research on computing in education*, 32(2):307–318, 1999.

[6] D. L. Butler and P. H. Winne. Feedback and self-regulated learning: A theoretical synthesis. *Review of Educational Research*, 65:245–281, 1995.

[7] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[8] M. H. Daniel Han and U. team. Unsloth, 2023.

[9] A. de Freitas, J. Coffman, M. de Freitas, J. Wilson, and T. Weingart. Falconcode: A multiyear dataset of python code samples from an introductory computer science course. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, SIGCSE 2023, page 938–944, New York, NY, USA, 2023. Association for Computing Machinery.

[10] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. Qlora: Efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, page 441, Red Hook, NY, USA, 2023. Curran Associates Inc.

[11] N. Ding, Y. Chen, B. Xu, Y. Qin, S. Hu, Z. Liu, et al. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, 2023.

[12] D. Dinucu-Jianu, J. Macina, N. Daheim, I. Hakimi, I. Gurevych, and M. Sachan. From problem-solving to teaching problem-solving: Aligning LLMs with pedagogy using reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 272–292, Suzhou, China, Nov. 2025. Association for Computational Linguistics.

[13] Z. Duan, N. Fernandez, A. Hicks, and A. Lan. Test case-informed knowledge tracing for open-ended coding tasks. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK '25, page 238–248, New York, NY, USA, 2025. Association for Computing Machinery.

[14] Z. Duan, N. Fernandez, and A. Lan. Kaser: Knowledge-aligned student error simulator for open-ended coding tasks, 2026.

[15] Z. Duan, N. Fernandez, A. B. L. Narayanan, M. Hassany, R. S. de Alencar, P. Brusilovsky, B. Akram, et al. Automated knowledge component generation for interpretable knowledge tracing in coding problems, 2025.

[16] D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, Q. Zhu, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638, 2025.

[17] M. C. Jadud. Methods and tools for exploring novice compilation behaviour. In *Proceedings of the second international workshop on Computing education research*, pages 73–84, 2006.

[18] J. Kasurinen and U. Nikula. Estimating programming knowledge with bayesian knowledge tracing. *ACM SIGCSE Bulletin*, 41(3):313–317, 2009.

[19] C. Koutcheme, N. Dainese, and A. Hellas. Direct repair optimization: Training small language models for educational program repair improves feedback. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 564–581, Vienna, Austria, July 2025. Association for Computational Linguistics.

[20] C. Koutcheme, S. Sarsa, A. Hellas, L. Haaranen, and J. Leinonen. Methodological considerations for predicting at-risk students. In *Proceedings of the 24th Australasian Computing Education Conference*, ACE '22, page 105–113, New York, NY, USA, 2022. Association for Computing Machinery.

[21] C. Koutcheme, J. Woodrow, and C. Piech. Aligning small language models for programming feedback: Towards scalable coding support in a massive global course. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.1*, SIGCSE TS 2026, page 610–616, New York, NY, USA, 2026. Association for Computing Machinery.

[22] H. Le, Y. Wang, A. D. Gotmare, S. Savarese, and S. C. H. Hoi. CodeRL: Mastering Code Generation through Pretrained Models and Deep Reinforcement Learning, Nov. 2022. arXiv:2207.01780 [cs].

[23] J. Leinonen, P. Denny, O. Kiljunen, S. MacNeil, S. Sarsa, and A. Hellas. Llm-itation is the sincerest form of data: Generating synthetic buggy code submissions for computing education. In *Proceedings of the 27th Australasian Computing Education Conference*, ACE '25, page 56–63, New York, NY, USA, 2025. Association for Computing Machinery.

[24] N. Liu, Z. Wang, R. Baraniuk, and A. Lan. Open-ended knowledge tracing for computer science education. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3849–3862, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[25] S. MacNeil, M. Rogalska, J. Leinonen, P. Denny, A. Hellas, and X. Crosland. Synthetic students: A comparative study of bug distribution between large language models and computing students. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 1*, SIGCSE Virtual 2024, page 137–143, New York, NY, USA,

2024. Association for Computing Machinery.

[26] N. Matsuda, W. W. Cohen, J. Sewall, G. Lacerda, and K. R. Koedinger. Evaluating a simulated student using real students data for training and testing. In *International Conference on User Modeling*, pages 107–116. Springer, 2007.

[27] M. Miroyan, R. Niousha, J. E. Gonzalez, G. Ranade, and N. Norouzi. Parastudent: Generating and evaluating realistic student code by teaching llms to struggle, 2025.

[28] S. Mukherjee, A. Mitra, G. Jawahar, S. Agarwal, H. Palangi, and A. Awadallah. Orca: Progressive learning from complex explanation traces of GPT-4. *arXiv preprint arXiv:2306.02707*, 2023.

[29] M. H. Nguyen, V.-A. Pădurean, A. Gotovos, S. Tschiatschek, and A. Singla. Synthesizing high-quality programming tasks with llm-based expert and student agents. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education*, pages 77–91, Cham, 2025. Springer Nature Switzerland.

[30] OpenAI. GPT-5 System Card. Technical report, OpenAI, 2025.

[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[32] T. Phung, V.-A. Pădurean, A. Singh, C. Brooks, J. Cambronero, S. Gulwani, A. Singla, et al. Automating human tutor-style programming feedback: Leveraging gpt-4 tutor model for hint generation and gpt-3.5 student model for hint validation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, LAK '24, page 12–23, New York, NY, USA, 2024. Association for Computing Machinery.

[33] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[34] S. Ren, D. Guo, S. Lu, L. Zhou, S. Liu, D. Tang, et al. Codebleu: a method for automatic evaluation of code synthesis, 2020.

[35] A. Ross, M. Srivastava, J. Blanchard, and J. Andreas. Modeling student learning with 3.8 million program traces, 2025.

[36] A. Scarlatos, R. S. Baker, and A. Lan. Exploring knowledge tracing in tutor-student dialogues using llms. In *Proceedings of the 15th Learning Analytics and Knowledge Conference, LAK 2025, Dublin, Ireland, March 3-7, 2025*. ACM, 2025.

[37] A. Scarlatos, N. Liu, J. Lee, R. Baraniuk, and A. Lan. Training llm-based tutors to improve student learning outcomes in dialogues. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education*, pages 251–266, Cham, 2025. Springer Nature Switzerland.

[38] A. Scarlatos, D. Smith, S. Woodhead, and A. Lan. *Improving the Validity of Automatically Generated Feedback via Reinforcement Learning*, page 280–294. Springer Nature Switzerland, 2024.

[39] J. Schulman and T. M. Lab. Lora without regret. *Thinking Machines Lab: Connectionism*, 2025. https://thinkingmachines.ai/blog/lora/.

[40] T. Sirkiä and J. Sorva. Exploring programming misconceptions: an analysis of student mistakes in visual program simulation exercises. In *Proceedings of the 12th Koli calling international conference on computing education research*, pages 19–28, 2012.

[41] Q. Team. Qwen3 technical report. Technical report, 2025.

[42] Unsloth AI. Lora hyperparameters guide, 2024. Accessed: 2025-12-23.

[43] K. VanLehn, S. Ohlsson, and R. Nason. Applications of simulated students: An exploration. *Journal of Artificial Intelligence in Education*, 5(2):135–175, 1994.

[44] L. Wang, A. Sy, L. Liu, and C. Piech. Deep knowledge tracing on programming exercises. In *Proceedings of the fourth (2017) ACM conference on learning@ scale*, pages 201–204, 2017.

[45] C. Watson and F. W. Li. Failure rates in introductory programming revisited. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, page 39–44, New York, NY, USA, 2014. Association for Computing Machinery.

[46] C. Watson, F. W. Li, and J. L. Godwin. Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *2013 IEEE 13th international conference on advanced learning technologies*, pages 319–323. IEEE, 2013.

[47] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, 2022.

[48] J. Woodrow, C. Piech, and S. Koyejo. Improving generative ai student feedback: Direct preference optimization with teachers in the loop. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 442–449, Palermo, Italy, July 2025. International Educational Data Mining Society.

[49] C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, et al. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*, 2024.

[50] Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, et al. Dapo: An open-source llm reinforcement learning system at scale, 2025.

[51] T. Zheng, G. Zhang, T. Shen, X. Liu, B. Y. Lin, J. Fu, et al. Opencodeinterpreter: Integrating code generation with execution and refinement. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 12834–12859. Association for Computational Linguistics, 2024.

The Medium is the Message: How Lecture Q&A Channel Structure Shapes Student Questioning

Meenakshi Mittal*
University of California, Berkeley
meenakshi.mittal@berkeley.edu

Kaden Tang*
University of California, Berkeley
kaden.tang1@berkeley.edu

Christopher Mach*
University of California, Berkeley
mach_chris@berkeley.edu

Narges Norouzi
University of California, Berkeley
norouzi@berkeley.edu

ABSTRACT

Student questions provide a vital signal of how learning unfolds, yet the nature of these questions varies systematically across delivery media. We analyze student help-seeking behavior across two contrasting lecture environments: a synchronous, public Q&A channel where students compete for limited "fixed-time" airwaves, and an asynchronous, private AI lecture support tool that supports "flexible-time" engagement. We introduce an automated, human-validated labeling scheme to characterize over 4,500 student questions across three dimensions: intent type, reasoning explicitness, and confidence. Our application of this scheme suggests that while synchronous channels seem to reinforce a fixed-time/variable-learning bottleneck, the asynchronous medium is more consistent with a flexible-time/fixed-learning model. When inquiry is decoupled from the live lecture stream, the asynchronous channel is associated with denser, more conversational question streams (1.93 turns vs. 1.06 on average) and a significant increase in definition-oriented question types. We show that medium structure appears to meaningfully shape the length and volume of student questioning, shifting question-asking from survival in a live queue toward more iterative conceptual engagement.

Keywords

Question-Answering, Lecture, Educational Technology

1. INTRODUCTION

Student questions offer a direct window into how learning unfolds. They reflect what students find confusing, what they believe matters, and what they feel comfortable admitting they do not understand. In large classes, where instructors and Teaching Assistants (TAs) cannot engage deeply with every student in real time, the questions students ask become a practical signal of where support is needed.

A useful lens for thinking about lecture support is the contrast between *fixed-time* instruction and *self-paced* learning time. Traditional lectures largely follow a fixed schedule, whereas students vary widely in how much time and clarification they require to reach the same level of understanding. Prior work argues for shifting educational experiences toward more flexible time infrastructures to support consistent learning outcomes across students [6, 7, 8]. This framing is often discussed in the context of formative and summative assessments, but it also applies naturally to lecture support. When time is fixed and attention span is limited, students may ask fewer questions, ask different types of questions, or avoid asking altogether; when time is flexible and the interaction is private, students may ask clarification and follow-up questions, watch the lecture in multiple sessions, and pursue understanding iteratively.

At the same time, the conditions under which students ask questions are changing quickly. AI-powered education tools are now widely used alongside traditional course channels such as office hours, discussion forums, and live lecture Q&A. These tools can scale access to help, but they also change basic properties of help-seeking: who students believe is answering their question, whether questions are public or private, how easy it is to ask follow-up questions, and whether students can pause and rewind the material while forming their question. As a result, the questions instructors observe may differ not only because students need help with different concepts, but also because the medium may shape what they choose to ask and how they express it.

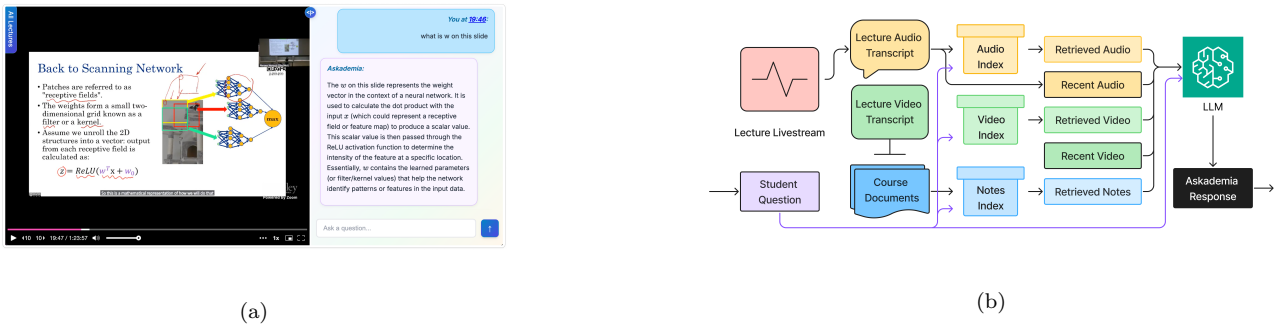
In this paper, we study student information-seeking behavior across two lecture support media with contrasting structures. The first is a synchronous, public live lecture Q&A channel (moderated on Slido[1]), where students post questions that are visible to peers and expect a TA response within the constraints of lecture pacing and a shared queue. The second is Askademia [20], an asynchronous, private AI lecture support tool used while students watch recorded lectures. In this medium, students know an AI system will an-

*Equal contribution.

Meenakshi Mittal, Kaden Tang, Christopher Mach, and Narges Norouzi. The Medium is the Message: How Lecture Q&A Channel Structure Shapes Student Questioning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 564–570. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039893

[1] https://www.slido.com/



(a) (b)

Figure 1: (a) Askademia interface embedded in the lecture experience, enabling grounded question asking at specific lecture moments. (b) Askademia system pipeline for retrieving relevant context and generating grounded responses.

swer them, their questions are private, and they can pause, rewind, and ask multiple follow-up questions.

Askademia [20][2] is a context-aware question-answering tool developed and deployed at UC Berkeley. It processes lecture video and audio to produce searchable, textual representations of spoken utterances and on-screen text, retrieves relevant lecture moments and course notes based on a student question, maintains a sliding window of recent context, and uses a Large Language Model (LLM) to generate a grounded response (Figure 1b). The student-facing interface (Figure 1a) supports questions tied to specific lecture moments and enables rapid back-and-forth as students refine their questions while reviewing material.

We ask essential questions with important implications for both tool design and instructional practice:

RQ1 How do student questions differ between fixed-time, synchronous TA lecture Q&A and self-paced, asynchronous AI lecture support in terms of what students ask and how they express reasoning and confidence?

RQ2 How do question volume, timing, conversation length, and user activity differ across media, and what do these differences suggest about how students use each channel?

To investigate these questions, we introduce a labeling scheme that characterizes questions along three dimensions: (1) intent type (logistical, definition-oriented, analytical, and meta), (2) whether students explicitly or implicitly express their reasoning process, and (3) confidence (neutral, high confidence, and low confidence). We apply this scheme at scale to quantify differences in question content and expression across synchronous and asynchronous media. Finally, we compare question volume, conversation length, and user activity to characterize how question flow differs between live public Q&A and private asynchronous AI support.

2. RELATED WORK

2.1 Student Help-Seeking and Behavior

Student questions are shaped by more than content difficulty: help-seeking is tied to motivation, confidence, and perceived cost (e.g., effort, embarrassment, or fear of asking a "bad" question) [4]. Digital learning tools aim to lower these barriers and increase engagement by changing how and when students can ask for help [2, 3]. As LLM-based tools become more common for student learning support, recent studies emphasize that tool choice and context affect how learners seek help, including whether students externalize uncertainty, ask incremental clarifications, or adopt different strategies [12, 27, 5]. Our work compares how question characteristics shift between a synchronous public medium and an asynchronous private AI medium.

2.2 Lecture-Time and Course Q&A Channels

Large courses face a bottleneck: questions arrive continuously, but instructional bandwidth and time are limited [15, 17]. This motivates the design of course Q&A channels to manage scale, redundancy, and timeliness. Tools such as Slido impose distinctive constraints: questions are posted in a shared, public queue, and participation is shaped by lecture pacing and limited response capacity. Prior work in computing education documents both the promise and the practical friction of these at-scale question channels [15, 17]. Our comparison between synchronous, public lecture Q&A and asynchronous, private AI-enabled Q&A examines how these constraints are reflected in the questions students ask and how they participate.

2.3 Retrieval-Augmented and Multimodal AI Course Assistants

Recent educational assistants use LLMs to support student questions in course environments [28, 23, 9]. These systems are often deployed as virtual assistants to reduce logistical overhead and scale students' access to course support [15, 17, 11, 14]. They have become more contextualized and course-aligned using Retrieval-Augmented Generation (RAG) [13, 18, 1]. In parallel, lecture and video tooling have explored how to represent what happened in a lecture through transcripts and visual signals [16]. More recent multimodal methods improve robustness by aligning noisy Automated Speech Recognition (ASR) with visual context [24] and by incorporating video frames into scientific transcription [26]. Broader progress in multimodal and visual question answering further motivates treating lecture content as a multimodal substrate that can be queried [25, 10].

[2] https://www.askademia.org/

2.4 Analyzing Student Questions

Prior work uses qualitative coding of question intent types and discourse features, paired with reliability checks, as well as automated NLP approaches for categorization at scale. With the growth of LLMs, prompting-based labeling has become a practical way to approximate consistent categorization over large corpora of student discourse [27, 5]. Studies of AI course support also highlight risks around consistency, misalignment with instructional norms, and equity when scaling support [19, 9]. Our paper introduces a compact labeling scheme and applies automated, human-validated labeling to compare patterns across synchronous, public Q&A and asynchronous, private AI Q&A. Rather than evaluating answer quality or system components, our work asks how information-seeking behavior differs when students interact with an AI system privately and asynchronously, compared with a synchronous, public channel.

3. METHODS

3.1 Lecture Support Media

We study student information-seeking behavior in two large upper-division Computer Science (CS) and Data Science (DS) courses at the University of California, Berkeley, with a combined enrollment of around 1,500 students. In both courses, students could ask questions via synchronous and asynchronous channels. In the synchronous medium, students posed questions electronically to TAs during the lecture via Slido. Students contributed to a shared channel of questions, where they could see contributions from other students and TAs. Each lecture lasted approximately 80 minutes, and students could attend live in-person or via Zoom. In the asynchronous medium, students directly asked Askademia while watching recorded lectures on the Askademia platform. Each student contributed to an individual channel of questions, could pause, rewind, and speed up lectures, and could not see other students' conversations.

Per IRB approval for this study,[3] All data included in our analyses were collected and used in accordance with the approved protocol. This protocol covered both sources of student questions considered in our analyses, namely synchronous questions submitted through Slido during lecture and asynchronous questions submitted through Askademia. Our analysis includes **992** synchronous questions (607 CS, 385 DS) asked via Slido across 56 lectures and **4,511** (2,721 CS, 1,790 DS) asynchronous questions from Askademia users who consented to the use of their anonymized questions for research.

3.2 Observable Characteristics

3.2.1 Labeling Scheme

Two annotators conducted exploratory analysis on a random sample of 100 questions and developed a three-dimensional labeling scheme: **Intent Type**, **Reasoning Process**, and **Confidence**. We treat this as a compact, study-specific framework rather than a standardized educational taxonomy. We selected these dimensions because they capture both what students ask and how they ask it, which aligns with our goal of comparing question behavior across synchronous and asynchronous media. The labeling criteria are described below:

[3]This study was approved by the UC Berkeley Institutional Review Board under Protocol ID 2025-09-18906.

Table 1: Representative Examples of Question Labels

Label	Example
	Intent Type
Logistical	"Which textbook are we using?"
Definition	"What is a partial derivative"
Analytical	"Why square the residuals instead of taking an absolute value?"
Meta	"draw me a picture"
	Reasoning Process
Implicit	"Does the denominator of softmax function include the numerator?"
Explicit	"Why does high bias matter? can't we just add in a factor to account for it and try and min the variance?"
	Confidence
Low	"Im confuse, if i have y = mx + b, and i have y(x) = 0, then won't it be 0 = mx +b and thus one solution?"
Neutral	"How is clustering different from classification?"
High	"Bias is the expected difference between predicted and actual though right"

- **Intent Type:** The type of content. Logistical pertains to course operations and policies; Definition pertains to identifying course concepts; Analytical pertains to explaining course concepts in depth; and Meta refers to all questions unrelated to the course. Questions could be labeled as both Analytical and Definition.
- **Reasoning Process:** Whether a student expresses their reasoning process implicitly or explicitly. Explicit questions reveal a student's train of thought, while Implicit questions do not.
- **Confidence:** How confidently a student asks a question. Low-Confidence indicates confusion or skepticism; Neutral questions are asked without uncertainty or over-sureness; High-Confidence questions are typically requests for confirmation and validation.

Two annotators labeled a random sample of 30 questions from both modalities. Initial inter-rater reliability was 60% for intent type, 93% for reasoning process, and 83% for confidence. For each question where there was disagreement, annotators discussed their decisions until an agreement was reached. Afterward, one of the two annotators labeled a random sample of 100 questions.

3.2.2 Automated Labeling

We developed an automated labeling system by prompt engineering GPT-4o [21] for classification. The LLM was provided with a labeling scheme and our set of 30 questions labeled by both annotators. We instructed the LLM to provide an explanation for each generated label, and used hyperparameter values of 0.1 for temperature and top-p, which control the creativity of model responses. When evaluated on our test set of 100 labeled questions, our classifier achieved 81%, 94%, and 83% agreement with human labels for intent type, reasoning process, and confidence, respectively. We used this LLM classifier to label all student questions.

3.2.3 Question Length and Sentiment

We measured length and sentiment as additional characteristics of student questions. To determine sentiment, we used pysentimiento [22], a transformer-based library for sentiment analysis. Each question was classified as Positive, Negative, or Neutral.

3.3 Question Volume and Frequency

We recorded the number of questions per lecture in each medium, the conversation lengths, and the total number of questions asked by each unique user. In the synchronous medium, each question opens a thread where students can post follow-up questions and receive additional TA responses; conversation history is measured as the number of student-TA turns within that thread. In the asynchronous medium, information seeking is continuous in a chat-like interface; we treat a question as a follow-up if it is asked within one minute of a previous question. We selected this one-minute window as a simple heuristic to capture short bursts of back-and-forth clarification in a chat setting. To reduce sensitivity to this choice, we also report a stricter alternative in which follow-up questions are linked only when they share the same lecture timestamp, indicating that a student paused the lecture video to engage further with a specific moment in the recording. We further analyze timestamp-anchored interaction patterns in the asynchronous medium, including clusters of questions asked at the same lecture timestamp and questions asked after rewinding the lecture.

4. RESULTS

4.1 RQ1: Question Content and Expression

We study how question content and expression differ across media by analyzing five characteristics: intent type, reasoning process, confidence, sentiment, and question length.

Table 2: Proportion of each intent type for questions asked in synchronous and asynchronous media, with p-values from a two-proportions z-test. Results are grouped by course.

Intent	Course	Synch	Asynch	Diff	p
Logistical	CS	0.102	0.033	0.069	< 0.001
	DS	0.242	0.064	0.178	< 0.001
Definition	CS	0.245	0.402	-0.157	< 0.001
	DS	0.255	0.368	-0.113	< 0.001
Analytical	CS	0.616	0.513	0.103	< 0.001
	DS	0.475	0.423	0.052	0.062
Meta	CS	0.048	0.072	-0.024	0.029
	DS	0.047	0.153	-0.106	< 0.001

Intent type. Table 2 shows the proportion of each intent type for both media. Questions labeled as both Analytical and Definition contributed to proportion estimates for both categories. For both courses, the proportions of questions that were Logistical and Analytical were higher in the synchronous medium than in the asynchronous one. In comparison, for both courses, the proportions of student questions that were Definition and Meta were significantly lower in the synchronous medium than the asynchronous one.

Reasoning process. Across media, student reasoning patterns were similar. For the CS course, 10.87% of questions expressed explicit reasoning in the synchronous medium compared to 11.44% in the asynchronous one. For the DS course, 8.31% of questions expressed explicit reasoning in the synchronous medium compared to 9.05% in the asynchronous one. These differences were not statistically significant.

Confidence. For the CS course, the proportion of high-confidence questions was lower in the synchronous medium (0.04) than in the asynchronous (0.09), with $p < 0.001$. Similarly, for the DS course, the proportion of high-confidence questions in the synchronous medium (0.021) was roughly half of that in the asynchronous (0.041), although this difference was not statistically significant. For low- and medium-confidence (neutral) questions, differences between media were marginal.

Sentiment. A higher proportion of questions in CS course expressed negative sentiment in the synchronous medium (0.11) compared to the asynchronous (0.075), with $p = 0.004$. However, the proportion of negative questions in the DS course was similar for both media, ranging from 0.119 in the synchronous medium to 0.111 in the asynchronous one. In only the CS course, a lower proportion of questions were neutral in the synchronous medium (0.885) compared to the asynchronous one (0.916) with $p = 0.014$. For positive questions, differences between media were not statistically significant in both courses.

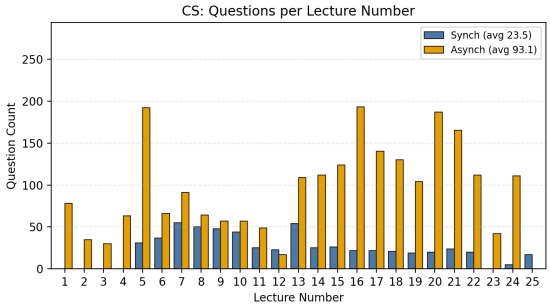
Question length. We report results from a two-sample t-test. For the CS course, the mean question length was slightly greater in the synchronous medium (15.53 words) than in the asynchronous one (14.91 words). For the DS course, mean question length was significantly greater in the synchronous medium (16.15 words) than the asynchronous one (12.86 words), with $p < 0.001$.

4.2 RQ2: Question Volume and Frequency

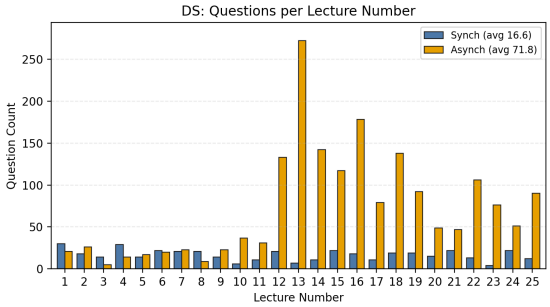
To study how interaction modality shapes question flow, we compare synchronous in-lecture questions (Synch TA) with asynchronous questions (Asynch Askademia) along three timing/frequency dimensions: per-lecture volume, conversation structure, and per-user participation concentration.

Per-lecture volume. As shown in the Figure 2, Asynch Askademia consistently receives substantially more questions per lecture than Synch TA within each course. In the CS course, the mean questions per lecture was 93.12 (Asynch) vs. 23.52 (Synch), with median questions per lecture 91 vs. 22. In the DS course, mean questions per lecture were 71.84 (Asynch) vs. 16.64 (Synch), with medians 49 vs. 18.

Figure 2 also shows a noticeable mid-semester surge in asynchronous question volume, aligning with the midterm period when students revisit past material. In contrast, Synch TA volume shows a flatter pattern over the same span, with no comparable spike. This divergence suggests that stu-



(a) CS course



(b) DS course

Figure 2: Distribution of questions per lecture, Synch TA vs. Asynch Askademia, shown separately for the CS and DS courses.

dents used the asynchronous medium for extended review and studying, while the live Slido channel may be used more for immediate, in-the-moment clarifications.

Conversation structure. Asynchronous questions are submitted with an associated lecture timestamp, allowing us to characterize how students concentrate questions around particular moments. We define a "paused" conversation as a set of two or more questions, with the same lecture timestamp, from a single student. Across **4,511** asynchronous questions, **2,300** are part of a "paused" conversation (51.0% of all Asynch questions). This behavior has no direct analog in the live TA setting, but highlights a distinct form of demand: students often want to pause at a specific point in the lecture and ask several tightly coupled questions to reach understanding, reinforcing the value of an always-available support channel anchored to fixed points in time.

We also observe **230** "rewind" questions, in which a student asks a question at an earlier lecture timestamp after having asked a question later in the lecture. This further supports the interpretation that asynchronous help-seeking is embedded in non-linear lecture consumption: students pause and move back and forth through the recording, revisiting earlier moments to resolve confusion when it arises.

Figure 3 indicates longer conversations in the asynchronous medium. Mean conversation length is 1.93 (Asynch) vs. 1.06 (Synch). This remains true under a stricter follow-up definition that only links questions with the same lecture timestamp (i.e., "paused" conversations), rather than linking questions within 60s; the asynchronous mean conversation length is still higher (1.54 vs. 1.06), with $p < 0.001$. Asynchronous conversations more often extend beyond single-question interactions, whereas synchronous interactions are predominantly one-shot. This pattern is consistent with students treating the asynchronous tool as a conversational agent: they can ask an initial question, read the response, and then refine or extend their inquiry without worrying about taking up public airtime or interrupting the lecture flow.

Participation concentration. Figure 4 indicates substantially more active users in Asynch Askademia than in the Synch medium. In the Synch medium, only 31 users asked

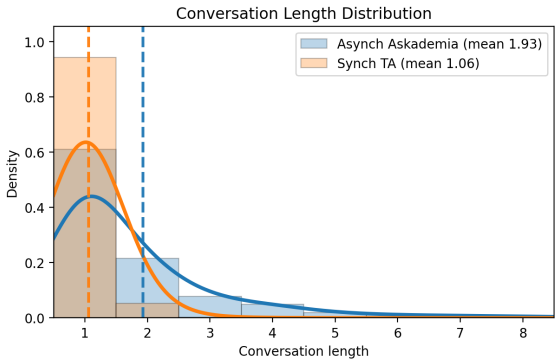


Figure 3: Conversation-length distribution (1 = no follow-up, 2 = one follow-up, etc.)

at least 5 questions throughout the whole semester, whereas in Asynch, 121 users asked at least 5 questions—almost **4**× as many relatively active users. The percentile markers in the figure show the same pattern of heavier engagement in Asynch: Asynch has p90 = 48.0 and p99 = 221.5, compared with Synch, which has p90 = 5.2 and p99 = 27.9. The Synch distribution is visibly concentrated in the lowest bins, suggesting that most students either ask a few questions or rely on the shared channel rather than repeatedly posting. In contrast, Asynch exhibits a much longer right tail, consistent with many more students returning to the tool across the term and using it as a sustained support resource while learning at their own pace. This broader set of active users suggests that a private, self-paced medium may extend meaningful information seeking beyond a small set of highly vocal students who participate in public live Q&A.

5. DISCUSSION

Labeling as a lens on medium differences. Our framework reveals systematic shifts in the types of uncertainty students externalize. The synchronous TA channel contained a larger share of logistical and analytical questions. In contrast, the asynchronous AI channel contained more definition-oriented and meta questions, along with a higher share of high-confidence, confirmation-style questions. One possible interpretation is that students sometimes used the asynchronous tool to validate their understanding of lecture

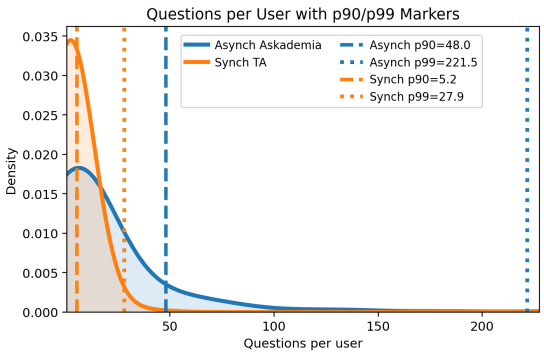


Figure 4: Distribution of the number of questions per user.

concepts rather than only to seek first-pass explanations. Reasoning explicitness remained low in both settings, suggesting that the two media differed more in what students asked than in how they articulated their reasoning. These label distributions suggest that the medium may shape which types of questions are easier to ask, socially appropriate, and likely to receive a useful response.

Participation and timing. Beyond content, the two media differed in scale and participation. The asynchronous medium received substantially more questions per lecture, showed stronger temporal variation, and was used repeatedly by a broader set of students. In contrast, the synchronous TA channel was more concentrated among a smaller group of frequent question askers. While we cannot attribute these differences to a single cause, they are consistent with medium affordances that may reduce barriers to asking, alongside other possible influences such as self-selection into support channels, differences in student expectations, and variation in when students seek help. Though we did not directly test this, shorter asynchronous questions may also align with students using the tool for brief, iterative clarification rather than composing longer, standalone questions. Similarly, some asynchronous questions may have been asked during review rather than first exposure, whereas synchronous questions were asked while students encountered the material live. If so, some observations may reflect not only channel structure but also differences in when students sought help.

Fixed-time instruction, variable-time learning. "Paused" conversation patterns, "rewind" questions, and a higher rate of multi-turn interaction in the asynchronous setting highlight a mode of help-seeking that is difficult to realize in live lecture Q&A. Students appear to pause at specific moments, ask clusters of related questions, and iteratively refine understanding before proceeding. They also sometimes return to earlier lecture moments after moving later in the recording. Unlike the synchronous setting, where questions must be posed under real-time lecture constraints, the asynchronous medium decouples help-seeking from the live flow of instruction. Students can spend variable time at moments of confusion while the instructional timeline remains fixed in the background. This supports a pause-and-repair workflow without the pacing costs of interrupting a live lecture, which may contribute to both longer conversation structure and higher overall demand.

These results motivate treating Q&A channels as part of the instructional environment rather than interchangeable interfaces. Overall, the medium structure was associated with differences in what students asked, who participated, and whether help-seeking took the form of a one-shot request or a more iterative process. Future work should test the generalizability of these patterns and connect medium-specific traces, including confirmation-heavy and "paused" usage, to downstream measures of engagement and learning.

Limitations. Our analysis is limited by the covariates available in the dataset. We did not have access to student-level background measures such as prior GPA, demographics, attendance mode, or prior usage history, nor to course-level covariates such as lecture difficulty or assessment timing. As a result, we could not adjust for potentially important differences in which students used each support channel, and unmodeled confounders, including self-selection and differences in when students chose to seek help, may also contribute to the patterns we observe.

More fundamentally, our comparison does not isolate the effect of any single factor. The two channels differ simultaneously along structural dimensions such as synchronicity, privacy, responder identity, and lecture-viewing context, as well as harder-to-measure sociotechnical ones such as students' familiarity with and acceptance of AI tools, their expectations about response quality and authority, their perceptions of the AI's institutional role, and the social costs of asking. The observed differences may therefore reflect a bundle of channel affordances, interaction norms, and user expectations rather than the independent effect of the interaction medium alone. Our findings are also drawn from two computing courses in one institutional context, which may limit generalizability. Future work with richer student- and course-level data, and designs that vary these features more independently, could better disentangle these explanations.

6. CONCLUSION

Our findings suggest that the transition from synchronous, human-led Q&A to asynchronous, AI-driven support is associated with a shift from fixed-time/variable-learning to flexible-time/fixed-learning. In traditional synchronous lectures, the rigid temporal structure may function as a bottleneck, placing students in a competitive, public environment where learning remains "variable" for those who cannot keep pace with the live stream. In contrast, the asynchronous medium supports "flexible time" when inquiry is decoupled from the lecture's live delivery, allowing students to treat conceptual mastery as a "fixed" goal. The asynchronous medium is associated with students engaging in denser, more iterative, and definition-oriented dialogues, which may reflect lower social costs and fewer time constraints. These patterns suggest that medium structure may support a more equitable instructional model, in which time spent is the variable and depth of understanding is the constant.

7. REFERENCES

[1] H. A. Alawwad, A. Alhothali, U. Naseem, A. Alkhathlan, and A. Jamal. Enhancing textbook question answering task with large language models and retrieval augmented generation, 2024.

[2] A. Ali. Using padlet as a pedagogical tool. *Journal of Learning Development in Higher Education*, 2021.

[3] J.-M. Campillo-Ferrer, P. Miralles-Martínez, and R. Sánchez-Ibáñez. Gamification in higher education: Impact on student motivation and the acquisition of social and civic key competencies. *Sustainability*, 2020.

[4] Y. F. Cheong, F. Pajares, and P. S. Oberman. Motivation and academic help-seeking in high school computer science. *Computer Science Education*, 2004.

[5] W. Gan, Z. Qi, J. Wu, and J. C.-W. Lin. Large language models in education: Vision and opportunities. In *2023 IEEE International Conference on Big Data (BigData)*, 2023.

[6] D. Garcia, A. Fox, S. Russell, E. Ambrosio, N. Terrell, M. Silva, M. West, C. Zilles, and F. Shakir. A's for all (as time and interest allow). In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, volume 1, 2023.

[7] D. Garcia, C. McMahon, Y. Garcia, M. West, and C. B. Zilles. Achieving "a's for all (as time and interest allow)". In *Proceedings of the Ninth ACM Conference on Learning @ Scale*, 2022.

[8] D. Garcia, C. McMahon, Y. Garcia, M. West, and C. B. Zilles. Software support for "a's for all". In *Proceedings of the Ninth ACM Conference on Learning @ Scale*, 2022.

[9] Y. Hicke, A. Agarwal, Q. Ma, and P. Denny. Ai-ta: Towards an intelligent question-answer teaching assistant using open-source llms, 2023.

[10] M. F. Ishmam, M. S. H. Shovon, M. F. Mridha, and N. Dey. From image to language: A critical analysis of visual question answering (vqa) approaches, challenges, and opportunities. *Information Fusion*, 106, 2024.

[11] M. Kazemitabaar, R. Ye, X. Wang, A. Z. Henley, P. Denny, M. Craig, and T. Grossman. Codeaid: Evaluating a classroom deployment of an llm-based programming assistant that balances student and educator needs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 2024.

[12] H. Kumar, M. Reza, J. Mitchell, I. Musabirov, L. Zhang, and M. Liut. Understanding help-seeking behavior of students using llms vs. web search for writing sql queries, 2024.

[13] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[14] R. Liu, C. Zenke, C. Liu, A. Holmes, P. Thornton, and D. J. Malan. Teaching cs50 with ai: Leveraging generative artificial intelligence in computer science education. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, volume 1, 2024.

[15] T. M. MacWilliam and D. J. Malan. Scaling office hours: managing live q&a in large courses. *Journal of Computing Sciences in Colleges*, 28, 2013.

[16] C. Mahipal, L. Angrave, Y. Xie, B. Chatterjee, H. Wang, and Z. Qian. "what did i just miss?!" presenting classtranscribe, an automated live-captioning and text-searchable lecture video system, and related pedagogical best practices. In *Proceedings of the 2019 ASEE Annual Conference & Exposition*, 2019.

[17] S. Mirhosseini, A. Z. Henley, and C. Parnin. What is your biggest pain point?: An investigation of cs instructor obstacles, workarounds, and desires. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, volume 1, 2023.

[18] C. Mitra, M. Miroyan, R. Jain, V. Kumud, G. Ranade, and N. Norouzi. Retllm-e: Retrieval-prompt strategy for question-answering on student discussion forums. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024.

[19] M. Mittal, A. Bailey, V. Phelps, M. Miroyan, C. Mitra, R. Jain, R. Niousha, G. Ranade, and N. Norouzi. Raising the bar: Automating consistent and equitable student support with llms. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2*, volume 2, 2025.

[20] M. Mittal, G. Tyagi, A. Bailey, G. Ranade, and N. Norouzi. Askademia: A real-time ai system for automatic responses to student questions. In *Artificial Intelligence in Education*, 2025.

[21] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, et al. Gpt-4 technical report, 2023.

[22] J. M. Pérez, M. Rajngewerc, J. C. Giudici, D. A. Furman, F. Luque, L. Alonso Alemany, and M. V. Martínez. pysentimiento: A python toolkit for opinion mining and social nlp tasks, 2021.

[23] L. Qi, J. D. Zamfirescu-Pereira, T. Kim, B. Hartmann, J. DeNero, and N. Norouzi. A knowledge-component-based methodology for evaluating ai assistants, 2024.

[24] N. Shvetsova, A. Kukleva, X. Hong, C. Rupprecht, B. Schiele, and H. Kuehne. Howtocaption: Prompting llms to transform video annotations at scale. In *Computer Vision – ECCV 2024*, 2024.

[25] S. Uppal, S. Bhagat, D. Hazarika, N. Majumder, S. Poria, R. Zimmermann, and A. Zadeh. Multimodal research in vision and language: A review of current and emerging trends. *Information Fusion*, 77, 2022.

[26] M. Wang, Y. Wang, T.-T. Vu, E. Shareghi, and G. Haffari. Exploring the potential of multimodal llm with knowledge-intensive multimodal asr. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

[27] S. Wang, T. Xu, H. Li, C. Zhang, J. Liang, J. Tang, P. S. Yu, and Q. Wen. Large language models for education: A survey and outlook, 2024.

[28] J. D. Zamfirescu-Pereira, L. Qi, B. Hartmann, J. DeNero, and N. Norouzi. 61a bot report: Ai assistants in cs1 save students homework time and reduce demands on staff. (now what?). In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*, volume 1, 2025.

To What Extent Can Early Keystroke Patterns Predict Knowledge Gain in Mathematics

Oscar Blessed Deho
Centre for Change and Complexity in Learning (C3L)
Adelaide University
Australia
oscar.deho@adelaide.edu.au

Ryan S. Baker
Centre for Change and Complexity in Learning (C3L)
Adelaide University
Australia
ryan.baker@adelaide.edu.au

Jiayi Zhang
Penn Center for Learning Analytics
University of Pennsylvania
United States
joycez@upenn.edu

Digory Smith
Eedi
United Kingdom
digory.smith@eedi.com

Simon Woodhead
Eedi
United Kingdom
simon.woodhead@eedi.com

ABSTRACT

Early identification of students who may struggle to learn from practice could enable timely instructional support. This study investigates whether keystroke dynamics during initial problem-solving attempts can signal subsequent knowledge gain in mathematics. We analyzed keystroke patterns from students' first two problems following instruction on a skill, examining associations with knowledge gain operationalized through Bayesian Knowledge Tracing. Students who ultimately learned more exhibited systematically different early behaviors: slower, more variable typing with greater revision activity. Effect sizes ranged from small to moderate (Cliff's $\delta = 0.15$–0.38), with timing features showing strongest associations. Machine learning models achieved modest but above-chance predictive accuracy (regression: RMSE = 0.36, $r = 0.42$; classification: AUC-ROC = 0.68, balanced accuracy = 0.67). While individual keystroke features explained limited variance (3.6–11.6%), results suggest that fine-grained behavioral traces from students' earliest practice attempts may provide meaningful early signals about learning trajectories, potentially enabling proactive instructional adaptation before patterns of persistent struggle emerge.

Keywords

Keystroke dynamics, Knowledge gain, Early prediction

1. INTRODUCTION

Intelligent tutoring systems (ITSs) have demonstrated the potential to significantly improve student learning outcomes [14]. By personalizing instruction and feedback, these systems can help learners acquire new knowledge and skills more effectively than traditional one-size-fits-all instruction. However, an important goal for such systems is not merely to optimize immediate performance (e.g., getting questions correct), but to ensure that students are actually learning—making durable knowledge gains [13]. In practice, a correct answer does not always equate to genuine understanding. For instance, some learners might arrive at correct answers by rote application of superficial strategies, such as relying heavily on hints, pattern matching, or completing problems through trial and error, without developing a deeper understanding of the underlying concepts or procedures. Such behavior may lead to high performance in the short term while yielding smaller gains in robust, long-term learning. This contrast between performance and learning highlights the need for adaptive platforms that focus on cultivating knowledge growth (e.g., long-term retention and transfer; [13]), rather than merely supporting immediate problem correctness.

A related challenge for intelligent tutoring systems concerns when different kinds of instructional decisions should be made. Most ITSs already provide rapid, fine-grained support at the inner-loop level—for example, responding immediately to an incorrect step with hints or feedback [17]. This form of in-the-moment guidance is often effective, but it is not always sufficient. Some students fail to make substantive progress despite repeated inner-loop assistance, resulting in wheel-spinning, where they continue working over a substantial time without moving closer to mastery [4]. Addressing such situations can require intervention by a teacher or outer-loop decisions by the learning environment [17], such as providing support with prerequisite skills. In most existing systems, however, these types of interventions only occur when the instructor realizes that a student has been struggling for quite some time. Although prior work has explored early prediction of eventual wheel-spinning [15, 5, 18], these predictions still rely on several problems of student interaction. This raises a natural question: can the need for deeper support be inferred even earlier—during a student's initial encounters with a skill? If so, an ITS could begin adapting instruction not after prolonged difficulty, but from the very beginning of work on the skill.

Early prediction of learning outcomes requires access to in-

Oscar Deho, Ryan Baker, Digory Smith, Jiayi Zhang, and Simon Woodhead. To What Extent Can Early Keystroke Patterns Predict Knowledge Gain in Mathematics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 571–577. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039945

formation that reflects how students engage with tasks as learning unfolds. Beyond final answers, fine-grained interaction traces—such as timing, attempts, and patterns of interaction—can reveal how a student is approaching a problem in real time [12]. Keystroke-level information provides even finer-grained detail, capturing the exact sequence and timing of keys pressed as a student constructs a response. This level of detail has been argued to shed light on the cognitive and metacognitive processes involved in problem solving [9]. For example, long pauses before initiating a response may reflect planning or recall, while rapid sequences of edits and deletions may indicate uncertainty or guess-and-revise strategies. In this sense, how a student types may provide insight into hesitation, confidence, and strategy use—factors that are closely related to learning [1, 16].

Building on this motivation, prior research in educational data mining has explored early prediction of learning outcomes using students' initial interactions [4, 6, 15, 5, 18]. These studies draw on data such as keystrokes, clickstream logs, and tutor transaction sequences to anticipate short- and long-term outcomes. For instance, Beck and Gong's work on wheel-spinning examined a student's first few attempts on a new skill to identify students unlikely to attain mastery, flagging those at risk of continuous failure and prompting possible changes in learning support. Other studies have focused on course-level or test outcomes. Casey [6] developed classifiers to detect at-risk programming students early in the semester; the inclusion of keystroke features improved the model's ability to predict which students would not pass the course. Conijn et al. [7] examined keystroke logs from the beginnings of writing assignments to forecast final essay quality. While their predictive models only modestly outperformed baselines, they demonstrated the viability of using typing process data to identify students who might need writing support before the draft is completed. Researchers have also shown that even a short window of student activity can yield insight into far-future outcomes. Gao et al. [10] found that data from the first 2–5 hours of student usage of an educational game and an ITS provided a valuable signal about students' performance on end-of-year assessments months later. In their study, a machine learning model using only the initial few hours of log data was able to predict year-end test scores with notable accuracy, indicating that early behaviors encapsulate information about eventual learning results. Together, this body of literature illustrates the promise of early prediction: from fine-grained keystroke patterns to aggregated usage metrics, a range of early indicators have been leveraged to forecast outcomes such as course failure, essay scores, and standardized test performance.

While prior studies demonstrate that early interaction data, including keystrokes, can be informative for predicting later performance or task outcomes, they offer limited insight into whether such early behaviors relate to subsequent knowledge gain during skill practice. Work incorporating keystroke features has examined outcomes such as essay quality, programming performance, or mastery likelihood, rather than learning improvement across a sequence of problems. Consequently, it remains unclear whether keystroke behaviors observed during students' initial encounters with a new skill are related to how much they ultimately learn through continued practice. This gap matters because proactive support requires identifying not only students who struggle to answer correctly, but also those who make limited learning gains even when their early answers are correct.

To fill this gap, we address two research questions. First, how are students' early keystroke patterns during their initial encounters with a skill associated with subsequent knowledge gain? Second, to what extent can early keystroke patterns predict students' eventual knowledge gain on that skill? To investigate these questions, we analyze students' keystroke patterns from the first two problems of a skill and relate them to subsequent knowledge gain. Using Bayesian Knowledge Tracing (BKT), we operationalize knowledge gain as the difference between the student's BKT-estimated mastery level at the start and end of practice on a given skill. We employ both statistical analyses and machine learning approaches. We examine group differences in early keystroke features between high-gain and low-gain students, then train predictive models to assess whether early keystroke features contain a meaningful signal about students' eventual knowledge gain. One modeling task estimates the amount of learning a student will achieve, while a complementary classification task evaluates whether these early behaviors can distinguish students who ultimately show higher versus lower knowledge gains.

2. METHODS

2.1 Data

2.1.1 Source and Context

This study used de-identified data from an online mathematics learning platform with adaptive practice. Each topic begins with a short diagnostic assessment, after which students receive targeted instruction through worked examples and videos before entering the worksheet practice phase. We focus on this worksheet phase, where students independently solve problems using open-ended typed responses that generate keystroke sequences and timing information. All personally identifiable information was removed prior to data access. The final dataset comprised 19,155 fully matched student–question transactions, each representing a single practice problem.

2.1.2 Correctness Labelling

The worksheet data did not include correctness labels. To generate these at scale, we developed and validated an automated correctness labeler using GPT-4o, prompted to assess whether a student's response was mathematically equivalent to an expert reference answer under explicit equivalence rules (e.g., commutativity, numerical approximation, and formatting variations). Ground truth was established via double human annotation of 200 responses, yielding high inter-rater reliability ($\kappa = 0.95$); disagreements were resolved through social moderation. The automated labeler closely matched human judgments ($\kappa = 0.95$, precision = 0.97, recall = 0.95) and was subsequently used to label all 19,155 responses.

2.1.3 Keystroke Data and Feature Engineering

Each worksheet question answered generated a sequence of keystrokes and associated metadata, including information

Table 1: Example of selected columns from a single row of the raw data (← = left-arrow cursor movement; BKSP = Backspace).

Question	Input	Keystroke	Reference	Time(s)
2548 − 362	2186	← 218y BKSP 6	2186	167

Table 2: Features and their definitions.

Feature	Definition
Keystroke edit distance	Levenshtein distance between the full raw keystroke sequence and the final submitted answer.
Seconds per keystroke	Total time spent on the response divided by the total number of keystrokes.
Keystroke entropy	Shannon entropy of the keystroke sequence.
Correction density	Number of correction keys (Backspace, Delete) divided by log-transformed total time.
Correction proportion	Number of correction keys divided by total keystrokes.
Keystroke efficiency ratio	Final answer length divided by total keystrokes.
Average seconds between keys	Mean inter-keystroke time (seconds) for a question attempt.
Variation in seconds between keys	Standard deviation of inter-keystroke times (seconds) for a question attempt.

about time spent, editing actions, and overall input behavior. Table 1 illustrates an example of a single row of data showing the information captured for a student's response to a worksheet question. From these data, a set of features was derived to capture temporal, corrective, and compositional aspects of students' typing activity. Table 2 presents a detailed summary of all engineered features.

2.2 Analysis

2.2.1 Deriving Knowledge Gain from BKT

To quantify learning within each skill, we applied Bayesian Knowledge Tracing (BKT) [8] to students' sequential response data. BKT models student learning as a dynamic process within a Hidden Markov Model framework, where each student transitions between two latent states: not mastered and mastered across successive items answered within the same skill. While prior work has noted limitations in BKT's accuracy at predicting subsequent performance [11], we adopted BKT not for predictive accuracy but as a relative indicator of student mastery based on prior performance. For this purpose, it is useful that BKT provides a situation-independent estimate of mastery that does not incorporate item difficulty or information from other skills, unlike more recent knowledge tracing models. BKT's ability to account for guessing and slipping further supports its use for mastery estimation in this context. BKT parameters were fit using a Brute Force/Grid Search algorithm [3], with a minimum value of 0.01 applied to all probabilities and a maximum value of 0.3 imposed on guess and slip parameters to prevent model degeneracy [2]. The resulting parameter estimates were used to compute predicted probabilities of mastery for each student after each question. For each student–skill pair, we extracted the probability of mastery before any question was answered, $P(L_0)$, and after the final question, $P(L_T)$; knowledge gain was then defined as $P(L_T) - P(L_0)$. Positive values indicate increased mastery across practice, while negative values reflect performance below the initial estimate. To focus on skills where meaningful learning occurred, we retained only skills with an average gain greater than 0.1. These knowledge gain values served as the target variable for all subsequent analyses.

2.2.2 Data Preparation for Analysis

After retaining only skills with average knowledge gain above 0.1, the dataset was reduced from 19,155 to 1,212 transactions. To capture early engagement, predictors were computed from the first two questions per skill by averaging numeric features across those attempts. This two-question window balances early identification with robustness, avoiding reliance on a single initial response that may reflect task familiarization rather than learning difficulty. Applying this restriction yielded a final dataset of 232 student–skill records from 190 (unique) students across seven skills. The dataset includes all keystroke features in Table 2 and a continuous BKT-based knowledge gain label, which ranged from -0.41 to 0.99 ($M = 0.19$, $SD = 0.40$). For classification, 53% of observations fell above the median knowledge gain and 47% below, which we treat as proxies for successful and unsuccessful learning.

2.2.3 Exploratory Analysis

To investigate the relationship between early keystroke features and subsequent knowledge gain, we conducted a set of exploratory statistical analyses to characterize the nature and direction of these relationships. We tested for significant differences between students categorized as successful learners (knowledge gain at or above the skill-specific median) and unsuccessful learners (knowledge gain below the skill-specific median) using the Mann–Whitney U test. p-values were corrected for multiple comparisons using the Benjamini-Hochberg false discovery rate (FDR) procedure, and effect sizes were computed using Cliff's Delta. Following this group comparison, continuous associations between each keystroke feature and numerical knowledge gain were assessed using Spearman's rank correlation. To further explore these relationships, the data were divided into four quartiles of knowledge gain (Q1–Q4), allowing inspection of median keystroke feature values within each quartile and revealing progressive trends as knowledge gain increased. These analyses characterize individual feature–outcome relationships, while Section 2.2.4 evaluates their combined predictive power.

2.2.4 Predictive Modelling

To evaluate the predictive power of students' early keystroke patterns, we conducted two predictive modeling tasks: a regression task to predict the magnitude of knowledge gain and a classification task to identify whether learning outcomes were successful. All models were evaluated using five-fold cross-validation, stratified at the student level to ensure that data from a given student did not appear in both training and test sets within a fold. This approach prevents data leakage and supports generalization to un-

Table 3: Comparison of keystroke features between successful (S) and unsuccessful (U) learning groups via Mann-Whitney U test (group alpha = 0.05)

Feature	S_{Mdn}	U_{Mdn}	δ	p-value
average seconds between keys	1.15	0.62	**0.38**	< 0.001
variation in seconds between keys	1.02	0.40	**0.34**	< 0.001
seconds per keystroke	5.03	2.78	**0.33**	< 0.001
keystroke edit distance	1.50	1.00	0.27	0.001
correction density	0.00	0.00	0.22	0.001
correction proportion	0.00	0.00	0.20	0.002
keystroke efficiency ratio	0.71	0.80	-0.19	0.015
keystroke entropy	2.00	1.91	0.15	0.042

Table 4: Spearman correlations (ρ) between keystroke features and knowledge gain (group alpha = 0.05)

Feature	ρ	p-value
average seconds between keys	**0.34**	< 0.001
variation in seconds between keys	0.30	< 0.001
seconds per keystroke	0.30	< 0.001
correction density	0.25	< 0.001
keystroke edit distance	0.24	< 0.001
correction proportion	0.23	< 0.001
keystroke entropy	0.19	0.005
keystroke efficiency ratio	-0.15	0.020

Table 5: Median feature values across quartiles of knowledge gain. Q1= Lowest, Q4= Highest

Feature	Q1	Q2	Q3	Q4
average seconds between keys	0.58	0.66	1.08	1.16
keystroke edit distance	1.00	1.50	1.00	2.00
seconds per keystroke	2.78	2.68	4.87	6.16
keystroke entropy	1.91	1.79	1.95	2.20
correction density	0.00	0.00	0.00	0.13
correction proportion	0.00	0.00	0.00	0.04
variation in seconds between keys	0.39	0.50	0.85	1.23
keystroke efficiency ratio	0.80	0.73	0.76	0.62

seen students. Within each fold, hyperparameter optimization was performed on the training set to identify the best-performing configuration for each model.

The regression task aimed to predict the continuous value of a student's knowledge gain from early keystroke features, serving an exploratory purpose by modeling the relationship between keystroke behavior and learning magnitude. The evaluated models included Random Forest Regressor, Extra Trees Regressor, Decision Tree Regressor, Linear Regression, Ridge Regression, and M5Prime. Model performance was evaluated using linear correlation and root mean squared error (RMSE), capturing both the strength of association between predicted and observed knowledge gain and the magnitude of prediction error. All reported results were averaged across cross-validation folds. The classification task predicted whether a student's knowledge gain was above or below the median for a given skill. This task evaluates the practical diagnostic utility of early keystroke features for distinguishing between high-gain and low-gain learners, with implications for targeted intervention. The target variable was binarized relative to the median per skill to account for differences in difficulty across skills. Models evaluated included Logistic Regression, Decision Tree, Random Forest, RIPPER, and K*. Performance was measured using the Area Under the Receiver Operating Characteristic curve (AUC-ROC), the Area Under the Precision-Recall curve (AUC-PR), F1-score, standard accuracy (ACC), and balanced accuracy (BACC).

3. RESULTS

3.1 How is Early Keystroke Patterns associated with Knowledge Gain?

To understand the relationship between early keystroke features and knowledge gain, we began by comparing students classified as successful learners with their unsuccessful peers. The Mann–Whitney U test (with Benjamini–Hochberg correction) identified significant differences for all eight features (Table 3). Effect sizes (Cliff's Delta) ranged from 0.15 to 0.38, indicating small to moderate group differences. The largest effects were found for *average seconds between keys*, *variation in seconds between keys*, and *seconds per keystroke*. For these latency-based features, the effect size range ($\delta = 0.33$–0.38) corresponds to approximately a 67% to 69% probability—based on the standard Cliff's Delta derivation—that a randomly selected student from the successful learning group would exhibit a higher feature value than one from the unsuccessful group. Again, as shown in Table 3, students with higher knowledge gains consistently showed higher median values for features related to latency, variability, and editing (e.g., *average seconds between keys*, *keystroke edit distance*), suggesting that slower, more variable, and more corrective early typing behaviors are associated with successful learning. Corresponding findings were obtained when examining continuous associations. Spearman's rank correlations were significant and positive for all features, with coefficients (ρ) ranging from 0.19 to 0.34 (Table 4). This indicates that individual keystroke features explained between 3.6% and 11.6% (ρ^2) of the variance in knowledge gain ranks, with latency-related features again showing the strongest associations. This general upward trend was also visible in the quantile analysis (Table 5), which showed the median values for features such as *average seconds between keys* increasing from the lowest quartile (Q1) to the highest quartile (Q4). These statistical analyses indicate that significant relationships exist between early keystroke features and knowledge gain, motivating the subsequent investigation into their combined predictive power.

3.2 To what extent can we predict Knowledge Gain from Early Keystroke Patterns?

Following the statistical analysis, we evaluated the combined predictive power of the early keystroke features using machine learning models. For both the regression and classification tasks, a suite of different models was tested (see Tables 6 and 7). Although the primary goal of this study is exploratory, the top-performing model is conventionally selected during model selection. We therefore focus our reporting on the performance of the best performing model for each task in terms of the each evaluation metrics used. In the regression task (Table 6), the Random Forest Regressor exhibited the strongest performance among the evaluated models. It achieved the lowest root mean squared error (RMSE = 0.36). We normalized the RMSE using the observed range of the knowledge gain variable (−0.41 to 0.99). After normalization, the RMSE corresponds to 26% of this range. This means that, on average, the model's predictions are off by about one quarter of the total spread of knowledge gain values in the data. The same model also produced the highest correlation between predicted and observed knowledge gains ($r = 0.42$). For the classification task (Table 7), the Random Forest classifier was the top-performing model. It achieved an AUC-ROC of 0.68, indicating that—based only on keystroke patterns from the first two questions—the model has a 68% chance of correctly distinguishing between a randomly chosen eventual successful learner and a randomly chosen unsuccessful learner. The model also achieved an F1-score of 0.71, indicating a strong balance between precision and recall, and a balanced accuracy of 0.67.

Table 6: Regression results for predicting knowledge gain

Model	RMSE	r
Random Forest Regressor	**0.36**	**0.42**
Extra Trees Regressor	0.37	0.40
Decision Tree Regressor	0.39	0.29
Linear Regression	0.40	0.25
Ridge Regression	0.39	0.25
M5prime	0.38	0.37

Table 7: Classification results for predicting above/below median knowledge gain

Model	ROC	PR	F1	BACC	ACC
Logistic Regression	0.63	0.64	0.60	0.61	0.61
Decision Tree	0.66	0.63	0.71	0.66	0.67
Random Forest	**0.68**	**0.67**	**0.71**	**0.67**	**0.67**
RIPPER	0.49	0.53	0.26	0.51	0.49
K*	0.63	0.64	0.63	0.60	0.60

4. DISCUSSION

4.1 Association Between Early Keystroke Patterns and Knowledge Gain

Our results indicate that students who achieved higher knowledge gains engaged with the first two post-instruction mathematics problems in ways that differed systematically from their lower-gaining peers. Specifically, higher-gain students exhibited slower response construction (longer average seconds between keys, longer seconds per keystroke), greater variability in pacing (higher variation in seconds between keys), and more evidence of answer modification and revision (higher keystroke edit distance, correction density, and correction proportion). The observed effect sizes were small to moderate (Cliff's $\delta = 0.15$–0.38), with the strongest associations found for latency-based features. These patterns were consistent across multiple analytical approaches: group comparisons, correlation analyses, and quartile examinations all converged on the finding that slower, more variable, and more corrective early typing behaviors were associated with greater subsequent knowledge gain.

These findings suggest—though do not definitively establish—that students who achieved higher knowledge gains exhibited different behavioral patterns during their first two practice problems compared to lower-gaining students. One plausible interpretation is that slower, more variable typing with greater revision activity reflects more effortful cognitive processing. In mathematics problem solving, pauses between keystrokes could indicate time spent on mental calculation, deliberation about solution strategies, or verification of intermediate steps. Similarly, greater editing activity and a lower ratio of final answer length to total keystrokes may indicate that students reconsidered their initial responses rather than committing to their first attempt.

Notwithstanding, alternative interpretations must be considered. Slower, more variable keystroke patterns could also arise from factors unrelated to productive cognitive engagement, such as lower typing proficiency, confusion or uncertainty that may not always lead to learning, or individual differences in problem-solving tempo. Our analyses cannot distinguish whether these keystroke patterns reflect productive struggle that facilitates learning or whether both the keystroke patterns and subsequent learning are independently driven by underlying factors such as prior knowledge, metacognitive skills, or motivation. Students who already possess stronger foundational knowledge or better self-regulation may naturally exhibit more deliberate, revisionary behavior while also being better positioned to learn from practice. Without experimental manipulation, the observed associations remain correlational.

Moreover, while the correlations we observed were statistically significant, individual keystroke features explained only modest proportions of the variance in knowledge gain (3.6%–11.6%), suggesting that these behavioral signals capture some, but not all, of the factors that differentiate higher- and lower-gaining students. The moderate effect sizes similarly indicate that group differences, while consistent, are not absolute: there is substantial overlap in the keystroke distributions of successful and unsuccessful learners. This overlap suggests that keystroke patterns alone provide an incomplete picture of the learning process.

4.2 Predictive Value of Early Keystroke Patterns

The machine learning analyses provide additional perspective on the practical utility of early keystroke features for anticipating learning outcomes. In the regression task, the best-performing model (Random Forest) achieved an normalized RMSE corresponding to approximately 26% of the observed range of knowledge gain values and a correlation of $r = 0.42$ between predicted and actual gains. In the classification task, the Random Forest classifier distinguished between above- and below-median learners with an AUC-

ROC of 0.68, balanced accuracy of 0.67, and F1-score of 0.71. These results indicate that early keystroke patterns contain a detectable signal about subsequent knowledge gain, but the predictive accuracy is far from perfect.

To contextualize these findings, it is useful to consider what they mean in practical terms. An AUC-ROC of 0.68 indicates that, given two randomly selected students—one who will ultimately show higher knowledge gain and one who will show lower gain—the model has a 68% chance of correctly identifying which is which based solely on their first two problems' keystroke patterns. While this exceeds chance performance (50%), it also means the model will make incorrect orderings approximately one-third of the time. Similarly, the regression model's prediction error of 26% of the knowledge gain range suggests that individual predictions may deviate substantially from actual outcomes in many cases. The classification model's balanced accuracy of 0.67 further indicates that roughly two-thirds of students are classified correctly, leaving one-third misclassified. These levels of predictive performance are comparable to those reported in related early prediction tasks in educational contexts. Prior work predicting wheel-spinning from early problem attempts [4] and forecasting course outcomes from initial interaction data [6, 10] has similarly shown modest but above-chance accuracy. The present findings extend this pattern to the specific case of predicting knowledge gain during skill practice, confirming that early behavioral signals are informative but not deterministic of learning outcomes.

4.3 Implications for Adaptive Learning Systems

Despite the modest predictive accuracy, our findings may have practical value for adaptive instructional systems, particularly when considered in light of the costs and benefits of different types of errors. Prior work in educational data mining has largely focused on detecting sustained unproductive patterns, such as wheel-spinning, where students fail to progress toward mastery despite extended practice [4, 15]. These detection systems typically operate after students have completed many problems, by which point valuable learning time may have been lost. The present results suggest that behavioral differences between higher- and lower-gaining students may be detectable much earlier—specifically, within the first two problems following instruction, before extended practice has occurred. This temporal advantage creates potential opportunities for earlier adaptation, though several important caveats apply. First, the utility of early prediction depends on the availability of effective interventions. If an adaptive system identifies a student as being likely to make limited knowledge gains based on rapid keystroke patterns with minimal revision, the appropriate response is not obvious. Prompting such students to slow down, explain their reasoning, or verify their work might be beneficial if their direct response construction reflects superficial engagement, but could be counterproductive if it indicates genuine fluency or if the additional cognitive load interferes with their learning process. Conversely, students showing slower, more variable patterns with substantial revision might benefit from encouragement or targeted scaffolding, but only if their behavior reflects struggle rather than successful work that happens to be more deliberate.

Furthermore, the modest predictive accuracy means that system-level decisions based on these features should be designed to accommodate uncertainty. Rather than triggering interventions based on early keystroke patterns alone, adaptive systems might use these signals as one input among many—combined with correctness, prior performance, time on task, and other indicators—to inform more nuanced responses. For example, a system might use early keystroke patterns to adjust the probability threshold for offering optional support rather than relying on rigid decision rules. Alternatively, these patterns could inform the sequencing or selection of subsequent problems, steering students toward items that provide appropriate challenge based on estimated learning trajectories.

4.4 Contribution and Open Questions

From a theoretical perspective, our findings contribute to an emerging body of work suggesting that fine-grained behavioral traces, such as keystrokes, can provide insight into learning as it unfolds [1, 6]. The observed associations between early keystroke features and subsequent knowledge gain are consistent with the idea that behaviors reflecting deliberation, self-monitoring, and revision may be detectable in real time and may relate to learning outcomes. However, several important questions remain unresolved. First, the causal mechanisms underlying the observed associations are not established. It remains unclear whether the keystroke patterns themselves reflect cognitive processes that causally contribute to learning, or whether both keystroke behavior and learning outcomes are independently influenced by factors such as prior knowledge, motivation, or self-regulation. Second, the specific cognitive processes that give rise to different keystroke patterns remain inferential. While slower, more variable typing has been interpreted here as potentially reflecting planning or checking, alternative explanations such as retrieval difficulty or hesitation are also plausible. Finally, the temporal dynamics of keystroke behavior and learning are not well understood. Keystroke patterns may evolve across practice, and patterns later in learning may provide different or stronger signals than those observed at the outset. Longitudinal analyses could help clarify how keystroke behavior changes over time and how these changes relate to knowledge gain trajectories.

4.5 Limitations

Several limitations should be noted. First, the data were drawn from a single online mathematics platform, which may limit generalizability. Second, knowledge gain was operationalized using Bayesian Knowledge Tracing estimates, which are model-based inferences rather than direct measures of learning. While BKT is widely used and validated [11], measurement error may attenuate observed associations. Third, the analysis focused on a specific set of keystroke features. Other features capturing different aspects of response construction may provide additional information. Finally, this study examined association and prediction rather than intervention. Experimental studies are needed to determine whether instructional adaptations informed by early keystroke patterns can causally improve learning outcomes.

4.6 Conclusion

In summary, our findings show that early keystroke patterns during the first two problems of skill practice are as-

sociated with subsequent knowledge gain in mathematics, with higher-gaining students exhibiting slower, more variable, and more revision-oriented answer construction. Machine learning models trained on these early features achieved modest but above-chance accuracy in predicting learning outcomes. While these results are consistent with theoretical accounts emphasizing the distinction between performance and learning, they also highlight important limitations and uncertainties. Nevertheless, the findings suggest that fine-grained behavioral data from the earliest moments of independent practice may offer valuable information for understanding and supporting learning trajectories, motivating future work on causal mechanisms, intervention design, and generalizability across domains and contexts.

4.7 Acknowledgments

This work was supported by the Learning Engineering Virtual Institute (LEVI)

5. REFERENCES

[1] V. M. Baaijen, D. Galbraith, and K. de Glopper. Keystroke analysis: Reflections on procedures and measures. *Written Communication*, 29(3):246–277, 2012.

[2] R. S. D. Baker, A. T. Corbett, and V. Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian knowledge tracing. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 406–415, Berlin, Heidelberg, 2008. Springer.

[3] R. S. D. Baker, A. T. Corbett, S. M. Gowda, A. Z. Wagner, B. A. MacLaren, L. R. Kauffman, A. P. Mitchell, and S. Giguere. Contextual slip and prediction of student performance after use of an intelligent tutor. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*, pages 52–63, Berlin, Heidelberg, 2010. Springer.

[4] J. E. Beck and Y. Gong. Wheel-spinning: Students who fail to master a skill. In *International conference on artificial intelligence in education*, pages 431–440. Springer, 2013.

[5] A. F. Botelho, A. Varatharaj, T. Patikorn, D. Doherty, S. A. Adjei, and J. E. Beck. Developing early detectors of student attrition and wheel spinning using deep learning. *IEEE Transactions on Learning Technologies*, 12(2):158–170, 2019.

[6] K. Casey. Using keystroke analytics to improve pass–fail classifiers. *Journal of Learning Analytics*, 4(2):189–211, 2017.

[7] R. Conijn, C. Cook, M. van Zaanen, and L. Van Waes. Early prediction of writing quality using keystroke logging. *International Journal of Artificial Intelligence in Education*, 32(4):835–866, 2022.

[8] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995.

[9] D. Galbraith and V. M. Baaijen. Aligning keystrokes with cognitive processes in writing. In *Observing Writing: Insights from Keystroke Logging and Handwriting*, pages 306–325. 2019.

[10] G. Gao, A. Leon, A. Jetten, J. Turner, H. Almoubayyed, S. Fancsali, and E. Brunskill. Predicting long-term student outcomes from short-term edtech log data. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 631–641, 2025.

[11] T. Gervet, K. R. Koedinger, J. Schneider, and T. Mitchell. When is deep learning the best approach to knowledge tracing? *Journal of Educational Data Mining*, 12(3):31–54, 2020.

[12] E. Kleinman, M. Shergadwala, Z. Teng, J. Villareale, A. Bryant, J. Zhu, and M. S. El-Nasr. Analyzing students' problem-solving sequences: A human-in-the-loop approach. *Journal of Learning Analytics*, 9(2):138–160, 2022.

[13] K. R. Koedinger, A. T. Corbett, and C. Perfetti. The knowledge–learning–instruction framework: Bridging the science–practice chasm to enhance robust student learning. *Cognitive Science*, 36(5):757–798, 2012.

[14] W. Ma, O. O. Adesope, J. C. Nesbit, and Q. Liu. Intelligent tutoring systems and learning outcomes: A meta-analysis. *Journal of Educational Psychology*, 106(4):901–918, 2014.

[15] N. Matsuda, S. Chandrasekaran, and J. C. Stamper. How quickly can wheel spinning be detected? In *Edm*, pages 607–608. ERIC, 2016.

[16] L. Van Waes and M. Leijten. Fluency in writing: A multidimensional perspective on writing fluency applied to l1 and l2. *Computers and Composition*, 38:79–95, 2015.

[17] K. VanLehn. The behavior of tutoring systems. *International journal of artificial intelligence in education*, 16(3):227–265, 2006.

[18] C. Zhang, Y. Huang, J. Wang, D. Lu, W. Fang, J. Stamper, S. Fancsali, K. Holstein, and V. Aleven. Early detection of wheel spinning: Comparison across tutors, models, features, and operationalizations. *International Educational Data Mining Society*, 2019.

Tracing Conceptual Overgeneralization in Middle School Students' Reasoning about Quantum Randomness

Jihyun Rho
University of Florida
j.rho@ufl.edu

Jiwon Kim
Purdue University
jwkim@purdue.edu

Zeynep G. Akdemir-Beveridge
University of Connecticut
zeynep.akdemir@uconn.edu

Muhsin Menekse
Southern Methodist University
menekse@smu.edu

ABSTRACT

students because it requires students' conceptual change between classical and quantum explanatory frameworks for randomness. Conceptual change in such contexts may not only involve replacing prior knowledge, but also risks overgeneralization, where newly learned concepts are extended beyond their intended scope. In this study, we investigate how students reorganize their reasoning about randomness across classical and quantum contexts following instruction. We analyzed 14,296 open-ended responses collected from pre-test and post-test assessments administered to 1,787 middle school students as part of a quantum-infused science curriculum. Student explanations were classified into classical, quantum, or non-informative reasoning using a large language model fine-tuned on a human-coded subset. A binomial mixed-effects model revealed asymmetric overgeneralization dynamics across problem contexts. Following instruction, overgeneralization decreased in quantum problem contexts, indicating improved alignment between quantum explanations and quantum phenomena. In contrast, overgeneralization increased in classical contexts, reflecting the extension of newly learned quantum reasoning into situations where classical explanations were sufficient. These findings highlight the bidirectional nature of conceptual overgeneralization and underscore the importance of instructional designs that support context-sensitive integration of new scientific concepts.

Keywords

Conceptual change, overgeneralization, quantum randomness, LLM-based classification.

1. INTRODUCTION

Research on conceptual change has emphasized that students' learning of new concepts is not a process of replacement, but one of gradual reorganization of existing knowledge structure [6,10]. Within this framework, overgeneralization can be understood as a pattern that emerges during conceptual change, reflecting students' attempts to coordinate prior knowledge and newly learned information.

Jihyun Rho, Jiwon Kim, Zeynep G. Akdemir-Beveridge, and Muhsin Menekse. Tracing Conceptual Overgeneralization in Middle School Students' Reasoning about Quantum Randomness. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 578–584. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039823

Prior work primarily framed overgeneralization as a phenomenon driven by prior knowledge constraining the interpretation of new information [19,20]. From this perspective, students rely on well-established prior knowledge to make sense of a new domain, sometimes leading to explanations that do not fully align with the new domain. However, this overgeneralization also occurs in the reversed direction. As students acquire new information, this newly learned information may itself be overextended, shaping or changing reasoning even in contexts where prior knowledge would otherwise be sufficient [22]. Overgeneralization, therefore, can reflect not only resistance to abandoning old frameworks but also the excessive generalization of newly acquired information beyond its intended scope.

This bidirectional view of overgeneralization is particularly relevant in domains where instruction introduces fundamentally different explanatory principles, such as quantum randomness. When learning quantum randomness, students need to reconcile probabilistic reasoning grounded in classical systems with non-classical sources of indeterminacy [4,17]. In this process, students may incorrectly apply classical randomness to a quantum context, but they may also extend quantum randomness back into classical situations, thereby reshaping their prior conceptual frameworks.

Despite its theoretical importance, overgeneralization during conceptual change has often been examined indirectly, primarily through accuracy-based measures. Such approaches can obscure whether students' responses reflect stable understanding, transitional reasoning, or systematic overextension across contexts. To address this gap, our study examines patterns of overgeneralizations across classical and quantum contexts before and after the instruction, focusing on how students' explanations shift across conceptual boundaries.

2. RELATED WORKS

2.1 Quantum Curriculum for Middle School

Recent years have seen growing interest in introducing quantum information science and engineering (QISE) concepts into K-12 education [15], driven largely by efforts to expand the future quantum workforce. National reports, including those from the National Quantum Coordination Office, have emphasized the importance of developing age-appropriate quantum learning pathways that integrate with existing science and computing curricula, rather than treating quantum topics as isolated and advanced content. In response to this need, initiatives such as The National Q-12 Education Partnership have emerged. However, their curricular recommendations primarily target the high school level. Correspondingly, existing empirical research on K-12 quantum education has

overwhelmingly focused on high school students (e.g., [5,16]), with comparatively little attention to middle school students.

Despite its imbalance, a growing body of evidence suggests that middle school students are cognitively capable of engaging with core ideas in modern physics, including foundational quantum concepts [1,7,13]. For example, prior work has demonstrated a saturation effect in post-test performance, indicating that no intrinsic cognitive barriers prevent middle school students from understanding key Einsteinian ideas such as curved spacetime or photon momentum [7]. Moreover, longitudinal findings show that students were able to recall central concepts (e.g., "space is curved," "light comes as photons") one to three years after instruction, suggesting durable conceptual understanding rather than short-term memorization [13]. Together, these findings challenge assumptions that quantum concepts are inherently inaccessible at the middle school level.

Building on this evidence, a small but growing literature on middle school quantum education has focused on two primary areas: (1) the development of effective pedagogical interventions and (2) teacher education and professional development for quantum curriculum implementation. A portion of this work has examined teacher preparation. Survey and interview studies with middle and high school teachers consistently show that the successful introduction of quantum topics depends heavily on teacher self-efficacy, conceptual confidence, and access to sustained support [2,11]. This highlights the role of professional development (PD) in co-designing their quantum curriculum [1]. Teachers often express strong motivation to engage with quantum science due to its perceived relevance and future-oriented appeal, while simultaneously reporting concerns about curricular alignment and their own content knowledge.

In parallel, research on instructional design has explored how middle school students can meaningfully engage with quantum concepts by deemphasizing mathematical formalism and instead foregrounding conceptual, experiential, and interactive learning modalities [7,13,14,21]. For instance, one study employed a gamified puzzle environment to teach quantum logic gates, finding that students required substantial scaffolding and frequently bypassed textual explanations, which motivated designs that embed learning directly into gameplay mechanics. Other work has leveraged interdisciplinary approaches, such as using music education to introduce quantum atomic models to 12-13-year-olds [21]. This interdisciplinary approach resulted in high post-test performance. Finally, the Einstein-First project exemplifies a complementary approach that replaces traditional Newtonian instruction with Einsteinian concepts through concrete physical models. Interventions in the project included using Nerf guns to represent photons (demonstrating momentum and uncertainty) and lycra sheets with marbles to visualize curved spacetime [7,13]. These studies support the feasibility of introducing modern physics concepts at the middle school level.

2.2 Teaching quantum randomness

However, demonstrating the feasibility of teaching quantum concepts at the middle school level does not imply that all quantum ideas are equally accessible. Among basic quantum concepts [9], the probabilistic nature of quantum phenomena is central for understanding the behavior of quantum entities. Prior research suggests that students often approach quantum physics with intuitions grounded in everyday experience, treating quantum entities as classical objects that follow definite trajectories through space and time [12]. This reliance on everyday experiences and classic objects creates substantial conceptual barriers when students encounter quantum phenomena that require fundamentally different explanatory assumptions.

Specifically, researchers investigated how students interpret probability and randomness in quantum contexts and categorized students' interpretations into three types, including a classic type, a mixed type, and a quantum type [17]. The classical type refers to students' interpretation where randomness is treated as epistemic uncertainty caused by incomplete information or imperfect measurement. The mixed type refers to an interpretation where students assume hidden information exists but is practically inaccessible. Lastly, in the quantum type, students treat individual outcomes as inherently unpredictable while recognizing that collective behavior can be predicted statistically. This framework highlights that the critical distinction between classical and quantum randomness lies not in whether individual outcomes are difficult to predict, but in the source of that unpredictability. In the classical view, unpredictability arises from incomplete information. Given sufficient knowledge of all relevant variables, outcomes are determinable, in principle. In the quantum view, unpredictability is intrinsic to the system. Even with complete knowledge of the system's state, individual outcomes remain fundamentally indeterminate.

2.3 Conceptual Change and Overgeneralization

Conceptual change research characterizes learning as a process of knowledge integration that unfolds gradually, during which students actively reorganize existing ideas rather than simply replacing them [6,10]. Within this framework, overgeneralization has been widely documented as a mechanism through which students extend familiar principles beyond their valid scope. Much of this work has focused on cases in which entrenched prior knowledge constrains the interpretation of new information, such as the application of whole-number reasoning to rational numbers [19], linearity assumptions in non-linear domains [20], or deterministic intuitions in probabilistic and stochastic systems [4]. In these accounts, overgeneralization is often interpreted as evidence of robust prior knowledge that resists accommodation. However, other strands of overgeneralization can also arise when newly learned concepts are overextended or inappropriately activated in contexts where previously established forms of reasoning would have been sufficient [4,8,22]. From this perspective, overgeneralization can reflect not only resistance to conceptual change, but also productive sense-making during the reorganization of knowledge structures, which occur in both directions. While prior knowledge constrains new learning, newly acquired concepts are overextended into familiar domains.

Thus, because conceptual change involves ongoing reorganization of knowledge, overgeneralization is often difficult to capture with summary performance measures alone. Accordingly, many conceptual change studies have examined students' strategies and explanations in depth. For example, prior research has analyzed students' written explanations, interview responses, and problem-solving justifications to characterize the reasoning strategies students apply across tasks [4,19,20]. However, such fine-grained analyses are typically time-intensive and therefore difficult to scale, particularly when studying large populations or multiple task contexts. As a result, relatively little work has systematically examined patterns of reasoning across contexts using large collections of student explanations.

Addressing this gap requires analytic approaches that can simultaneously preserve sensitivity to students' explanatory reasoning and operate at scale across multiple task contexts. Recent work in

educational data mining has demonstrated the feasibility of using large language models to automatically assess rich, open-ended student data [3,18]. Therefore, in the present study, we adopt a Large Language Model-assisted analytic approach to examine large collections of student explanations across classical and quantum randomness contexts.

3. RESEARCH QUESTIONS

Building on prior works, the present study examines how students' reasoning about randomness reorganizes after their first formal exposure to quantum randomness in a middle school science setting. This investigation is situated within a larger evaluation of a quantum-infused curriculum that introduces quantum randomness through instruction on radioactive decay. This topic already involves classical probability within existing science standards. By comparing students' written explanations across pre- and post-test items designed to elicit classical and quantum interpretations of randomness, we analyze whether overgeneralization dynamics shift asymmetrically across contexts as students integrate new conceptual knowledge.

Accordingly, we address the following research question:

RQ) How do overgeneralization patterns differ across classical and quantum contexts before and after curriculum, and what asymmetries emerge in these dynamics?

4. DATASETS

4.1 Quantum-Infused Science Curriculum

We co-designed the 7-day-long quantum education curriculum for middle school students with middle school science teachers. We aim to align the curriculum with classroom constraints and be accessible to typical middle school students. The first half of the curriculum introduces randomness in an Earth and Space Science context, starting with instruction on radioactive decay and radiometric dating. Students begin with an archaeology-themed challenge (e.g., estimating the age of remains) and then learn how Carbon-14 decays into Nitrogen-14 over time, using half-life as a statistical model for decay rates rather than a deterministic prediction of individual events. In the following lessons, students engage with isotopes, decay types, and hands-on modeling activities (e.g., popcorn and coin-flip analogies) to build an intuitive distinction between what can be predicted in aggregate (i.e., the decay rate) versus what cannot be predicted at the level of a single atom (i.e., which nucleus decays next). At this stage, students are expected to understand this unpredictability as classical randomness, arising from practical complexity (i.e., too many variables to track). They are not yet expected to understand any fundamental indeterminacy in the system itself.

The second half of the curriculum then extends beyond existing standards-based science content by explicitly introducing the concept of quantum randomness. They now explore randomness in the context of quantum indeterminacy and learn how quantum random number generators produce unpredictable outcomes that cannot be explained by hidden variables or measurement limitations. Students apply this idea in an activity that involves generating binary numbers, converting them to hexadecimal codes using real quantum-generated data from an online quantum lab, creating random colors, and using them in their artwork. This final lesson positions quantum randomness as an extension of students' prior reasoning about probability, while discussing boundary conditions for when classical explanations are sufficient versus when quantum explanations are needed.

4.2 Participants and Dataset

Participants were middle school students enrolled in regular science classes at seven public middle schools in the Midwestern United States. Across the schools, a total of 1787 students participated in both the pre-test and post-test. Approximately 25.4 % of students were in 6th grade, 37.0% were in 7th grade, and 37.6% were in 8th grade. Female students comprised 49.3% of the sample, while male students comprised 50.7%. Students were recruited through classroom participation during regular instructional activities, and no prior instruction on quantum randomness was assumed.

Each student responded to four assessment items (i.e., two items asking the classical randomness concept and two items asking the quantum randomness concept) in the pre-test and the same four items in the post-test, resulting in a total of 8 responses per student. The example of the question is shown in Figure 1 below. Across all participants, this yielded 14,296 total responses in the full dataset used for large-scale analysis. This study was reviewed and approved by the Institutional Review Board at Purdue University (IRB-2022-1016). We randomly selected a subset of responses from 100 students for detailed human coding and model calibration. This subset comprised 800 responses (4 items × pre/post × 100 students). We used this subset to establish coding reliability, train, and evaluate automated classification models. The remaining responses formed the dataset for population-level analyses of conceptual reasoning and overgeneralization dynamics.

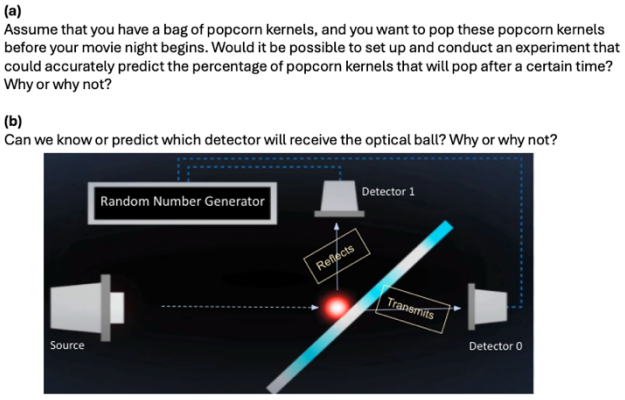


Figure 1. Example test items used in both pre-test and post-test. (a) is a classical randomness item asking whether the outcome of popcorn popping can be accurately predicted in advance and why. (b) is a quantum randomness item depicting a single-photon beam splitter setup, asking whether one can know or predict which detector will register the optical signal and why.

5. ANALYSIS

5.1 Coding Scheme

We developed the coding scheme through an iterative, theory-informed process. We first derived an initial set of codes from the theoretical distinction between classical randomness and quantum randomness, informed by prior research on middle school students' understanding of quantum randomness [17]. We then applied these preliminary codes to a subset of student responses to identify ambiguities, overlaps, and borderline cases. Through iterative discussion, we refined the code definitions to ensure conceptual clarity, mutual exclusivity, and applicability across both classical and quantum contexts. The refinement was done with the understanding that coding of student responses did not aim to assess correctness per se, but rather to characterize the dominant

conceptual basis underlying each student's response. Crucially, the coding distinguished responses based on the attributed source of unpredictability, rather than on the acknowledgment of unpredictability itself. For example, in the popcorn item in Figure 1, attributing unpredictability to practical complexity (e.g., varying moisture or heat) was coded as classical, whereas treating individual outcomes as fundamentally indeterminate was coded as quantum. The latter constitutes overgeneralization because popcorn popping is governed by deterministic physical processes that are, in principle, predictable given sufficient information. This process yielded a three-category coding scheme, summarized in Table 1.

Using this coding scheme, two trained researchers independently coded a subset of students' open-ended responses. This human-coded dataset consisted of responses from 100 students, each answering four items in both pre-test and post-test, yielding a total of 800 responses. The overall inter-rater reliability (Cohen's κ) was $\kappa = 0.8227$, indicating strong agreement beyond chance. Specifically, 0.8143 for classical, 0.8450 for quantum, and 0.8227 for non-informative.

5.2 LLM-Based Response Classification

To enable large-scale analysis of student reasoning, we used a large language model (LLM) to classify the remaining student responses that were not manually coded. We fine-tuned a pretrained instruction-following LLM (*Meta-LLaMA-3.1-8B-Instruct*) using parameter-efficient fine-tuning with low-rank adaptation (LoRA). We split the human-coded student responses described above into training and validation sets. The model was trained on this training set.

We evaluated the trained model's performance using the validation set exclusively held out for model evaluation. To account for class imbalance across conceptual categories, we report both macro-averaged and weighted F1 scores, along with category-level precision and recall. After validation, we applied the trained model to fully code the remaining full dataset of 13,078 responses after excluding empty responses.

5.3 Operationalizing Overgeneralization

We operationalized overgeneralization as cross-contextual misalignment between the problem context and the category of conceptual reasoning expressed in student responses. For each response, we compared the coded conceptual category with the problem context. We coded overgeneralization as present when a student applied quantum reasoning to a classical problem context or classical reasoning to a quantum problem context.

We excluded responses coded as non-informative from overgeneralization analyses because they did not provide sufficient evidence of a coherent conceptual framework. This binary overgeneralization variable served as the primary outcome in subsequent statistical analyses examining change over test time (i.e., pre-test and post-test) and problem contexts.

5.4 Statistical Analysis

To examine changes in overgeneralization across test time (i.e., pre-test and post-test) and problem contexts, we employed a binomial mixed effects modeling approach. The dependent variable was the binary overgeneralization indicator. Fixed effects included the test time (pre vs. post), problem context (classical vs. quantum), and their interaction. Random intercepts were included for students and test items to account for repeated measurements and problem-level variability.

Table 1. Three-category coding scheme used to classify students' reasoning of randomness in classical and quantum test items.

Code	Definition	Example
Classical	Randomness is treated as epistemic rather than fundamental. Outcomes are assumed to be determined by underlying physical causes and, in principle, predictable given complete information. Uncertainty is attributed to limited knowledge, measurement constraints, or statistical aggregation.	"Yes, you would have to do a lot of re-testing and starting the whole experience over again. You could look in your microwave and see how many pop in a minute."
Quantum	Randomness is treated as fundamental to the physical system. Even with complete knowledge of the system, the outcome of an individual measurement is considered inherently unpredictable, reflecting intrinsic indeterminacy rather than lack of information.	"We cannot predict which detector will receive the optical ball. It is random, so we don't know if detector 1 or detector 0 will get it."
Non-Informative	Responses that are off-topic, vague, or non-conceptual, including expression of uncertainty or yes/no answers without explanatory reasoning.	"I don't know", "idk", "No, because I don't want to wait to watch the movie."

6. RESULTS

6.1 Validating LLM-based Classification Against Human Annotation

Before analyzing the overgeneralization pattern in the full dataset, we assessed the LLM-based classification performance against the human-coded validation set. The model achieved an accuracy of 0.8841, a macro F1 score of 0.8316, and a weighted F1 score of 0.8803. The code category-specific level's F1, precision, and recall scores were shown in Table 2. Precision and recall scores indicated that the model was most reliable in distinguishing students' classical reasoning and quantum reasoning.

Table 2. Classification performance of the LLM-based model by conceptual category on the human-coded validation set.

Code	Precision	Recall	F1
Classical	0.885	0.955	0.919
Quantum	0.750	0.600	0.667
Non-Informative	0.962	0.862	0.909

6.2 Distributed Patterns of Conceptual Reasoning

Prior to instruction, students' pre-test responses in the classical problem context were predominantly classified as classical reasoning (75.32%), with only a smaller proportion reflecting quantum reasoning (4.83%). In contrast, responses to the quantum problem context showed a substantial reliance on classical reasoning (68.13%), indicating that students largely interpreted quantum phenomena using classical conceptual reasoning before the instruction. Table 3 summarizes the distribution of conceptual reasoning across problem context and test times.

Table 3. Pre-test and post-test distributions of classical, quantum, and non-informative reasoning by problem context.

Problem context	Test time	Classical	Quantum	Non-Informative
Classical	Pre	75.32%	4.83%	19.85%
	Post	64.78%	15.45%	19.77%
Quantum	Pre	68.13%	14.44%	17.43%
	Post	56.49%	29.23%	14.28%

Following the instruction on quantum randomness, the overall distribution of conceptual reasoning shifted in the post-test (Figure 2). In the quantum problem context, the proportion of quantum reasoning increased to 29.23%, accompanied by a decrease in classical reasoning to 56.49%, suggesting increased uptake of non-classical reasoning.

Notably, a parallel shift also occurred in the classical problem context. Despite the classical nature of the problems, the proportion of responses classified as quantum reasoning increased from 4.83% in the pre-test to 15.45% in the post-test.

6.3 Overgeneralization Dynamics Across Problem Context and Test Time

Our binomial mixed-effects model revealed meaningful effects of both test time and problem context on students' overgeneralization patterns. First, the main effect of time showed a positive posterior mean (β = 1.31, 95% credible interval = [1.24, 1.37]), indicating that, on average, overgeneralization became more likely following instruction. The 95% credible interval for this effect did not include zero, providing posterior evidence for an overall increase in overgeneralization across problem contexts. The main effect of context also showed a positive posterior mean (β = 4.22, 95% credible interval = [4.15, 4.29]), suggesting substantially higher baseline overgeneralization rates in the quantum problem context compared to the classical problem context.

Critically, the interaction between test time and problem context showed a negative posterior mean (β = -2.22, 95% credible interval = [-2.31, -2.13]). This pattern indicates that instructional effects on overgeneralization diverged across problem contexts. Specifically, while overgeneralization decreased in the quantum problem context following instruction, it increased in the classical problem context, reflecting a problem-context dependent reorganization of students' conceptual reasoning. These posterior estimates from the mixed-effects model are reported in Table 4.

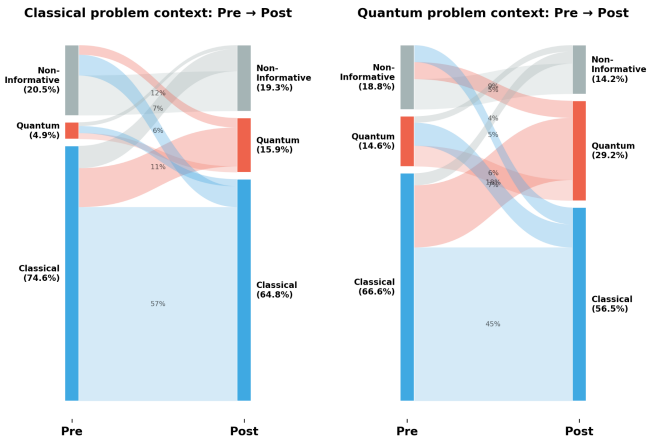


Figure 2. Sankey plot showing transitions in students' response from pre-test to post-test in classical (left) and quantum (right) problem contexts. Each band represents the flow of responses from a pre-test to a post-test.

Table 4. Posterior estimates from the binomial mixed-effects model predicting overgeneralization.

	Posterior mean (β)	Posterior standard deviation	95% Credible interval (CrI)
Intercept	-2.680	0.028	[-2.74, -2.63]
Test time	1.305	0.034	[1.24, 1.38]
Problem context	4.217	0.035	[4.15, 4.29]
Test time × Problem context	-2.219	0.045	[-2.31, -2.13]

Figure 3 summarizes model-predicted probabilities of overgeneralization across test time and problem context, derived from the binomial mixed-effects model and transformed from the logit scale. In the classical problem context, predicted overgeneralization increased from 6.42% in pre-test to 20.19% in post-test. In contrast, in the quantum problem context, predicted overgeneralization decreased from 82.31% in the pre-test to 65.11% in the post-test. These estimates reveal a qualitative asymmetry. This pattern indicates an asymmetric shift in explanatory tendencies across problem contexts: post-instruction responses showed reduced context-reasoning mismatch in the quantum problems but increased mismatch in classical problems. These changes mirror the interaction effect observed in the mixed-effects model, suggesting that shifts in overgeneralization differed systematically by problem contexts.

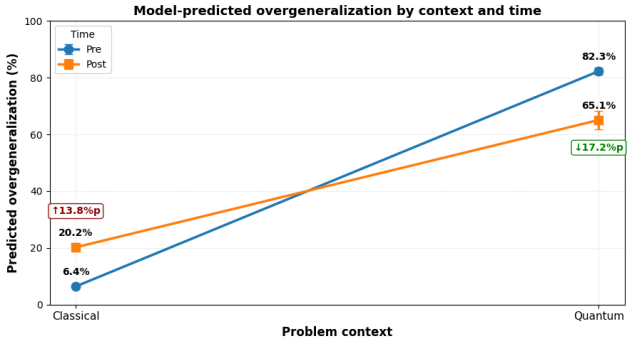


Figure 3. Model-predicted overgeneralization rates by problem context (classical vs. quantum) and time (pre- vs. post-test). Annotations indicate the change in percentage points from pre- to post-test within each context.

7. DISCUSSION

This study examined how middle school students' reasoning about randomness reorganizes across classical and quantum problem contexts following the curriculum implementation, with a particular focus on overgeneralization. From an educational data mining perspective, our work demonstrates how large-scale analyses of open-ended student explanations can uncover systematic patterns of conceptual change and overgeneralization that are challenging to observe through correctness-based metrics alone. By integrating LLM-based classification with mixed-effects modeling, we illustrate how EDM methods can be used to model fine-grained shifts in students' explanatory reasoning across contexts and time in quantum education.

Firstly, our results provide empirical support for a bidirectional account of overgeneralization during conceptual change in quantum education, revealed through large-scale analysis of open-ended responses. Overgeneralization of classical reasoning in quantum problem contexts decreased after students completed the curriculum, while overgeneralization of quantum reasoning in classical contexts increased. Prior work has largely framed overgeneralization as the transfer of entrenched prior knowledge into newly learned domains (e.g., [18, 19]). Consistent with this view, students initially relied heavily on classical reasoning when interpreting quantum randomness. Following the curriculum, such an initial reliance decreased in quantum contexts, indicating improved understanding of quantum randomness. However, newly acquired quantum reasoning was increasingly applied even to classical contexts where classical explanations were sufficient. From an EDM standpoint, this asymmetric pattern represents the value of modeling context-dependent reasoning trajectories rather than treating learning gains as uniform improvements. Large-scale classification of explanations enables detection of such directional shifts, which would likely be obscured in accuracy-only analyses.

Secondly, the observed bidirectional overgeneralization has implications for data-informed curriculum design in middle school quantum education. The increase in quantum reasoning in classical contexts suggests that students actively integrate newly learned frameworks into existing knowledge structures, even when this integration exceeds normative boundaries. Similar challenges in distinguishing explanatory frameworks have been reported in prior studies of quantum reasoning and conceptual change (e.g., [3, 16]). In our curriculum, classical and quantum randomness were intentionally taught together to align with state standards and the pre-set annual teaching plan, with instruction beginning from classical probability and concluding with quantum randomness. While such a curricular design successfully and practically promoted quantum reasoning, it may also have increased the likelihood of quantum overgeneralization in classical contexts. These findings suggest that EDM approaches can play a critical role in evaluating instructional trade-offs and thus help curriculum designers identify when instructional success in one domain introduces unintended conceptual spillover into another. Explicit scaffolds that clarify boundary conditions may therefore be necessary to stabilize students' reasoning across classical and quantum science contexts.

Last but not least, this study demonstrates the methodological potential of LLM-assisted educational data mining for studying conceptual change using open-ended student responses. Prior work has shown that LLMs can reliably assess rich student explanations at scale (e.g., [2, 17]). Building on this work, we show how validated LLM-based classifications can be integrated with statistical modeling to examine population-level patterns of reasoning reorganization. This approach is particularly valuable for underrepresented student populations, such as middle school students in quantum education, where large datasets of open-ended responses have rarely been explored and are difficult to analyze manually.

Nevertheless, limitations remain. First, the three-category coding scheme, while aligned with the theoretical distinction central to our research question, may not capture finer-grained variations such as mixed or transitional reasoning. To capture such details of students' conceptual trajectories, future studies could explore how to sophisticate coding schemes. Second, automated classification can introduce uncertainty into coding results, particularly for the quantum category, where the model achieved a recall of 0.60, suggesting that quantum reasoning may be underestimated in our analyses. This limitation likely makes our central finding conservative, as the true extent of quantum reasoning in classical contexts may exceed what is reported. Future work could address this through different strategies, such as oversampling the quantum minority class during training and applying class-weighted loss functions to improve classification performance under class imbalance. Lastly, our analysis focused on short-term instructional effects; longitudinal EDM analyses are needed to examine how overgeneralization patterns evolve over longer timescales and across instructional contexts.

8. CONCLUSIONS

Our study contributes to research on conceptual change by empirically demonstrating that overgeneralization during learning is not a unidirectional phenomenon but can occur as a bidirectional phenomenon. In the context of learning quantum randomness, students showed both reduced inappropriate reliance on classical reasoning in quantum problem contexts and increased application of quantum reasoning in classical problem contexts following instruction. These asymmetric shifts highlight that conceptual change involves not only replacing or constraining prior knowledge but also negotiating the boundaries of newly acquired concepts. From an educational perspective, our findings suggest that successful instruction should be evaluated not only by gains within the target domain but also by how newly learned concepts interact with students' existing knowledge across contexts. Specifically, for quantum education at the middle school level, this implies a need for instructional designs that explicitly address the scope and limits of quantum explanations, helping students differentiate when quantum reasoning is appropriate and when classical explanations remain sufficient.

9. ACKNOWLEDGEMENTS

This study is supported by the National Science Foundation (Award Number: 2422937) and the National Defense Education Program (Award Number: HQ00342110014). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the National Defense Education Program.

10. REFERENCES

1. Zeynep Gonca Akdemir. 2024. Design, Implementation, and Evaluation of a Quantum-Infused Middle-School Level Science Teaching and Learning Sequence.
2. Zeynep Gonca Akdemir, Muhsin Menekse, Erica W Carlson, Nicholas Dang, Mahdi Hosseini, and Dongyang Li. 2024. Supporting Middle School Students' Learning Outcomes and Engagement with NGSS-Aligned Quantum-Infused Science Curriculum. *ASEE Annual Conference & Exposition.* 10.18260/1-2--48039
3. Sami Baral, Eamon Worden, Wen-Chiang Lim, Zhuang Luo, Christopher Santorelli, and Ashish Gurung. 2024. Automated Assessment in Math Education: A Comparative Analysis of LLMs for Open-Ended Responses. 732–737.
4. T. Bouchée, L. de Putter - Smits, M. Thurlings, and B. Pepin. 2022. Towards a better understanding of conceptual difficulties in introductory quantum physics courses. *Studies in Science Education* 58, 2: 183–202.
5. Berit Bungum, Maria Vetleseter Bøe, and Ellen Karoline Henriksen. 2018. Quantum talk: How small-group discussions may enhance students' understanding in quantum physics. *Science Education* 102, 4: 856–877.
6. Michelene TH Chi. 2009. Three types of conceptual change: Belief revision, mental model transformation, and categorical shift. In *International handbook of research on conceptual change*. Routledge, 89–110, Routledge.
7. Rahul K Choudhary, Alexander Foppoli, Tejinder Kaur, David G Blair, Marjan Zadnik, and Richard Meagher. 2018. Can a short intervention focused on gravitational waves and quantum physics improve students' understanding and attitude? *Physics Education* 53, 6: 065020.
8. Yi Ding, Guangtian Zhu, Qiuxin Bian, and Lei Bao. 2024. Analysis of students' conceptual change in learning Newton's third law with an integrated framework of model analysis and knowledge integration. *Physical Review Physics Education Research* 20, 2: 020141.
9. Directorate-General for Communications Networks, Content and Technology (European Commission), Quantum Flagship, Greinert, Franziska, and Müller, Rainer. 2025. *European competence framework for quantum technologies (CFQT): reference framework for planning, mapping and comparing QT related educational activities, personal qualification and job requirements*. Publications Office of the European Union. Retrieved January 28, 2026 from https://data.europa.eu/doi/10.2759/8917117
10. Andrea A DiSessa. 2014. *A history of conceptual change research*.
11. Apekshya Ghimire, Jaya Shivangani Kashyap, Emily Edwards, Diana Franklin, and Chandralekha Singh. 2025. Investigating high school and pre-high school teachers' perceptions and experiences introducing quantum concepts: a survey of QuanTime and other quantum-related activities. *EPJ Quantum Technology* 12, 1: 89.
12. Fabian Hennig, Kristóf Tóth, Moritz Förster, and Philipp Bitzenbauer. 2024. A new teaching-learning sequence to promote secondary school students' learning of quantum physics using Dirac notation. *Physics Education* 59, 4: 045007. https://doi.org/10.1088/1361-6552/ad353d
13. Tejinder Kaur, David Blair, Warren Stannard, David Treagust, Grady Venville, Marjan Zadnik, Warwick Mathews, and Dana Perks. 2020. Determining the Intelligibility of Einsteinian Concepts with Middle School Students. *Research in Science Education* 50, 6: 2505–2532.
14. Tianle Liu, David Gonzalez-Maldonado, Danielle B. Harlow, Emily E. Edwards, and Diana Franklin. 2023. Qupcakery: A Puzzle Game that Introduces Quantum Gates to Young Learners. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (SIGCSE 2023), 1143–1149. https://doi.org/10.1145/3545945.3569837
15. Christopher Monroe, Michael G Raymer, and Jacob Taylor. 2019. The US national quantum initiative: From act to action. *Science* 364, 6439: 440–442.
16. Sara Satanassi, Paola Fantini, Roberta Spada, and Olivia Levrini. 2021. Quantum Computing for high school: an approach to interdisciplinary in STEM for teaching. *Journal of Physics: Conference Series* 1929, 1: 012053. https://doi.org/10.1088/1742-6596/1929/1/012053
17. Kristóf Tóth, Fabian Hennig, Joaquin Marc Veith, and Philipp Bitzenbauer. 2025. Secondary school learners' reasoning about quantum randomness in the context of single-photon interferometer and double-slit experiments. *Eurasia Journal of Mathematics, Science and Technology Education* 21, 8: em2686.
18. Nhat Tran, Benjamin Pierce, Diane Litman, Richard Correnti, and Lindsay Clare Matsumura. 2024. Analyzing Large Language Models for Classroom Discussion Assessment. https://doi.org/10.48550/arXiv.2406.08680
19. Xenia Vamvakoussi and Stella Vosniadou. 2004. Understanding the structure of the set of rational numbers: a conceptual change approach. *Learning and Instruction* 14, 5: 453–467. https://doi.org/10.1016/j.learninstruc.2004.06.013
20. Wim Van Dooren, Dirk De Bock, An Hessels, Dirk Janssens, and Lieven Verschaffel. 2004. Remedying secondary school students' illusion of linearity: a teaching experiment aiming at conceptual change. *Learning and Instruction* 14, 5: 485–501. https://doi.org/10.1016/j.learninstruc.2004.06.019
21. Hugo Vieira and Carla Morais. 2022. Musical analogies to teach middle school students topics of the quantum model of the atom. *Journal of Chemical Education* 99, 8: 2972–2980.
22. Joseph Jay Williams, Tania Lombrozo, and Bob Rehder. 2013. The hazards of explanation: Overgeneralization in the face of exceptions. *Journal of Experimental Psychology: General* 142, 4: 1006.

A demonstration of the Quick Red Fox (QRF) system: An application for conducting Data-Driven Classroom inter-views

Luc Paquette
University of Illinois Urbana-Champaign
lpaq@illinois.edu

Jaclyn Ocumpaugh
University of Houston
jocumpau@central.uh.edu

ABSTRACT

In this demonstration, we present the Quick Red Fox (QRF) system, a system design to assist researchers in conducting Data-Driven Classroom Interviews (DDCIs). DDCIs enables researchers to collect rich qualitative interview data at key moments of the students' interaction with a digital learning environment. Using the QRF system, researchers can use EDM methods to analyze logs of student interaction data, identify interview triggers and provide classroom interviewers with real-time interview requests on a handheld Android app. We describe the main components of the QRF system (the QRF android app, the QRF server and the QRF dashboard) and demonstrate the usage of QRF in the context of WHIMC, a Minecraft-based learning system in which students learn about astronomy.

Keywords

Data-driven interviews, digital learning environments, mixed-method research.

1. INTRODUCTION

The analysis of student interaction log data using EDM approaches has been successfully used to achieve a range of educational goals ranging from the use of Discovery with Models to improve our understanding of student learning processes (as described in [1]), to generate data-driven student support (such as data-driven feedback, recommendations or interventions; [2], [3]), to design student and teacher facing learning dashboards ([4], [5]), and to inform the design of learning tasks and learning environments ([6], [7], [8]). In this demonstration, we present the Quick Red Fox (QRF) system, a system that leverages EDM methodologies to support the collection of rich in-the-moment qualitative data. Using this system, researchers can leverage data-driven analyses of students' interaction log data to collect Data-Driven Classroom Interviews (DDCI) at key moments in the students' usage of digital learning environments.

2. DATA-DRIVEN CLASSROOM INTERVIEWS (DDCI)

The purpose of QRF is to provide researchers with immediate, and contextualized students reactions to learning systems. Unlike other methods that attempt to capture *in situ* data, such as Think Aloud methods [9], QRF allows us to target specific learning events that are detectable in a learning software so that researchers can observe and interview students while that event is still fresh in their minds. In that way, we can ensure that interviewers are able to provide students with questions about those experiences that help us to address underlying research questions, and we can do so without asking the student to carry the metacognitive load of figuring out what would be relevant to the research question, which is sometimes a criticism of methods like Think Alouds [10].

DDCIs are often very short (3-7 minutes). Questions may vary depending upon the underlying research question, and interviewers are encouraged to develop a range of question strategies before entering the field, as they are likely to have many DDCIs with each student over the course of a fieldwork experience. For example, if students are using the system for an hour a day, it might not be unusual for a single student to be interviewed more than once, and if the fieldwork efforts extend over a week, the same interviewer may talk to one student a dozen times.

Interviewers are trained to take a friendly but non-authoritative approach where they express interest about the students' experience [11]. This allows them to pivot to specific questions about the students' immediate experience without becoming bogged down with questions about how the learning system works or with requests from the student to have the interviewer explain the learning material. Instead, if the student is struggling, the interviewer might ask what strategies they have already tried, which strategies they might still need to try, and more generally how the student usually goes about dealing with learning obstacles.

This approach allows the interviewers to establish student trust, so that they are willing to make suggestions about what might improve their learning or engagement. As such, the approach lends itself well to investigating research questions about both the difficulties that students might be having at any given moment and about any assets students bring to the learning context [12].

3. THE QUICK RED FOX (QRF) SYSTEM

The QRF system provides multiple inter-related components (the QRF android app, the QRF server and the QRF dashboard) that can be used together to facilitate the implementation of DDCIs using an existing digital learning system (see Figure 1). QRF is made available as an open-source system and its user manual provides instructions about how to access the QRF source code and the steps required to setup a QRF server (see "Appendix 2: Installing the QRF server" of the user manual [11] for detailed instructions).

Luc Paquette, and Jaclyn Ocumpaugh. A demonstration of the Quick Red Fox (QRF) system: An application for conducting Data-Driven Classroom inter-views. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 585–588. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040109

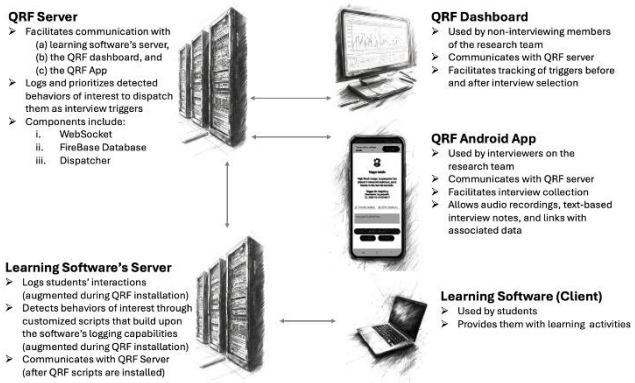


Figure 1. Overview of the QRF system, its different components and the interactions between each component.

3.1 The QRF Android App

The QRF Android App is the tool used by interviewers to conduct DDCIs. Interviewers are first asked to login to the app with their username. Once logged in, the app waits for the QRF server to send it an interview trigger (see left of Figure 2). The right side of Figure 2 shows the app's interface after it receives an interview trigger. It provides the interviewer with information about the trigger that was just received, including the trigger's description, which student triggered the interview prompt and the time at which the trigger was detected.

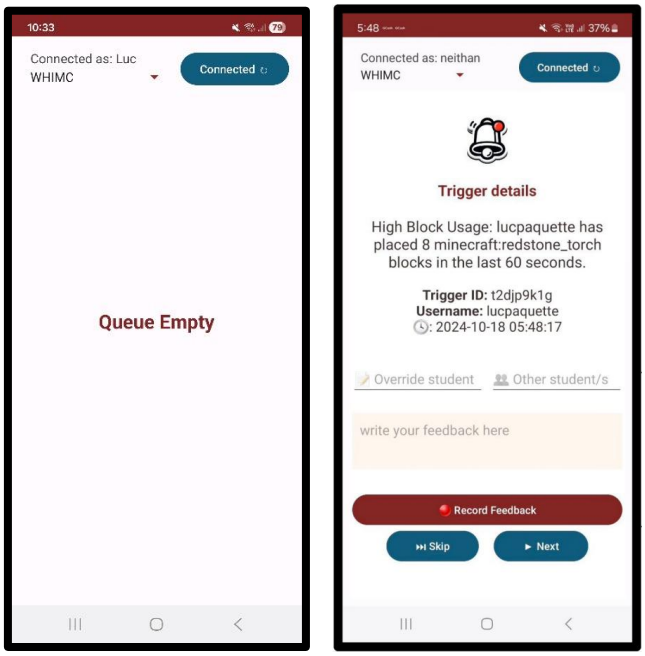


Figure 2. The QRF app. On the left, the interviewer connected to the QRF server and is waiting for an interview trigger. On the right, the QRF server sent an interview trigger to the interviewer.

Once an interview prompt is provided to the interviewer, they can review the prompt to decide whether they would prefer to skip this interview, for example when the student's attention is already occupied and they should not be disturbed. If the interviewer decides to proceed with the interview, it can be recorded directly through the app by pushing the "Record Feedback" button (we recommend using an external, directional microphone that can help with reducing background noise). Audio recordings are stored locally on the phone. In addition, the QRF app can be configured to automatically send the recordings to a data server. Interviewers can also add text notes to provide additional contextual information about the interview. They can "override" the interviewed student if they have observed a salient event from another student that they wish to interview as soon as possible. Through our past interviews, we have also observed that neighboring students sometimes join ongoing interviews. In such a case, the app allows the interviewer to make explicit note of which "Other student/s" joined the interview. Finally, once an interview is completed, the interviewer pushes the "Next" button. The app then sends information about the interview to the QRF server and goes back to waiting for the next available interview prompt.

3.2 The QRF server

The central component of the QRF server is the dispatcher program which is responsible for receiving interview triggers from the learning system, keeping track of active triggers for each student and dispatching triggers to classroom interviewers.

Communication between the learning environment, the dispatcher program and the QRF app uses the websocket communication protocol. When the learning environment detects a situation that was identified as an interview trigger, it sends information about this trigger (its description, what students triggered it, and its priority) to the dispatcher. The dispatcher receives the triggers and organizes them in a queue, based on their priority (a numerical value provided by the learning environment along with the trigger). Triggers are maintained in the queue for a predetermined amount of time (can be adjusted by the researchers) before they timeout and are removed from the queue. This mechanism prevents interviewers from being asked to interview students for a trigger that happened too long ago and is no longer relevant.

When the QRF app notifies the dispatcher program that an interviewer is available, the dispatcher selects the trigger with the highest priority and sends it to the QRF app. In addition to the numerical priority of the triggers, the dispatcher implements mechanisms to avoid frequent repeated interviews of the same student. The dispatcher can be configured to enforce a minimum waiting period between the end of an interview and the next interview request for the same student. This mechanism is designed to minimize situations where a student might experience interview fatigue and maximize the chances that all students are interviewed during data collection.

Throughout this process, the dispatcher program logs all the received triggers to a Firebase database to keep records of which trigger was received, for which student and at what times. It also logs information about which interviewer (if any) received the trigger on their QRF app, and whether the interviewer decided to skip the interview request. If an interview is conducted, the dispatcher also logs metadata about the interview (e.g., when did the interview start, how long did it last, did the interviewer add any text note to the interview, etc.)

3.3 The QRF dashboard

The QRF dashboard is a web browser-based interface that allows the research team to monitor interview triggers and interviews during data collection. The dashboard can be configured to interface with both the Firebase database (storing information about the detected interview triggers and the resulting interviews) and the data server used to store audio recordings (if a data server is used).

The dashboard displays a table with information about each trigger that has been sent to the dispatcher program. It is automatically updated at regular intervals to display the most recent triggers. During data collection, the dashboard is typically used by the research team as a way to verify that the QRF system is functioning as expected (e.g., triggers are being received by the dispatcher and logged in the database), and to monitor which triggers are being received by the dispatcher and which ones are sent to the interviewers. Such real-time monitoring of the received trigger can be useful to identify whether any adjustments of trigger priorities might be warranted during data collection to increase the likelihood that a rare, but important trigger will be forwarded to interviewers. If audio recordings are being uploaded to a data server, the dashboard can also be configured to access the audio recordings, allowing the research team to playback interviews from the dashboard.

4. DDCI IMPLMENTATION IN WHIMC

In this demo, we will show an example of how the QRF system has been integrated with WHIMC (What-if Hypothetical Implementations in Minecraft) [13], a Minecraft-based learning system in which students learn about astronomy while exploring virtual recreation of hypothetical Earths (e.g., what would Earth be like if there were no moon). The WHIMC system includes two main types of learning activities: exploration of what-if worlds and building a space habitat on Mars.

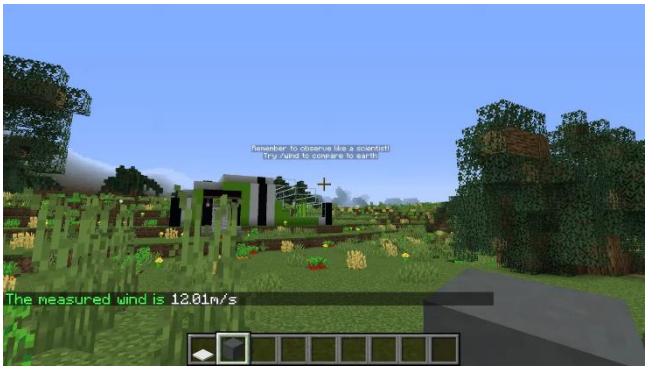


Figure 3. Screen captured from the WHIMC system. The student used one of the available measurement tools to measure the speed of the wind in the current world.



Figure 4. Screen captured from the WHIMC system. The student made a new observation about the color of the sun in this world.

During what-if exploration worlds, students are asked to explore a virtual recreation of a what-if Earth and to investigate how this Earth differs from ours. During their exploration, students can, at any moment, use scientific measurement tools (e.g. Figure 3) to collect data about different aspects of the what-if world (e.g., its gravity, wind speed, temperature, etc.). They can also write down observations (e.g. Figure 4) that are displayed in the world, at the coordinates where the observation was made. Throughout their exploration students may encounter various non-player characters (NPCs) and points of interest (e.g., a greenhouse, a windmill, etc.) that provides them with clues about the differences between the virtual world they are exploring and our Earth, and about which measurement tool might provide them with relevant information about these differences. Students typically explore each exploration world for a short period of time (around 20 minutes).

During the Mars habitat building activity, students interact with the recreation of a part of Mars. They work in small groups (2-3 students) with the goal of building structures that could be used as a habitat on Mars. Students are given instructions about characteristics of a habitat that would make it appropriate for life on Mars (e.g., does the habitat have a source of food and water, does it have a source of electric power, etc.) Students typically work on their Mars habitat for 3-4 hours across 2 days.

As part of this demo, participants will be able to interact with the WHIMC learning environment. They will be shown how their actions in WHIMC (e.g., using a scientific tool near an NPC or using a specific type of block during Mars habitat building) activate predetermined interview triggers requests. They will be able to interact with the QRF app to record short interviews and see how logs and audio recordings of their interviews are being reported on the QRF dashboard. The demo will also show an example of a simple Python script illustrating how triggers are sent to the QRF server from the learning environment.

5. ACKNOWLEDGMENTS

This study was supported by the National Science Foundation (NSF; DRL-2301172). Any conclusions expressed in this material do not necessarily reflect the views of the NSF.

6. REFERENCES

[1] Baker. R.S., and Yacef, K. 2009. The state of educational data mining in 2009: A review and future visions. *Journal of Educational Data Mining*, 1, 1, 3-17. DOI=https://doi.org/10.5281/zenodo.3554657

[2] Jia, Q., Young, M., Xiao, Y., Cui, J., Liu, C., Rashid, P., and Gehringer, E. 2022. Automated feedback generation for student project reports: A data-driven approach. *Journal of Educational Data Mining*, 14, 3, 132-161. DOI=https://doi.org/10.5281/zenodo.7304954

[3] Paaßen, B., Hammer, B., Price, T. W., Barnes, T., Gross, S., and Pinkwart, N. 2017. The continuous hint factory - Providing hints in vast and sparsely populated edit distance spaces. *Journal of Educational Data Mining*, 10, 1, 1-35. DOI=https://doi.org/10.5281/zenodo.3554697

[4] Borchers, C., Peng, C., Lyu, Q., Carvalho, P. F., Koedinger, K. R., and Aleven, V. 2025. Student perceptions of adaptive goal setting recommendations: A design prototyping study. In *International Conference on Artificial Intelligence in Education*, Cham: Springer Nature Switzerland, Palermo, Italy, 244-251. DOI=https://doi.org/10.1007/978-3-031-98462-4_31

[5] De Menezes, M. F., Netto, J. F. D. M., Lopes, A. M. M., Silva, R. C., and Menezes, R. A. 2023. The proposal of a

dashboard for analysis and visualization of educational data of intelligent tutoring systems. In *2023 IEEE Frontiers in Education Conference (FIE)*, IEEE, College Station, TX, USA, 1-8. DOI=https://doi.org/10.1109/FIE58773.2023.10343000

[6] Pelánek, R., Effenberger, T., and Kukučka, A. 2022. Towards design-loop adaptivity: Identifying items for revision. *Journal of Educational Data Mining*, 14, 3, 1-25. DOI=https://doi.org/10.5281/zenodo.7357331

[7] Kerr, D. 2015. Using data mining results to improve educational video game design. *Journal of Educational Data Mining*, 7, 3, 1-17. DOI=https://doi.org/10.5281/zenodo.3554723

[8] Liu, R., and Koedinger, K.R., 2017. Closing the loop: Automated data-driven cognitive model discoveries lead to improved instruction and learning gains. *Journal of Educational Data Mining*, 9, 1, 25-41. DOI=https://doi.org/10.5281/zenodo.3554625

[9] Ericsson, K.A., and Simon, H.A. 1980. Verbal reports as data. *Psychological review*, 87, 3, 215-251. DOI= https://psycnet.apa.org/doi/10.1037/0033-295X.87.3.215

[10] Boren, T., and Ramey, J. 2000. Thinking aloud: Reconciling theory and practice. *IEEE Transactions on Professional Communication*, 43, 3, 261-278. DOI= https://doi.org/10.1109/47.867942

[11] Ocumpaugh, J., Paquette, L., Baker, R.S., Barany, A., Ginger, J., Casano, N., Zambrano, A., Liu, X., Wei, Z., Zhou, Y., Liu, S., Hutt, S., Andres, A., Nasiar, N., Giordano, C., van Velsen, and M., Mogessie, M. 2025. The Quick Red Fox gets the best Data Driven Classroom Interviews: A manual for an interview app and its associated methodology. https://arxiv.org/pdf/2511.13466

[12] Ocumpaugh, J., Roscoe, R.D., Baker, R.S., Hutt, S., and Aguilar, S.J. 2024. Toward asset-based instruction and assessment in artificial intelligence in education. *International Journal of Artificial Intelligence in Education*, 34, 1559-1598. DOI= https://doi.org/10.1007/s40593-023-00382-x

[13] Lane, HC., Gadbury, M., Ginger, J., Yi, S., Comins, N., Henhapl, J., and Rivera-Rogers, A. 2022. Triggering STEM interest with Minecraft in a hybrid summer camp. *Technology, Min, and Behavior*, 3, 4, 580-597. DOI= https://doi.org/10.1037/tmb0000077

Accounting for Classroom Effects in Student-Level Randomized Online Experiments: The Ordered LOOP Estimator

Yanping Pei
Worcester Polytechnic Institute
ypei@wpi.edu

Adam Sales
Worcester Polytechnic Institute
asales@wpi.edu

Johann Gagnon-Bartsch
University of Michigan
johanngb@umich.edu

Eamon Worden
Worcester Polytechnic Institute
elworden@wpi.edu

Neil Heffernan
Worcester Polytechnic Institute
nth@wpi.edu

ABSTRACT

Digital learning platforms enable large-scale randomized controlled trials (RCTs) in real classroom settings, where students are nested within classrooms and outcomes are shaped by both individual characteristics and shared environments. Although randomization at the student level helps balance these factors, differences across classrooms may still persist, motivating the need for covariate adjustment.

In this paper, we study whether and how accounting for classroom structure improves treatment effect estimation. Standard approaches, such as fixed- and random-effects models, adjust for classroom variation and can improve precision, but they rely on parametric assumptions and may fail to capture complex or nonlinear relationships.

We build on the Leave-One-Out Potential outcomes (LOOP) estimator, which uses machine learning to flexibly model outcomes while preserving unbiasedness. However, LOOP struggles with high-cardinality categorical variables, such as classroom identifiers. To address this limitation, we propose *Ordered LOOP*, a method designed to better incorporate such variables.

Through simulations and real data applications, we show that accounting for classroom structure improves precision. Ordered LOOP performs particularly well when outcome heterogeneity is complex or nonlinear, while remaining competitive in simpler settings. Overall, our results highlight the importance of modeling classroom structure and demonstrate that Ordered LOOP provides a flexible, robust, and unbiased approach for analyzing hierarchical educational data.

Yanping Pei, Adam Sales, and Johann Gagnon-Bartsch. Accounting for Classroom Effects in Student-Level Randomized Online Experiments: The Ordered LOOP Estimator. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 589–593. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039790

Keywords

Causal inference, A/B testing, Online learning platforms, Hierarchical modeling, Covariate adjustment

1. INTRODUCTION

Digital learning platforms allow researchers to conduct large-scale A/B tests and collect rich student-level data in real classroom settings. As their use grows, so does the need for methods that deliver valid and precise causal estimates in real-world environments. A key challenge is the hierarchical structure of educational data: students are nested within classrooms, teachers, and schools, and outcomes reflect both individual characteristics and shared learning environments [2].

Although student-level randomization is intended to balance these factors, real-world experiments often fall short. For example, analyses of ASSISTments experiments [6] show that many classrooms include only a single participating student [11], limiting the effectiveness of randomization and motivating covariate adjustment. Prior work [2] shows that accounting for classroom structure using fixed- or random-effects models can improve precision. However, these approaches rely on parametric assumptions and may fail to capture complex, nonlinear, or interactive relationships, especially when group structures are unknown.

To address these limitations, we study the Leave-One-Out Potential outcomes (LOOP) estimator [4, 14], which uses machine learning to flexibly model outcomes while preserving unbiasedness. Random forests are well suited to this framework because they naturally produce out-of-bag predictions aligned with the leave-one-out structure. However, standard implementations struggle with high-cardinality categorical variables, such as classroom identifiers, due to computational constraints.

We propose *Ordered LOOP*, an extension that enables effective adjustment for such variables. We evaluate the method using simulations based on real-world data and real experimental datasets [12] . Our results show that accounting for classroom structure improves precision, and that Ordered LOOP performs particularly well when classroom effects are complex or nonlinear, while remaining competitive in sim-

pler settings. In these complex settings, it outperforms common alternatives, including OLS, OLS with classroom fixed effects, and hierarchical linear models with random effects.

The remainder of the paper is organized as follows. The next section reviews the potential outcomes framework, LOOP, and Ordered LOOP. Section 3 describes the simulation design, Section 4 presents the results, and Section 5 concludes.

2. CAUSAL ESTIMATOR

Consider a randomized controlled trial with N units indexed by $i = 1, \ldots, N$. Treatment assignments $tr_i \in \{0, 1\}$ are independent across units, with $P(tr_i = 1) = p_i \in (0, 1)$. For each unit, we observe outcome Y_i and covariates Z_i. Under the potential outcomes framework [8], unit i has potential outcomes t_i and c_i, and the observed outcome is $Y_i = tr_i t_i + (1 - tr_i)c_i$. The individual treatment effect is $\tau_i = t_i - c_i$, and the average treatment effect is $\bar{\tau} = \frac{1}{N}\sum_{i=1}^{N} \tau_i$.

2.1 Leave-One-Out Potential outcome

Define the counterfactual expected outcome as $m_i = (1 - p_i)t_i + p_i c_i$, with estimate $\hat{m}_i$. Let the inverse probability weight be $U_i = \frac{1}{p_i}$ if $tr_i = 1$ and $U_i = \frac{1}{1-p_i}$ if $tr_i = 0$. The unit-level estimator $\hat{\tau}_i = (Y_i - \hat{m}_i)U_i$ is unbiased for τ_i when $\hat{m}_i$ is independent of tr_i [14]. The average treatment effect is estimated by $\hat{\tau} = \frac{1}{N}\sum_{i=1}^{N} \hat{\tau}_i$, and the estimator has bounded variance when $\hat{m}_i \approx m_i$ [14].

In practice, $\hat{m}_i$ is obtained via leave-one-out prediction: for each unit i, a model trained on the remaining $N - 1$ units produces predictions $\hat{t}_i$ and $\hat{c}_i$, with $\hat{m}_i = (1 - p_i)\hat{t}_i + p_i\hat{c}_i$. Flexible methods such as random forests can capture nonlinearities and interactions while preserving unbiasedness, and naturally provide out-of-bag predictions aligned with this procedure.

2.2 Ordered LOOP

Standard random forest implementations face significant computational challenges when handling high-cardinality categorical predictors. For a predictor with k categories, the classical splitting rule considers $2^{k-1} - 1$ possible partitions, which quickly becomes infeasible even for moderate k. Unlike continuous or ordinal variables, these splits cannot be represented by a single threshold. In practice, implementations often encode category assignments using bit representations, imposing hard limits (e.g., 32 or 64 levels) on the number of supported categories.

A common solution is to impose an ordering on categorical levels. For regression and binary classification, ordering categories by their mean response or class proportion yields the same optimal split as exhaustive search, while reducing the number of candidate splits from $2^{k-1} - 1$ to $k - 1$ [1, 3, 7]. When this ordering is computed once and reused throughout tree construction, splits can be implemented as simple thresholds, leading to substantial computational gains with minimal loss in predictive performance [13].

We adopt the `ranger` implementation, which supports this approach via `respect.unordered.factors = "order"`. In this setting, factor levels are ordered once based on their mean response (for regression) or outcome class proportions (for classification), and are not reordered at each split [5], where "class" refers to the response category rather than classroom groupings.

Incorporating ordered factors into LOOP introduces two challenges. First, the imputed regression function $\hat{m}_i$ must remain independent of each observation's outcome. To ensure this, we use K-fold cross-fitting: the data are randomly partitioned into K folds, and for each fold, separate models are trained on the remaining $K - 1$ folds for the treated and control groups. Out-of-fold predictions $\widehat{\mu}_1(tr) = \mathbb{E}[Y(1) \mid tr]$ and $\widehat{\mu}_0(tr) = \mathbb{E}[Y(0) \mid tr]$ are then obtained for the held-out observations.

Second, arm-specific K-fold training can produce categorical levels in the test fold that are absent from the corresponding training data, leading to unstable predictions under ordered encoding. To address this, we adopt a row-level fallback strategy: for observations with unseen levels, we use predictions from a numeric-only imputation model (standard LOOP), while retaining ordered-`ranger` predictions for all other observations.

This hybrid approach leverages high-cardinality categorical information while ensuring stability and preserving the flexibility and unbiasedness of the LOOP estimator. When high-cardinality categorical variables are informative about the outcome, Ordered LOOP can exploit the induced ordering structure to reduce variance, effectively lowering the standard error and improving precision relative to standard LOOP. In settings where such structure is weak or absent, its performance remains comparable to LOOP.

3. SIMULATION DESIGN

Evaluating treatment effect estimators requires ground truth, which is unavailable in real experiments. We therefore simulate realistic settings using *remnant* [4, 9, 10] logs, which consist of observational student–assignment records aligned with experimental contexts.

3.1 Focal Measures

We use three variables: (i) student performance (inverse mastery speed, bounded in $[0, 1/3]$, with 0 indicating non-completion [2]), (ii) prior performance (prior percent correct), and (iii) class membership (de-identified classroom ID).

3.2 Data Generation

We generate synthetic A/B tests using a bootstrap-style resampling procedure applied to the remnant data. This approach preserves the empirical distributions of class sizes, class-level heterogeneity, and student-level noise, while allowing controlled specification of treatment effects and outcome models.

We consider two data-generating processes. First, we replicate the baseline design of Closser et al. [2]. Second, we construct a more realistic setting in which class membership interacts with covariates and induces nonlinear outcome relationships. In this setting, standard fixed- and random-effects models are misspecified, allowing us to assess whether flexible machine learning methods can better capture the underlying structure.

3.2.1 Linear relationship

Under the linear specification, outcomes are generated as

$$\begin{aligned}\text{true_outcome} = \text{class_coef} + 0.16 \times \text{prior_percent_correct} \\ + \text{effect_size} \times \text{condition} + \eta. \end{aligned} \quad (1)$$

Parameters are calibrated from the remnant data. We fit three models relating inverse mastery speed to prior percent correct: (i) OLS, (ii) OLS with class fixed effects, and (iii) a mixed-effects model with class-level random intercepts. The class-specific intercepts from the fixed-effects model define class_coef, capturing empirical classroom heterogeneity without imposing distributional and independence assumptions. The coefficient on prior performance is set to the average estimate across the three models (0.16).

Student-level noise η is generated by resampling residuals from the mixed-effects model, which provides a parsimonious decomposition of within- and between-class variation.

Each simulated dataset is constructed by sampling n_{classes} classrooms with replacement from the remnant data and retaining all associated observations. For each classroom, we draw a class-specific effect by resampling from the empirical distribution of the estimated class-level coefficients class_coef obtained from the observed data.

At the observation level, we resample η and assign treatment independently with probability 0.5. Outcomes are then generated according to Equation (1).

3.2.2 Nonlinear relationship

Under the nonlinear specification, we introduce a class-relative threshold effect,

$$\begin{aligned}\text{true_outcome} = \text{class_coef} + 0.16 \times \text{prior_percent_correct} \\ - I\{\text{prior_percent_correct} > \text{class_coef}\} \\ + I\{\text{prior_percent_correct} < \text{class_coef}\} \\ + \text{effect_size} \times \text{condition} + \eta. \end{aligned} \quad (2)$$

This component captures class-relative dynamics in which outcomes depend on a student's performance relative to the class baseline. Higher-performing students may experience smaller gains (e.g., ceiling effects), while lower-performing students may benefit more from instruction, inducing a nonlinear pull toward the class mean.

Overall, this simulation framework preserves key empirical features of the remnant data while enabling controlled comparisons of estimator performance under both linear and nonlinear outcome-generating mechanisms.

4. RESULTS

The models evaluated in this study are summarized in Table 1 and are selected to address our central question: whether and how treatment effect estimation should account for the clustering of students within classrooms. We consider three classes of methods. First, we include approaches that ignore classroom structure, namely the two-sample t test and standard LOOP; the latter cannot incorporate classroom indicators when the number of categories is large (e.g., more than 53 levels). Second, we examine parametric models with fixed or random intercepts. Third, we evaluate Ordered LOOP, a machine-learning approach that flexibly captures classroom heterogeneity. Our goal is to assess when Ordered LOOP improves upon these alternatives.

4.1 Simulation Studies

All simulations fix the true treatment effect at 0.2 and are repeated 100 times.

4.1.1 Linear setting.

Table 2 reports results under the linear data-generating process. All methods recover the true effect accurately, but differ in precision. Methods that account for classroom structure achieve smaller standard errors than those that ignore it.

The random-effects model achieves the lowest variance, as expected under correct model specification. Ordered LOOP ranks second, indicating that it remains competitive even when the true outcome model is linear and fixed- or random-effects models are correctly specified.

In contrast, the standard LOOP estimator with covariate adjustment performs worse than the baseline t test. To better understand this result, we examined the imputation step $\hat{m}$ in LOOP and found that the random forest used for outcome prediction explains a negative proportion of the outcome variance (-25.02%). This indicates that the prediction model fails to capture useful signal in this setting, which in turn explains why covariate adjustment through LOOP does not improve—and in fact degrades—performance relative to the simple t test.

Overall, in the linear scenario, our findings are consistent with those of Closser et al. [2]: methods that explicitly account for classroom structure outperform approaches that ignore group-level heterogeneity, even when treatment is randomized at the student level. At the same time, Ordered LOOP remains a competitive alternative without requiring parametric assumptions about classroom effects.

4.1.2 Nonlinear setting.

Table 3 summarizes results under the nonlinear data-generating process. In this setting, parametric models are misspecified, and performance differences become more pronounced.

All methods recover the true effect accurately again. Ordered LOOP achieves the lowest standard error, followed by standard LOOP, demonstrating the advantage of flexible machine learning methods in capturing nonlinear and interaction effects. In contrast, fixed- and random-effects models are less efficient due to model misspecification.

Notably, standard LOOP performs well despite not explicitly including classroom indicators, suggesting that flexible modeling can partially compensate for missing group structure. However, incorporating classroom information through Ordered LOOP yields additional gains.

Across both settings, accounting for classroom structure consistently improves precision. Ordered LOOP performs ro-

Table 1: Modeling methods overview

Method	Equation	Assumptions
t-test	$Y_{ij} = \beta_0 + \beta_1 tr_{ij} + \epsilon_{ij}$	$\mathbb{E}(\epsilon_{ij} \mid tr_{ij}) = 0$
OLS + fixed class intercepts	$Y_{ij} = \beta_{0j} + \beta_1 tr_{ij} + \beta_2 Z_{ij} + \epsilon_{ij}$	$\mathbb{E}(\epsilon_{ij} \mid tr_{ij}, Z_{ij}, \text{Class}_j) = 0$
HLM + random class intercept	$Y_{ij} = \beta_{0j} + \beta_1 tr_{ij} + \beta_2 Z_{ij} + \epsilon_{ij}$ where $\beta_{0j} = \gamma_{00} + \eta_j$	$\epsilon_{ij} \sim \mathcal{N}(0, \sigma)$ and $\eta_j \sim \mathcal{N}(0, \tau)$
LOOP	$\hat{\tau} = \frac{1}{N} \sum_{i=1}^{N} \hat{\tau}_i$ where $\hat{\tau}_i = (Y_i - \hat{m}_i)U_i$ ignoring classes	Unbiased with good precision when $\hat{m}_i \perp U_i$ and $\hat{m}_i \approx m_i$
Ordered LOOP	Same as LOOP while $\hat{m}_i$ is estimated using ordered random forests in `ranger` with classes	Same as LOOP

Table 2: Average treatment effect estimates and standard errors under the linear outcome model.

Method	t-test	FE	RE	LOOP	Ordered LOOP
Estimate	0.1995	0.1995	0.1994	0.1992	0.1994
Std.Error	0.0030	0.0029	**0.0025**	0.0033	**0.0028**

Note: Lowest standard errors are shown in bold.

Table 3: Average treatment effect estimates and standard errors under the nonlinear outcome model.

Method	t-test	FE	RE	LOOP	Ordered LOOP
Estimate	0.2003	0.2003	0.2003	0.1995	0.1996
Std.Error	0.0063	0.0059	0.0050	**0.0041**	**0.0033**

bustly, remaining competitive in linear settings and outperforming alternatives in nonlinear settings, highlighting its effectiveness across both simple and complex outcome-generating processes.

4.2 Real-World Applications

4.2.1 Remnant data application

We construct a semi-synthetic dataset by assigning treatment at random, $tr_i \sim$ Bernoulli(0.5), since the remnant data contain no real treatment. Results are reported in Table 4. As expected, all estimates are close to zero.

Table 4: Average treatment effect estimates and standard errors on remnant data.

Method	t-test	FE	RF	LOOP	Ordered LOOP
Estimate	-0.0004	-0.0002	-0.0002	−0.0001	−0.0000
Std.Error.	0.0016	**0.0015**	**0.0014**	0.0016	**0.0015**

Table 4 shows that methods accounting for classroom structure outperform those that ignore it: the hierarchical linear model achieves the smallest standard error, followed by Ordered LOOP and OLS with classroom fixed effects. Ordered LOOP is unbiased by construction, whereas the random-effects model relies on distributional assumptions that may not hold in practice.

4.2.2 LLM-based experiment.

We further evaluate all methods on a real experiment involving GPT-4–generated explanations [12]. Results are reported in Table 5.

Table 5: Average treatment effect estimates and standard errors on LLM-generated data.

Method	t-test	FE	RE	LOOP	Ordered LOOP
Estimate	0.0163	0.0083	0.0065	0.0102	0.0086
Std.Error	0.0083	0.0083	0.0112	0.0082	**0.0078**

The point estimates differ across methods, and because this is a real A/B test, the true effect is unknown. Theoretically, the t-test, fixed-effects model, LOOP, and Ordered LOOP are unbiased, while the random-effects model may be biased under misspecification.

Accounting for classroom clustering generally improves precision, but model misspecification can substantially increase variance. Ordered LOOP achieves the lowest standard error, outperforming the t-test baseline (0.0078 vs. 0.0083). In contrast, the random-effects model exhibits the largest variance, with a standard error 44% higher than that of Ordered LOOP (0.0112 vs. 0.0078), likely due to misspecification of its underlying assumptions. This illustrates a key limitation of parametric approaches: when the true data-generating process is unknown, misspecification can reduce robustness and lead to poorer performance. In contrast, more flexible, machine-learning–based methods that rely less on parametric assumptions tend to produce more stable and efficient estimates in such settings.

Across simulations and real data, two patterns emerge: accounting for classroom structure improves efficiency, and Ordered LOOP provides a robust and flexible approach that matches correctly specified parametric models and outperforms them when such models are difficult to specify. We therefore recommend Ordered LOOP for reliable performance across a wide range of settings.

5. DISCUSSION AND CONCLUSION

We study treatment effect estimation in online educational experiments with student-level randomization, focusing on classroom-level heterogeneity. Across simulations and real-world applications, methods that account for classroom structure, including fixed effects, random effects, and Ordered LOOP, consistently outperform those that ignore it, indicating that randomization alone may be insufficient for precise estimation.

When classroom effects are approximately linear, fixed- and random-effects models perform best under correct specifi-

cation [2]. However, when effects are nonlinear or interact with covariates, parametric models may be misspecified and lose efficiency. In such settings, Ordered LOOP flexibly captures complex relationships while remaining competitive in simpler cases and retaining interpretability for estimating the average treatment effect.

Overall, accounting for clustering is essential for precision. Ordered LOOP offers a robust and flexible approach without requiring strong parametric assumptions. Future work could combine LOOP-style imputation with parametric models to further improve efficiency and interpretability.

6. ACKNOWLEDGMENTS

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305D210031. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education.

7. REFERENCES

[1] L. Breiman, J. Friedman, R. A. Olshen, and C. J. Stone. *Classification and regression trees*. Chapman and Hall/CRC, 2017.

[2] A. H. Closser, A. Sales, and A. F. Botelho. Should we account for classrooms? analyzing online experimental data with student-level randomization. *Educational technology research and development*, 72(5):2865–2894, 2024.

[3] W. D. Fisher. On grouping for maximum homogeneity. *Journal of the American statistical Association*, 53(284):789–798, 1958.

[4] J. A. Gagnon-Bartsch, A. C. Sales, E. Wu, A. F. Botelho, J. A. Erickson, L. W. Miratrix, and N. T. Heffernan. Precise unbiased estimation in randomized experiments using auxiliary observational data. *Journal of Causal Inference*, 11(1):20220011, 2023.

[5] T. Hastie, R. Tibshirani, J. Friedman, et al. The elements of statistical learning, 2009.

[6] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[7] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.

[8] D. B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

[9] A. Sales, A. Botelho, T. Patikorn, and N. T. Heffernan. Using big data to sharpen design-based inference in a/b tests. In *Proceedings of the eleventh international conference on educational data mining*, 2018.

[10] A. C. Sales, B. B. Hansen, and B. Rowan. Rebar: Reinforcing a matching estimator with predictions from high-dimensional covariates. *Journal of Educational and Behavioral Statistics*, 43(1):3–31, 2018.

[11] D. Selent, T. Patikorn, and N. Heffernan. Assistments dataset from multiple randomized controlled experiments. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pages 181–184, 2016.

[12] E. Worden, K. Vanacore, A. Haim, and N. Heffernan. Scaling effective ai-generated explanations for middle school mathematics in online learning platforms. In *Proceedings of the Twelfth ACM Conference on Learning@ Scale*, pages 40–49, 2025.

[13] M. N. Wright and I. R. König. Splitting on categorical predictors in random forests. *PeerJ*, 7:e6339, 2019.

[14] E. Wu and J. A. Gagnon-Bartsch. The loop estimator: Adjusting for covariates in randomized experiments. *Evaluation review*, 42(4):458–488, 2018.

Automating Literature Screening with Multi-Agent Consensus Seeking: A Preliminary Study

Deliang Wang
Faculty of Education
The University of Hong Kong
Hong Kong SAR
wdeliang@connect.hku.hk

Xian Chen
School of Environment, Education and Development
The University of Manchester
UK
metaxianchen@gmail.com

Jiaying (Joy) Yu
Ontario Institute for Studies in Education
The University of Toronto
Canada
joyy.yu@mail.utoronto.ca

Gaowei Chen
Faculty of Education
The University of Hong Kong
Hong Kong SAR
gwchen@hku.hk

ABSTRACT

The automation of systematic reviews (SRs) with large language models (LLMs) offers substantial potential to reduce the labor-intensive demands of manual screening. Yet, single-agent LLMs are prone to unilateral bias and critical omissions (false negatives), particularly when dealing with the contextualized and often subjective constructs characteristic of educational research. This study proposes a heterogeneous dual-agent, consensus-seeking workflow that emulates human double-blind screening followed by inter-coder reconciliation. Using DeepSeek-V3.2 and Qwen3.5-35B as independent screening agents, the system first conducts separate title–abstract evaluations and then applies an automated negotiation mechanism to resolve disagreements. Tested on a dataset of 2,107 articles extracted from the Web of Science across five SRs, the multi-agent framework successfully outperformed both single-agent baselines in accuracy, recall, and F2 scores. Concurrently, it maintained strong workload reduction, accurately excluding 81% of irrelevant records from human review. Variance analyses further demonstrated that the debate mechanism improved the stability of screening performance across heterogeneous data subsets. Although the absolute performance metrics remain moderate—highlighting the inherent difficulty of fully automating literature screening for complex constructs—these preliminary findings provide promising empirical support. They establish multi-agent collaboration as a more rigorous and reliable paradigm for advancing LLM-assisted evidence synthesis in education.

Keywords

Large Language Models, Multi-Agent Systems, Systematic Review, Literature Screening, Consensus Mechanism

Deliang Wang, Xian Chen, Joy Yu, and Gaowei Chen. Automating Literature Screening with Multi-Agent Consensus Seeking: A Preliminary Study. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 594–599. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040068

1. INTRODUCTION

A literature review, as a scholarly paper, provides an overview of current knowledge on a given topic by summarizing and synthesizing the substantive findings of existing research, as well as highlighting theoretical and methodological contributions [13]. Accordingly, literature reviews serve to establish what is known and unknown about a particular phenomenon and to identify areas where further research is required. Based on differences in their search and analysis characteristics, literature reviews can be classified into several types, including rapid reviews, scoping reviews, systematic reviews, meta-analyses, and umbrella reviews, among others [8].

Among these, systematic reviews are among the most frequently employed approaches. They aim to collate all empirical evidence that meets pre-specified eligibility criteria in order to answer specific, clearly formulated research questions [2]. By employing explicit and reproducible systematic methods—such as the PRISMA framework [15])—systematic reviews seek to minimize bias and provide reliable findings from which conclusions can be drawn and informed decisions made. Due to their transparency, accountability, and replicability, systematic reviews have gained increasing popularity within the field of education.

Typically, systematic reviews involve a standardized set of procedures. Retrieved studies are screened for eligibility using predetermined inclusion and exclusion criteria, a process strictly requiring at least two human reviewers working independently to ensure reliability and mitigate individual bias [16]. However, it has been found that conducting a systematic review takes approximately 67 weeks from registration to publication [3]. In research areas characterized by rapid technological change, such a prolonged timeframe may render an unpublished systematic review outdated. Consequently, researchers have begun exploring automated methods to replace human effort in the labor-intensive screening phases [12].

The emergence of large language models (LLMs) presents promising opportunities to expedite this process. Recently, [6] systematically evaluated 18 different LLMs for title-and-abstract screening, demonstrating their substantial potential to reduce workload. However, their findings also highlighted significant variability among models; relying on a single LLM (a single-agent approach) often yields inconsistent accuracy, risking the exclusion of relevant studies or the over-inclusion of irrelevant ones. Furthermore, assigning a single LLM to conduct the screening fundamentally violates the methodological requirement of having multiple independent reviewers, raising concerns about the dependability and credibility of the automated screening process.

To address the limitations of single-agent approaches, recent advancements in qualitative data analysis have introduced multi-agent systems (MAS) (e.g., [17, 18]. In a multi-agent framework, multiple LLMs are instantiated with distinct roles, allowing them to collaborate, cross-verify, and engage in social behaviors such as debating and consensus-seeking [1, 26]. By simulating human collaborative workflows, multi-agent systems can effectively reduce the inherent biases and hallucinations of individual models, yielding far more robust and rigorous analytical outcomes.

Despite these advancements, there is a lack of empirical research examining the application of multi-agent LLM systems to systematic reviews, particularly within the education domain. It remains unclear whether simulating the human dual-reviewer workflow using heterogeneous LLMs (i.e., different underlying models acting as independent coders) can resolve the inconsistency issues identified in prior studies. Therefore, in this preliminary study, we aim to investigate the effectiveness of leveraging a multi-agent framework to facilitate the title-abstract screening process for educational systematic reviews. Specifically, we employ distinct LLMs to act as independent coders, evaluating their individual screening performance and their ability to reach a consensus through simulated inter-coder discussion, thereby paving the way for a more reliable, automated approach to evidence synthesis.

2. RELATED WORK

2.1 LLMs and Multi-Agent Systems in Qualitative Research

LLMs are increasingly utilized in qualitative data analysis (QDA). Early research explored single LLMs as research assistants, demonstrating their ability to simulate thematic analysis [5] and develop codebooks in educational contexts [25, 11]. While single-agent approaches improve efficiency and clarify coding definitions [25], their performance heavily depends on prompting strategies and construct complexity [11, 22, 21, 23]. Furthermore, they remain susceptible to inherent biases and hallucinations when processing complex logic.

To overcome these limitations, research is shifting toward LLM-based multi-agent systems (MAS). By assigning specific roles to different LLMs, MAS can automate complex workflows. For example, [18] utilized task-specific agents for text summarization and theme extraction, showing that this "division of labor" accelerates QDA and ensures analytical validity by approximating the rigor of human research teams.

Crucially, MAS can simulate human "triangulation" and peer-review processes via built-in collaboration. Introducing "debate" and "reflection" workflows enables agents to exhibit consensus-seeking behaviors and reach agreements through negotiation [26, 4]. In thematic analysis, employing multi-agent collaboration (e.g., independent coder roles) significantly enhances the credibility and dependability of findings compared to single agents [17]. Collectively, these mechanisms are critical for improving the rigor of automated QDA.

2.2 Automating Systematic Reviews via LLM Agents

LLMs also present opportunities to accelerate the highly time-consuming systematic review (SR) process, alleviating the burden of manual screening and data extraction. Despite methodological challenges like model bias and hallucinations, researchers are exploring end-to-end automated systems that integrate screening and bias assessment functionalities.

For title and abstract screening, single LLMs show promise but yield mixed results. While GPT-4 achieved a 100% recall rate in environmental science SRs [14], studies evaluating multiple cutting-edge models in medical SRs noted significant discrepancies in false negative and positive rates, alongside substantial instability in identifying relevant literature [19, 6]. Furthermore, extraction accuracy for complex variables remains highly domain-dependent [9].

Despite this potential, two salient research gaps remain. First, the efficacy of LLMs for SRs in the education domain is underexplored. Unlike objective clinical metrics, educational literature involves highly contextualized and subjective constructs, leaving it unclear whether LLMs can reliably apply complex inclusion/exclusion criteria here. Second, most automated SR research relies on single-agent frameworks. This unilateral decision-making is susceptible to inherent biases and fundamentally violates the SR methodological requirement of having at least two independent reviewers. It remains unknown whether a multi-agent system—simulating human dispute discussion and consensus-seeking through independent LLM coders—can surpass single agents. Consequently, empirical research is needed to validate the feasibility of multi-agent collaborative architectures for title-abstract screening in educational SRs.

3. METHOD

3.1 Data Source

In this preliminary study, we searched Computers & Education—a leading journal in the field of education—for systematic reviews with titles containing the phrase "systematic review". Our initial goal was to replicate the literature search process by adhering to the search strategies described in these systematic reviews. However, we found that many reviews did not specify the exact search date, making replication challenging. Additionally, for some systematic reviews, even when we strictly followed the reported search strategies, our search results did not correspond with those presented

Table 1: Five systematic reviews whose literature search was replicated in our study.

Reference	Topic	# Total studies identified in WoS	# Included WoS studies
[7]	Immersive virtual reality serious games for evacuation training and research	249	9
[10]	Speech-recognition chatbots for language learning	235	49
[20]	Computational thinking through programming in K-12 education	643	59
[24]	Teaching and learning robotics content knowledge in K-12 education	373	6
[27]	Impact of esports participation on the development of 21st century skills in youth	607	49

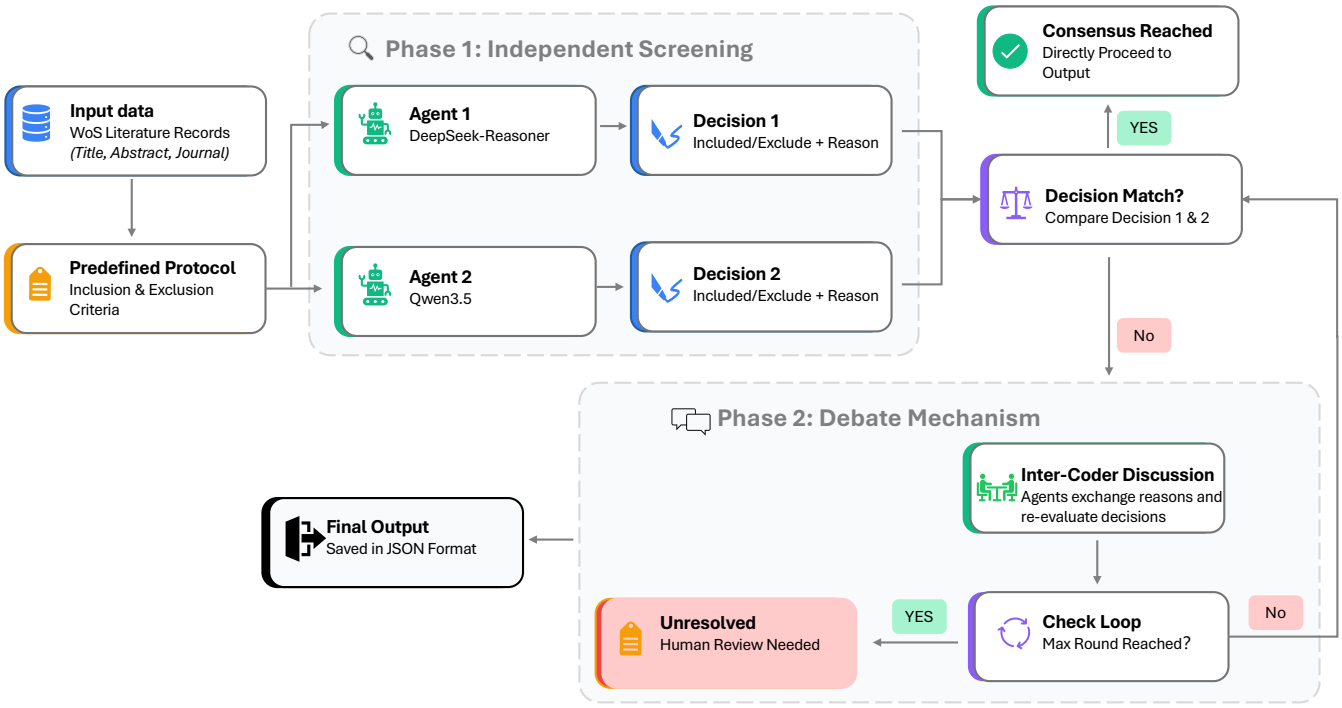


Figure 1: A dual-agent consensus-seeking workflow for systematic literature screening.

in the articles. Consequently, we revised our approach: we limited our analysis to systematic reviews that clearly reported both their search strategies and search dates. We also acknowledged that some variation between our search results and those reported in the articles was likely.

To avoid duplicate records from multiple databases, we restricted our search to the Web of Science (WoS) database. Ultimately, we selected five systematic reviews for this exploratory and preliminary study and replicated their literature search results using WoS. Table 1 summarizes the basic information for these reviews, including their topics, the total number of studies identified in WoS, and the number of WoS studies ultimately included in each review.

3.2 A dual-agent consensus-seeking workflow for systematic literature screening

In this study, we designed a dual-agent consensus-seeking workflow to simulate the rigorous human double-blind screening process for systematic literature reviews, as illustrated in Figure 1. To perform the title-abstract screening, we instantiated two distinct agents powered by heterogeneous large language models (i.e., DeepSeek-V3.2 and Qwen3.5-35b).

In Phase 1, the two agents independently coded each article and generated unilateral decisions regarding its inclusion or exclusion for subsequent full-text review. Both agents were governed by a unified system prompt that assigned them the persona of an expert academic researcher conducting a preliminary systematic review screening. Through zero-shot prompting, the models were provided with strict inclusion and exclusion criteria alongside a standardized output format (e.g., JSON). Importantly, to facilitate an objective evaluation of our workflow's efficacy, these screening criteria were directly extracted from the previously published systematic review. This design choice ensured an unbiased benchmark, allowing us to evaluate the agents' performance

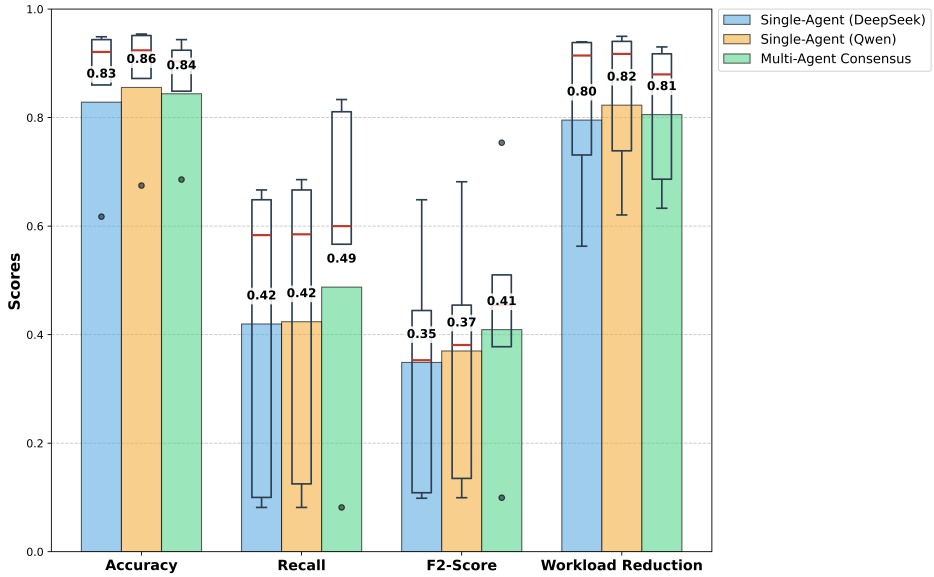


Figure 2: The aggregated global performance (represented by the bar heights) alongside the cross-sectional variance across the five systematic review (illustrated by the overlaid boxplots).

against an established human-coded ground truth.

In Phase 2, the workflow automatically cross-verified the decisions generated by the two agents. If a consensus was reached (i.e., both agents agreed to include or exclude), the final decision was recorded, and the system proceeded to the next article. However, in the event of a disagreement, the debate mechanism was triggered. During a debate round, each agent was provided with the counterpart's previous decision and rationale, prompting them to reflect, cross-examine the discrepancies, and re-evaluate their own stance. If this inter-coder discussion led to a consensus, the loop terminated successfully. If the conflict persisted, the agents engaged in further debate iterations until a predefined maximum number of rounds (i.e., $N = 3$ in our study) was reached. Articles that remained unresolved after the maximum debate rounds were flagged for human intervention by a senior researcher, perfectly mirroring the standard dispute-resolution protocol of traditional systematic reviews.

3.3 Evaluation

To rigorously evaluate the performance of the single-agent baselines (Phase 1) against the multi-agent consensus framework (Phase 2), we compared the classification decisions generated by DeepSeek and Qwen against the human-annotated ground truth (i.e., the ultimate inclusion or exclusion decisions established by the original systematic review).

Given the highly imbalanced nature of title-abstract screening in systematic reviews—where negative samples (excluded articles) vastly outnumber positive ones—we employed four specific evaluation metrics: Recall, F2-Score, Accuracy, and Workload Reduction. Recall calculates the proportion of genuinely relevant studies successfully identified for inclusion. This metric is paramount in our context, as minimizing the omission of relevant literature (i.e., false negatives) is the primary objective of automated screening. To comprehensively evaluate the model's discriminative power while heavily penalizing missed articles, we utilized the F2-Score, which assigns twice the weight to recall compared to precision. Accuracy measures the overall proportion of studies correctly classified by the agents. Finally, Workload Reduction quantifies the practical utility of the automated system by calculating the proportion of total retrieved articles that were correctly excluded (true negatives), reflecting the actual human screening effort saved.

4. PRELIMINARY RESULTS

Due to page limits, Figure 2 reports the aggregated global performance (bar heights) together with the cross-sectional variance across the five independent data chunks (overlaid boxplots).

As shown in Figure 2, the Multi-Agent Consensus system clearly outperformed the single-agent baselines in identifying positive samples. The dual-agent debate mechanism achieved the highest global Recall of 0.49, outperforming both DeepSeek (0.42) and Qwen (0.42). This 16.6% relative improvement suggests that the inter-coder discussion and re-evaluation phase successfully recovered false negatives that were initially missed under unilateral decision-making. Consistent with this pattern, the multi-agent system also obtained the highest F2-score (0.41), compared with DeepSeek (0.35) and Qwen (0.37).

Although Recall was the primary objective, the automated system must also substantially reduce human screening effort. For Workload Reduction, all three configurations performed strongly, effectively filtering out irrelevant records. The Multi-Agent system achieved a Workload Reduction of 0.81 (i.e., safely excluding 81% of all articles from human review), closely comparable to Qwen (0.82) and DeepSeek (0.80). Overall Accuracy ranged from 0.83 to 0.86 across models, indicating a consistently strong ability to identify

true negatives in the highly imbalanced dataset.

The overlaid boxplots in Figure 2 also highlight the intrinsic challenges of automated screening in subjective, imbalanced educational datasets. Accuracy and Workload Reduction exhibited compact interquartile ranges, indicating stable performance across the five data chunks. In contrast, Recall and F2-score displayed larger variability, suggesting that model sensitivity is context-dependent and fluctuates with the specific subset of literature. Notably, however, the Multi-Agent Consensus system raised the median Recall (red line within the box) to approximately 0.60, higher than both single-agent baselines. This upward shift in the median distribution provides additional evidence that the debate mechanism not only improves global performance but also enhances the robustness of screening across heterogeneous samples.

Despite these promising results, 92 studies in the corpus still failed to reach a consensus after three discussion rounds.

5. DISCUSSION

This study proposed a dual-agent consensus-seeking workflow to automate title-abstract screening in educational systematic reviews. Our preliminary findings demonstrate that simulating a human-like inter-coder debate significantly improves global Recall (0.49) and F2-Score compared to single-agent baselines (0.42), while successfully reducing the human screening workload by over 80%. This performance gain highlights the effectiveness of the multi-agent debate mechanism in rectifying unilateral LLM biases and rescuing critical false negatives.

Despite these improvements, the absolute Recall of approximately 50% indicates that fully autonomous LLM screening remains challenging in the education domain. We attribute this to the highly contextualized and subjective nature of educational constructs. Unlike explicit clinical metrics in medical reviews, educational inclusion criteria (e.g., "21st-century skills" or "pedagogical integration") often require nuanced, human-level interpretative reasoning that zero-shot prompting struggles to fully capture. Thus, future iterations should incorporate few-shot prompting with human-annotated examples to calibrate the agents' sensitivity.

6. REFERENCES

[1] A. Barany, N. Nasiar, C. Porter, A. F. Zambrano, A. L. Andres, D. Bright, M. Shah, X. Liu, S. Gao, J. Zhang, et al. Chatgpt for education research: Exploring the potential of large language models for qualitative codebook development. In *International conference on artificial intelligence in education*, pages 134–149. Springer, 2024.

[2] J. Bettany-Saltikov and R. McSherry. How to do a systematic literature review in nursing: A step-by-step guide, 3/e. 2024.

[3] R. Borah, A. W. Brown, P. L. Capers, and K. A. Kaiser. Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the prospero registry. *BMJ open*, 7(2):e012545, 2017.

[4] H. Chen, W. Ji, L. Xu, and S. Zhao. Multi-agent consensus seeking via large language models. *arXiv preprint arXiv:2310.20151*, 2023.

[5] S. De Paoli. Performing an inductive thematic analysis of semi-structured interviews with a large language model: An exploration and provocation on the limits of the approach. *Social Science Computer Review*, 42(4):997–1019, 2024.

[6] F. M. Delgado-Chaves, M. J. Jennings, A. Atalaia, J. Wolff, R. Horvath, Z. M. Mamdouh, J. Baumbach, and L. Baumbach. Transforming literature screening: The emerging role of large language models in systematic reviews. *Proceedings of the National Academy of Sciences*, 122(2):e2411962122, 2025.

[7] Z. Feng, V. A. González, R. Amor, R. Lovreglio, and G. Cabrera-Guerrero. Immersive virtual reality serious games for evacuation training and research: A systematic literature review. *Computers & Education*, 127:252–266, 2018.

[8] M. J. Grant and A. Booth. A typology of reviews: an analysis of 14 review types and associated methodologies. *Health information & libraries journal*, 26(2):91–108, 2009.

[9] T. Jansen, L. W. Liebenow, U. Mertens, F. T. Schmidt, J. F. Lohmann, J. Fleckenstein, and J. Meyer. Data extraction by generative artificial intelligence: Assessing determinants of accuracy using human-extracted data from systematic review databases. *Psychological Bulletin*, 151(10):1280, 2025.

[10] J. Jeon, S. Lee, and H. Choe. Beyond chatgpt: A conceptual framework and systematic review of speech-recognition chatbots for language learning. *Computers & Education*, 206:104898, 2023.

[11] X. Liu, A. F. Zambrano, R. S. Baker, A. Barany, J. Ocumpaugh, J. Zhang, M. Pankiewicz, N. Nasiar, and Z. Wei. Qualitative coding with gpt-4: Where it works better. *Journal of Learning Analytics*, 12(1):169–185, 2025.

[12] I. J. Marshall and B. C. Wallace. Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic reviews*, 8(1):163, 2019.

[13] M. Newman and D. Gough. Systematic reviews in educational research: Methodology, perspectives and application. *Systematic reviews in educational research: Methodology, perspectives and application*, pages 3–22, 2019.

[14] B. Nykvist, B. Macura, M. Xylia, and E. Olsson. Testing the utility of gpt for title and abstract screening in environmental systematic evidence synthesis. *Environmental Evidence*, 14(1):7, 2025.

[15] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, et al. The prisma 2020 statement: an updated guideline for reporting systematic reviews. *bmj*, 372, 2021.

[16] A. Pollock and E. Berge. How to do a systematic review. *International Journal of Stroke*, 13(2):138–156, 2018.

[17] T. Qiao, C. Walker, C. Cunningham, and Y. S. Koh. Thematic-lm: a llm-based multi-agent system for large-scale thematic analysis. In *Proceedings of the*

ACM on Web Conference 2025, pages 649–658, 2025.

[18] Z. Rasheed, M. Waseem, A. Ahmad, K.-K. Kemell, W. Xiaofeng, A. N. Duc, and P. Abrahamsson. Can large language models serve as data analysts? a multi-agent assisted approach for qualitative data analysis. *arXiv preprint arXiv:2402.01386*, 2024.

[19] M. Ruan, J. Fan, M. Liu, Z. Meng, X. Zhang, and C. Zhang. Artificial intelligence for the science of evidence synthesis: how good are ai-powered tools for automatic literature screening? *BMC Medical Research Methodology*, 25(1):199, 2025.

[20] C. Tikva and E. Tambouris. Mapping computational thinking through programming in k-12 education: A conceptual model based on a systematic literature review. *Computers & Education*, 162:104083, 2021.

[21] D. Wang and G. Chen. Evaluating the use of bert and llama to analyse classroom dialogue for teachers' learning of dialogic pedagogy. *British Journal of Educational Technology*, 56(6):2671–2704, 2025.

[22] D. Wang, D. Shan, Y. Zheng, K. Guo, G. Chen, and Y. Lu. Can chatgpt detect student talk moves in classroom discourse? a preliminary comparison with bert. In *Proceedings of the 16th international conference on educational data mining*, pages 515–519, 2023.

[23] D. Wang, Y. Zheng, J. Li, and G. Chen. Parameter-efficiently fine-tuning large language models for classroom dialogue analysis. *IEEE Transactions on Learning Technologies*, 18:542–555, 2025.

[24] L. Xia and B. Zhong. A systematic review on teaching and learning robotics content knowledge in k-12. *Computers & Education*, 127:267–282, 2018.

[25] A. F. Zambrano, X. Liu, A. Barany, R. S. Baker, J. Kim, and N. Nasiar. From ncoder to chatgpt: From automated coding to refining human coding. In *International conference on quantitative ethnography*, pages 470–485. Springer, 2023.

[26] J. Zhang, X. Xu, N. Zhang, R. Liu, B. Hooi, and S. Deng. Exploring collaboration mechanisms for llm agents: A social psychology view. *arXiv preprint arXiv:2310.02124*, 2023.

[27] Y. Zhong, K. Guo, J. Su, and S. K. W. Chu. The impact of esports participation on the development of 21st century skills in youth: A systematic review. *Computers & Education*, 191:104640, 2022.

Beyond Idealized Cooperation: Epistemic Noise in Collaborative Problem Solving

Videep Venkatesha
Colorado State University

Ethan Seefried
Colorado State University

Changsoo Jung
Colorado State University

Nathaniel Blanchard
Colorado State University

{Videep.Venkatesha, Ethan.Seefried, Changsoo.Jung, Nathaniel.Blanchard} @colostate.edu

ABSTRACT

AI agents that support collaborative learning must model the phenomena that drive productive group work, including communicative acts, belief formation, trust dynamics, and reasoning quality, as they unfold in real time. Existing collaborative datasets capture idealized cooperation under shared epistemic goals, but the dynamics pervasive in real classrooms, such as misconceptions, unreliable contributions, overconfident errors, and their downstream effects on group reasoning, remain largely absent and unlabeled. We propose social deduction games, specifically Secret Hitler, as a proxy testbed where collaborative problem solving and controlled injection of unreliable information co-occur with researcher-accessible ground truth. The game's role structure produces contributions that are functionally equivalent to classroom misconceptions and misdirection, but with known provenance, enabling traceable study of how unreliable information propagates through groups. We present a multi-layer annotation framework that captures communicative acts using a validated persuasion strategy taxonomy, player-reported belief states yielding first- and second-order Theory of Mind data, and post-hoc epistemic vigilance assessments of whether belief updates were warranted given available evidence. We pair this framework with multi-person egocentric sensing using Meta Aria research glasses alongside exocentric recording, providing synchronized capture of gaze, attention, and social signals from every participant's perspective. Pilot data from two 6-player, 9-round sessions demonstrates that failed epistemic vigilance, belief fragmentation, and traceable misinformation effects emerge naturally and are captured by the framework.

Keywords

epistemic vigilance, collaborative problem solving, multimodal learning analytics, Theory of Mind, social deduction games, belief dynamics

Videep Venkatesha, Ethan Seefried, Changsoo Jung, and Nathaniel Blanchard. Beyond Idealized Cooperation: Epistemic Noise in Collaborative Problem Solving. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 600–604. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039891

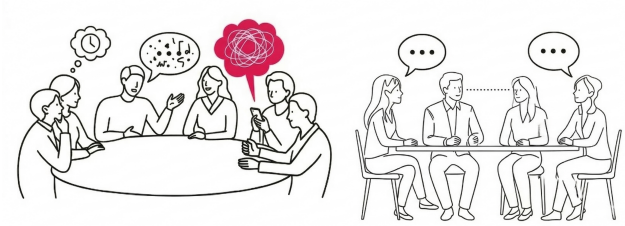


Figure 1: Real collaboration (left) involves heterogeneous contributions, competing attention, and participants who may be disengaged or misinformed—dynamics pervasive in collaboration but largely absent from controlled datasets. Idealized settings (right) capture only cooperative interaction. Our testbed produces the dynamics on the left with the ground truth needed to study them.

1. INTRODUCTION

AI agents that facilitate collaborative learning must model the phenomena that drive productive group work as they unfold: the *communicative acts* members perform, the *beliefs* they hold and revise, the *reasoning quality* with which they evaluate one another's contributions, and the *trust dynamics* that determine whose voice carries weight [2, 15, 18]. Developing models that capture these phenomena requires extensive iteration on sensing, annotation, and inference that would consume scarce instructional time if conducted in classrooms. Instead, we need proxy testbeds: controlled environments where groups collaborate naturally, producing multimodal signals (speech, gaze, gesture, posture), but where data collection serves system development rather than student learning.

Existing testbeds such as the DELI corpus [9] and Physics Playground [18] capture multi-party deliberation and collaborative problem solving (CPS), but under shared epistemic goals where all participants work toward the right answer. Real collaboration is messier: students unknowingly spread misconceptions, assert incorrect solutions with unwarranted confidence, freeload, or dominate discussions unproductively [4]. These dynamics of *unreliable contributions*—not intentional sabotage, but the natural heterogeneity of contributor quality—are pervasive in classrooms [10, 14] yet rare in existing datasets.

We propose social deduction games, specifically *Secret Hitler*,

as a proxy testbed that fills this gap. Players are secretly assigned to cooperative or disruptive teams and engage in structured rounds of discussion, voting, and policy enactment. For the majority of players, the game *is* collaborative problem solving: pooling incomplete evidence, evaluating reliability of claims, and coordinating decisions under genuine uncertainty about who is trustworthy [15, 18]. What makes the setting valuable is that a minority of players are structurally incentivized to introduce unreliable information—spreading misinformation, asserting false claims with confidence, and undermining group reasoning. These contributions are *functionally equivalent* to the misconceptions, overconfident errors, and unproductive dominance documented in classroom groups [3]: the collaborative challenge they pose is the same, even though the underlying intent differs. Crucially, every piece of unreliable information has known provenance (role assignments provide ground truth), and its effects on group beliefs are traceable through our annotation framework.

We introduce a multi-layer annotation framework targeting progressively deeper social-cognitive constructs. Layer 1 classifies communicative acts using a validated persuasion strategy taxonomy [11]. Layer 2 collects player-reported belief states after each round, yielding first- and second-order Theory of Mind data grounded in Baker et al.'s [2] framework. Layer 3 infers epistemic vigilance post-hoc [16], the capacity to evaluate communicated information for source reliability and content plausibility rather than accepting it uncritically, assessing whether belief updates were warranted given available evidence. The cascading structure of communicative act → belief update → vigilance assessment—captures the trajectory from speech to social-cognitive outcome that a classroom agent must track. We pair this framework with multimodal sensing using Meta Aria egocentric research glasses [7], instrumenting every player to capture synchronized video, eye tracking, and inertial data from first-person perspectives.

Our contributions are:

- A multi-layer annotation framework integrating communicative act classification, player-reported belief states (first- and second-order Theory of Mind), and post-hoc epistemic vigilance inference for coding group social-cognitive dynamics.
- A positioning of Secret Hitler as a CPS environment with controlled injection of unreliable information with known provenance, addressing a gap left by datasets capturing only idealized cooperation.
- Pilot data from two 6-player sessions demonstrating that failed vigilance, belief fragmentation, and traceable misinformation effects emerge naturally and are captured by the framework.

2. SECRET HITLER AS A CPS TESTBED

In Secret Hitler, 5 to 10 players are secretly assigned roles as Liberals or Fascists, with one Fascist designated as Hitler. Each round, a President nominates a Chancellor; the group votes to approve or reject; and if approved, the government enacts a policy from a constrained deck. Liberals win by enacting five Liberal policies or assassinating Hitler; Fascists win by enacting six Fascist policies or electing Hitler as Chancellor.

For the majority of players (Liberals), the game *is* collaborative problem solving: they must externalize private information, build shared representations of the game state, and negotiate coordinated action, all without knowing who their allies are. This maps onto the joint problem space [6, 15, 19] operationalized as "constructing shared knowledge" in the General Competency Model [18]. Critically, the game enforces the information asymmetry that the hidden profile literature identifies as the hardest part of real collaboration [12, 17], a challenge most CPS testbeds sidestep. Discussion phases require players to surface disagreements, provide evidence, and converge on action [3, 8], instantiating the argumentative structure that predicts decision quality [9, 13, 20].

What makes the setting distinctive is not that it introduces dynamics foreign to CPS, but that it makes *visible and labeled* the kinds of unreliable contributions that pervade real collaboration but are rarely captured in research settings. Barron [4] documented how classroom collaborations fail when members push incorrect solutions with unwarranted confidence or dismiss valid contributions. Students regularly spread misconceptions, sometimes unknowingly, sometimes through carelessness and the group's ability to detect and correct these contributions determines whether collaboration succeeds [4]. These dynamics are pervasive [10, 14] yet absent from controlled CPS datasets designed to elicit cooperation under shared goals [18]. Secret Hitler produces functionally equivalent dynamics through its role structure: Fascist players introduce misinformation, assert false claims with confidence, and selectively withhold evidence. The collaborative challenge this poses of forcing the group to evaluating contributions of uncertain reliability, pooling incomplete evidence, and reaching decisions despite misleading information, is the same challenge groups face whenever collaboration involves distributed knowledge and heterogeneous contributor quality. We do not claim that game players are cognitively equivalent to struggling students; we claim that the epistemic problem is shared. The methodological advantage is that the ground truth is known: role assignments let us track who introduced unreliable information, who accepted it (Layer 2), and whether that acceptance was warranted (Layer 3). In a classroom, the same dynamics occur but leave no traceable record.

The testbed also provides structured temporal episodes (nomination → discussion → vote → reveal), rich multimodal interaction in a facetoface small group, and repeatability across sessions. Our annotation framework (Section 3) captures what existing deliberation datasets cannot: the downstream effects of unreliable contributions on *individual* beliefs, traced from specific communicative acts to specific belief changes in specific individuals.

3. MULTI-LAYER ANNOTATION FRAMEWORK

3.1 Multi-Layer Annotation Framework

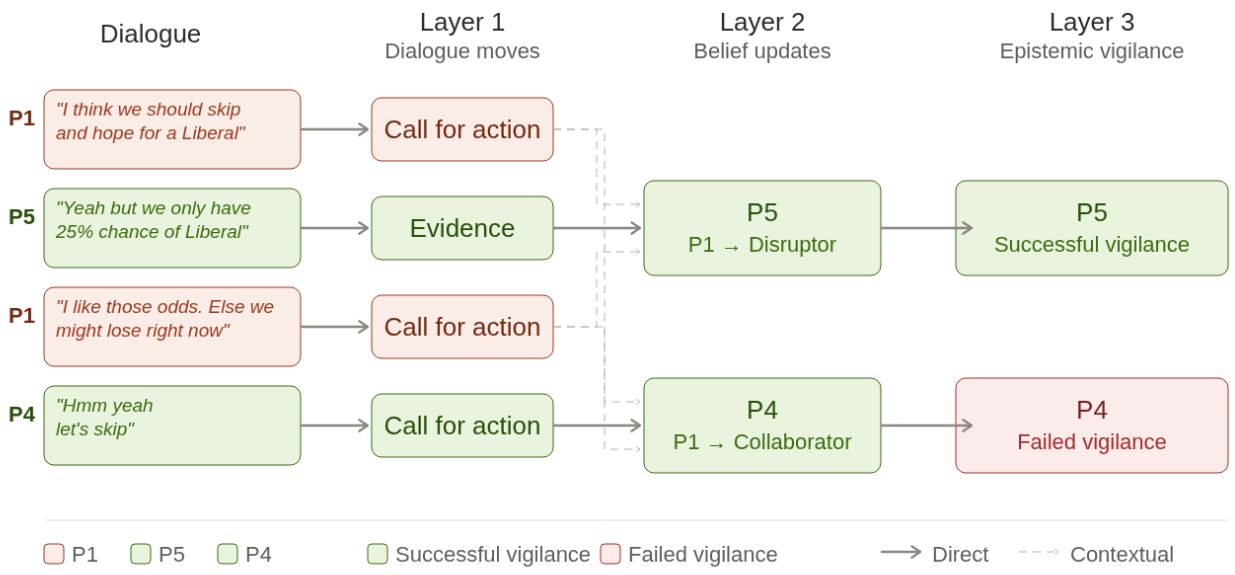


Figure 2: Three layer annotation of a discussion phase. Utterances are classified by communicative function (Layer 1), linked to belief updates (Layer 2), and assessed against ground truth as successful or failed epistemic vigilance (Layer 3). Color indicates true role (red = adversarial, green = collaborative; hidden during play).

The framework is hierarchical: each layer targets a progressively deeper dimension of group social-cognitive dynamics. Layers 2 and 3 operationalize two cognitive science frameworks: Baker et al.'s [2] Bayesian Theory of Mind and Sperber et al.'s [16] epistemic vigilance which have been influential but not previously applied to multimodal behavioral annotation in group settings.

3.2 Layer 1: Communicative Acts

We adopt the six-category taxonomy from Lai et al. [11], annotated at the utterance level: *Identity Declaration*, *Accusation*, *Interrogation*, *Call for Action*, *Defense*, and *Evidence*. This layer captures what communicative act a speaker performs, the information an agent needs to classify conversational moves in real time. Lai et al. report Krippendorff's $\alpha > 0.6$ across all categories, establishing feasibility for reliable annotation.

3.3 Layer 2: Player-Reported Belief States

At the end of each round, players report two types of beliefs:

First-order ToM. Each player records their belief about every other player's role (Liberal, Fascist, or uncertain). Since researchers know the true assignments, these reports yield calibration data: how accurately does each player's belief state track reality as evidence accumulates?

Second-order ToM. Each player records what they think others believe about *their own* role. Comparing Player A's second-order report against Player B's actual first-order report yields a direct measure of Theory of Mind accuracy [2].

These belief reports serve as supervision signals for training models that infer beliefs from observable behavior—gaze patterns, speech acts, voting behavior, and postural cues—since a classroom agent cannot ask students to report beliefs directly.

3.4 Layer 3: Epistemic Vigilance

This layer is inferred post-hoc using Layers 1 and 2 together with ground-truth role assignments [16]. For each communicative act targeting a specific player, we examine the receiver's belief trajectory:

- **Successful vigilance**: belief moved toward truth (e.g., correctly increasing suspicion of a Fascist after inconsistent behavior)
- **Failed vigilance**: belief moved away from truth (e.g., persuaded by a misleading claim, or failing to update when evidence warranted it)

Figure 2 illustrates this cascading structure with an excerpt from a pilot session. P1 (true role: Disruptive) advocates for an irrational move of skipping a turn and hoping for a favorable outcome. P5 correctly objects by presenting evidence that the odds are poor, updating their belief that P1 is a Disruptor showing successful vigilance. P4, however, accepts P1's reasoning and agrees to skip, maintaining the belief that P1 is a trustworthy Collaborator despite the weak justification: failed vigilance. The same exchange thus produces both outcomes in different receivers, and the frame-

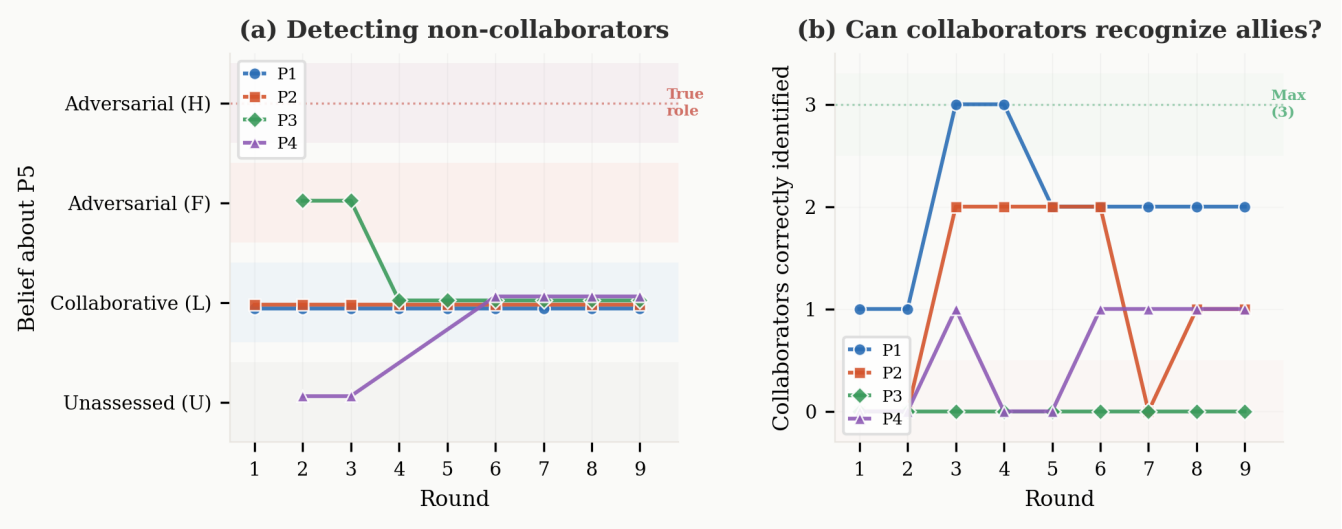


Figure 3: **(a)** Four collaborative players' beliefs about P5 (true role: Disruptive) across rounds. No collaborator ever correctly classified P5. P3 briefly suspected P5 (Rounds 2–3) but reversed by Round 4—a belief update *away* from truth, illustrating failed vigilance. **(b)** Number of fellow collaborators each player correctly identified per round (max = 3).

work captures exactly where and why the group's reasoning diverged.

4. SENSING AND DATA CAPTURE

All players wear Meta Aria Gen 1 research glasses [7], capturing egocentric RGB video, eye-tracking data, and IMU signals. Group audio is recorded via the glasses' microphones, and an overhead camera provides an exocentric reference view. Game state is logged by a facilitator.

Most multimodal learning analytics systems rely on exocentric sensing—ceiling-mounted cameras and external microphones [1]. While scalable, these setups cannot reliably recover participant-relative signals: whether someone is looking *at you*, directing a statement toward you, or attending to the current speaker. Our testbed instruments *every player* with egocentric glasses, producing synchronized capture from all participants' viewpoints. This enables gaze-derived features aligned with our annotation layers: identifying each player's attention target at utterance onset (linking Layer 1 acts to perceptual targets), characterizing gaze sequences during deceptive vs. truthful communication [5], and computing group-level attention divergence across all players at each timestep. Both egocentric and exocentric streams are captured simultaneously, enabling empirical comparison of whether first-person signals carry more information about the social-cognitive constructs in Layers 1–3.

5. PILOT DATA

To demonstrate that the target phenomena emerge naturally, we report observations from a pilot session (6 players, 9 rounds, 2 Disruptive/Adversarial and 4 Collaborative). All players wore Aria glasses and completed belief reports after each round.

Detecting Failed Vigilance. Figure 3(a) tracks collaborative players' evolving beliefs about P5, whose true role was Disruptive. Despite 9 rounds of discussion and evidence, no collaborator correctly identified P5. P3 initially suspected P5 (Rounds 2 and 3) but reversed toward Collaborative by Round 4, constituting *failed vigilance* in our Layer 3 framework: P3 possessed a partially correct signal but abandoned it, likely in response to P5's communicative acts (Layer 1) that undermined the suspicion.

Active but Counterproductive Participation. Figure 3(b) reveals a complementary pattern. P3 correctly identified zero allies across all 9 rounds, placing every collaborator in the Disruptive category—the profile of a group member who is actively reasoning but systematically misdirected. This is analogous to a student who is engaged but whose contributions fragment rather than strengthen shared understanding. Distinguishing active-but-counterproductive participation from constructive collaboration is precisely the capability our Layer 3 assessment operationalizes. P1, by contrast, peaked at identifying all three allies (Rounds 3–4) before declining, suggesting group-level belief fragmentation as unreliable information accumulated.

6. DISCUSSION AND CONCLUSION

The pilot data, though limited to a single session, demonstrates two properties. First, the multi-layer framework produces *linked annotations*: specific communicative acts traced to specific belief changes in specific individuals, assessed for epistemic warrant which is a cascading structure not available in existing CPS [18] or deliberation [9] datasets. Second, the testbed produces unreliable contributions with known ground truth, enabling study of how misinformation propagates through groups and what behavioral signals distinguish listeners who detect it from those who do not.

We frame the testbed as addressing a prior question: is it feasible to infer social-cognitive constructs from multimodal behavioral signals at all? The signals captured here are nat-

uralistic with overlapping speech, occlusions, and rapid gaze shifts, but with clean supervision unavailable in educational settings. We do not claim models trained on game data will deploy directly in classrooms; what transfers is the inference machinery and the insight that proxy testbeds with clean supervision can accelerate model development impractical to iterate on in instructional settings.

7. LIMITATIONS

The pilot comes from a single 6-player session; we do not claim generalizability. Player-reported beliefs are subject to social desirability and retrospective rationalization. Layer 3 assessments depend on binary ground truth cleaner than any classroom would provide. The game's competitive structure may elicit more deliberate reasoning than typical collaboration, and the sensing setup requires research-grade hardware whose value is in model development, not deployment.

8. ACKNOWLEDGMENTS

This material is based in part upon work supported by the U.S. National Science Foundation (NSF) under award DRL 2454151 (Institute for Student-AI Teaming). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

9. REFERENCES

[1] K. Ahuja, D. Kim, F. Xhakaj, V. Varga, A. Xie, S. Zhang, J. E. Townsend, C. Harrison, A. Ogan, and Y. Agarwal. Edusense: Practical classroom sensing at scale. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–26, 2019.

[2] C. L. Baker, J. Jara-Ettinger, R. Saxe, and J. B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4):0064, 2017.

[3] B. Barron. Achieving coordination in collaborative problem-solving groups. *The journal of the learning sciences*, 9(4):403–436, 2000.

[4] B. Barron. When smart groups fail. *The journal of the learning sciences*, 12(3):307–359, 2003.

[5] R. Dewhurst, M. Nyström, H. Jarodzka, T. Foulsham, R. Johansson, and K. Holmqvist. It depends on how you look at it: Scanpath comparison in multiple dimensions with MultiMatch, a vector-based approach. *Behavior Research Methods*, 44(4):1079–1100, 2012.

[6] P. Dillenbourg and D. Traum. Sharing solutions: Persistence and grounding in multimodal collaborative problem solving. *The Journal of the Learning Sciences*, 15(1):121–151, 2006.

[7] J. Engel, K. Somasundaram, M. Goesele, A. Sun, A. Gamino, A. Turner, A. Talattof, A. Yuan, B. Souti, B. Meredith, et al. Project aria: A new tool for egocentric multi-modal ai research. *arXiv preprint arXiv:2308.13561*, 2023.

[8] Y. Hayashi. The power of a "maverick" in collaborative problem solving: An experimental investigation of individual perspective-taking within a group. *Cognitive science*, 42:69–104, 2018.

[9] G. Karadzhov, T. Stafford, and A. Vlachos. Delidata: A dataset for deliberation in multi-party problem solving. *Proceedings of the ACM on Human-Computer Interaction*, 7(CSCW2):1–25, 2023.

[10] S. J. Karau and K. D. Williams. Social loafing: A meta-analytic review and theoretical integration. *Journal of Personality and Social Psychology*, 65(4):681–706, 1993.

[11] B. Lai, H. Zhang, M. Liu, A. Pariani, F. Ryan, W. Jia, S. A. Hayati, J. Rehg, and D. Yang. Werewolf among us: Multimodal resources for modeling persuasion behaviors in social deduction games. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6570–6588, 2023.

[12] L. Lu, Y. C. Yuan, and P. L. McLeod. Twenty-five years of hidden profiles in group decision making: A meta-analysis. *Personality and Social Psychology Review*, 16(1):54–75, 2012.

[13] D. Moshman and M. Geil. Collaborative reasoning: Evidence for collective rationality. *Thinking & Reasoning*, 4(3):231–248, 1998.

[14] S. L. Piezon and W. D. Ferree. Perceptions of social loafing in online learning groups: A study of public university and us naval war college students. *International Review of Research in Open and Distributed Learning*, 9(2):1–17, 2008.

[15] J. Roschelle and S. D. Teasley. The construction of shared knowledge in collaborative problem solving. In *Computer supported collaborative learning*, pages 69–97. Springer, 1995.

[16] D. Sperber, F. Clément, C. Heintz, O. Mascaro, H. Mercier, G. Origgi, and D. Wilson. Epistemic vigilance. *Mind & language*, 25(4):359–393, 2010.

[17] G. Stasser and W. Titus. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of personality and social psychology*, 48(6):1467, 1985.

[18] C. Sun, V. J. Shute, A. Stewart, J. Yonehiro, N. Duran, and S. D'Mello. Towards a generalized competency model of collaborative problem solving. *Computers & Education*, 143:103672, 2020.

[19] S. D. Teasley, F. Fischer, A. Weinberger, K. Stegmann, P. Dillenbourg, M. Kapur, and M. Chi. Cognitive convergence in collaborative learning. In *Proceedings of the 8th International Conference for the Learning Sciences*, volume 3, pages 360–367. 2008.

[20] S. E. Toulmin. *The Uses of Argument*. Cambridge University Press, 2 edition, 2003.

Bias in MOOC Completion Prediction: A Fairness Analysis Across African Countries and Socio-Economic Contexts

Jade Mai Cock
LIP6, CNRS, Sorbonne University
Paris, France
Jade-Mai.Cock@lip6.fr

Sébastien Lallé
LIP6, CNRS, Sorbonne University
Paris, France
sebastien.lalle@lip6.fr

Mélina Verger
INSA Lyon, LIRIS
Lyon, France
melina.verger@insa-lyon.fr

François Bouchet
LIP6, CNRS, Sorbonne University
Paris, France
francois.bouchet@lip6.fr

ABSTRACT

Low completion rates are a well-documented challenge in MOOCs, and Machine Learning (ML)-based predictors have been proposed as a way to detect at-risk learners and support them through personalised interventions. Yet, evidence shows that ML models trained on such data can be biased against certain groups of learners, often the historically discriminated ones. In particular, usage patterns can substantially vary across world regions due to contextual factors, which can affect predictions. In this work, we analyse ML models trained on a MOOC dataset including African and OECD learners to examine whether, and for whom, biases arise. Our findings show that learners from Middle and Western Africa and from countries with low HDI and low literacy rates are affected by significant biases in ML predictions. Our findings highlight the need to systematically evaluate ML unfairness in African educational contexts.

Keywords

Fairness, Biases, Machine learning, MOOCs, Africa

1. INTRODUCTION

The widespread adoption of ML in education, e.g., for predicting course success or dropout risk, has raised concerns about their fairness towards historically discriminated groups [7]. ML unfairness arises when model behaviours and accuracy substantially differ across learner groups [8, 21]. When such models guide important decisions like program admission, resource allocation, and interventions for at-risk learners, they risk perpetuating existing biases and hindering the academic outcomes for specific groups of learners [4].

Recent educational data mining research has explored fairness in ML models on educational datasets [3, 10, 11, 26, 29]. Yet, surveys and opinion papers highlight a key limitation: an overwhelming amount of studies focuses on North American and European learners, with few considering African contexts [8, 17, 21]. Yet, African educational contexts have distinctive features, including the high prevalence of mobile or offline learning [16], stark rural-urban contrasts [14], unequal access to connectivity, electricity and computers [16], widespread reliance on open digital spaces and digital universities to mitigate infrastructure limitations [24], decreasing gender parity with higher levels of educational level increases [1], and overcrowded classrooms [12]. These may affect the target population and platform interactions, potentially reflecting contextual constraints rather than learners' engagement, abilities or interests.

Understanding this gap is critical for fairness-aware analysis, particularly in MOOCs which serve international and highly diverse learners, and are widely adopted in Africa [12, 16, 28]. While ML studies to predict drop-out on MOOC data is extensive [2, 18], fairness analyses are scarce [10, 13, 22]. As a result, the promise of democratizing access to knowledge across continents through MOOCs cannot be fully realized if known contextual differences are not accounted for from the course design stage onward. In this respect, indiscriminately combining data from African and Western learners may exacerbate existing biases rather than mitigate them.

Within prior work, the only openly available MOOC dataset used in a fairness study is the GDP-MOOC dataset published by Lallé et al. [13]. This dataset involves students from 23 African countries, mostly located in Western, Northern and Middle Africa. This work [13] showed that ML models can accurately predict MOOC completion, but were biased against African learners. However, they treated all African learners as a single homogeneous group. Building on this, we propose a fine-grained fairness analysis of biases across African countries and socio-economic characteristics in this dataset. Beyond MOOCs, ML biases have been analysed for automatic scoring of Nigerian essays (which accounted for only $< 0.03\%$ of the data) [6], and in the PISA 2015 data, but without specific results for Africa (Al-

Jade Mai Cock, Sébastien Lallé, Mélina Verger, and François Bouchet. Bias in MOOC Completion Prediction: A Fairness Analysis Across African Countries and Socio-Economic Contexts. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 605–610. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040101

geria and Tunisia) [15]. This led to this research question: *Which specific countries or socio-economic characteristics help explain the biases toward Africa of ML models predicting course success in the GDP-MOOC dataset?*

2. METHOD

2.1 Data and Machine Learning Setup

We used the GDP-MOOC open dataset in this paper (`https://github.com/Mocahteam/GdPMOOC_Fairness`) [13]. It originates from a French MOOC on Project Management lasting five weeks (2022 offering) delivered on the OpenEdX platform. The dataset includes weekly aggregated statistical summaries (e.g. counts) derived from OpenEdX log data, the students' country, and their final grade. Course completion (pass/fail) is defined as achieving a final grade of at least 50%. The GDP-MOOC dataset corresponds to the Spring 2022 offering. It includes complete data for 912 learners (460 are in Africa, 452 in OECD countries in Western Europe and North America), among whom 380 (39%) successfully passed the course. Learners provided informed consent for their data to be used for research purposes, and all data are fully anonymized per the EU's General Data Protection Regulation (GDPR).

We replicate the ML pipeline proposed in [13] and shared in their GitHub to train three supervised ML models and predict in a binary way whether learners would *succeed* or *fail* to obtain the certificate at the end of the MOOC. The features used to train the models are the concatenation of weekly action counts f_w^l for each learner $l \in \mathcal{L}$. We train Logistic Regression (LR), Naive Bayes (NB), and Stochastic Gradient Boosting (SGB) on these features, selected respectively for classification performance, fairness, and accuracy–fairness trade-off [13]. We apply learner-level 5-fold stratified cross-validation with feature selection. The full dataset description and ML pipeline are reported in [13], and here we focus on the fairness analysis.

2.2 Bias Analyses

As stated above, ML models trained on the GDP-MOOC dataset exhibited significant unfairness toward African learners [13]. To better characterize these biases, we conduct a subgroup analysis based on the country of African learners (N=460). Their distribution (shown in Table 1) is highly uneven, with the top six countries accounting for over two-thirds of learners and all others having fewer than 20 each. Consequently, we opt to address our research question in two complementary ways: first by focusing on the six countries with over 20 learners (Ivory Coast, DR Congo, Benin, Cameroon, Burkina Faso and Senegal); and second by studying larger groups of countries based on UN-based socio-economic indicators. This second approach highlights higher-level biases that span similar countries. In addition, it allows us to include all of African learners into meaningful and relevant larger analytical groups. The country characteristics (listed in Table 2) are based on five country-level UN-based indicators.

UN Regions: The five sub-regions of Africa defined by the UN Geoscheme, a statistical way to group countries by main geographical areas. This is to understand at the high level whether specific parts of the continent receive more unfair predictions. The data are from the UN Statistical Division[1].

HDI: The four Human Development Index groups as defined by the UN Development Program, a widely accepted way to quantify human development in terms of life expectancy, quality of education, and standard of living. This allows assessing whether ML models perform equitably across contexts with unequal structural opportunities. The HDI groups are based on the Human Development Report for 2022[2].

Literacy: The levels of literacy among adults, as compiled by the UNESCO, which is used as a proxy for the access to fundamental education. The 2022 data are extracted from the IUS Bulk Data[3].

Education: The quality of education (from elementary to university), which is compiled by the UNESCO across a large battery of indicators as part of its Sustainable Development Goals (SDG-4). This is to evaluate whether lower quality of education in a country hinders the overall usage of the MOOCs, and in turn, the fairness of the models. The data are extracted from the IUS Bulk Data[3].

GDI: We consider the Gender Development Index as measured by the UN Development Program, since lower levels of gender development may influence how men and women engage with online learning resources such as MOOCs, potentially affecting ML fairness. Using the raw GDI values, we construct two categories, "low" and "high" GDI, by applying the African median as the cutoff threshold. The data are extracted from the Human Development Report for 2022[2].

We also considered the countries' GDP and access to the Internet, but discarded them as both metrics were highly correlated to the HDI (Pearson $r > .7$).

Fairness metrics. We measure the potential biases across the different groups described in Table 2 through two complementary metrics, namely the group-level AUC, and the MADD. For the first one, we compute the AUC of the models in the test sets for each of the groups, by selecting only learners from a specific group in the test sets (e.g., "West Africa") and computing the AUC of the models' outputs for those learners. This is a simple and commonly used approach to gauge how a model's accuracy varies across groups [10, 21, 23]. For a model to be fair, its AUC should be similar across the different groups of learners. As for the MADD, it leverages the histograms of probabilities output by a model for two groups to compute the areas where the histograms do not overlap [27]. This approach allows to measure how a given characteristics influence the models' outputs, regardless of its performances, as well as toward whom the output probabilities are inflated. To be fair, a models' outputs should not be related to any countries or characteristics it does not know about. Because the MADD involves comparing two groups in a pairwise manner, we use it by comparing each target group against all of the other learners (e.g., "Western Africa" learners vs. everybody else). AUC ranges from 0 (worst performance) to 1 (best performance) and MADD ranges from 0 (fairest) to 2 (most unfair). Both AUC and MADD are independent from

[1] `https://unstats.un.org/unsd/methodology/m49/`
[2] `https://hdr.undp.org/system/files/documents/global-report-document/hdr2023-24reporten.pdf`
[3] `https://databrowser.uis.unesco.org/resources/bulk`

Table 1: Number of learners (n) and passing rates (pr) per African country in the GDP-MOOC dataset.

Country	n (pr)	Country	n (pr)	Country	n (pr)	Country	n (pr)
Ivory Coast	82 (22%)	Togo	16 (31%)	Niger	10 (20%)	Burundi	3 (0%)
DR Congo	74 (36%)	Guinea	14 (14%)	Tchad	10 (3%)	Mauritania	2 (50%)
Benin	60 (40%)	Congo Republic	13 (23%)	Tunisia	9 (22%)	Central African Rep.	2 (0%)
Cameroon	42 (36%)	Mali	13 (8%)	Algeria	8 (38%)	Rwanda	1 (0%)
Burkina Faso	40 (28%)	Gabon	11 (27%)	Madagascar	4 (0%)	Mozambique	1 (0%)
Senegal	33 (27%)	Morocco	8 (38%)	Djibouti	4 (0%)		

Table 2: Country characteristics, sample size per group within Africa and passing rates. We report all groups for completeness, however, only groups with ≥ 20 learners are included in the analysis.

Characteristics	Groups (#learners [passing rate])
Sub-regions of Africa	Northern (25 [32%]), Western (270 [27%]), Eastern (13 [0%]) , Middle (152 [34%]), Southern (0)
HDI	Low (369 [28%]), Medium (63 [33%]), High (28 [29%]), Very high (0)
Literacy	Low (231 [26%]), Medium (155 [33%]), High (41 [27%]), Very high (33 [27%])
Quality of education	Low (79 [22%]), Lower middle (292 [30%]), Upper middle (65 [31%]), High (24 [33%])
Gender Gap Index	Low (281 [33%]), High (179 [22%])

the decision threshold of the ML models, i.e., the probability cutoff above which a binary classifier labels an instance as positive. Thus, they can provide a more robust and stable assessment than metrics tied to a single threshold. In addition, AUC and MADD are complementary, as AUC deals with the likelihood of correct predictions across groups and MADD with the independence of the models' decisions toward group membership. Both metrics have been used in EDM fairness studies, e.g., [19, 23, 27, 29].

3. RESULTS

Table 3 reports on the evaluation of the ML models' fairness over the countries and characteristics. For the MADD, we indicate with a +/- sign whether the model's behaviour is in favour/biased (i.e., tends to give higher/lower probabilities of success in the MOOC) toward the target group. To formally compare these results, we apply a separate Friedman test for each unique combination of metric (group-level AUC and MADD), ML model (LR, SGB, and NB) and factor (countries or characteristics). Each test assesses differences in the metric across the levels of a single factor within one ML model, with repeated measures defined by the AUC/MADD values computed at each week and fold of cross-validation. To account for multiple comparisons, we apply the Benjamini–Hochberg method (BH) to adjust all p-values simultaneously across the full set of Friedman tests, making the correction stricter. We then examine significant main effects using pairwise Wilcoxon signed-rank tests with adjustment, and indicate in bold in Table 3 the values for which the tests show significant unfairness.

Sub-regions. Table 3 shows contrasting results. LR favors North Africa and is more biased against Middle Africa, while NB favors Western Africa and is more biased against Northern Africa. The Friedman tests show a significant main effect of sub-regions on AUC for RF ($p = 0.008, \eta^2 = 0.30$), and on MADD for LR, SGB and NB ($p = 0.0001, \eta^2 = 0.84$, $p = 0.0001, \eta^2 = 0.75$, $p = 0.001, \eta^2 = 0.67$, respectively). The post-hoc pairwise tests indicate that LR is significantly more biased toward Middle Africa on both metrics ($p < 0.03$). For SGB and NB, there is no significant difference in AUC, but the main effects and pairwise tests for MADD also show significantly more biases toward Middle Africa. These results provide evidence that the Middle Africa sub-region countries would experience some levels of unfairness with all models, especially LR, even though its passing rate is the highest (34%) among the three regions.

HDI. Table 3 shows that the AUC for High HDI countries is substantially higher than for the other groups with LR and SGB, but not NB. The Friedman tests only shows a significant main effect of HDI on AUC for LR ($p = 0.049, \eta^2 = 0.19$) and NB ($p = 0.038, \eta^2 = 0.20$), confirming the trends that LR is significantly less accurate for Low HDI and NB for high HDI. For all models, there is also a significant main effect in MADD ($p < 0.0001$, $\eta^2 > 0.66$ in all cases), consistently showing more biases for Low and Medium HDI groups, according to the post-hoc tests. These results show that LR exhibits clear biases for low HDI countries.

Literacy. The Friedman tests show no significant main effect of literacy groups on AUC, but significant main effects for all models on MADD ($p < 0.0001, \eta^2 > 0.51$). For all of these effects, the pairwise comparisons always show significant biases toward Low and Medium levels of Literacy, as compared to High (and surprisingly Very high for LR). These results show that literacy can capture ML unfairness in this dataset, but only in terms of MADD.

Quality of education. The Friedman tests show no main effects on AUC, but on MADD for LR and SGB ($p < 0.0001$, $\eta^2 > 0.52$). This suggests that the quality of education impacts to some extent the models' behaviors, but not their performance. The post-hoc tests for MADD indicate more biases toward the Low education groups ($p < 0.05$) across the models. More surprisingly, the same effect is found for the High education group (as compared to Lower middle and Upper middle), possibly due to its small size.

GDI. The Friedman tests reveal no main effect of GDI on

Table 3: Results per group and ML model. Bold means significantly more biased toward that group.

Factor	Values	LR		SGB		NB	
		AUC	MADD	AUC	MADD	AUC	MADD
Sub-regions	Northern Africa	0.82	1.03+	0.73	0.95+	0.69	0.28+
	Western Africa	0.75	**0.81-**	0.73	**0.59-**	0.75	0.64+
	Middle Africa	**0.73**	**0.47-**	0.74	**0.40-**	0.70	**0.22-**
HDI	Low	**0.74**	**0.75-**	0.73	**0.54-**	0.74	**0.66-**
	Medium	0.75	**0.73-**	0.75	**0.65-**	0.73	**0.22-**
	High	0.87	0.94+	0.78	0.86+	**0.63**	0.19+
Literacy	Low	0.76	**0.78-**	0.73	**0.58-**	0.74	**0.63-**
	Medium	0.73	**0.49-**	0.72	**0.44-**	0.73	**0.29-**
	High	0.84	0.82+	0.78	0.74+	0.69	**0.18-**
	Very high	0.71	**0.90-**	0.74	0.68-	0.77	0.45+
Quality of education	Low	0.72	**0.79-**	0.73	**0.62-**	0.72	0.52-
	Lower middle	0.76	0.62+	0.74	0.48+	0.74	0.52+
	Upper middle	0.78	0.78+	0.77	0.66+	0.68	0.27+
	High	0.71	**1.01-**	0.66	**0.96-**	0.83	0.40+
GDI	Low	0.75	0.53-	0.74	0.41-	0.73	0.42-
	High	0.75	0.85-	0.72	0.68-	0.75	0.57-
Countries	Benin	0.78	**0.75-**	0.76	0.61-	0.72	0.37-
	Burkina Faso	**0.71**	**0.84-**	0.71	**0.79-**	0.75	0.35-
	Cameroon	0.76	**0.89-**	0.76	**0.75-**	**0.69**	**0.23-**
	DR Congo	0.72	**0.57-**	0.71	0.56-	0.72	0.16-
	Ivory Coast	0.75	**0.66-**	**0.69**	**0.77-**	0.76	0.66-
	Senegal	**0.71**	**0.90-**	0.74	**0.88-**	0.77	0.45+
	All Africa	**0.75**	0.86-	**0.73**	0.60-	0.73	0.70-
	All OECD	0.81	0.86+	0.77	0.60+	0.70	0.70+

AUC and MADD, suggesting that no unfair treatments was observed in the models' outputs for this factor. This could stem from the fact that gender gaps are on the high side on the UN gender gap scales, meaning that this factor does not capture in sufficient details the nuances in gender gap across African countries, or conversely that all model can be trained fairly across GDI groups.

Countries. Table 3 shows that none of the African countries reach an AUC above 0.8, and that performances vary substantially across models. Friedman tests reveal a significant main effect of country on AUC for LR ($p = 0.006, \eta^2 = 0.25$), SGB ($p = 0.008, \eta^2 = 0.24$) and NB ($p = 0.002, \eta^2 = 0.29$). Post-hoc tests further indicate that LR is significantly less accurate for all African countries as compared to OECD ones, SGB for learners from Ivory Coast, and NB for learners from Cameroon. This shows that no model can perform reliably across African countries. There are also main effects of country on MADD for LR, SGB and NB. Post-hoc tests show that LR is more biased against all African countries as compared to OECD ones, SGB against all countries but Benin and DR Congo, and NB against Cameroon. Lastly, similarly to [13] when comparing African versus OECD countries (last two rows of Table 3), significant biases are observed for LR ($p = 0.002, \eta^2 = 0.19$) and SGB ($p = 0.039, \eta^2 = 0.11$), confirming that these models are more biased against Africa, even though African learners constitute the majority of the learners in this dataset.

4. DISCUSSION

Our results do show several patterns of unfairness in our ML predictors, thus answering our research question affirmatively. Three main trends of unfairness emerge in our results. First, we found evidence of biases toward learners from Middle Africa, a region that, to our knowledge, has received no prior attention in the ML fairness literature. We thus contribute to this work by providing strong evidence of biases in ML predictors trained on an international educational dataset for this part of Africa. Within Middle Africa, learners from Cameroon are exposed to biases in all models, especially NB. This calls for investigating further the possible reasons that could shape these learners' data in future work. In particular, Cameroon is characterized by strong multilingualism, severe infrastructural deficiencies in rural schools, and high dropout rates in primary and secondary education among girls and students from lower-income households [20]. Second, we uncovered biases against Western Africa, at least with LR and SGB. In related work on AI fairness in education, solely one study [6] looked at Nigerian learners (in Western Africa), without finding evidence of unfairness, possibly due to their small number of Nigerian data points. Within this region of Africa, we found in particular that learners from Burkina Faso, Senegal and Ivory Coast are exposed to varied levels of biases in the ML models, further reinforcing the per-African country analysis we propose in this paper. In a recent institutional report on deploying a MOOC at scale in Western Africa, educators from these countries reported specific issues that could explain different usage patterns of MOOCs resources [25].

Specifically, educators reported Internet connectivity issues leading to student dropouts in Burkina Faso; a clear lack of experience with remote learning in Ivory Coast; and the need for offline access to the MOOC in Senegal to enable participation for all learners. For Ivory Coast, we also found very low levels of interaction across the board, further showing that students from this country may have struggled with the MOOC's activities, which could have in turn make it more difficult for ML models to learn from these data. Third, we also found evidence of unfairness that appears to be matching existing struggles, namely countries with low HDI and low levels of adult literacy. To our knowledge, no prior studies have uncovered these trends, which, worth noticing, are not explained by sample size, since many biases affect large groups (e.g., Low HDI, Middle Africa). Overall, our findings highlight the need to broaden fairness research in ML to more diverse contexts, including replicating existing fairness studies in African settings.

Furthermore, as for the ML models, we found that LR generates substantial unfairness towards several groups, and in general is more favourable to OECD learners, which is consistent with the findings in [13] that LR is the most accurate but the most biased model. While this previous work also found that NB is the fairest model, we actually found that NB can be biased as well (e.g., toward Cameroon), showing that examining in depth unfairness is crucial to actually deploy an ML model. This confirms that different models trained on the same data can generate their own algorithmic biases.

Overall, the absence of correlation between the observed unfairness and subgroup pass rates, proportional representation, or absolute numbers of successful learners suggests that the sources of bias in our setting differ from typical representational and measurement biases, typically identified in Western educational contexts [4]. This observation complements the regional patterns discussed above and supports the hypothesis that commonly used modelling features, largely designed around assumptions and behavioural norms characteristic of Western learning environments, may not transfer well to African contexts. In particular, prior work has shown that while infrastructural constraints constitute a major barrier to MOOC adoption (often exceeding challenges related to self-discipline [9]), course contextualization, including the extent to which content reflects learners' lived realities, also plays a critical role in shaping engagement [5]. Alongside these factors, variations in digital literacy, constrained access to reliable infrastructure, and educational trajectories may shape learner behaviours in ways that are not adequately captured by current feature representations. As a result, existing modelling approaches may fail to encode meaningful indicators of learning and engagement, thereby contributing to systematic disparities in predictions. These findings highlight the need for context-sensitive feature design and fairness interventions explicitly accounting for linguistic, cultural, and infrastructural differences.

In addition to potential biases introduced by the ML models, the diversity of learning contexts and modalities for accessing the MOOC in African countries likely contributed to greater feature heterogeneity, which in turn affected model training. In the GDP-MOOC dataset, for instance, African learners interacted significantly less with videos (fewer videos played, and fewer actions such as pausing, seeking, or enabling transcripts) than OECD learners, according to a t-test with amount of video interaction as the dependent measure ($p < .05$). Within African subgroups, learners from the DR Congo showed the lowest levels of video interaction. These findings are consistent with prior work (see Introduction) emphasizing diverse modalities and challenges for accessing remote courses in Africa (e.g., mobile learning, access through digital open spaces), which may help explain some of the patterns observed here.

5. CONCLUSION

In this work, we explored the biases in ML predictors of MOOC completion, focusing on African learners, by leveraging the publicly available GDP-MOOC dataset, which includes logged events, outcomes, and learners' countries, with about half of them being African. To the best of our knowledge, our study is the first that focuses on eliciting complex patterns of unfairness within Africa in educational data.

Overall, our findings show diverse forms of unfairness that vary by ML model, highlighting the importance of investigating ML unfairness for subgroups of African learners. As these findings show no association with subgroup pass rates, proportional representation, or absolute numbers of successful learners, they suggest that the observed biases cannot be attributed solely to representation effects.

These findings provide strong evidence for the need to control for the fairness of ML predictors in education before deploying them in Africa, and provide insights about the specific groups needing more attention. However, due to the GDP-MOOC being in French, most of the African students came from countries with a sizable French-speaking population, typically Western, Middle, and Northern Africa, and future work should consider other parts of Africa and multilingual MOOCs. Another limitation is that we focused on one offering of a single MOOC, hence future work should aim at replicating the results over a larger and more balanced pool of MOOC learners. Lastly, while we provide possible reasons for the observed biases in the ML models, including structural and economical challenges documented in African studies about MOOCs, future work is still needed to link algorithmic unfairness with actual underlying causes.

With the rise of ML-driven tools in education, it is crucial to carefully consider their robustness for African learners, a continent with millions of learners who have received little attention so far in existing ML fairness studies in education. If we are to fulfill the promise that knowledge can reach a wider audience, we must pay close attention to cultural diversity at every stage of the educational and ML pipeline.

Acknowledgements. We thank Rémi Bachelet and the MOOC GdP's team for the data collection.

6. REFERENCES

[1] E. R. Aikins and J. Cilliers. Education. `https://futures.issafrica.org/thematic/06-education/`, 2026. Published online at futures.issafrica.org. Updated October 11, 2024. Accessed February 9, 2026.

[2] S. Alghamdi, B. Soh, and A. Li. A comprehensive review of dropout prediction methods based on multivariate analysed features of mooc platforms. *Multimodal Technologies and Interaction*, 9(1):3, 2025.

[3] H. Anderson, A. Boodhwani, and R. S. Baker. Assessing the fairness of graduation predictions. In *Proceedings of The 12th International Conference on Educational Data Mining*, 2019.

[4] R. S. Baker and A. Hawn. Algorithmic bias in education. *International Journal of Artificial Intelligence in Education*, pages 1052–1092, 2021.

[5] S. Boga and R. McGreal. Introducing moocs to africa: New economy skills for africa program. 2014.

[6] B. Bridgeman, C. Trapani, and Y. Attali. Considering fairness and validity in evaluating automated scoring. In *Proceedings of the Annual meeting of the National Council on Measurement in Education*, 2009.

[7] J. Buolamwini and T. Gebru. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, pages 77–91, 2018.

[8] S. V. Chinta, Z. Wang, Z. Yin, N. Hoang, M. Gonzalez, T. L. Quy, and W. Zhang. Fairaied: Navigating fairness, bias, and ethics in educational ai applications. *arXiv preprint arXiv:2407.18745*, 2024.

[9] C. E. Franco Yáñez, D. Nigmonova, and W. Panichpathom. Demoocratization of education?: Massive open online courses (moocs) and the opportunities and challenges for developing countries. 2014.

[10] J. Gardner, C. Brooks, and R. Baker. Evaluating the fairness of predictive student models through slicing analysis. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 225–234, 2019.

[11] Q. Hu and H. Rangwala. Towards fair educational data mining: A case study on detecting at-risk students. In *Proceedings of The 13th International Conference on Educational Data Mining (EDM)*, 2020.

[12] A. I. Jouicha, K. Berrada, R. Bendaoud, S. Machwate, A. Miraoui, and D. Burgos. Starting moocs in african university: The experience of cadi ayyad university, process, review, recommendations, and prospects. *IEEE Access*, 8:17477–17488, 2020.

[13] S. Lallé, F. Bouchet, M. Verger, and V. Luengo. Fairness of mooc completion predictions across demographics and contextual variables. In *International Conference on Artificial Intelligence in Education*, pages 379–393. Springer, 2024.

[14] R. Lembani, A. Gunter, M. Breines, and M. T. B. Dalu. The same course, different access: the digital divide between urban and rural distance education students in south africa. *Journal of Geography in Higher Education*, 44(1):70–84, 2020.

[15] X. Li, D. Song, M. Han, Y. Zhang, and R. F. Kizilcec. On the limits of algorithmic prediction across the globe. *arXiv preprint arXiv:2103.15212*, 2021.

[16] V. Maphosa and M. Maphosa. Opportunities and challenges of adopting moocs in africa: A systematic literature review. *Massive Open Online Courses-Current Practice and Future Trends*, 2023.

[17] B. Memarian and T. Doleck. Fairness, accountability, transparency, and ethics (fate) in artificial intelligence (ai) and higher education: A systematic review. *Computers and Education: Artificial Intelligence*, 5:100152, 2023.

[18] P. M. Moreno-Marcos, C. Alario-Hoyos, P. J. Muñoz-Merino, and C. D. Kloos. Prediction in moocs: A review and future research directions. *IEEE transactions on Learning Technologies*, 12(3):384–401, 2018.

[19] N. Nasiar, C. Belitz, H. Lee, F. Stinar, R. Baker, J. Ocumpaugh, S. Fancsali, S. Ritter, and N. Bosch. Investigating algorithmic bias in affect detectors with constructed categories of student identity. In *Proceedings of the International Conference on Computers in Education*, 2025.

[20] A. Ndindeng. Factors influencing under-education in cameroon: A comprehensive analysis. *African Educational Research Journal*, 12(3):182–201, 2024.

[21] N. Pham, H. P. Ngoc, and A. Nguyen-Duc. Fairness for machine learning software in education: A systematic mapping study. *Journal of Systems and Software*, 219:112244, 2025.

[22] L. Sha, M. Raković, A. Das, D. Gašević, and G. Chen. Leveraging class balancing techniques to alleviate algorithmic bias for predictive tasks in education. *IEEE Transactions on Learning Technologies*, 15(4):481–492, 2022.

[23] V. Švábenskỳ, M. Verger, M. Rodrigo, C. Monterozo, R. Baker, M. L. Saavedra, S. Lallé, and A. Shimada. Evaluating algorithmic bias in models for predicting academic performance of filipino students. In *Proceedings of the 17th Educational Data Mining Conference*, 2024.

[24] K. Sylla, G. M. M. Nkwetchoua, and F. Bouchet. How does the use of open digital spaces impact students success and dropout in a virtual university?. *International Association for Development of the Information Society*, 2022.

[25] C. Tkaczuk, L. Jeannin, and A. Jaillet. Rapport du mooc certicescol – session 2 – année 2021 – cy cergy paris université, inspé versailles, cy bonheurs, 2025.

[26] M. Verger, C. Fan, S. Lallé, F. Bouchet, and V. Luengo. A comprehensive study on evaluating and mitigating algorithmic unfairness with the madd metric. *Journal of Educational Data Mining*, 16(1):365–409, 2024.

[27] M. Verger, S. Lallé, F. Bouchet, and V. Luengo. Is your model" madd"? a novel metric to evaluate algorithmic fairness for predictive student models. In *Proceedings of the International Conference on Educational Data Mining*, pages 91–102, 2023.

[28] A. A. Yunusa, I. N. Umar, and B. Bervell. Massive open online courses (moocs) in sub-saharan african higher education landscape: a bibliometric review. *MOOC*, pages 1–25, 2021.

[29] A. F. Zambrano, J. Zhang, and R. S. Baker. Investigating algorithmic bias on bayesian knowledge tracing and carelessness detectors. In *Proceedings of the 14th International Conference on Learning Analytics & Knowledge*, 2024.

Cognitive and Metacognitive Process Differences in High and Low Writing Performers : A Trace-Based SRL Analysis of Web Log Data

Hyeeun Kim[1], Minju Kang[1], Jeongyun Han[2], Young Hoan Cho[1]
[1]Learning Sciences Research Institute, Department of Education, Seoul National University
[2]Graduate School of Education, University of Seoul
Kim5039@snu.ac.kr, minjuk@snu.ac.kr, hanjy@uos.ac.kr, yhcho95@snu.ac.kr

ABSTRACT

This study examined how self-regulated learning (SRL) processes differed between high- (n = 54) and low-performing (n = 57) writers during a 30-minute learning task. Adapting Fan et al.'s (2022) trace-based SRL protocol, we mapped 124,398 web log events onto eight SRL processes and further distinguished progress monitoring (timer/checklist checking) from comprehension monitoring (self-generated questioning). Group differences were tested using Mann–Whitney U tests with Benjamini–Hochberg correction across 5-minute windows, and monitoring–control coupling was examined through first-order Markov transitions. The groups did not differ in overall metacognitive frequency, but the low-performing group showed higher metacognitive activity at 20–25 min, mainly driven by comprehension-monitoring events that fed into quiz-response loops rather than strategic adaptation. In contrast, the high-performing group transitioned from Elaboration to Organisation earlier in the task. Although effect sizes were small to small-to-medium and several findings were marginal after correction, the results suggest that the effectiveness of monitoring–control coupling, rather than monitoring frequency alone, may distinguish writing performers.

Keywords

self-regulated learning, process mining, metacognitive monitoring, trace data, educational data mining

INTRODUCTION

Self-regulated learning (SRL) is a dynamic process in which learners actively set goals, select strategies, and monitor their progress toward desired outcomes [14, 15]. In technology-enhanced learning environments, SRL processes can be traced through timestamped log data that record learner interactions [2, 13]. Recent advances have enabled researchers to detect theoretically meaningful SRL processes from trace data using process mining techniques [4, 11].

A central yet under-investigated question is how the temporal deployment of metacognitive processes relates to learning outcomes. Prior studies have shown that self-regulated learners outperform less-regulated peers [1] , but most analyses treat metacognition as a unitary construct, without distinguishing between functionally different types of monitoring. This is problematic because monitoring frequency alone may not indicate effective self-regulation; what matters is whether monitoring triggers appropriate control actions [9, 14]. We restrict our scope to this monitoring–control coupling question in a single writing task, and do not make broader claims about motivational orientations such as mastery- versus performance-approach goals.

We address this gap by (a) adapting Fan et al.'s (2022) trace-based SRL measurement protocol, grounded in Bannert's (2007) framework, to a writing task context; (b) extending their process library by distinguishing *progress monitoring* from *comprehension monitoring*; and (c) examining monitoring–control coupling between groups using Markov transition analysis. Three research questions guide this study:

RQ1. How do cognitive and metacognitive SRL processes differ in temporal deployment between high and low writing performers?

RQ2. Does the temporal concentration of Organisation and Elaboration processes differ between groups across 5-minute time windows?

RQ3. Do progress monitoring and comprehension monitoring differ between groups in temporal distribution and downstream transitions?

METHOD

Participants and Task

111 sixth-grade elementary-school students (age 11) participated in a 30-minute web-based learning task that involved planning, video watching with embedded questions, and essay writing. Participants were recruited through promotional postings in nationwide online parent communities, and parental consent was obtained before any data were collected. Based on writing quality ratings, participants were classified into a high-performing group (n = 54) and a low-performing group (n = 57). A total of 124,398 timestamped events across 11 activity types were recorded. The study was reviewed and approved by the institutional review board of the Seoul National University (IRB No. 2507/001-014); both parental written informed consent and child assent were obtained prior to data collection, and log data were de-identified before analysis.

SRL Process Coding

We adapted the trace-based measurement protocol of Fan et al. (2022), which operationalises Bannert's (2007) theoretical

Hyeeun Kim, Minju Kang, Jeongyun Han, and Young Hoan Cho. Cognitive and Metacognitive Process Differences in High and Low Writing Performers: A Trace-Based SRL Analysis of Web Log Data. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 611–615. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039829

framework of metacognition and cognition in hypermedia learning. The protocol maps raw trace data to learning actions (action library) and then to SRL processes (process library) through theoretically grounded sequence rules. Because the platform actions logged in our writing environment differ from the multi-channel hypermedia actions in Fan et al.'s original study, we re-constructed the action library to fit our 11 platform-logged actions and then mapped each action to the closest SRL process in Bannert's (2007) taxonomy; where the process requires temporal ordering, the mapping was applied to a two-step action sequence rather than to a single action. The resulting process library (Table 1) maps the 11 actions to eight SRL processes: five metacognitive (Orientation, Planning, Progress Monitoring, Comprehension Monitoring, Evaluation) and three cognitive (First-reading, Elaboration, Organisation). All mapping decisions were independently reviewed by the authors against Bannert's (2007) definitions before the process counts were computed.

Building on Fan et al.'s trace-based SRL framework, we retained Monitoring (MC.M) as a higher-order category but differentiated it into two theoretically meaningful subtypes. This refinement was based on the referent of monitoring rather than on surface action types alone. Progress Monitoring (MC.M_prog), comprising CHECK_TIMER, USE_CHECKLIST, and VIDEO → CHECKLIST, reflects learners′ monitoring of task progress, time, and procedural requirements, which is consistent with Zimmerman′s (2000) account of self-monitoring during performance. In contrast, Comprehension Monitoring (MC.M_comp), comprising QUESTION_VIDEO, captures learners' voluntary generation of questions about video content and can be interpreted as monitoring one's internal understanding. This interpretation aligns with reading and metacomprehension research, where self-questioning is viewed as an indicator of recognizing uncertainty or a comprehension gap [6, 10].

While Fan et al. (2022) treated monitoring-related traces as a single category, distinguishing MC.M_prog from MC.M_comp allows us to examine whether monitoring has different regulatory value depending on what learners monitor: task progress or internal understanding. This distinction is further grounded in Nelson and Narens's (1990) monitoring-control framework and the COPES model [5], which suggest that monitoring contributes to regulation when the information being monitored informs subsequent adaptive control.

Statistical Approach

Following Fan et al. (2022), who noted that SRL process distributions are typically non-normal and zero-inflated, all group comparisons used Mann–Whitney U tests. We retain this rank-based test as our primary inferential procedure for two reasons. First, the U statistic compares the full rank ordering of observations and tests stochastic dominance, not equality of medians; it therefore remains a valid group comparison even in cells where both group medians coincide at zero, since the rank-level difference between zero and non-zero observations is preserved in U. Second, the rank-biserial r derived from U ($r = 1 - 2U / n_1 n_2$) [8] is a standardised effect size that does not depend on the centre or scale of the distribution and is therefore directly interpretable for zero-inflated counts; $|r| \geq .20$ is treated as a small-to-medium effect, and Cohen's d is reported alongside r for interpretive reference. Across all 5- and 10-minute-window comparisons within each research question we applied a Benjamini–Hochberg false-discovery-rate correction (q = .05)[3], and both raw and corrected p values are reported; marginal effects ($.05 < p < .10$) are treated as exploratory and emphasised through effect sizes rather than p values. For temporal analysis, SRL process ratios were computed per 5-minute window, and first-order Markov transition probabilities were computed to examine monitoring–control coupling, providing an event-conditional view that is by construction free of the zero-inflation issue.

RESULTS

RQ1: Temporal Deployment of SRL Processes

Table 2 presents median SRL process durations across 10-minute windows. In the first 10 minutes, both groups primarily engaged in Planning and Elaboration, with the high group showing marginally greater Elaboration (Mdn: 247 vs. 183 s, $U = 1868$, $p = .052$, $r = –.21$). The critical divergence emerged at 10–20 min: the high group showed significantly greater Organisation (Mdn: 246 vs. 173 s, $U = 1953$, $p = .015$, $r = –.27$). In the 20–30 min window, the low group showed significantly greater Elaboration ($U = 1227$, $p = .006$, $r = +.20$) and marginally greater Progress Monitoring (Mdn: 13 vs. 5 s, $U = 1260$, $p = .099$, $r = +.18$), the latter likely reflecting anxiety-driven time checking when behind schedule.

Figure 1 shows dominant SRL process proportions across 5-minute windows. By 15–20 min, 81% of high performers had Organisation as their dominant process versus 61% of low performers, who maintained Elaboration longer (33% vs. 15% at 15–20 min).

Table 1. Process library mapping learning actions to SRL processes. Adapted from the measurement protocol of Fan et al. (2022), grounded in Bannert's (2007) theoretical framework. *MC.M_comp is our extension, not present in the original framework.

SRL Process	Learning Actions	Definition
MC.O Orientation	READ_INSTRUCTION NAV to INSTRUCTION	Orientation on task requirements and learning goals [2, 4].
MC.P Planning	PLAN_* ; NAV to PLAN	Planning by arranging activities and determining strategies[4].
MC.M_prog	CHECK_TIMER; USE_CHECKLIST;	Monitoring progress against time and task requirements. Progress monitoring of task completion and time management [15].
MC.M_comp*	QUESTION_VIDEO	Self-generated questioning indicating comprehension gap recognition. Monitoring without effective control when followed by repeated loops [6, 10, 14].
MC.E Evaluation	ANSWER_QUIZ	External evaluation of content understanding via platform-provided comprehension checks.
LC.F First-reading	WATCH_VIDEO	Initial information acquisition from instructional video.
HC.E Elaboration	USE_MEMO;	Connecting and recording content; note-taking during video as elaboration [7].

SRL Process	Learning Actions	Definition
	VIDEO → MEMO	
HC.O Organisation	WRITE_ESSAY; MEMO → ESSAY	Text production integrating learned information into coherent essay structure [4].

Table 2. SRL process duration (Mdn, seconds) by group and 10-min window. Mann–Whitney U test with rank-biserial r. Bold r values indicate |r| ≥ .20. p values are Benjamini–Hochberg corrected within each window. H = High (n = 54); L = Low (n = 57).

SRL Process	0–10 min				10–20 min				20–30 min			
	H	L	U	r	H	L	U	r	H	L	U	r
MC.O	11.3	17.9	1220	+0.21	0.0	0.0	1598	-0.04	0.0	0.0	1570	-0.02
MC.P	87.5	102.8	1336	+0.13	0.0	0.0	1543	0.00	0.0	0.0	1568	-0.02
MC.M_prog	34.2	33.2	1431	+0.07	11.3	20.0	1340	+0.13	5.1	13.1	1260	+0.18
MC.M_comp	0.0	0.0	1530	+0.01	0.0	0.0	1423	+0.08	0.0	0.0	1456	+0.05
MC.E	10.6	0.0	1710	-0.11	12.4	22.7	1321	+0.14	0.0	0.0	1299	+0.16
LC.F	80.1	94.3	1384	+0.10	39.9	45.6	1450	+0.06	0.0	0.0	1711	-0.11
HC.E	247.1	183.4	1868	-0.21	114.4	158.6	1459	+0.05	0.0	0.0	1227	+0.20
HC.O	0.0	0.0	1542	0.00	245.7	172.5	1953	-0.27	441.2	329.6	1860	-0.21

Note. Cells in which both groups display Mdn = 0.0 arise from zero-inflation, driven by the phase-structured task design (e.g., Planning concentrates in the opening minutes and Organisation in the closing minutes), by the voluntary low-base-rate nature of comprehension monitoring (QUESTION_VIDEO; see §4.2), and by two-step sequence rules in the process library that further restrict the non-zero rate. Mann–Whitney U is rank-based and remains valid in such cells: U and r summarise the rank-level difference between groups (driven by the proportion of non-zero observations) rather than the tied central tendency.

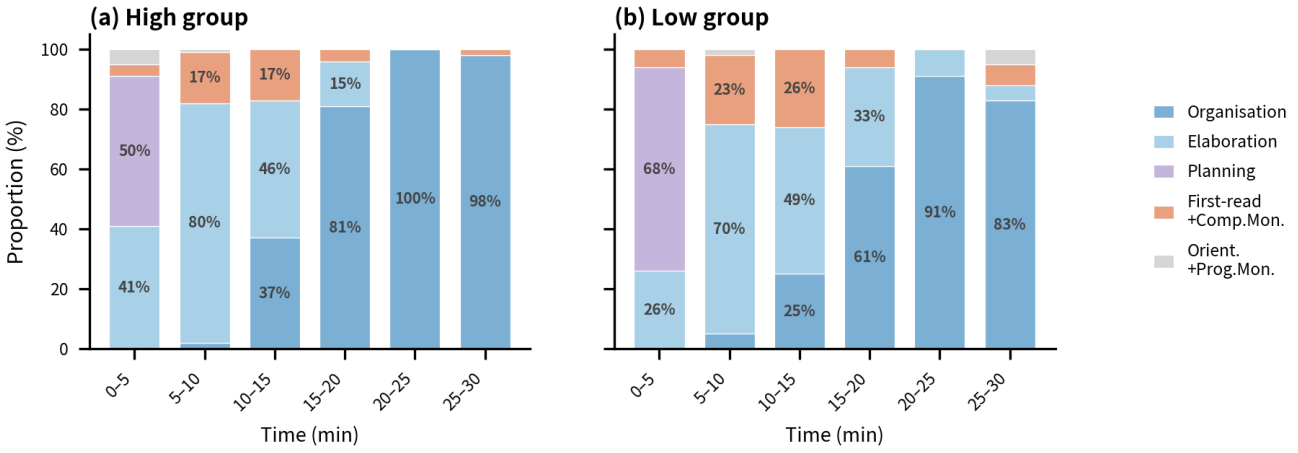


Figure 1. Dominant SRL process proportions across 5-min windows for high (a) and low (b) groups.

RQ2: Temporal Concentration

Organisation concentration (Figure 2a) showed a marginally significant difference at 15–20 min ($U = 1868$, $p = .052$, $r = –.21$). Elaboration (Figure 2b) showed a crossover: the high group had marginally higher Elaboration at 0–5 min ($U = 1825$, $p = .085$, $r = –.19$) but significantly lower Elaboration at 20–25 min ($U = 1127$, $p = .015$, $r = +.18$) and 25–30 min ($U = 720$, $p = .014$, $r = +.14$), indicating earlier completion of the Elaboration-to-Organisation phase transition.

RQ3: Monitoring and Control Coupling

Total metacognition ratio (Figure 2d) was significantly higher for the low group at 20–25 min ($U = 1015$, $p = .020$, $r = +.26$). Comprehension Monitoring (Figure 2c) showed elevated trends for the low group, particularly at 10–15 min (Mdn: 0% for both groups, but the low group's distribution had a substantially heavier right tail). Progress Monitoring was marginally higher for the low group at 20–25 min ($U = 1086$, $p = .061$, $r = +.21$).

Post-monitoring transition analysis (Figure 3) revealed differential monitoring–control coupling. After Progress Monitoring, both groups transitioned primarily to Organisation and Navigation, with similar patterns. However, after Comprehension Monitoring, the low group transitioned to Evaluation (quiz loop) at a higher rate (35.5% vs. 27.3%), while the high group showed higher Navigation probability (24.2% vs. 15.1%), suggesting strategic disengagement from unproductive comprehension checking.

The transition WRITE_PLAN → NAVIGATE was significantly more frequent in the low group (Mdn: 1 vs. 0, $U = 1155$, $p = .010$, $r = +.25$), suggesting greater post-planning disorientation. Conversely, WRITE_ESSAY → NAVIGATE was significantly more frequent in the high group ($U = 1869$, $p = .037$, $r = –.21$),

reflecting more active navigation during writing.

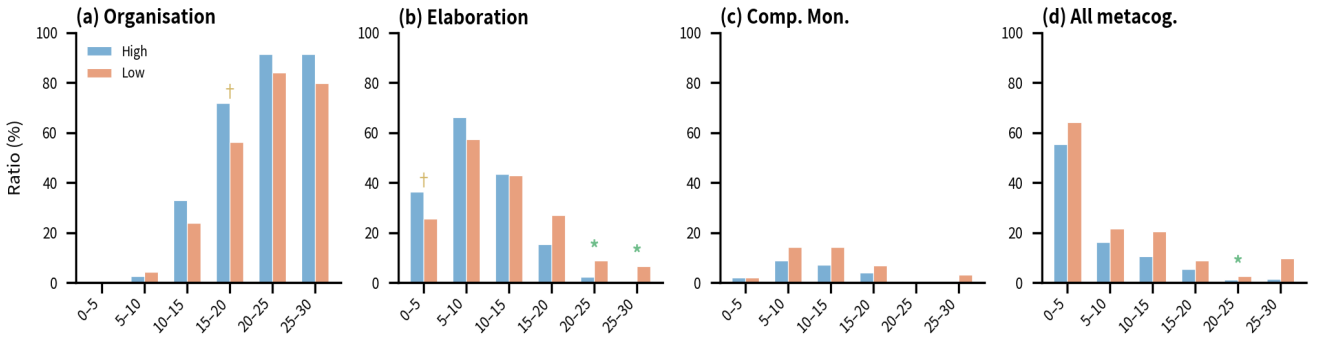


Figure 2. SRL process concentration across 5-min windows. (a) Organisation. (b) Elaboration. (c) Comprehension Monitoring. (d) Total metacognition. * $p < .05$, † $p < .10$.

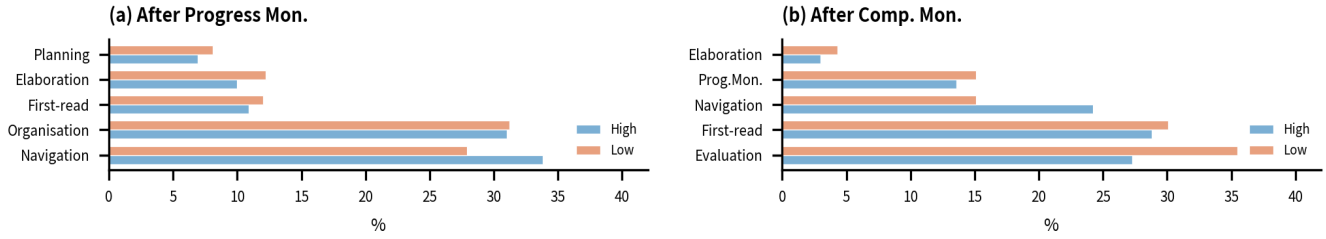


Figure 3. Post-monitoring transition probabilities. (a) After Progress Monitoring: similar patterns. (b) After Comprehension Monitoring: low group → Evaluation (quiz loop); high group → Navigation (strategic redirection).

DISCUSSION

Applying a trace-based SRL measurement protocol adapted from Fan et al. (2022) to web log data from a single writing task, this study finds that, within the bounds of this task, writing performance differences are associated with the functional effectiveness of metacognitive processes rather than with their raw frequency.

Monitoring without effective control. Our most theoretically significant finding is the differential monitoring–control coupling between groups. The low group engaged in more Comprehension Monitoring (self-questioning during video), yet this monitoring disproportionately led to Evaluation (quiz-answering) rather than strategic redirection. This exemplifies what Winne and Hadwin (1998) describe as a breakdown in the monitoring–control loop: learners detect a comprehension gap but respond passively rather than adapting strategy [14]. That the total metacognition ratio was significantly *higher* for the low group at 20–25 min (p = .020) directly challenges the common assumption that more metacognition implies better self-regulation.

The Elaboration-to-Organisation phase transition. The Elaboration crossover pattern, in which Elaboration is higher early and lower late for the high group, demonstrates a phase transition aligned with Kellogg's (1996) writing model [7]. The high group completes this transition earlier not because they elaborate less overall but because their early elaboration is more intensive, enabling a faster transition to text production.

Progress Monitoring as a lagging indicator. The marginally higher Progress Monitoring in the low group at 20–25 min likely represents reactive time checking when behind schedule, rather than proactive self-regulation. This cautions against equating monitoring frequency with quality, consistent with Schraw's (1998) distinction between monitoring accuracy and frequency [12].

Methodological Contributions

This study extends the trace-based SRL measurement protocol of Fan et al. (2022) in two ways [4]. First, our decomposition of Monitoring into productive and comprehension subtypes reveals functionally distinct processes with opposing relationships to performance, a distinction invisible when Monitoring is treated as a unitary category. Second, the post-monitoring transition analysis operationalises the monitoring–control coupling concept from Winne and Hadwin's (1998) COPES model in a data-driven manner, providing an empirical method for assessing whether monitoring leads to effective strategy adaptation [14].

Limitations

Several limitations qualify these findings. First, causal direction cannot be established from cross-sectional trace data; the observed Elaboration-to-Organisation timing differences may function either as antecedents or consequences of writing quality. Second, WRITE_ESSAY frequency is partly confounded with text quantity, creating potential circularity with the quality-based grouping. Future work should use independent baseline writing measures and consider sequence balancing to reduce the influence of overall activity volume on SRL process counts. Third, several SRL-process duration variables were zero-inflated, partly due to the phase-structured task design, the low-base-rate nature of QUESTION_VIDEO, and the restrictive two-step sequence rules. Although the Mann–Whitney U test can capture rank-level group differences under these conditions, medians are less informative when many observations are zero. Future studies could use hurdle or zero-inflated models to distinguish the presence of a process from its duration conditional on occurrence. Fourth, some effects were marginal and small to small-to-medium in size; even with Benjamini–Hochberg correction, the findings should be interpreted as preliminary and require replication with larger samples. Fifth,

the process library relied only on single-action and two-step sequence detection. As Fan et al. (2022) showed, richer multi-channel data such as eye-tracking or think-aloud protocols can improve SRL-process detection granularity, and our mapping may under-capture higher-order regulatory episodes. Finally, because the data came from a single writing task and learner population, cross-task and cross-domain validation is needed before the proposed monitoring decomposition can be treated as a general SRL measurement contribution.

ACKNOWLEDGMENTS

This study was supported by the research grant (0767-20250004) from Learning Sciences Research Institute at Seoul National University.

REFERENCES

[1] Bannert, M. and Reimann, P. 2012. Supporting self-regulated hypermedia learning through prompts. *Instructional Science*. 40, 1 (Jan. 2012), 193–211. https://doi.org/10.1007/s11251-011-9167-4.

[2] Bannert, M., Reimann, P. and Sonnenberg, C. 2014. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and Learning*. 9, 2 (Aug. 2014), 161–185. https://doi.org/10.1007/s11409-013-9107-6.

[3] Benjamini, Y. and Hochberg, Y. 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological)*. 57, 1 (Jan. 1995), 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

[4] Fan, Y., Lim, L., van der Graaf, J., Kilgour, J., Raković, M., Moore, J., Molenaar, I., Bannert, M. and Gašević, D. 2022. Improving the measurement of self-regulated learning using multi-channel data. *Metacognition and Learning*. 17, 3 (Dec. 2022), 1025–1055. https://doi.org/10.1007/s11409-022-09304-z.

[5] Greene, J.A. and Azevedo, R. 2007. A Theoretical Review of Winne and Hadwin's Model of Self-Regulated Learning: New Perspectives and Directions. *Review of Educational Research*. 77, 3 (Sept. 2007), 334–372. https://doi.org/10.3102/003465430303953.

[6] Joseph, L.M., Alber-Morgan, S., Cullen, J. and Rouse, C. 2016. The Effects of Self-Questioning on Reading Comprehension: A Literature Review. *Reading & Writing Quarterly*. 32, 2 (Mar. 2016), 152–173. https://doi.org/10.1080/10573569.2014.891449.

[7] Kellogg, R.T. 1996. A model of working memory in writing. *The Science of Writing*. Routledge.

[8] Kerby, D.S. 2014. The Simple Difference Formula: An Approach to Teaching Nonparametric Correlation. *Comprehensive Psychology*. 3, (Jan. 2014), 11.IT.3.1. https://doi.org/10.2466/11.IT.3.1.

[9] Nelson, T.O. and Narens, L. 1994. Why investigate metacognition? *Metacognition: Knowing about knowing*. 1–25.

[10] Ozuru, Y., Kurby, C.A. and McNamara, D.S. 2012. The effect of metacomprehension judgment task on comprehension monitoring and metacognitive accuracy. *Metacognition and Learning*. 7, 2 (Aug. 2012), 113–131. https://doi.org/10.1007/s11409-012-9087-y.

[11] Saint, J., Whitelock-Wainwright, A., Gašević, D. and Pardo, A. 2020. Trace-SRL: A Framework for Analysis of Microlevel Processes of Self-Regulated Learning From Trace Data. *IEEE Transactions on Learning Technologies*. 13, 4 (Oct. 2020), 861–877. https://doi.org/10.1109/TLT.2020.3027496.

[12] Schraw, G. 1998. Promoting general metacognitive awareness. *Instructional Science*. 26, 1 (Mar. 1998), 113–125. https://doi.org/10.1023/A:1003044231033.

[13] Siadaty, M., Gašević, D. and Hatala, M. 2016. Associations between technological scaffolding and micro-level processes of self-regulated learning: A workplace study. *Computers in Human Behavior*. 55, (Feb. 2016), 1007–1019. https://doi.org/10.1016/j.chb.2015.10.035.

[14] Winne, P.H. and Hadwin, A.F. 1998. Studying as self-regulated engagement in learning. (1998), 277–304.

[15] Zimmerman, B.J. 2002. Becoming a Self-Regulated Learner: An Overview. *Theory Into Practice*. 41, 2 (May 2002), 64–70. https://doi.org/10.1207/s15430421tip4102_2.

Comparing RAG and GraphRAG for Page-Level Retrieval Question Answering on a Math Textbook

Eason Chen
Carnegie Mellon University
eason.tw.chen@gmail.com

Chuangji Li
Carnegie Mellon University
chuangjl@andrew.cmu.edu

Eric Li
Carnegie Mellon University
shizhuol@andrew.cmu.edu

Zimo Xiao
Carnegie Mellon University
zimox@andrew.cmu.edu

Jionghao Lin
The University of Hong Kong
jionghao@hku.hk

Ken Koedinger
Carnegie Mellon University
koedinger@cmu.edu

ABSTRACT

Large language models (LLMs) show promise as educational aids but often lack alignment with specific course materials. We investigate Retrieval-Augmented Generation (RAG) and GraphRAG for page-level question answering on an undergraduate mathematics textbook. Using a curated dataset of 477 question-answer pairs, each tied to a specific textbook page, we compare five embedding-based RAG models, a BM25 baseline, and GraphRAG across two metrics: retrieval accuracy (whether the correct page is retrieved) and answer quality (F1 score). Our results show that embedding-based RAG outperforms GraphRAG for page-level retrieval, with `voyage-3-large` achieving 99.4% accuracy at top-10 (bootstrap 95% CI for top-1: [.644, .728]). BM25 proves a strong baseline, outperforming several embedding models. Error analysis reveals that 63.3% of top-1 failures retrieve same-chapter content, suggesting pedagogical relevance even in failure cases. GraphRAG retrieves excessive context (∼47K tokens vs. ∼3.7K for RAG), reducing generation quality. We further replicate key experiments using an open-source local LLM (Qwen3.5-35B-A3B), finding that RAG benefits are proportionally larger for weaker models (+39% vs. +16% relative F1 improvement), an important result for cost-sensitive educational deployments. These findings inform the design of AI tutoring systems that reference specific textbook pages.

Keywords

Retrieval-Augmented Generation, GraphRAG, Mathematics Education, Question Answering, AI Tutoring

1. INTRODUCTION

During self-paced study, students frequently need to locate specific textbook pages, for example to revisit a definition they have forgotten, review a proof they found confusing, or find the relevant section for a homework problem. Large language models (LLMs) have shown notable capabilities in domains such as mathematics [1, 27], yet they often lack alignment with specific course content and may produce hallucinated sources [15, 4]. For instance, prior work has shown that LLMs tend to overestimate question quality without precise course alignment [21] and struggle with complex mathematical problems in digital learning games [24]. Retrieval-Augmented Generation (RAG) addresses these limitations by combining information retrieval with LLM generation, grounding responses in domain-specific documents [10, 31]. Recent work has demonstrated RAG's effectiveness in building LLM-based tutors across various educational settings [12, 20, 9, 16, 33, 18], though challenges remain around groundedness, user trust, and retrieval transparency [6, 17].

GraphRAG [8] extends standard RAG by constructing knowledge graphs from the corpus, capturing entity relationships and enabling more structured retrieval [5]. While research has shown GraphRAG can enhance question-answering performance in some settings [13, 19], it may also introduce excessive or irrelevant content in fine-grained retrieval tasks [28, 32]. In educational contexts, it is critical that AI tutors not only provide correct answers but also precisely reference the pages students need [7, 29], enabling verification and deeper learning [6, 17].

We address the following research question: *"To what extent can an AI-based retrieval system identify the correct textbook page for a question derived from that specific page?"* We compare embedding-based RAG, a BM25 sparse-retrieval baseline, and GraphRAG on a custom dataset of 477 question-answer pairs from an undergraduate mathematics textbook [23], evaluating retrieval accuracy, generated answer quality (F1 score), and error patterns. Our contributions include: (1) a systematic comparison of seven retrieval approaches (five embedding models, BM25, and GraphRAG) for page-level educational QA; (2) an error analysis showing that most retrieval failures surface pedagogically relevant content from nearby pages; (3) evidence that BM25 remains a competitive baseline, outperforming several neural embedding models; and (4) practical recommendations for deploying RAG-based AI tutors.

2. METHODS

2.1 Dataset

Eason Chen, Chuangji Li, Shizhuo Li, Zimo Xiao, Jionghao Lin, and Ken Koedinger. Comparing RAG and GraphRAG for Page-Level Retrieval Question Answering on a Math Textbook. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 616–621. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039806

We used the undergraduate textbook *An Infinite Descent into Pure Mathematics* [23] as our corpus. The textbook contains 628 pages covering topics in pure mathematics. Each page was converted from PDF to LaTeX-based Markdown via GPT Vision OCR [26, 11]. We generated one question-answer pair per page (628 total) using `gpt-4o-mini`, prompting it to produce questions representative of what students might ask during self-study, for example requesting clarification of a definition, asking how to apply a theorem, or seeking the proof of a result. Two authors, who completed undergraduate coursework using this textbook, then manually reviewed all pairs, filtering out front/back matter (reducing to 528) and eliminating poorly formed or educationally irrelevant questions, yielding a final set of 477 curated pairs that reflect the types of page-specific queries students encounter when studying mathematics independently. Each pair is labeled with its source page number, enabling precise evaluation of page-level retrieval.

2.2 RAG Pipeline

Figure 1 illustrates our pipeline, consisting of three stages: (1) *Indexing*, where pages are embedded as vectors (RAG), indexed as term-frequency vectors (BM25), or encoded as relational entities (GraphRAG); (2) *Retrieval*, where the top-k most relevant pages or entities are identified; and (3) *Generation*, where the query, prompt, and retrieved content are fed to an LLM for answer generation.

2.3 Models and Baselines

We selected five embedding models from the December 2024 Massive Text Embedding Benchmark (MTEB) leaderboard [22, 14]: `voyage-3-large` and `nv-embed-v2` were chosen for their high rankings; `gte-large` is commonly used as a fine-tuning base; `text-embedding-3-large` was included for its widespread adoption; and `multilingual-e5` for its multilingual capabilities. We also include a **BM25** sparse-retrieval baseline using Okapi BM25 with default parameters, to contextualize neural embedding gains against a classical term-matching approach that requires no training or fine-tuning. For each model, we tested retrieval at $k \in \{1, 3, 5, 10\}$.

For GraphRAG [8], we adapted the framework to expose `document_ids` for each entity and text unit, enabling page-level traceability. Specifically, we modified various GraphRAG components to store and output fields such as `document_ids` and `entity_ids`, allowing our system to link each retrieved item back to a corresponding page number. We tested GraphRAG with both `gpt-4o-mini` and `o3-mini` as the underlying LLM. Note that since GraphRAG retrieves entire entities and community summaries rather than ranked pages, we cannot directly control the number of pages returned; accuracy is evaluated based on whether the correct page's entities appear in the retrieved data.

2.4 Evaluation Metrics

We evaluate with two metrics: (1) **Retrieval Accuracy**, whether the correct source page appears in the retrieved set, averaged across all 477 queries; and (2) **F1 Score**, word-overlap F1 between the generated answer (produced by `gpt-4o-mini` given retrieved context) and the ground-truth answer. The F1 score captures the balance between precision (fraction of generated words appearing in the reference) and recall (fraction of reference words appearing in the generated output). We report bootstrap 95% confidence intervals (10,000 resamples) for key comparisons to assess statistical significance. We also conducted an error analysis of retrieval failures, categorizing top-1 misses by page distance and chapter membership to understand whether failures are near-misses or far-misses. Additionally, we tested LLM-based re-ranking of retrieved pages using `gpt-4o-mini`, where the model reorders retrieved pages by estimated relevance to the query.

Table 1: Retrieval accuracy for embedding-based RAG models, BM25, and GraphRAG. GraphRAG retrieves entities rather than ranked pages, so only overall accuracy is reported.

Model	Top-1	Top-3	Top-5	Top-10
voyage-3-large	**.686**	**.910**	**.958**	**.994**
nvidia/nv-embed-v2	.585	.843	.912	.964
BM25	.579	.780	.855	.929
OpenAI emb-3-large	.549	.811	.893	.933
gte-large	.461	.711	.799	.881
multilingual-e5	.457	.702	.795	.870
GraphRAG (o3-mini)		.914		
GraphRAG (4o-mini)		.845		

3. RESULTS

3.1 Retrieval Accuracy

Table 1 presents retrieval accuracy results. Among all models, `voyage-3-large` achieves the highest accuracy across all top-k settings, reaching 99.4% at top-10. Bootstrap 95% CIs for top-1 accuracy confirm that `voyage-3-large` ([.644, .728]) significantly outperforms `nv-embed-v2` ([.541, .629]), as these intervals do not overlap at top-1 or top-3.

BM25 proves a strong sparse-retrieval baseline, achieving top-1 accuracy of .579, outperforming both `gte-large` (.461) and `multilingual-e5` (.457) at all k values despite using no learned representations. This highlights the importance of including classical baselines in retrieval evaluations. Despite strong MTEB leaderboard rankings, `multilingual-e5` underperformed on this domain-specific task, underscoring that benchmark performance does not always transfer to specialized mathematical domains.

GraphRAG achieves 84.5% with `gpt-4o-mini` and 91.4% with `o3-mini`. All five embedding models surpass GraphRAG's accuracy at top-5 or above. The gap between the two GraphRAG configurations suggests that the underlying LLM quality substantially affects graph-based retrieval performance.

3.2 Generated Answer Quality

Table 2 shows F1 scores for answer generation. The no-retrieval baseline (`gpt-4o-mini` alone) achieves 0.475. All RAG configurations improve upon this, with F1 scores ranging from 0.514 to 0.552, confirming that retrieval grounding consistently benefits domain-specific QA. GraphRAG achieves F1 of $\sim$0.524, comparable to but slightly below the best embedding-based results.

Interestingly, `nv-embed-v2` achieves the highest top-1 F1

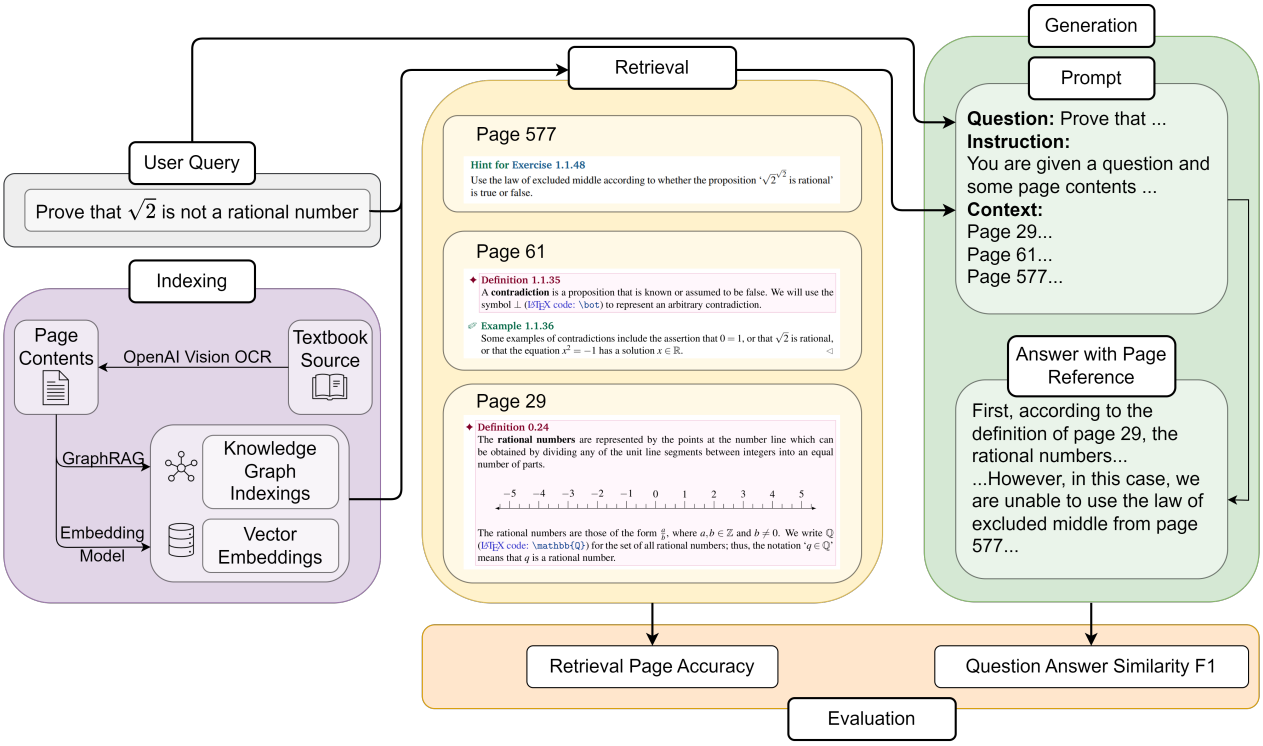


Figure 1: Our RAG pipeline: Indexing, Retrieval, and Generation stages with two evaluation metrics.

Table 2: F1 scores for generated answers. Baseline is `gpt-4o-mini` without retrieval (F1 = 0.475).

Model	Top-1	Top-3	Top-5	Top-10
voyage-3-large	.523	.543	.544	.547
nvidia/nv-embed-v2	**.537**	.542	.541	.539
OpenAI emb-3-large	.531	.549	.533	.543
gte-large	.526	.533	.550	**.552**
multilingual-e5	.514	.534	.541	.535
GraphRAG (o3-mini)		.524		
GraphRAG (4o-mini)		.525		
No retrieval		.475		

(.537) despite lower retrieval accuracy than `voyage-3-large`, suggesting retrieval accuracy and generation quality do not always correlate, because near-miss retrievals still provide useful context. Increasing k generally does not improve F1, as additional pages introduce noise. GraphRAG's ∼47K tokens of context per query (vs. ∼3.7K for top-5 RAG) overwhelms the generator, suggesting that *retrieval precision* matters more than *recall* for generation quality.

3.3 Re-Ranking

LLM-based re-ranking with `gpt-4o-mini` yielded mixed results. Re-ranking top-5 pages degraded `voyage-3-large`'s top-1 accuracy from .686 to .593, while re-ranking top-10 caused further degradation across all models. Critically, we observed hallucinated page numbers in re-ranking outputs: the LLM occasionally referenced pages not in the retrieved set. These findings suggest that LLM-based re-ranking is unreliable for page-level retrieval, particularly with larger candidate sets.

3.4 Error Analysis

We examined the 150 top-1 failures from `voyage-3-large`, categorizing each by page distance and chapter membership. Of these, 46.0% retrieved an adjacent page (distance ≤ 2), 17.3% retrieved a same-chapter but non-adjacent page, and 35.3% were far misses (different chapter). Overall, 63.3% of failures retrieved same-chapter content, with a median page distance of just 3 pages. This suggests that even "incorrect" retrievals often surface pedagogically relevant material. For example, a student asking about a proof may be directed to the adjacent page containing the theorem statement, content that is still valuable for building understanding and that an instructor might recommend reviewing first. The high proportion of near-misses reflects the textbook's pedagogical structure, where definitions, theorems, and proofs are deliberately organized across consecutive pages to support incremental learning. The 35.3% far misses, where shared terminology across chapters causes cross-topic confusion, represent the genuinely problematic failures that could misdirect students.

3.5 Open-Source LLM Replication

To assess whether our findings generalize beyond commercial APIs, we replicated key experiments using Qwen3.5-35B-A3B [30], an open-source 36B Mixture-of-Experts model running locally on 2× NVIDIA TITAN RTX GPUs with Q4_K_M quantization via Ollama. For retrieval, we tested `nomic-embed-text` [25], an open-source embedding model, also served locally.

Table 3: F1 scores: commercial API vs. open-source local LLM.

Configuration	gpt-4o-mini	Qwen3.5-35B
No retrieval	0.475	0.276
Best RAG + generation	0.552	0.384
Relative improvement	+16%	+39%

Table 3 compares generation quality (F1) between the commercial `gpt-4o-mini` and the local Qwen3.5 setup. Without retrieval, Qwen3.5 achieves substantially lower F1 (0.276 vs. 0.475). However, adding RAG with `nomic-embed-text` retrieval improves Qwen3.5's F1 to 0.384, a **39% relative improvement**, compared to only 16% for `gpt-4o-mini` with its best embedding model. This suggests that *weaker models benefit disproportionately from retrieval grounding*, as they rely more heavily on external context to compensate for limited parametric knowledge.

For retrieval accuracy, `nomic-embed-text` achieved top-1 accuracy of 45.1%, top-3 of 71.3%, and top-10 of 88.7%, below the best commercial embeddings but still serviceable. Notably, BM25 (61.2% top-1) again outperforms this neural embedding model, reinforcing BM25's strength as a zero-cost baseline.

We also attempted GraphRAG v3.0.6 indexing with Qwen3.5, successfully extracting 4,528 entities, 10,828 relationships, and 1,107 communities. However, query-time generation failed due to incompatibilities between GraphRAG's OpenAI-compatible API expectations and Qwen3.5's thinking-mode architecture. This demonstrates that while local LLM indexing for GraphRAG is feasible, end-to-end integration with open-source models remains challenging.

4. DISCUSSION AND CONCLUSION

Our findings yield several key insights for designing RAG-based educational AI systems.

RAG consistently improves over baseline LLMs. All RAG configurations produced higher F1 scores than the no-retrieval baseline (0.475 vs. 0.514 to 0.552), confirming that grounding LLM responses in textbook content improves answer quality for domain-specific question answering. This improvement is meaningful in educational settings where students benefit from answers grounded in their assigned course materials rather than the LLM's general pre-training data, which may contain incorrect or outdated mathematical content [3, 2].

Embedding-based RAG outperforms GraphRAG for page-level retrieval. GraphRAG retrieves excessive context (∼47K tokens vs. ∼3.7K for top-5 RAG), leading to lower precision and comparable or lower F1 [28, 32]. This reflects a *task-method mismatch*: GraphRAG was designed for global summarization and multi-hop reasoning, not page-level lookup. For educational applications where students need a specific page reference, the simpler embedding-based approach is more appropriate. However, GraphRAG may excel at cross-concept questions (e.g., "How do induction and well-ordering relate across chapters?") where its entity-based structure could surface connections that page-level retrieval would miss.

BM25 is a competitive baseline. BM25 outperforms two of five embedding models at all k values, partly due to high lexical overlap (72% unigram) between questions and source pages. This underscores the importance of including classical baselines, as neural embeddings do not always yield proportional gains, particularly when questions and documents share substantial vocabulary.

Most retrieval failures are near-misses with pedagogical value. Our error analysis reveals that 63.3% of top-1 failures retrieve same-chapter content (median distance: 3 pages). From an educational perspective, this means a student directed to the "wrong" page is still reading closely related material in most cases, analogous to how a tutor might say "look at the previous page first" before addressing the specific question. The top-1 accuracy of 68.6% may appear insufficient, but the near-miss pattern means the system rarely sends students to entirely irrelevant content.

Re-ranking introduces more risk than benefit. LLM-based re-ranking degraded performance for top-performing retrievers and introduced hallucinated page references, suggesting general-purpose LLMs struggle to distinguish between mathematically similar pages. Deployments requiring re-ranking should explore fine-tuned cross-encoder models or constrained decoding.

Open-source LLMs are viable but benefit more from RAG. RAG provides proportionally larger gains for weaker models (+39% vs. +16% relative F1 improvement), suggesting that retrieval grounding partially compensates for reduced model capability. This is significant for educational settings where commercial API costs are prohibitive, and aligns with EDM 2026's "Across Borders" theme, because institutions can deploy useful textbook-grounded QA systems on local hardware rather than relying on commercial APIs. Our results show that a locally hosted open-source model combined with BM25 retrieval (requiring no GPU for the retrieval component) can deliver meaningful QA quality at near-zero marginal cost.

Practical recommendations. (1) Use top-5 RAG retrieval as default (accuracy .958, ∼3.7K tokens); (2) present top-3 candidate pages to students (accuracy >91%); (3) avoid LLM re-ranking due to hallucination risk; (4) consider adjacent-page windowing to capture theorem-proof pairs spanning page boundaries.

Limitations and Future Work. While questions were curated by course-familiar authors to reflect realistic student queries, their high lexical overlap with source pages (72% unigram) may inflate retrieval accuracy relative to more diverse student phrasings; future work should supplement with authentic questions from discussion forums or office hours. Word-overlap F1 is a weak proxy for math QA, as mathematically equivalent expressions can score poorly; LLM-as-judge or symbolic equivalence checking would better capture correctness. Extending to multiple STEM textbooks, investigating hybrid BM25+dense retrieval, and conducting classroom studies measuring whether page references improve comprehension and student trust are important next steps. Notably, `multilingual-e5`'s underperformance despite strong general benchmarks raises questions about deploying these systems for non-English textbooks, an important consideration for

educational equity across linguistic borders.

5. REFERENCES

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[2] E. Chen, S. Judicke, K. Beigh, X. Tang, I. Wang, N. Yuan, Z. Xiao, C. Li, S. Li, R. Luttmer, S. Singh, and K. R. Koedinger. Chat-based support alone may not be enough: Comparing conversational and embedded llm feedback for mathematical proof learning. *arXiv preprint arXiv:2602.18807*, 2026.

[3] E. Chen, S. Judicke, K. Beigh, X. Tang, Z. Xiao, C. Li, S. Li, R. Luttmer, S. Singh, and K. R. Koedinger. Generative ai alone may not be enough: Evaluating ai support for learning mathematical proof. *arXiv preprint arXiv:2509.16778*, 2025.

[4] E. Chen, D. Wang, L. Xu, C. Cao, X. Fang, and J. Lin. A systematic review on prompt engineering in large language models for k-12 stem education. *arXiv preprint arXiv:2410.11123*, 2024.

[5] X. Chen, S. Jia, and Y. Xiang. A review: Knowledge reasoning over knowledge graph. *Expert systems with applications*, 141:112948, 2020.

[6] S. Chiesurin, D. Dimakopoulos, M. A. Sobrevilla Cabezudo, A. Eshghi, I. Papaioannou, V. Rieser, and I. Konstas. The dangers of trusting stochastic parrots: Faithfulness and trust in open-domain conversational question answering. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 947–959, Toronto, Canada, July 2023. Association for Computational Linguistics.

[7] W. Dai, Y.-S. Tsai, J. Lin, A. Aldino, H. Jin, T. Li, D. Gašević, and G. Chen. Assessing the proficiency of large language models in automatic feedback generation: An evaluation study. *Computers and Education: Artificial Intelligence*, 7:100299, 2024.

[8] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, and J. Larson. From local to global: A graph rag approach to query-focused summarization, 2024.

[9] T. Feng, S. Liu, and D. Ghosal. Courseassist: Pedagogically appropriate ai tutor for computer science education, 2024.

[10] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang. Retrieval-augmented generation for large language models: A survey, 2024.

[11] A. Ghiriti, W. Göderle, and R. Kern. Exploring the capabilities of gpt4-vision as ocr engine. In *International Conference on Theory and Practice of Digital Libraries*, pages 3–12. Springer, 2024.

[12] Z. F. Han, J. Lin, A. Gurung, D. R. Thomas, E. Chen, C. Borchers, S. Gupta, and K. R. Koedinger. Improving assessment of tutoring practices using retrieval-augmented generation. *arXiv preprint arXiv:2402.14594*, 2024.

[13] X. He, Y. Tian, Y. Sun, N. Chawla, T. Laurent, Y. LeCun, X. Bresson, and B. Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37:132876–132907, 2024.

[14] Hugging Face. MTEB leaderboard. `https://huggingface.co/spaces/mteb/leaderboard`, 2025. [Accessed 20-12-2024].

[15] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

[16] G. Lang and T. Gürpınar. AI-Powered Learning Support: A study of retrieval-augmented generation (RAG) chatbot effectiveness in an online course. *Information Systems Education Journal*, 23(2):4–13, 2025.

[17] Z. Levonian, C. Li, W. Zhu, A. Gade, O. Henkel, M.-E. Postle, and W. Xing. Retrieval-augmented generation to improve math question-answering: Trade-offs between groundedness and human preference, 2023.

[18] H. Li, T. Xu, C. Zhang, E. Chen, J. Liang, X. Fan, H. Li, J. Tang, and Q. Wen. Bringing generative ai to adaptive learning in education. *arXiv preprint arXiv:2402.14601*, 2024.

[19] J. Lin, S. Mai, B. Bu, M. He, and X. Wang. Research on the application of stem practical teaching based on rag knowledge graph and large models. In *7th International Conference on Educational Technology Management*, pages 520–527, 2024.

[20] C. Mitra, M. Miroyan, R. Jain, V. Kumud, G. Ranade, and N. Norouzi. Retllm-e: Retrieval-prompt strategy for question-answering on student discussion forums. In *Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24)*, 2024.

[21] S. Moore, H. A. Nguyen, N. Bier, T. Domadia, and J. Stamper. Assessing the quality of student-generated short answer questions using gpt-3. In *European conference on technology enhanced learning*, pages 243–257. Springer, 2022.

[22] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

[23] C. Newstead. *An Infinite Descent into Pure Mathematics*. 2024. Last updated on Wednesday 29th May 2024.

[24] H. A. Nguyen, H. Stec, X. Hou, S. Di, and B. M. McLaren. Evaluating chatgpt's decimal skills and feedback generation in a digital learning game. In *European conference on technology enhanced learning*, pages 278–293. Springer, 2023.

[25] Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*, 2024.

[26] OpenAI. Openai vision model documentation. `https://platform.openai.com/docs/guides/vision`, 2024. [Accessed 20-02-2025].

[27] e. a. OpenAI. Gpt-4 technical report, 2024.

[28] B. Peng, Y. Zhu, Y. Liu, X. Bo, H. Shi, C. Hong, Y. Yan, and Y. Li. Graph retrieval-augmented generation: A survey. *ACM Transactions on Information Systems*, 2025.

[29] A. Scarlatos, D. Smith, S. Woodhead, and A. Lan. Improving the validity of automatically generated feedback via reinforcement learning. In *International Conference on Artificial Intelligence in Education*, pages 280–294. Springer, 2024.

[30] Q. Team. Qwen3.5 technical report, 2026. Accessed: 2026-03-28.

[31] S. Wang, T. Xu, H. Li, C. Zhang, J. Liang, J. Tang, P. S. Yu, and Q. Wen. Large language models for education: A survey and outlook, 2024.

[32] Q. Zhang, S. Chen, Y. Bei, Z. Yuan, H. Zhou, Z. Hong, J. Dong, H. Chen, Y. Chang, and X. Huang. A survey of graph retrieval-augmented generation for customized large language models. *arXiv preprint arXiv:2501.13958*, 2025.

[33] C. Q. Zhao, J. Cao, E. Chen, K. R. Koedinger, and J. Lin. Slideitright: Using ai to find relevant slides and provide feedback for open-ended questions. In *International Conference on Artificial Intelligence in Education*, pages 378–392, 2025.

Constrained Multimodal Neural Networks for Interpretable Adaptation Rule Extraction in Game-Based Learning

Ange Tato
Université Laval
Québec, Canada
ange-adrienne.nyamen-tato@fse.ulaval.ca

Roger Nkambou
Université du Québec à Montréal
Montréal, Canada
nkambou.roger@uqam.ca

ABSTRACT

Establishing adaptation rules is a key step in designing effective adaptive systems and is traditionally carried out by domain experts. This paper presents a novel approach based on a constrained multimodal neural network (NN) to extract actionable adaptation rules from learner data. Unlike post-hoc explainability methods, the architecture is intrinsically interpretable through non-negative weight constraints, enabling stable and globally consistent identification of adaptation levers. The extracted patterns are validated using a decision tree and operationalized into concrete rules regulating feedback, NPC reactions, and emotional scaffolding in a serious game. An empirical evaluation comparing adaptive and non-adaptive versions shows improved learning outcomes (with marginal statistical significance, $p = 0.058$) and significantly enhanced emotional states conducive to learning ($p < 0.001$).

Keywords

Adaptation Rules, Socio-moral Reasoning, Neural Networks, Decision Tree, Game-Based Learning

1. INTRODUCTION

Adaptive game-based learning systems personalize learning by adapting to learners' cognitive, behavioral, and affective states [1]. Central to these systems are *adaptation rules* that govern feedback and pedagogical interventions, yet they are typically expert-defined, limiting scalability and responsiveness to multimodal data [14].

Prior work has explored adaptive mechanisms using knowledge tracing, Bayesian models, and deep learning [11, 6], but often focuses on task selection while overlooking affective and interactional dimensions critical for engagement in serious games [15]. In open-ended environments, adaptation must also regulate feedback, emotional engagement, and character interactions [20].

Ange Tato, and Roger Nkambou. Constrained Multimodal Neural Networks for Interpretable Adaptation Rule Extraction in Game-Based Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 622–626. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039675

Neural networks (NNs) can model complex multimodal data but remain difficult to translate into actionable pedagogical rules [8, 4]. Existing rule extraction methods rely on post-hoc explanations or expert-driven design. In contrast, we propose a constrained multimodal NN that embeds interpretability through non-negative weights, enabling stable and globally consistent rule extraction.

This paper makes three contributions: (1) a constrained multimodal NN architecture with intrinsic interpretability; (2) a rule extraction pipeline combining weight analysis and decision tree validation; and (3) an empirical evaluation in a serious game context.

More broadly, we show that constraining neural architectures can transform them into interpretable rule generators, bridging data-driven modeling and actionable pedagogical decision-making.

2. LESDILEMMES: GAME CONTEXT

LesDilemmes [17, 16] is a serious game designed to assess and develop socio-moral reasoning in youths (ages 8–18). Players engage with dilemmas, justify decisions, and evaluate NPC responses spanning five reasoning levels from the So-Moral framework [3]: (1) Authoritarian, (2) Egocentric, (3) Interpersonal, (4) Societal, and (5) Evaluative. Intermediate levels (1.5–4.5) capture developmental transitions. These levels are modeled as developmental constructs independent of age, enabling joint analysis across participants.

The learner model integrates affective state, cognitive profile, and socio-moral reasoning. Game dynamics combine social feedback (e.g., "likes") with pedagogical rules governing NPC reactions and feedback.

Expert-defined rules regulate interactions. For instance, agreement with a higher-level NPC ($r_n > r_p$) triggers positive feedback, while learners below level 4 receive encouragement and scaffolding toward higher reasoning. These rules may co-occur, reflecting authentic pedagogical practice.

3. CONSTRAINED MULTIMODAL NEURAL NETWORK

3.1 Architecture

We use a multilayer perceptron (MLP) as a structured aggregation mechanism for multimodal learner data. All weights are constrained to be non-negative, ensuring intrinsic interpretability [5]. All inputs are normalized to $[0, 1]$.

Given a neuron with inputs x_1, x_2 and non-negative weights w_1, w_2:

$$y = f(w_1 \cdot x_1 + w_2 \cdot x_2 + b) \quad (1)$$

where f is ReLU. Under non-negativity, feature importance is directly reflected by weight magnitude ($w_1 > w_2 \Rightarrow x_1$ is more influential).

The model integrates three modalities (Fig. 1): (1) emotions (9 features), (2) NPC evaluations (5 binary features), and (3) gameplay context (reasoning level of the first NPC visited). The output layer uses Softmax to predict $[1, 0]$ (reasoning ≤ 3) or $[0, 1]$ (reasoning > 3). A dropout layer prevents overfitting given the limited dataset.

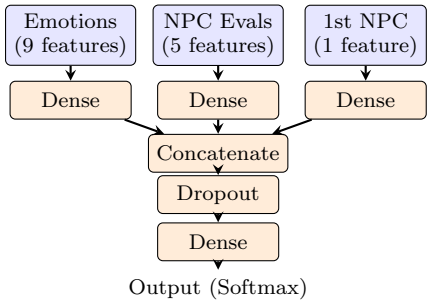


Figure 1: Constrained multimodal NN architecture for rule extraction.

3.2 Feature Importance Computation

To mitigate stochastic variability, 20 training runs are performed. Feature importance $A_{i,j}$ for input i predicting output j is computed using softmax normalization of penultimate-layer weights:

$$a_{i,j} = \frac{e^{w_{i,j}}}{\sum_{i=1}^{n} e^{w_{i,j}}} \quad (2)$$

$$A_{i,j} = \frac{\sum_k a_{i,j,k}}{\sum_k \sum_j a_{i,j,k}} \quad (3)$$

where k indexes training runs. Higher $A_{i,j}$ indicates greater influence on the prediction of output j.

This design transforms the neural network from a predictive black box into a structured aggregation mechanism whose weights directly encode interpretable relationships between learner features and outcomes. As a result, the model supports global rule extraction rather than local post-hoc explanations.

4. RULE EXTRACTION AND VALIDATION

4.1 Neural Network Analysis

Table 1 reports the penultimate-layer weights and averaged importance $A_{i,j}$ across 20 runs. *Emotions* consistently show the highest importance ($A_{i,1} = 0.475$, $A_{i,2} = 0.689$), outweighing NPC evaluations and first NPC visited.

The predictive performance of the model reached an average classification accuracy between 65% and 70% depending on the training run, which is consistent with the exploratory nature of the study and the limited dataset size. Importantly, the NN is not used as a final predictive model but as a mechanism for identifying stable explanatory patterns across modalities—which is why the final adaptation rules combine NN importance rankings with decision tree thresholds. This hybrid strategy compensates for the NN's inability to directly specify threshold values while leveraging its capacity to model nonlinear multimodal interactions.

Table 1: CMNN penultimate layer weights from 3 runs. $A_{i,j}$ averaged over 20 runs.

	Run 1		Run 2		Run 3		$A_{i,j}$	
Modality	[1,0]	[0,1]	[1,0]	[0,1]	[1,0]	[0,1]	[1,0]	[0,1]
Emotions	.007	.812	.000	.751	.925	.739	.475	.689
Eval NPCs	.066	.000	.323	.040	.043	.378	.367	.276
1st NPC visited	.067	.607	.742	.099	.116	.071	.158	.035

Table 2 details importance per emotion. **Arousal** is the most influential ($A_i = 0.234$), followed by sadness (0.165), surprise (0.155), and happiness (0.120). Table 3 shows that NPCs at levels 2 and 5 are most predictive ($A_i = 0.215$ and 0.273 respectively).

Table 2: Emotion branch weights and importance A_i (20 runs).

Emotion	Run 1	Run 2	Run 3	A_i
Neutral	.080	.093	.000	.001
Happy	.000	.475	.000	.120
Sad	.672	.013	.333	.165
Angry	.069	.702	.000	.090
Surprised	.000	.034	.583	.155
Scared	.138	.000	.041	.001
Disgusted	.000	.016	.001	.106
Valence	.346	.006	.045	.128
Arousal	.039	.404	.578	**.234**

Table 3: NPC evaluation weights and importance A_i (20 runs).

NPC Level	Run 1	Run 2	Run 3	A_i
Eval 1	.939	.186	.000	.198
Eval 2	.078	.755	.000	**.215**
Eval 3	.052	.000	.938	.167
Eval 4	.028	.588	.000	.147
Eval 5	.920	.369	.000	**.273**

4.2 Decision Tree Validation

A decision tree (Fig. 2) trained on the same dataset validates the NN findings and provides threshold values for operationalization. Key extracted rules:

- Low *arousal* $\Rightarrow$ higher probability of reasoning level > 3
- Agreement with NPC level 2 + low disgust $\Rightarrow$ reasoning level > 3
- No agreement with NPC level 2 + low surprise $\Rightarrow$ reasoning level < 3
- No agreement with NPC level 2 + agreement with NPC level 3 $\Rightarrow$ reasoning level > 3

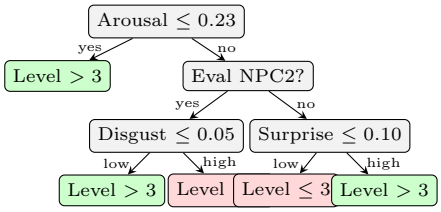


Figure 2: Simplified decision tree for socio-moral reasoning prediction.

These findings motivate concrete game adaptations: background music is changed according to arousal and valence, and players are required to visit all NPCs (including levels 2 and 5) rather than stopping after three.
While the decision tree is trained on the same dataset, it provides an independent, symbolic approximation of the learned relationships, enabling the identification of explicit thresholds and facilitating operational deployment. One advantage of using the NN is that, as the data grows, the model will still be able to highlight features that need to be observed and modified to improve the mastery of the knowledge, compared to the decision tree alone.

5. EMPIRICAL EVALUATION

5.1 Dataset and Protocol

29 youths (ages 8–18) played the non-adaptive version of *LesDilemmes* across 9 dilemmas, yielding 261 observations. Verbal justifications were transcribed via Google Speech-to-Text and analyzed with a data mining algorithm. Emotional data were collected via *Facereader* (7 basic emotions + valence + arousal). An adaptive version incorporating the extracted rules was subsequently developed and evaluated against the baseline.

5.2 Learning Outcomes

Table 4 reports paired t-test results. Players showed significant improvement from pre-test (mean 1.70) to in-game performance (2.30, $p = 0.029$) and post-test (2.49, $p < 0.001$). The non-significant difference between in-game and post-test ($p = 0.445$) indicates that gains persist after gameplay.

Table 4: Paired-samples t-test results for learning in *LesDilemmes*.

Comparison	Mean diff.	SD	t	N	p
Pre → Post	−0.789	0.706	−4.33	14	< .001
Game → Post	−0.233	1.188	−0.78	15	.445
Pre → Game	−0.491	1.056	−2.32	24	.029

Comparing adaptive (A) vs. non-adaptive (NA) versions (Table 5), the pre–post gain was 0.789 for A vs. 0.333 for NA ($p = 0.058$). Note that A participants had a lower pre-test baseline (1.70 vs. 2.18), which is partly explained by the NA group being older on average.

5.3 Emotional Impact of Adaptation

Fig. 3 compares emotional profiles between conditions. The adaptive version yields significantly higher valence ($p < 0.001$), confirming that adaptation rules oriented toward positive affect were effective despite the reading-heavy nature of the

Table 5: Adaptive (A) vs. non-adaptive (NA) comparison.

Measure	Version	N	Mean	SD
Pre→Post gain	A	15	0.789	0.706
	NA	29	0.333	0.749
Pre-test	A	15	1.700	0.575
	NA	29	2.178	0.675
Post-test	A	16	2.536	0.684
	NA	29	2.511	0.648

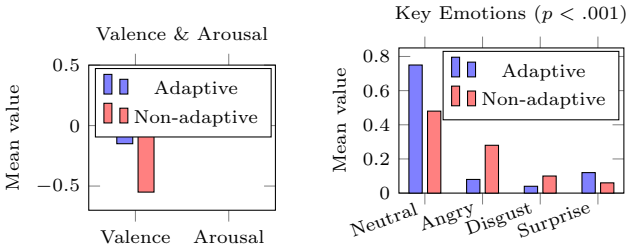


Figure 3: Emotional comparison between adaptive (A) and non-adaptive (NA) versions ($p < 0.001$). The adaptive version produces significantly higher valence, lower anger and disgust, and higher surprise.

game. The emotions neutral and anger are typically more prevalent during reading activities and tend to manifest with greater intensity when measured via Facereader [18]. In *LesDilemmes*, the predominant activity is reading—especially true in the adaptive version with additional learning and encouragement messages—and the main character has no direct control over the environment, only making decisions and evaluating others through button clicks. Therefore, negative emotions being more frequent than positive ones is entirely expected. Anger and disgust were significantly reduced in the adaptive condition, while neutrality and surprise increased. Surprise is particularly relevant: research shows surprising stimuli are remembered faster and recalled more accurately [7], and it promotes deeper cognitive engagement [2].

6. ADAPTATION RULES

A central contribution of this work is the translation of NN-derived patterns and decision tree thresholds into concrete, actionable rules deployed in *LesDilemmes*. Table 6 summarizes the five main rules implemented in the adaptive version, along with their source (NN or DT), the targeted learner state, and the game action triggered.

These rules complement expert-defined rules by targeting emotional regulation as a primary lever—an aspect absent from the original design. The NN-derived rules are globally stable (extracted from 20 runs), making them robust to stochastic training variability and suitable for deployment without per-learner retraining.

7. DISCUSSION

A key insight of this work is that architectural constraints can play a central role in making neural models usable in educational settings. Rather than relying on post-hoc interpretability, constraining the hypothesis space enables the

Table 6: Operationalized adaptation rules in *LesDilemmes*.

Condition	Source	Game Action	Goal
Valence < 0	NN (arousal)	Switch to cheerful music	Mood lift
Valence ≥ 0	NN (valence)	Play soft background music	Maintain state
Agrees with NPC $r_n > r_p$	Expert rule	NPC shows happy face + thumbs-up	Reinforce
$r_p < 4$	Expert rule	Display encouraging + scaffolding message	Scaffold
Player skips NPCs 2 or 5	NN (Eval 2/5)	Force exploration of remaining NPCs	Ensure exposure

extraction of stable and pedagogically meaningful rules that can be directly operationalized.

7.1 Interpretability by Design

A key methodological contribution of this work is the distinction between *intrinsic* and *post-hoc* interpretability. Techniques such as LIME [12] and SHAP [10] explain individual predictions after training, providing local approximations that may vary across instances. In contrast, non-negative weight constraints make the trained model itself interpretable at a global level: because all weights $w_i \geq 0$ and inputs $x_i \in [0, 1]$, the relative magnitude of weights directly encodes feature importance without auxiliary approximation.

This property is especially valuable in educational contexts, where adaptation rules must be stable, auditable, and pedagogically justifiable. A rule derived from a globally consistent model carries more credibility for instructional designers than a saliency map computed locally for a single learner episode.

7.2 Role of Emotions in Socio-Moral Learning

The prominence of affective features—particularly arousal, surprise, and happiness—in predicting reasoning level has important theoretical implications. It aligns with the cognitive-affective learning model [19], which posits that emotional states modulate attention, working memory, and long-term retention. High arousal without positive valence can hinder deliberate reasoning, while moderate arousal paired with positive affect creates an optimal state for engagement and higher-order thinking. Positive emotions also increase learner engagement, facilitate learning, and contribute to better long-term retention [13].

The observed increase in surprise in the adaptive condition is consistent with theories of *productive confusion* [7, 2]: unexpected feedback or NPC responses prompt learners to re-evaluate their assumptions, fostering deeper moral reflection. Designing NPCs to occasionally challenge learners with unexpected high-level reasoning—rather than always providing confirmatory feedback—may therefore be a principled lever for supporting reasoning progression.

7.3 Comparison with Related Approaches

Unlike deep knowledge tracing [11], which models latent skill states through LSTM networks but provides little interpretability, our constrained MLP explicitly surfaces which input dimensions drive predictions. Compared to Bayesian student models [6], which require strong prior assumptions, our approach learns directly from multimodal data without structural priors, while still producing transparent outputs. The hybrid NN–decision tree pipeline combines the representational power of neural learning with the symbolic clarity of tree-based rules—a complementary strategy supported by recent work on rule extraction from neural networks [8]. Unlike narrative personalization [20] and goal recognition [9] approaches that require manually authored rules, our rules emerge directly from data via the constrained architecture, making the approach scalable to new game contexts.

7.4 Limitations and Future Directions

Several limitations constrain the generalizability of the current results. First, the sample size ($N = 29$, yielding 261 observations) is small. Results should therefore be interpreted cautiously given the limited sample size and group differences, and viewed as preliminary evidence of the effectiveness of the extracted adaptation rules. Furthermore, the marginal significance of the adaptive gain ($p = 0.058$) warrants cautious interpretation. A pre-registered, adequately powered study is needed to confirm the learning benefit of adaptation. Second, the reasoning level labels depend on automatic speech transcription (Google Speech-to-Text) followed by algorithmic coding, introducing noise that could affect both model training and outcome measurement. Third, the age heterogeneity of participants (8–18 years) introduces developmental variance that may interact with adaptation mechanisms in ways not yet captured by the model.

Future work will address these limitations through: (1) larger and more homogeneous samples stratified by age group; (2) improved multimodal fusion incorporating gaze and physiological signals alongside facial expressions; (3) online model updating to enable within-session personalization rather than rule extraction from a fixed prior dataset; and (4) extension of the framework to other serious game contexts beyond socio-moral reasoning, such as scientific argumentation and collaborative problem-solving.

7.5 Broader Implications for Adaptive ITS

A key insight is that architectural constraints shift the neural model from a predictive black box to a knowledge elicitation mechanism, yielding stable, operationalizable rules without post-hoc interpretability. Three implications follow for adaptive ITS. First, interpretability and predictive power need not be traded off: 65–70% accuracy is maintained alongside globally consistent importance rankings. Second, the modular pipeline—NN for *which* features matter, decision tree for *when* and *at what thresholds*—allows independent updates as data or targets change. Third, globally extracted rules (20 averaged runs) are robust to stochastic variability, enabling real-time deployment without per-session retraining. The approach generalizes to any multimodal dataset with a learner outcome variable: architecture

choices adapt to available sensors while the non-negativity constraint preserves interpretability by construction.

8. CONCLUSION

This paper introduced a data-driven pipeline for extracting actionable adaptation rules from a constrained multimodal neural network in a game-based learning context. By enforcing non-negative weight constraints, the architecture transforms the NN into a globally interpretable aggregation mechanism whose weights directly encode feature importance across emotional, interactional, and gameplay modalities—without relying on post-hoc approximations.

Applied to *LesDilemmes*, the approach identifies arousal, surprise, and happiness as dominant predictors of higher socio-moral reasoning, and interactions with NPCs at levels 2 and 5 as the most informative social levers. These findings, validated by a decision tree analysis, were operationalized into five concrete adaptation rules governing background music, NPC feedback, and player navigation.

Empirical evaluation demonstrates that the resulting adaptive version shows larger pre–post learning gains with marginal statistical significance ($p = 0.058$; adaptive gain 0.789 vs. 0.333 for non-adaptive), substantially improves emotional climate ($p < 0.001$), and generates higher levels of surprise—an emotion associated with deeper cognitive engagement and better retention.

Beyond the specific game context, this work demonstrates that constraining neural architectures can transform them into interpretable rule extraction mechanisms, bridging the gap between data-driven learning and pedagogical design. This positions constrained multimodal neural networks as a practical and principled foundation for adaptive intelligent tutoring systems.

9. ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support provided by the Natural Sciences and Engineering Research Council of Canada (NSERC).

10. REFERENCES

[1] S. Adipat, K. Laksana, K. Busayanon, A. Asawasowan, and B. Adipat. Engaging students in the learning process with game-based learning: The fundamental concepts. *International Journal of Technology in Education*, 4(3):542–552, 2021.

[2] J. E. Adler. Surprise. *Educational Theory*, 58(2):149–173, 2008.

[3] M. H. Beauchamp, J. J. Dooley, and V. Anderson. A preliminary investigation of moral reasoning and empathy after traumatic brain injury in adolescents. *Brain injury*, 27(7-8):896–902, 2013.

[4] G. Bologna and Y. Hayashi. A comparison study on rule extraction from neural network ensembles, boosted shallow trees, and svms. *Applied computational intelligence and soft computing*, 2018(1):4084850, 2018.

[5] J. Chorowski and J. M. Zurada. Learning understandable neural networks with nonnegative weight constraints. *IEEE transactions on neural networks and learning systems*, 26(1):62–69, 2014.

[6] C. Conati. Bayesian student modeling. In *Advances in intelligent tutoring systems*, pages 281–299. Springer, 2010.

[7] M. I. Foster and M. T. Keane. The role of surprise in learning: Different surprising outcomes affect memorability differentially. *Topics in cognitive science*, 11(1):75–87, 2019.

[8] C. He, M. Ma, and P. Wang. Extract interpretability-accuracy balanced rules from artificial neural networks: A review. *Neurocomputing*, 387:346–358, 2020.

[9] K. R. Koedinger, E. Brunskill, R. S. Baker, E. A. McLaughlin, and J. Stamper. New potentials for data-driven intelligent tutoring system development and optimization. *Ai Magazine*, 34(3):27–41, 2013.

[10] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[11] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[12] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[13] L. Shen, M. Wang, and R. Shen. Affective e-learning: using" emotional" data to improve learning in pervasive learning environment. *J. Educ. Technol. Soc.*, 12(2):176–189, 2009.

[14] V. Shute and B. Towle. Adaptive e-learning. In *Aptitude*, pages 105–114. Routledge, 2018.

[15] L. Sun, M. Kangas, and H. Ruokamo. Game-based features in intelligent game-based learning environments: A systematic literature review. *Interactive Learning Environments*, 32(7):3431–3447, 2024.

[16] A. Tato and R. Nkambou. Infusing expert knowledge into a deep neural network using attention mechanism for personalized learning environments. *Frontiers in Artificial Intelligence*, 5:921476, 2022.

[17] A. Tato, R. Nkambou, A. Dufresne, and M. H. Beauchamp. Convolutional neural network for automatic detection of sociomoral reasoning level. *International Educational Data Mining Society*, 2017.

[18] V. Terzis, C. N. Moridis, and A. A. Economides. Measuring instant emotions during a self-assessment test: the use of facereader. In *Proceedings of the 7th international conference on methods and techniques in behavioral research*, pages 1–4, 2010.

[19] C. M. Tyng, H. U. Amin, M. N. Saad, and A. S. Malik. The influences of emotion on learning and memory. *Frontiers in psychology*, 8:235933, 2017.

[20] P. Wang, J. P. Rowe, W. Min, B. W. Mott, and J. C. Lester. Interactive narrative personalization with deep reinforcement learning. In *IJCAI*, pages 3852–3858, 2017.

Coordination Regimes in Student–GenAI Interaction: The Integration Trap and Toward Enactment-Aware Scaffolding*

Fangwei Lin
Faculty of Education
The University of Hong Kong
fangwei.lin@connect.hku.hk

Mengchen Zhang
Faculty of Education
The University of Hong Kong
mengchenzhang@connect.hku.hk

Zhihan Guo
Faculty of Education
The University of Hong Kong
zhihang330@connect.hku.hk

Long Zhang
School of Computer Science and Engineering
South China University of Technology
longzhang@scut.edu.cn

Khe Foon Hew
Faculty of Education
The University of Hong Kong
kfhew@hku.hk

ABSTRACT

How does student–GenAI interaction quality vary with academic achievement? We introduce *coordination regimes*, a descriptive lens informed by the control-theoretic model of self-regulated learning, to characterize how students align cognitive goals, prompting actions, and inquiry phases in dialogue with GenAI. We coded 2,243 dialogue turns from 136 undergraduates along three dimensions (Learning Intention, Prompt Communicative Function, and Cognitive Presence) and analyzed them using Epistemic Network Analysis (ENA), Sequential Pattern Mining (SPM), and ANCOVA controlling for prior GPA. High achievers showed tighter co-occurrence structures linking reasoning, prompting, and resolution-oriented inquiry, while lower-achieving students more often exhibited what we term the *integration trap*: integration without enactment, in which students consolidate understanding without progressing to resolution. Sequential analyses further revealed "delegate-then-verify" patterns among high achievers versus "delegate-then-retreat" patterns among low achievers. Artifact Submission (students returning with execution results for AI diagnosis) showed the largest group difference ($\chi^2 = 31.83$, $p < .001$) and emerged as a promising scaffolding target. After controlling for prior GPA, learning intention differences were attenuated, whereas cognitive presence differences remained significant ($p = .011$), suggesting that the enactment layer of student–GenAI interaction may be partially distinct from prior academic ability and potentially scaffoldable.

Keywords

Generative AI, Student-AI Interaction, Epistemic Network Analysis, Sequential Pattern Mining, Learning Analytics

1. INTRODUCTION

Generative AI (GenAI) tools offer new possibilities for personalized learning support [8, 10]. As these tools become embedded in coursework, a central question is not simply whether students use GenAI, but how interaction quality differs across learners. Prior work has examined student–AI interaction through isolated lenses: prompt types and usage frequencies [11, 4, 14], co-occurrence structures [9], or learner profiles [7]. These approaches yield useful insights but typically treat cognitive goals, communicative actions, and inquiry processes as separate targets.

This separation limits what we can learn about interaction quality. Effective student–GenAI interaction is not reducible to asking higher-level questions, using particular prompt forms, or reaching later inquiry phases in isolation. Rather, it depends on how these elements work together as a coordinated system during dialogue. To capture this, we introduce *coordination regimes*, a descriptive analytic lens informed by the control-theoretic model of self-regulated learning [3, 15, 17]. In this model, goal-directed learning involves a *reference value* (the goal), an *output function* (the action taken), and an *input function* (monitoring progress) [3]. We operationalize these functions as Learning Intention, Prompt Communicative Function (PCF), and Cognitive Presence, respectively (Figure 1). We call the resulting group-level pattern of code co-occurrences and sequential transitions across these three dimensions a *coordination regime*. The an-

*

Fangwei Lin, Mengchen Zhang, Zhihan Guo, Long Zhang, and Khe Foon Hew. Coordination Regimes in Student–GenAI Interaction: The Integration Trap and Toward Enactment-Aware Scaffolding. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 627–631. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040099

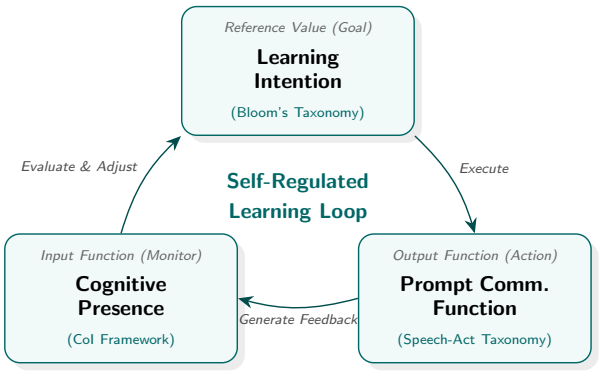


Figure 1: Operationalization of the control-theoretic SRL feedback loop. Each regulatory function is mapped to a conversational variable.

alytic value lies not in proposing new categories for Bloom's taxonomy or the Cognitive Presence framework, but in examining how goal, action, and monitoring indicators align or misalign within student–GenAI interaction.

We make three contributions. First, we offer *coordination regimes* as a conceptual lens for analyzing student–GenAI interaction as a coupled regulatory system rather than a set of independent variables. Second, we identify the *integration trap*, a recurring pattern in which lower-achieving students consolidate understanding without enacting or testing solutions. Third, we show that *Artifact Submission* (students returning with execution results for AI diagnosis) is a promising candidate target for enactment-aware scaffolding, supported by evidence that cognitive presence differences are not fully explained by prior academic ability.

We address three research questions:

RQ1. How do coordination regimes differ across achievement groups in the co-occurrence structure of Learning Intention, Prompt Communicative Function, and Cognitive Presence?

RQ2. How do turn-level sequential interaction patterns differ across groups?

RQ3. Do observed interaction differences persist after controlling for prior academic ability?

2. METHOD

The dataset comprised 136 third-year undergraduate students majoring in Statistics who were enrolled in a data mining course and used a GenAI assistant (Doubao, powered by ByteDance; a general-purpose conversational AI with code generation capabilities, comparable to ChatGPT in functionality) during Weeks 6–9 of a 16-week semester. The sample included 81 female students (59.6%) and 55 male students (40.4%). Students were stratified into three achievement tiers by final grade tertiles: G1 (High, $n = 52$), G2 (Mid, $n = 46$), G3 (Low, $n = 38$); smaller group sizes in lower tiers reflect more students excluded for absent or invalid dialogue records. The study was approved by the university ethics committee, and students provided informed consent for the use of their anonymized learning data. Dialogue records and grade information were de-identified before analysis, and all results are reported in aggregate. In total, 481 student-week observations containing 2,243 dialogue turns were collected. Each student contributed an average of 3.49 weekly observations ($SD = 1.04$).

Each turn was coded along three dimensions (Table 1). *Learning Intention* captures the prompt-expressed cognitive goal level (Remember through Create) via Bloom's revised taxonomy [2]. We use this term operationally to refer to enacted cognitive goals, not latent psychological intentions. *Prompt Communicative Function* (PCF) classifies prompts by their primary communicative function, i.e., what the student is trying to *do*, using six categories derived from speech act theory [12] and refined through pilot coding. *Cognitive Presence* captures the phase of inquiry via the Community of Inquiry framework [5]: Triggering, Exploration, Integration, and Resolution.

Inter-rater reliability was established on a random 20% sample ($n = 449$ turns) coded independently by two experts. Cohen's κ: Learning Intention = 0.72, PCF = 0.83, Cognitive Presence = 0.68. Disagreements were resolved through discussion, and the remaining turns were coded by the first author.

We used three complementary analyses. ENA [13] modeled co-occurrence structures (units: students; conversations: student-weeks; stanza window: entire conversation within each student-week; codes: 16 binary codes across three dimensions), capturing *which* codes connect within sessions. SPM [1] complemented ENA by capturing how codes unfold turn by turn; bigram and trigram frequencies were extracted from turn-level code sequences, expressed as proportions of all bigrams within each group, and compared across groups using chi-square tests with Benjamini–Hochberg correction (all p-values reported for SPM are BH-corrected). Groups were compared on ENA network positions using pairwise t-tests with Bonferroni correction. To test whether observed differences persisted beyond baseline ability, we ran ANCOVA controlling for prior-semester GPA. Together, these analyses aligned with the three research questions: ENA addressed RQ1 by comparing group-level co-occurrence structures, SPM addressed RQ2 by comparing turn-level bigram and trigram patterns, and ANCOVA addressed RQ3 by testing whether group differences persisted after controlling for prior-semester GPA.

3. RESULTS

3.1 Co-occurrence Structure (RQ1)

At the learning-intention dimension, differences were present but modest at the structural level, with stronger contrasts emerging in sequential analyses reported later. ENA networks showed that high achievers (G1) formed stronger connections along the Understand–Apply–Evaluate chain, while G2 and G3 showed more evenly distributed connections among lower-level goals (Remember–Understand–Apply) without reaching Evaluate (Figure 3). Group differences on ENA's primary dimension (SVD1) were not statistically significant after Bonferroni correction (G1 vs. G2:

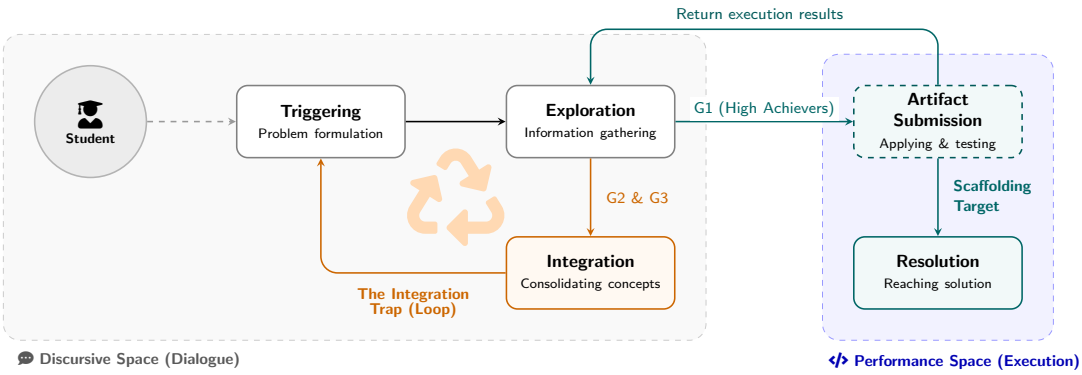


Figure 2: Conceptual model mapping the Integration Trap within the discursive space, and the pathway to resolution via Artifact Submission in the performance space. (G1, G2, and G3 refer to High, Mid, and Low achievement groups, respectively).

Table 1: Coding Framework: Three Dimensions of Student–GenAI Interaction

Dimension	Code	Definition	Source
Learning Intention	Remember	Recall or recognize factual information	Bloom's Revised Taxonomy [2]
	Understand	Explain or summarize concepts	
	Apply	Use knowledge in a concrete task	
	Analyze	Break down or compare components	
	Evaluate	Judge, test, or diagnose outputs	
	Create	Design or produce original artifacts	
Prompt Comm. Function (PCF)	Knowledge Query (KQ)	Ask for factual or conceptual information	Speech Act Theory [12]
	Task Delegation (TD)	Request AI to perform a task (e.g., write code)	
	Contextualized Query (CQ)	Query grounded in student's own context	
	Artifact Submission (AS)	Submit execution output for AI diagnosis	
	Elaboration (EF)	Follow up or refine a prior exchange	
	Output Shaping (OS)	Request format or style changes to output	
Cognitive Presence	Triggering	Recognize a problem or sense of puzzlement	Community of Inquiry [5]
	Exploration	Search for information or alternatives	
	Integration	Connect ideas into a coherent understanding	
	Resolution	Apply or test the solution in practice	

$d = 0.40$, $p = .050$; G1 vs. G3: $d = 0.40$, $p = .061$; G2 vs. G3: n.s.), consistent with the ANCOVA attenuation reported in RQ3.

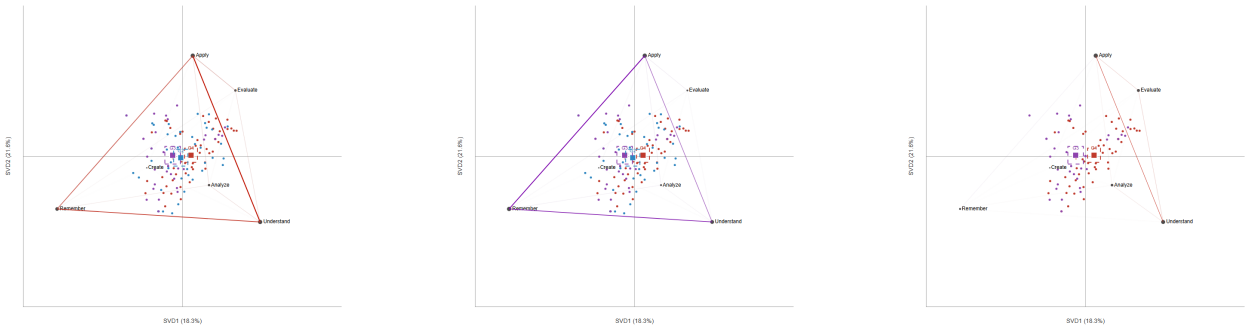
At the prompt-function dimension, five of the six PCF codes differed significantly across achievement groups (Table 2), but Artifact Submission (AS) stood out as the clearest behavioral marker. In AS turns, students submit their own execution results, such as error messages or code output, for AI diagnosis. AS showed the largest group difference of any code across all three dimensions (G1: 14.5%, G3: 5.1%, $\chi^2(2) = 31.83$, $p < .001$). This finding is reinforced by cross-dimensional patterns: 51.0% of AS turns co-occurred with Evaluate-level intentions and 6.9% reached the Resolution phase. By contrast, Knowledge Query, the most common PCF code overall, was dominated by Remember and Understand (77.9%), with Resolution near-absent (0.2%).

At the cognitive-presence dimension, the clearest structural separation across achievement groups appeared in inquiry-phase patterns, where lower-achieving students were displaced toward Integration rather than Resolution. ENA networks showed that high achievers (G1) leaned toward Resolution, whereas G2 and G3 clustered near Integration (Figure 4). These structural differences along the primary network dimension (SVD1) were statistically significant after Bonferroni correction (G1 vs. G2: $d = 0.64$, $p = .002$; G1 vs. G3: $d = 0.55$, $p = .011$; G2 vs. G3: n.s.). This provides evidence for what we term the *integration trap*: lower-achieving students sustained inquiry cycles but consolidated

Table 2: PCF Distribution by Achievement Group

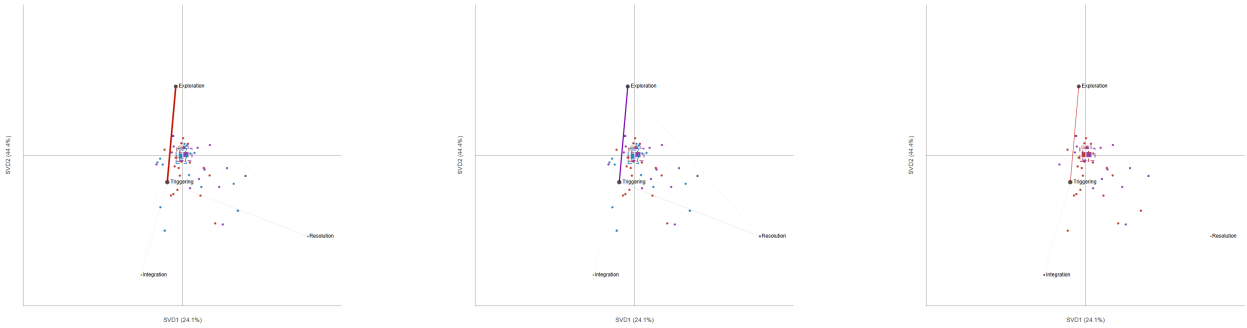
PCF Code	G1	G2	G3	$\chi^2(2)$	V
Knowledge Query	54.5%	65.9%	63.9%	26.78***	.109
Task Delegation	17.1%	17.7%	23.0%	7.88*	.059
Artifact Submission	14.5%	9.2%	5.1%	31.83***	.119
Contextualized Query	6.7%	3.0%	1.1%	28.40***	.113
Elaboration	6.0%	4.2%	5.5%	2.71	.035
Output Shaping	1.4%	0.0%	1.3%	9.14*	.064

Note. $^{*}p < .05$; $^{***}p < .001$. V = Cramér's V.



(a) G1 (High Achievers) (b) G3 (Lower Achievers) (c) Contrast (G1 − G3)

Figure 3: Learning Intention ENA. (a–b) Mean networks for G1 and G3; (c) contrast highlighting modest structural differences along the Understand–Apply and Apply–Evaluate edges.



(a) G1 (High Achievers) (b) G3 (Lower Achievers) (c) Contrast (G1 − G3)

Figure 4: Cognitive Presence ENA. (a–b) Mean networks for G1 and G3; (c) contrast showing G3's displacement toward Integration, illustrating the integration trap.

understanding rather than testing solutions in practice, a pattern of *integration without enactment*.

Despite these group differences, 75.1% of all coded turns remained in the Triggering phase across all groups, with Integration (3.2%) and Resolution (4.4%) rarely observed, consistent with prior findings that cognitive presence frequently stalls at lower phases [6].

3.2 Sequential Patterns (RQ2)

Sequential analyses revealed sharper contrasts, clarifying how coordination regimes unfold turn by turn.

For Learning Intention sequences, Understand→Apply appeared at 24.7% in G1 versus 8.6% in G3, and Apply→Evaluate at 20.1% versus 7.5%, indicating that high achievers progressed upward through Bloom's levels within their dialogues while lower achievers remained at foundational levels.

For prompt sequences, the contrasts were even sharper. TD→AS ("delegate-then-verify") occurred in G1 at four times the rate of G3 (3.7% vs. 0.9%, $p = .009$). Sustained debugging chains (AS→AS→AS) were present in G1 (3.5%) but entirely absent in G3. Conversely, TD→KQ ("delegate-then-retreat") was most frequent in G3 (12.1% vs. 5.6% in G1, $p < .001$). These patterns indicate that high achievers close the loop by verifying AI-generated solutions, while lower achievers return to conceptual questioning.

3.3 Does Prior Ability Explain the Differences? (RQ3)

To test whether these differences merely reflect pre-existing ability, we ran ANCOVAs controlling for prior-semester GPA.

The two dimensions behaved differently once baseline ability was accounted for. Learning Intention differences were no longer significant ($F(2, 132) = 2.30$, $p = .104$, $\eta_p^2 = .034$): *what* students asked about largely reflected their pre-existing competence. By contrast, Cognitive Presence differences remained significant ($F(2, 132) = 4.70$, $p = .011$, $\eta_p^2 = .066$): *how* students navigated the inquiry process differed across groups even after controlling for ability. That is, the group effect was fully attenuated for Learning Intention but retained for Cognitive Presence, suggesting that the enactment dimension of the coordination regime is not simply a by-product of prior ability and may be open to scaffolding.

4. DISCUSSION AND IMPLICATIONS

This study argues that differences in student–GenAI interaction are best understood as differences in *coordination regimes* (Figure 2). Lower-achieving students more often

exhibited the *integration trap*: consolidating understanding without enacting or testing solutions. Higher-achieving students more often closed the loop through *Artifact Submission*, returning with execution results that enabled diagnosis and movement toward resolution.

The ANCOVA results suggest a dimension-specific interpretation. Learning Intention differences were attenuated after controlling for prior-semester GPA, whereas Cognitive Presence differences remained, consistent with the possibility that goal selection is more tightly coupled to prior competence while enactment is not fully explained by prior ability and may be open to scaffolding. At the same time, our observational design does not support strong causal claims, and future experimental work is needed to test whether scaffolding enactment changes learning behavior or outcomes.

These findings have practical implications for learning analytics and GenAI-supported instruction. In particular, Artifact Submission appears promising as an observable trace-level marker and candidate design target because it is visible in dialogue logs, detectable automatically, and potentially responsive to real-time nudges. Rather than focusing only on prompt engineering [14, 16], educators and systems might test scaffolds that encourage enactment: running code, testing outputs, bringing back errors, and iterating on concrete artifacts. More broadly, assessment may need to place greater value on artifacts of enactment (such as tested code or verified outputs) than on artifacts of inquiry alone, such as chat transcripts.

4.1 Limitations and Future Work

While this study identifies coordination regimes in student–GenAI interaction, several boundary conditions should be considered. The study was conducted in a coding-heavy data mining course, where Artifact Submission often involved executable code, error messages, or output traces; future work should examine whether analogous enactment markers appear in less technical learning contexts. The measures also relied on dialogue logs. Learning Intention therefore captures prompt-expressed cognitive goals rather than latent psychological intentions, and the PCF coding scheme, although pilot-refined and reliable in this dataset, requires further validation across learning domains. Because achievement groups were defined within a single course, future studies should examine whether similar coordination patterns appear across courses, institutions, and learning tasks. Experimental work is also needed to test whether enactment-aware scaffolds change interaction patterns and learning outcomes.

5. ACKNOWLEDGMENTS

This work was supported by the Research Grants Council of Hong Kong Research Fellow Scheme (Reference no: RFS2223-7H02). Generative AI (e.g., ChatGPT-5.1) was used to improve the readability and clarity of the manuscript. After using this tool, the authors carefully reviewed the content. They fully accept responsibility for the publication's content.

6. REFERENCES

[1] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, pages 3–14. IEEE Computer Society, 1995.

[2] L. W. Anderson and D. R. Krathwohl. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Longman, New York, 2001.

[3] C. S. Carver and M. F. Scheier. *On the Self-Regulation of Behavior*. Cambridge University Press, 1998.

[4] H. Crompton and D. Burke. The educational affordances and challenges of chatgpt: State of the field. *TechTrends*, 68(2):380–392, 2024.

[5] D. R. Garrison, T. Anderson, and W. Archer. Critical inquiry in a text-based environment: Computer conferencing in higher education. *The Internet and Higher Education*, 2(2–3):87–105, 1999.

[6] D. R. Garrison and M. Cleveland-Innes. Facilitating cognitive presence in online learning: Interaction is not enough. *American Journal of Distance Education*, 19(3):133–148, 2005.

[7] H. Güner and E. Er. Ai in the classroom: Exploring students' interaction with chatgpt in programming learning. *Education and Information Technologies*, 30(9):12681–12707, 2025.

[8] E. Kasneci et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274, 2023.

[9] W. Li, H.-Y. Chang, A. Bradford, L. Gerard, and M. C. Linn. Combining natural language processing with epistemic network analysis to investigate student knowledge integration within an ai dialog. *Journal of Science Education and Technology*, 34(5):980–993, 2025.

[10] C. K. Lo. What is the impact of chatgpt on education? a rapid review of the literature. *Education Sciences*, 13(4):410, 2023.

[11] G. Sawalha, I. Taj, and A. Shoufan. Analyzing student prompts and their effect on ChatGPT's performance. *Cogent Education*, 11(1):2397200, 2024.

[12] J. R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[13] D. W. Shaffer, W. Collier, and A. R. Ruis. A tutorial on epistemic network analysis: Analyzing the structure of connections in cognitive, social, and interaction data. *Journal of Learning Analytics*, 3(3):9–45, 2016.

[14] J. White et al. A prompt pattern catalog to enhance prompt engineering with chatgpt. In *Proceedings of the 30th Conference on Pattern Languages of Programs*, PLoP '23, pages 1–31, Allerton Park, Monticello, IL, USA, 2023. ACM.

[15] P. H. Winne. A metacognitive view of individual differences in self-regulated learning. *Learning and Individual Differences*, 8(4):327–353, 1996.

[16] J. D. Zamfirescu-Pereira, R. Y. Wong, B. Hartmann, and Q. Yang. Why Johnny can't prompt: How non-AI experts try (and fail) to design LLM prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–21. ACM, 2023.

[17] B. J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory into Practice*, 41(2):64–70, 2002.

Detecting Self-Reported Cognitive-Affective States in Collaborative Problem Solving from Prosodic and Linguistic Features

Videep Venkatesha
Colorado State University

Sifatul Anindho
Colorado State University

Nathaniel Blanchard
Colorado State University

{videep.venkatesha,sifatul.anindho,nathaniel.blanchard} @colostate.edu

ABSTRACT

Detecting cognitive-affective states in collaborative problem-solving is challenging due to their subjective nature, social regulation, and the complexity of multiparty speech. While prior work has largely relied on observer-coded labels in individual learning contexts, self reported states may capture internal experiences that learners regulate and do not outwardly express, precisely the states where intervention could be most valuable. We investigate whether conversational speech alone provides sufficient signal to detect self reported cognitive-affective states during collaborative problem solving. Using retrospective video-cued recall, we obtained labels for seven states (Optimistic, Confused, Curious, Disengaged, Surprised, Frustrated, and Conflicted) from 27 participants (9 groups) completing a collaborative task. We systematically compared prosodic features (eGeMAPS) and semantic embeddings (SentenceTransformers) across multiple temporal window sizes using Leave One Group Out cross validation. Four of seven states were detectable above chance with statistical significance (permutation $p < 0.05$), with AUROCs ranging from 0.570 to 0.618. The remaining three states showed promising trends but did not reach significance, likely due to limited statistical power. No single feature modality dominated across states, and optimal detection windows varied substantially, from 5 seconds for transient states like Confused to 30 seconds for sustained states like Frustrated. These findings suggest that meaningful signal for self-reported affective states exists in conversational speech during collaborative problem-solving, motivating further investigation with larger samples and richer feature representations.

Keywords

Collaborative learning, affective computing, multimodal learning analytics, prosodic features

1. INTRODUCTION

Videep Venkatesha, Sifatul Anindho, and Nathaniel Blanchard. Detecting Self-Reported Cognitive-Affective States in Collaborative Problem Solving from Prosodic and Linguistic Features. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 632–636. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039901

Collaborative problem-solving (CPS) is a learning paradigm where two or more individuals work together to achieve a shared goal [1]. Through CPS, learners engage in social interactions such as joint reasoning, finding common ground, and coordination of strategies, which require the regulation of cognitive-affective states such as confusion and conflict. Prior research has shown that such states influence both group performance and individual learning outcomes [2]. Therefore, automatically detecting cognitive-affective states during collaborative problem-solving is useful for adaptive educational technologies to better support learners in real time.

However, automatic detection of these states in collaborative settings remains challenging. Prior work on affect detection has largely focused on individual learning contexts, where confusion and frustration can be inferred from dialogue in intelligent tutoring systems [3, 4], and multimodal signals including speech, facial expression, and gaze have improved detection [5–7]. Collaborative learning introduces additional complexity: speech overlaps and social masking make it difficult to map behavioral signals to affective states, and observable behavior does not always align with subjective experience. To obtain more ecologically valid labels, Anindho et al. [8] used retrospective cued-recall to capture cognitive-affective labels aligned to collaborative interaction, providing access to internally experienced states that may not be externally observable. Although retrospective labeling may introduce recall bias and some temporal imprecision, it may provide access to subjective states that are not captured by external observation. This brings forth an interesting question about the relationship between subjective states and observable behavior: how does self-reported affect manifest in external signals such as speech in collaborative problem-solving? To address this, we turn to a naturalistic collaborative setting in which both speech and self-reported states can be examined together.

This paper investigates the automatic detection of seven self-reported cognitive-affective states: Optimistic, Confused, Curious, Disengaged, Surprised, Frustrated, and Conflicted, from audio recordings of 27 participants (nine groups) completing a collaborative problem-solving task. Using retrospective video-cued self-report labels, we systematically compare prosodic features (eGeMAPS) and sentence embeddings (SentenceTransformers) across four temporal window sizes (5s, 10s, 20s, 30s). Our results establish preliminary baseline detection performance for each state, revealing that

four states show reliable above-chance detection (permutation $p < 0.05$): Optimistic, Curious, Disengaged, and Confused. Three additional states—Frustrated, Conflicted, and Surprised—have confidence intervals above chance, though statistical power limitations suggest these findings are preliminary. We find that no single feature modality dominates across all states, and that optimal temporal windows vary systematically by state (ranging from 5 seconds for Confused to 30 seconds for Frustrated). Despite the inherent difficulty of detecting subjective internal states from speech alone, our results demonstrate that meaningful signal exists in conversational audio for multiple cognitive-affective states, establishing an initial baseline for self-report-driven affect detection in collaborative learning environments.

2. METHODS

2.1 Participants and Task

This study examines 27 participants who were older than 18 and all fluent in English. They were recruited from the author's department with approval from the Institutional Review Board and compensated with 15 USD. The sample included 18 male, 7 female, and 2 non-binary participants, who self-identified as Asian (13), White (12), or Hispanic (2). Participants were organized into nine collaborative groups, with three participants per group. Each group completed the first part of the Weights Task, which is a collaborative problem-solving activity from Khebour et al. [9] in which participants are given five blocks, a balance scale, and told that one block weighs 10 grams. Their task is to determine the weights of the remaining four blocks through collaborative experimentation and reasoning. The task requires coordination, hypothesis testing, and shared decision-making, naturally eliciting a range of cognitive-affective states as participants encounter uncertainty, disagreement, and discovery [9]. Participants provided written consent from each participant and were informed that their audio and video recordings would be used for research purposes. Recordings are stored on secure, access-controlled servers and transcripts were de-identified prior to analysis. Figure 1 shows one of the groups collaborating during the problem-solving activity.



Figure 1: Our experimental setup during a collaborative problem solving session

2.2 Annotation of Cognitive-Affective States

Cognitive-affective states were annotated using retrospective video-cued recall. After completing the task, each participant individually watched a recording of their collaborative session and marked instances where they experienced specific cognitive-affective states. Seven states were adopted from previous research [10] and were made available for selection: Optimistic, Confused, Curious, Conflicted, Frustrated, Surprised, and Disengaged. Participants could report multiple states concurrently if they experienced overlapping cognitive-affective responses. Each annotation generated a timestamp aligned to the video timeline, along with participant and group identifiers. Annotations were completed independently to avoid social influence, and no external coders were involved in labeling. The resulting dataset therefore reflects participants' subjective, post-hoc recall of their internal cognitive–affective experience, rather than behavioral interpretations assigned by observers. An example conversational segment is presented below. Participant 3 labeled this moment as Frustrated.

> **P1:** So what if we tried 40? Which one did we say was 30?
> **P2:** Purple.
> **P1:** Perfect.
> **P1:** Okay, incorrect. So it is 50.

This segment illustrates the challenge of detecting internal states from speech alone: P1's self-reported frustration is not explicitly verbalized, and the prosodic signature must be extracted from brief utterances ("Okay, incorrect. So it is 50") that occur in the context of ongoing collaboration. The detection task requires distinguishing such moments from similar task-focused utterances spoken without frustration. Table 1 presents the distribution of labeled samples across states and window sizes. The distribution is imbalanced, with Optimistic and Confused being the most frequent states and Surprised, Disengaged, and Frustrated each having fewer than 20 samples. Windows are centered on the point of self-report: a 5-second window includes 2.5 seconds before and 2.5 seconds after the reported moment, a 10-second window includes 5 seconds before and after, and so on. We exclude any window without an associated transcript (i.e., periods of group silence). As window size decreases, the number of such samples increases.

Table 1: Number of positive samples per cognitive-affective state across four temporal window sizes.

State	5s	10s	20s	30s
Optimistic	73	81	88	90
Confused	57	58	58	59
Curious	42	49	52	54
Conflicted	18	19	19	19
Frustrated	16	16	16	16
Surprised	15	15	16	16
Disengaged	15	18	20	21
Total labeled	**236**	**256**	**269**	**275**

2.3 Feature Extraction

Prosodic features. Each participant wore an individual microphone, providing speaker-separated audio. We extracted

88 features using the eGeMAPS feature set via openSMILE [11], including F0 statistics, loudness, spectral features, voice quality measures (jitter, shimmer, HNR), and temporal descriptors. Features were computed over each temporal window.

Transcript features. Speech was transcribed using Whisper [12] and segmented into temporal windows. We obtained 768-dimensional sentence embeddings using `all-mpnet-base-v2` from SentenceTransformers [13], which outperformed BERT `[CLS]` embeddings across states.

Combined features. We additionally concatenated prosodic and transcript vectors (856 dimensions).

2.4 Classification Framework

We framed detection as a set of binary one-vs-rest classification problems, training a separate classifier for each of the seven cognitive-affective states. For each state, the positive class consisted of all windows labeled with that state (using the primary label only), and the negative class was formed by randomly sampling from all non-target windows at a 1:1 ratio. The negative class for each state consists of randomly sampled non-target windows, including both unlabeled windows and windows labeled with other states, to allow for a balanced dataset. This formulation tests whether each state has a discriminative signature that distinguishes it from both neutral collaboration and other affective states, a pragmatic choice for real-world affect detection systems that must distinguish target states from heterogeneous background activity. To reduce variance from negative sampling, we repeated the random undersampling with five different random seeds and averaged results across seeds. We evaluated Logistic Regression (L2, balanced weights), Random Forest (200 trees, balanced weights), and SVM with RBF kernel (balanced weights, Platt scaling). Features were standardized with parameters fit on training data only.

We used Leave-One-Group-Out (LOGO) cross-validation, holding out one group at a time, and report AUROC as the primary metric.

3. RESULTS

We report classification performance using AUROC as the primary metric, averaged across five negative resampling seeds. We report 95% bootstrap confidence intervals (2,000 iterations on pooled LOGO predictions) and permutation test p-values (1,000 iterations) to assess whether observed performance exceeds chance. Logistic Regression was the best or near-best model for the majority of configurations; we focus on Logistic Regression unless otherwise noted.

3.1 Detection Performance Across States

Table 2 reports the best AUROC per state across all window sizes and modalities, along with the optimal modality/window and permutation p-values. Three states met a significance criterion of CI excluding 0.50 and permutation $p < 0.05$: Optimistic (0.607, $p = 0.004$), Curious (0.600, $p = 0.016$), and Disengaged (0.618, $p = 0.023$). Confused reached significance at 5 seconds (0.570, $p = 0.049$). The remaining states of Frustrated (0.634), Conflicted (0.623), and Surprised (0.613), had CIs above chance but permutation p-values between 0.10 and 0.14, reflecting limited power with a 9-fold LOGO design. We additionally evaluated late fusion (averaging classifier probabilities from prosodic-only and transcript only models) and learned fusion (training a meta-classifier on concatenated probability outputs). Neither approach improved over the better single modality (mean AUROC difference: -0.01 for late fusion, $+0.00$ for learned fusion), suggesting that prosodic and linguistic features are largely redundant rather than complementary for most states.

Table 2: Best AUROC per state, with the corresponding optimal modality/window and permutation p-values.

State	n	AUROC	Modality	Window	p
Frustrated	16	0.634	Prosodic	30s	.108
Conflicted	18–19	0.623	Combined	30s	.136
Disengaged	15–21	0.618	Transcript	30s	.023
Surprised	15–16	0.613	Transcript	30s	.103
Optimistic	73–90	0.607	Transcript	10s	.004
Curious	42–54	0.600	Combined	20s	.016
Confused	57–59	0.570	Prosodic	5s	.049

Each state's optimal configuration differed in both temporal window size and feature modality. Across all 28 state–window combinations, transcript features were optimal 13 times, prosodic features 9 times, and combined features 6 times.

3.2 Effect of Temporal Window Size

Table 3 reports the best AUROC per state at each window size with bootstrap CIs. The optimal window varied substantially by state as seen by Figure 2. Four states achieved their best performance with 30 second windows (Frustrated, Conflicted, Disengaged, Surprised), while Optimistic peaked at 10 seconds and Confused at 5 seconds. Curious improved steadily from 5 to 20 seconds, then plateaued.

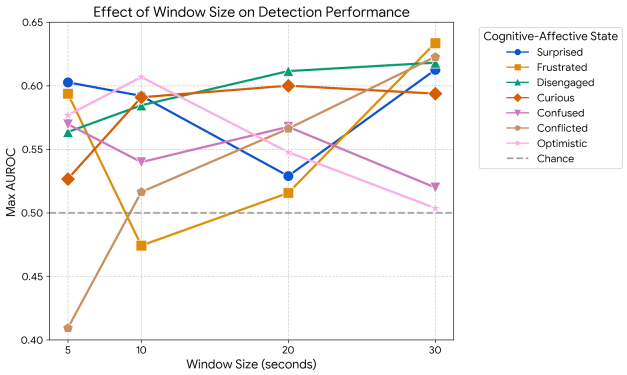


Figure 2: Interaction between temporal window size and detection performance. Max AUROC achieved for each state across 5s, 10s, 20s, and 30s windows. The crossing trajectories highlight distinct temporal dynamics: Optimistic and Confused decay with longer windows (suggesting transient expression), while Disengaged and Frustrated improve (suggesting sustained states).

The temporal trajectory differed qualitatively across states. Optimistic peaked early and declined ($0.577 \rightarrow 0.607 \rightarrow 0.548 \rightarrow 0.504$), with CIs excluding chance only at 5, 10, and 20 seconds. Curious showed consistent above-chance detection from 10 seconds onward, with significant permutation

Table 3: Best AUROC [95% CI] per state at each window size. Bold indicates the highest AUROC per state. † CI excludes 0.50; * permutation $p < 0.05$.

State	5s	10s	20s	30s
Frustrated	.594 [.51, .68]†	.474 [.39, .57]	.516 [.42, .61]	**.634** [.56, .72]†
Conflicted	.409 [.32, .49]	.516 [.44, .60]	.566 [.48, .63]	**.623** [.55, .70]†
Disengaged	.564 [.48, .66]	.585 [.50, .67]	.611 [.52, .68]†	**.618** [.54, .69]†*
Surprised	.603 [.51, .69]†	.592 [.49, .67]	.529 [.45, .63]	**.613** [.51, .69]†
Optimistic	.577 [.54, .62]†	**.607** [.57, .65]†*	.548 [.51, .58]†	.504 [.46, .54]
Curious	.527 [.47, .58]	.591 [.54, .64]†	**.600** [.55, .65]†*	.594 [.55, .64]†*
Confused	**.570** [.52, .62]†*	.540 [.49, .58]	.568 [.52, .62]†	.521 [.47, .56]

p-values at 20s ($p = 0.016$) and 30s ($p = 0.031$). Disengaged improved monotonically with window size but only reached significance at 30 seconds ($p = 0.023$).

4. DISCUSSION

This study investigated whether conversational speech contains sufficient signal to detect self-reported cognitive affective states during collaborative problem-solving. The central finding is encouraging: despite the inherent subjectivity of internal states and the social pressures that shape expression during collaboration, detectable signal exists for multiple states. We discuss the implications of this finding and directions for building on it.

4.1 Signal Exists for Subjective Internal States

We used *self-reported* labels capturing what participants said they internally experienced, rather than observer-coded labels capturing externally visible behavior. In collaborative settings, participants may regulate outward expression to maintain group harmony while still experiencing the underlying state [14, 15]. Our ability to still identify the signal despite this regulation suggests that speech-based detection of internal affective states during collaboration is feasible and worth further investment.

4.2 State-Specific Detection Configurations

A consistent finding across our systematic comparison is that each cognitive-affective state has a distinct optimal detection configuration. No single modality dominated: transcript embeddings were optimal for some states (Disengaged, Surprised, Optimistic), prosodic features for others (Frustrated, Confused), and the combination for two (Curious, Conflicted). Similarly, optimal temporal windows ranged from 5 seconds (Confused) to 30 seconds (Frustrated, Conflicted, Disengaged, Surprised).

The temporal patterns are particularly notable. Confused was best detected at short windows and degraded with longer context, while Frustrated and Disengaged improved monotonically up to 30 seconds. These patterns align with theoretical accounts distinguishing transient states from sustained episodes [16]—confusion as brief cognitive impasse that either resolves or escalates, frustration as accumulated difficulty over time. The state-specific temporal signatures suggest that affect detection systems for collaborative learning may need adaptive windowing strategies rather than fixed parameters.

The failure of multimodal fusion to improve over the best single modality (mean AUROC difference <0.01 across early, late, and learned fusion) indicates that prosodic and linguistic features are largely redundant for these states. This contrasts with typical findings in acted-speech emotion recognition [17], suggesting that the relationship between modalities may differ for self-reported states in naturalistic settings.

4.3 Limitations

The sample of nine collaborative groups limits both the reliability of leave-one-group-out estimates and the power of permutation tests, in addition to limiting generalization .

The retrospective self-report methodology introduces temporal imprecision and potential recall biases [18]. Participants may misalign labels when marking video, or preferentially recall salient moments. The binary one-vs-rest framing creates heterogeneous negative classes that include both unlabeled windows and windows labeled with other states.

Feature representations. Transcript embeddings were computed by concatenating all utterances within each window into a single string, which may not optimally capture turn-taking dynamics, overlapping speech, or the sequential structure of collaborative dialogue. Additionally, prosodic features were aggregated over fixed temporal windows, potentially missing finer-grained affective dynamics within windows.

Generalizability. Our findings are based on a single collaborative task (the Weights Task) with groups of three participants in a controlled laboratory setting. Task characteristics such as problem structure, duration, and social dynamics may influence which states emerge and how they are expressed. Detection performance may differ for other collaborative learning contexts (e.g., open-ended projects, peer tutoring, online collaboration) or different group sizes.

5. CONCLUSION

In this work, we used conversational speech to automatically detect self-reported cognitive-affective states during a collaborative problem-solving task, employing a retrospective video-cued recall approach to obtain affective labels without disrupting the task itself. Four of seven states were detectable above chance from prosodic and linguistic features alone: an encouraging result given the inherent subjectivity of internal states and the social regulation that characterizes collaborative settings. These results motivate future work incorporating richer features, additional modalities, and larger samples. We hope they inspire continued investment in self-report-driven approaches to affect detection in educational technology. We note that the intended application of this work is to assist educators and AI tutoring

systems in providing timely support; any deployment should ensure that inferred states are used to benefit the individual learner rather than for surveillance or social comparison.

6. ACKNOWLEDGMENTS

This material is based in part upon work supported by the U.S. National Science Foundation (NSF) under award DRL 2454151 (Institute for Student-AI Teaming). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

7. REFERENCES

[1] Friedrich Hesse, Esther Care, Juergen Buder, Kai Sassenberg, and Patrick Griffin. A framework for teachable collaborative problem solving skills. In *Assessment and teaching of 21st century skills: Methods and approach*, pages 37–56. Springer, 2014.

[2] Sidney D'Mello and Art Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, 2012.

[3] Sidney D'Mello and Art Graesser. Affect detection from human-computer dialogue with an intelligent tutoring system. In *Intelligent Virtual Agents (IVA 2006)*, volume 4133 of *Lecture Notes in Computer Science*, pages 54–67. Springer, 2006.

[4] Ryan S. J. d. Baker, Sujith M. Gowda, Michael Wixon, Jessica Kalka, Angela Z. Wagner, Aatish Salvi, Vincent Aleven, Gail W. Kusbit, Jaclyn Ocumpaugh, and Lisa Rossi. Towards sensor-free affect detection in cognitive tutor algebra. In *Proceedings of the 5th International Conference on Educational Data Mining (EDM)*, pages 126–133, 2012.

[5] Rafael A Calvo and Sidney D'Mello. Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on affective computing*, 1(1):18–37, 2010.

[6] Nigel Bosch, Sidney K D'Mello, Ryan S Baker, Jaclyn Ocumpaugh, Valerie Shute, Matthew Ventura, Lubin Wang, and Weinan Zhao. Detecting student emotions in computer-enabled classrooms. In *IJCAI*, volume 16, pages 4125–4129, 2016.

[7] Joseph Grafsgaard, Joseph Wiggins, Kristy Elizabeth Boyer, Eric Wiebe, and James Lester. Predicting learning and affect from multimodal data streams in task-oriented tutorial dialogue. In *Educational Data Mining 2014*, 2014.

[8] Sifatul Anindho, Videep Venkatesha, and Nathaniel Blanchard. A methodological framework for capturing cognitive-affective states in collaborative learning. *arXiv preprint arXiv:2507.01166*, 2025.

[9] Ibrahim Khebour, Richard Brutti, Indrani Dey, Rachel Dickler, Kelsey Sikes, Kenneth Lai, Mariah Bradford, Brittany Cates, Paige Hansen, Changsoo Jung, Brett Wisniewski, Corbyn Terpstra, Leanne Hirshfield, Sadhana Puntambekar, Nathaniel Blanchard, James Pustejovsky, and Nikhil Krishnaswamy. When text and speech are not enough: A multimodal dataset of collaboration in a situated task. *Journal of Open Humanities Data*, 2024.

[10] Sifatul Anindho, Videep Venkatesha, Mariah Bradford, Anne M Cleary, and Nathaniel Blanchard. An exploration of internal states in collaborative problem solving. In *International Conference on Human-Computer Interaction*, pages 135–150. Springer, 2025.

[11] Florian Eyben, Martin Wöllmer, and Björn Schuller. opensmile – the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 1459–1462. ACM, 2010.

[12] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pages 28492–28518, 2023.

[13] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3982–3992. Association for Computational Linguistics, 2019.

[14] Emily A Butler, Boris Egloff, Frank H Wilhelm, Nancy C Smith, Elizabeth A Erickson, and James J Gross. The social consequences of expressive suppression. *Emotion*, 3(1):48–67, 2003.

[15] Hanna Järvenoja, Sanna Järvelä, and Jonna Malmberg. Supporting groups' emotion and motivation regulation during collaborative learning. *Learning and Instruction*, 63:101211, 2019.

[16] Sidney D'Mello and Art Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, 2012.

[17] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, et al. The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism. In *Proceedings of INTERSPEECH*, pages 148–152, 2013.

[18] Michael D Robinson and Gerald L Clore. Belief and feeling: Evidence for an accessibility model of emotional self-report. *Psychological Bulletin*, 128(6):934–960, 2002.

Divergent Patterns of Attention and Exploratory Behavior Under Increasing Task Difficulty in an XR Diagnostic

Kukhyeon Kim
Chonnam National University,
edu.kukhyeon@gmail.com

Jeeheon Ryu
Chonnam National University
jeeheon@jnu.ac.kr

ABSTRACT
This study investigates how task difficulty shapes learners' self-reported attention and multimodal exploratory behaviors in an extended reality (XR) diagnostic simulation. Twenty-one medical students completed three strabismus diagnostic subtasks of increasing difficulty (ocular motility, cover-uncover, and prism cover tests) while gaze and head movement data were continuously recorded alongside self-reported attention measured after each task. Results revealed asymmetrical divergence: self-reported attention declined significantly with task difficulty, while gaze path rate and head movement rate increased significantly. This compensation strategy indicates heightened cognitive demand. Critically, behavioral indicators carry context-dependent cognitive meanings, high exploratory activity does not uniformly signal engagement but may indicate struggle. These findings are essential for designing adaptive XR diagnostic systems that interpret behavioral signals appropriately.

Keywords
extended reality, task difficulty, attention, gaze behavior, head movement

1. INTRODUCTION
Extended Reality (XR) has gained traction as a training tool in high-stakes professional domains, offering learners a level of immersion and high contextual realism that conventional digital environments cannot replicate [1]. In clinical education, XR-based diagnostic simulations allow repeated practice of complex procedural skills in safe, controlled contexts, yet they also impose concurrent demands on visual attention, spatial navigation, and clinical reasoning that can strain learners' limited cognitive resources [2].

A key challenge in evaluating XR-based training is that most studies rely on post-task outcome measures, such as accuracy or scores, which offer limited insight into how learners allocate cognitive resources in real time. Gaze and head movement data captured continuously during task performance offer a more direct window into these moment-to-moment processes [4][5]. However, the relationship between behavioral indicators and cognitive states is not straightforward: as task demands rise, learners may simultaneously report declining attention while displaying increased exploratory effort, a behavioral compensation pattern whose meaning shifts with difficulty context [6].

The present study addresses this gap using a within-subjects design in an XR strabismus diagnostic simulation. We examine (1) how self-reported attention changes across three difficulty levels, (2) how gaze and head movement indicators vary with task difficulty, and (3) how the relationship between attention and exploratory behavior shifts depending on the difficulty context. By integrating subjective and objective behavioral measures, this study aims to provide a context-sensitive account of learner cognitive states in immersive clinical training.

Kukhyeon Kim, and Jeeheon Ryu. Divergent Patterns of Attention and Exploratory Behavior Under Increasing Task Difficulty in an XR Diagnostic. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 637–641. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039802

2. THEORETICAL BACKROUND

2.1 XR Learning Environments and Cognitive Demands in Clinical Training
XR-based simulations have gained increasing attention in clinical education for their capacity to replicate realistic diagnostic scenarios while enabling safe, repeatable practice [2]. The Cognitive Affective Model of Immersive Learning (CAMIL) provides a theoretical account of how XR environments engage learners: technological features such as immersion and agency activate cognitive and affective processes, including cognitive load and self-regulation that mediate learning outcomes [1]. Critically, while immersion enhances engagement, it can simultaneously impose extraneous cognitive demands through the need to navigate three-dimensional space and coordinate visual and physical actions alongside core task performance. This dual nature makes it particularly important to characterize how learners' cognitive states change as task demands escalate within XR simulations.

2.2 Task Difficulty, Attention, and Behavioral Compensation
According to Cognitive Load Theory, human working memory is limited, and rising task complexity increases intrinsic load, progressively depleting the cognitive resources available for sustained attention [3]. As task demands increase, learners are expected to report decreased attentional focus, which can be reliably measured using validated subjective mental effort scales [7]. However, the relationship between subjective attention and observable behavior is not straightforward. Rather than reducing exploratory effort in

proportion to attentional decline, learners may instead intensify their visual and physical search to compensate for depleted cognitive resources, a pattern termed behavioral compensation [4]. Whether this compensatory pattern emerges, and how its meaning shifts with difficulty context, remains an underexplored question. Crucially, behavioral compensation is conceptualized here as a strategic redirection of behavioral effort when attentional resources are depleted, distinguishing it from random or fatigue-driven movement: learners actively intensify exploration to gather diagnostic cues despite reduced subjective attention [10, 11]. This framework predicts not only main effects of difficulty on both attention and behavior, but also a context-dependent reversal of their relationship at high difficulty, where preserved attentional capacity becomes increasingly coupled with strategic exploratory effort [12].

2.3 Multimodal Behavioral Indicators: Gaze and Head Movement

Eye gaze and head movement provide continuous, non-intrusive indices of cognitive dynamics during task performance that retrospective self-reports cannot capture [5]. Gaze path rate reflects the volume and spatial extent of visual exploration, while gaze speed variability indexes instability in information-seeking dynamics. In XR specifically, head translation path rate captures embodied spatial exploration, as learners must physically reposition themselves to acquire visual information within three-dimensional space [8]. Critically, the functional meaning of these indicators may vary depending on difficulty context: the same level of gaze exploration may reflect unfocused, scattered search at low difficulty, but strategic, resource-dependent search at high difficulty [6]. Integrating subjective attention with multimodal behavioral data is therefore essential for accurate cognitive assessment.

.

3. METHOD

3.1 Participants

Twenty-one medical students (13 males, 8 females; third- or fourth year) participated in this study. All had completed relevant ophthalmology coursework and practical training in strabismus prior to participation. The study was approved by the Institutional Review Board (No. 1040198-221021-HR-128-02).

3.2 Procedure and Task Design

Participants completed three strabismus diagnostic subtasks using a HoloLens 2 head mounted display, following a 10-minute familiarization session and eye-tracking calibration. The tasks were administered in a fixed order mirroring standard clinical workflow with progressive cognitive demands: (1) ocular motility test (Level 1; low difficulty), in which participants positioned themselves approximately 40 cm from the virtual patient and moved a fixation stick along an H-pattern while tracking the patient's eye movements to determine the presence of strabismus; (2) cover-uncover test (Level 2; medium difficulty), in which participants alternately covered each of the virtual patient's eyes with an occluder and observed refixation movements to identify both the presence and the type of strabismus; and (3) prism cover test (Level 3; high difficulty), in which participants placed a prism over the non-deviating eye while applying the occluder to the strabismic eye, then iteratively adjusted the prism degree until eye movement was no longer observed in order to quantify the angle of deviation. The progressive difficulty structure reflects escalating demands in visual search, pattern recognition, and procedural reasoning beyond passive vision screening, and is consistent with established clinical training standards and was validated by ophthalmology specialists. Trial duration was limited to three minutes for Levels 1 and 2, and five minutes for Level 3. A within-subjects design was employed, with each participant completing all three tasks.

3.3 Measures

*S*elf-reported attention was assessed immediately after each task using a 7-point Likert scale [9], where higher scores indicate greater attentional focus. Four behavioral indicators were derived from continuous XR tracking logs. *Gaze path rate* (GPR) and *gaze speed variability* (SD of instantaneous gaze speed) served as indices of visual exploration volume and stability, respectively, computed from frame-to-frame gaze hit position data. *Head movement rate* (HMR) and *head angular speed variability* (SD of angular speed) served as indices of embodied exploration, derived from head position and rotation quaternion data sampled at each frame.

3.4 Analysis

Task difficulty was treated as an ordered three-level predictor. To address RQ1 and RQ2, within-person dependence was accounted for using participant-level random-intercept linear mixed models (LMM), a participant fixed-effects approach was substituted when random-effects structures were not estimable. To address RQ3, Pearson correlations between self-reported attention and each behavioral metric were computed separately at each difficulty level to examine context dependent shifts in the attention and exploration relationship. All analyses were conducted in Python (Google Colab), with $\alpha = .05$.

Note on the dataset. A complementary descriptive analysis of the same XR strabismus simulation dataset has been reported in the proceedings of iLRN 2026 [13]. That prior work focused exclusively on comparing behavioral indicators across difficulty levels. The present study extends this work by introducing self-reported attention as a subjective cognitive measure, examining the difficulty-contingent relationship between attention and exploratory behavior, and interpreting findings through Cognitive Load Theory to propose a behavioral compensation mechanism.

4. RESULTS

Table 1 summarizes descriptive statistics and model-based effects for all five outcome variables across the three difficulty levels.

4.1 RQ1. Effects of Task Difficulty on Self-Reported Attention

Self-reported attention decreased significantly as task difficulty increased ($\beta = -0.184$, $p = .008$, LMM). Mean scores declined monotonically from 6.20 ($SD = 0.62$) at Level 1 to 6.06 ($SD = 0.74$) at Level 2 and 5.83 ($SD = 0.54$) at Level 3, confirming that learners experienced progressively greater cognitive strain across the three diagnostic subtasks.

4.2 RQ2. Effects of Task Difficulty on Exploratory Behavior

Increased task difficulty was associated with significant increases across three of the four behavioral indicators. Gaze path rate rose approximately 2.85-fold from Level 1 to Level 3 ($\beta = 0.32$, $p < .001$, LMM), indicating that learners engaged in substantially broader visual scanning under higher demand. Gaze speed variability also increased significantly ($\beta = 0.78$, $p = .010$, fixed effects), reflecting greater instability in visual search dynamics. Head movement rate similarly increased with difficulty ($\beta = 0.01$, $p < .001$, fixed effects),

suggesting that learners engaged in greater physical repositioning as diagnostic demands escalated. Head angular speed variability showed an increasing trend but did not reach statistical significance (β = 0.04, p = .100). Together, these results indicate that despite declining subjective attention, learners intensified their exploratory effort across difficulty levels, a pattern consistent with behavioral compensation under increasing cognitive load. Notably, while the decline in self-reported attention was modest (5.9% reduction across levels), behavioral indicators showed substantially larger relative increases (gaze path rate: 185%; gaze speed variability: 50%; head movement rate: 50%), suggesting that objective behavioral metrics may capture cognitive demand shifts more sensitively than subjective reports.

4.3 RQ3. Difficulty-Contingent Shifts in the Attention–Exploration Relationship

The relationship between self-reported attention and GPR shifted in direction across difficulty levels (Table 2). At Levels 1 and 2, negative correlations were observed (r = -0.217 and r = -0.176, respectively), indicating that learners reporting lower attention exhibited broader visual exploration patterns, consistent with unfocused, scattered search. In contrast, at Level 3, the correlation reversed to positive (r = 0.342), despite a statistically significant decline in mean attention across the cohort (L1: M = 6.20 → L3: M = 5.83, β = -0.18, p = .008). This reversal is notable: the substantial increase in mean GPR (from 0.34 at L1 to 0.97 at L3) coincided with decreased attention, suggesting that visual exploration intensity increased not because of greater attentional resources, but possibly due to heightened task demand. Within this constrained attentional environment, the positive correlation (r = 0.342) indicates that learners who retained relatively higher attention tended to exhibit more extensive gaze-based exploration, a pattern consistent with a compensatory mechanism whereby preserved attentional capacity is redirected toward broader information gathering.

For HMR, negative correlations persisted across all three levels (Level 1: r = -0.182; Level 2: r = -0.300; Level 3: r = -0.055), with magnitude substantially attenuated at Level 3. This progressive dissociation between attention and head-based exploration under increasing demand suggests that gaze became the primary behavioral marker of cognitive resource allocation at peak difficulty, whereas head movement decoupled from attentional availability, possibly reflecting a shift toward more efficient, gaze-centered diagnostic strategies.

Table 1. Descriptive Statistics and Model Effects by Task Difficulty

Outcome	L1 M (*SD*)	L2 M (*SD*)	L3 M (*SD*)	β	p	Model
Attention	6.20 (0.62)	6.06 (0.74)	5.83 (0.54)	−0.18	.008	LMM
Gaze path rate	0.34 (0.22)	0.77 (0.52)	0.97 (0.84)	0.32	<.001	LMM
Gaze speed variability	2.88 (1.07)	4.13 (1.92)	4.33 (2.65)	0.78	.010	Fixed
Head movement rate	0.02 (0.01)	0.03 (0.01)	0.03 (0.02)	0.01	<.001	Fixed
Head rotation variability	0.15 (0.10)	0.22 (0.11)	0.23 (0.22)	0.04	.100	Fixed

Note. L1 = ocular motility; L2 = cover–uncover; L3 = prism cover. β = unstandardized coefficient. LMM = linear mixed model. Fixed = participant fixed effects.

Table 2. Pearson Correlations Between Attention and Exploratory Metrics by Difficulty Level

	Level 1	Level 2	Level 3
Attention – GPR	−0.217	−0.176	+0.342
Attention – HMR	−0.182	−0.300	−0.055

Note. GPR = gaze path rate; HMR = head movement rate. Bold indicates direction reversal.

5. DISCUSSION AND CONCLUSION

This study examined how task difficulty shapes self-reported attention and multimodal exploratory behaviors in an XR diagnostic simulation, and how the relationship between these measures shifts with difficulty context.

5.1 Declining Attention and Behavioral Compensation

Consistent with Cognitive Load Theory [3], self-reported attention declined monotonically as diagnostic task demands escalated. Concurrently, gaze path rate and head movement rate increased significantly, presenting an asymmetrical pattern in which subjective attentional resources were depleted while behavioral exploratory effort was amplified. This divergence is interpretable as a behavioral compensation strategy: despite declining attentional capacity, learners persistently expanded their visual search for diagnostic cues. The finding underscores that behavioral quantity (gaze/head movement) and attentional quality (self-reported focus) do not move in tandem, and that exploration metrics alone may misrepresent cognitive engagement without accounting for difficulty context and concurrent attentional constraints. An alternative interpretation, that increased eye movements at higher difficulty reflect visual strain rather than cognitive compensation, is less consistent with the observed selective patterns: gaze path rate increased significantly while head rotation variability did not, and the GPR–attention correlation reversed at Level 3 — patterns not predicted by a fatigue-based account[11].

5.2 Difficulty-Contingent Reversal of the Attention–Exploration Relationship

The most theoretically significant finding is the reversal of the attention−GPR correlation at Level 3. At lower difficulty levels, lower attention was associated with broader exploration, consistent with unfocused, scattered search. At the highest difficulty level, this relationship reversed within a substantially constrained attentional environment. Within this depleted-attention context, learners who retained relatively higher attentional resources tended to exhibit more extensive gaze-based exploration.

This pattern suggests a compensatory mechanism: when task difficulty depletes attentional capacity across the cohort, learners with preserved (relative) resources redirect their search effort toward more thorough visual exploration to compensate for degraded

cognitive focus. The reversal does not indicate that higher absolute attention predicts exploration; rather, it reflects how the same cognitive resource (attention) is allocated differently under distinct difficulty contexts.

This finding aligns with the CAMIL model's prediction that high cognitive demands amplify performance heterogeneity [1] and underscores the critical risk of misclassifying learner cognitive states when behavioral indicators are interpreted without considering concurrent difficulty level and attentional availability [5].

5.3 Implications, Limitations, and Conclusion

These findings carry direct implications for adaptive XR learning system design. Because the same behavioral signal carries different cognitive meanings across difficulty levels and attentional states, real-time learner state monitoring requires multimodal data that combines objective behavioral indicators (gaze, head movement) with subjective attention measures and explicit difficulty-level context. A learner exhibiting simultaneously low attention and high gaze exploration under peak demand may not be "strategic" but rather "compensatory"—attempting to maintain performance through expanded search despite depleted attentional capacity. Conversely, low exploration coupled with low attention under high demand may indicate cognitive overload and risk of disengagement, warranting targeted adaptive scaffolding.

Several limitations should be noted. The small sample size limits generalizability, and the modest strength of the attention–exploration relationship suggests substantial individual variability. The fixed task order, while ecologically valid, introduces potential order and fatigue effects. Future research should employ larger, multisite samples to confirm the replicability of the difficulty-contingent reversal, examine how these behavioral-attentional profiles relate to diagnostic accuracy outcomes, and investigate whether real-time multimodal monitoring can support adaptive difficulty adjustment in clinical XR training.

This study has several limitations. The modest sample size (n = 21) limits generalizability, though the consistent direction of effects across multiple indicators warrants larger-scale replication. The fixed task order, retained for ecological validity in line with standard clinical workflow, cannot fully separate difficulty effects from order or fatigue effects; counterbalanced designs are needed where clinically feasible. Self-reported attention, collected post-task, may not capture moment-to-moment fluctuations; future research should incorporate diverse cognitive load measures (e.g., pupillometry, EEG) to formally test the behavioral compensation hypothesis. Finally, future work should examine how these multimodal behavioral signatures predict diagnostic accuracy to validate their utility as adaptive assessment tools in XR training

In conclusion, this study provides empirical evidence that task difficulty produces divergent, difficulty-dependent trajectories in attention and exploratory behavior in XR. Critically, the functional meaning of behavioral indicators (gaze, head movement) is not static but context-dependent, shaped by both difficulty level and the learner's residual attentional capacity. These findings advance the theoretical understanding of learner cognitive states in immersive environments and offer a nuanced behavioral basis for designing more responsive, context-aware clinical training systems that account for the complex interplay between objective exploration and subjective attentional availability.

6. REFERENCES

[1] Makransky, G. and Petersen, G. B. 2021. The Cognitive Affective Model of Immersive Learning (CAMIL): A theoretical research-based model of learning in immersive virtual reality. *Educational Psychology Review* 33, 937–958. DOI= https://doi.org/10.1007/s10648-020-09586-2

[2] Laspro, M., Groysman, L., Verzella, A. N., Kimberly, L. L., and Flores, R. L. 2023. The use of virtual reality in surgical training: Implications for education, patient safety, and global health equity. *Surgeries* 4, 4, 635–646. DOI= https://doi.org/10.3390/surgeries4040056

[3] Sweller, J. 2019. Cognitive load theory and educational technology. *Educational Technology Research and Development* 68, 1, 1–16. DOI= https://doi.org/10.1007/s11423-019-09701-3

[4] Gorin, H., Patel, J., Qiu, Q., Merians, A., Adamovich, S., and Fluet, G. 2024. A review of the use of gaze and pupil metrics to assess mental workload in gamified and simulated sensorimotor tasks. *Sensors* 24, 6, 1759. DOI= https://doi.org/10.3390/s24061759

[5] Mu, S., Cui, M., and Huang, X. 2020. Multimodal data fusion in learning analytics: A systematic review. *Sensors* 20, 23, 6856. DOI= https://doi.org/10.3390/s20236856

[6] Schirm, J., Gómez-Vargas, A. R., Perusquía-Hernández, M., Skarbez, R. T., Isoyama, N., Uchiyama, H., and Kiyokawa, K. 2023. Identification of language-induced mental load from eye behaviors in virtual reality. *Sensors* 23, 15, 6667. DOI= https://doi.org/10.3390/s23156667

[7] Paas, F. G. W. C., van Merriënboer, J. J. G., and Adam, J. J. 1994. Measurement of cognitive load in instructional research. *Perceptual and Motor Skills* 79, 1, 419–430. DOI= https://doi.org/10.2466/pms.1994.79.1.419

[8] Lustig, A., Wilf, M., Dudkiewicz, I., and Plotnik, M. 2023. Higher cognitive load interferes with head-hand coordination: Virtual reality-based study. *Scientific Reports* 13., 17632. DOI= https://doi.org/10.1038/s41598-023-43337-x

[9] Georgiou, Y., & Kyza, E. A. 2017. The development and validation of the ARI questionnaire: An instrument for measuring immersion in location-based augmented reality settings. *International Journal of Human-Computer Studies, 98*, 24-37. https://doi.org/10.1016/j.ijhcs.2016.09.014

[10] Martinez-Cedillo, A. P., Gavrila, N., Mishra, A., Geangu, E., and Foulsham, T. 2025. Cognitive load affects gaze dynamics during real-world tasks. *Experimental Brain Research* 243, 4, 82. DOI= https://doi.org/10.1007/s00221-025-07037-4

[11] Walter, K., and Bex, P. 2021. Cognitive load influences oculomotor behavior in natural scenes. *Scientific Reports* 11, 12405. DOI= https://doi.org/10.1038/s41598-021-91845-5

[12] Nasri, M., Kosa, M., Chukoskie, L., Moghaddam, M., and Harteveld, C. 2024. Exploring eye tracking to detect cognitive load in complex virtual reality training. In *2024 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, 51–54. DOI= https://doi.org/10.1109/ISMAR-Adjunct64951.2024.00022

[13] Kim, K., Han, D., Yang, E., and Ryu, J. 2026. Analysis of Attention Based on Gaze and Head Movements under Task Difficulty in an Extended Reality Diagnostic Simulation. In

Do Online Quiz-Taking Behaviors Matter? An Event-Level Log Analysis in Computer Science Courses

Steve Geinitz
Metropolitan State University of Denver
geinitz@msudenver.edu

ABSTRACT

This work-in-progress study asks whether fine-grained online quiz process data can explain exam performance in computer science courses. Using event logs from two courses, we aggregated quiz behavior for $N = 138$ student-exam observations and investigated whether browser focus-switching ("page blur") and broader quiz-taking profiles added explanatory value beyond pre-semester GPA. We extracted ten behavioral features capturing blur frequency, timing, attempts, variability, and formative quiz performance, then applied random-forest feature ranking, hierarchical regression, and K-means clustering. Page blur frequency emerged as the strongest behavioral predictor and was found to be statistically significant in predicting summative exam performance beyond pre-semester GPA alone ($p = 0.041$). In contrast, interpretable behavioral clusters did not significantly improve model fit ($p = 0.148$). The results suggest that a specific attention-related process signal can be more informative than broader behavioral profiles, although its gain over an academic baseline is modest. These findings are used to motivate a future intervention in which page blur is embedded within a broader Effort-and-Engagement score is shared with students through personalized feedback.

Keywords

student assessment, learning management systems, learning analytics, process data, nudge interventions

1. INTRODUCTION

Online quizzes are now routine in education but are often observed only through scores. In unproctored settings, that leaves instructors with little visibility into how students interact with an assessment and whether they persevere with a challenging problem, switch away to another resource, or engage in repeated practice. Learning analytics makes these behaviors visible through learning management system (LMS) process data, but it remains unclear which signals are genuinely useful for explaining later summative performance.

Steve Geinitz. Do Online Quiz-Taking Behaviors Matter? An Event-Level Log Analysis in Computer Science Courses. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 642–646. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040117

That question is especially important when quizzes function as low-stakes, frequent assessments rather than one-shot checkpoints. In that setting, quiz logs capture not only performance but repeated opportunities for practice, retakes, and feedback before a higher-stakes exam. This makes them a useful context for analytics intended to support students early, while there is still time for behavior change.

The work herein focuses on one concrete signal: browser focus loss during a quiz attempt, recorded as a "page blur." We treat blur events as a low-cost, imperfect proxy for attention switching or external resource use. The study is also part of a broader work-in-progress effort to develop an Effort-and-Engagement (EE) score that combines quiz behaviors with other student-facing course indicators to support non-punitive, personalized feedback. Here, however, we narrow the contribution to two questions.

- **RQ1:** Does the frequency of browser focus-switching ("page blur" events) during quizzes predict final exam performance, after controlling for students' baseline academic ability (GPA)?
- **RQ2:** Can unsupervised clustering of quiz behaviors (including timing, attempts, and blurs) identify distinct student profiles that explain performance differences better than GPA alone?

2. RELATED WORK

Process-data research shows that event logs can reveal strategies that are not visible in correctness-only measures [7, 8]. In LMS settings, log event features can predict achievement, but recent reviews emphasize that these gains depend heavily on feature engineering, context, and strong baseline comparison [1, 13]. Process-mining and formative-assessment studies have also shown that quiz behavior can be summarized into interpretable patterns or clusters [2, 4, 9].

Attention and multitasking research provides the conceptual motivation for using page blur as a signal. Attention can be inferred from interaction traces [18], off-screen exam behaviors can be detected through more intrusive sensing [5], and multitasking is often associated with poorer academic outcomes [10, 12, 17]. At the same time, window-switching behaviors are context dependent and should not be treated as a standalone cheating detector or risk label [14, 16]. This tension motivates our focus on whether a specific blur signal is more useful than broader behavioral profiling.

3. METHODS

We analyzed quiz logs from the Canvas LMS for two upper-division computer science courses offered during Fall 2025. To align quiz behavior with major assessments, we aggregated data over two eight-week windows. Namely, quizzes preceding the midterm and quizzes preceding the final. Because students could contribute to one or both windows, the final dataset contained $N = 138$ student-exam observations from roughly 70 students. The outcome variable was the corresponding exam grade, with pre-semester GPA serving as the academic baseline. Many quizzes included randomized Canvas "Formula" items, which reduced the value of answer memorization across attempts and made repeated attempts more interpretable as continued engagement with the procedure.

For each observation, we extracted ten quiz-level features describing blur frequency and change, time on task, timing variability, number of attempts, days of engagement, and formative quiz performance. We first trained a random forest model to identify the most salient behavioral predictors, then estimated hierarchical linear models to compare a GPA-only baseline with a GPA+blur model and a GPA+cluster model. For clustering, we tested $K = 2$ through $K = 5$ on the full behavioral feature set and retained $K = 3$ because it yielded the most interpretable profiles. We also used 5-fold cross-validation as a secondary comparison between blur-based and cluster-based predictive configurations. The extracted features were:

- **avg_page_blurred_mean**: average page-blur frequency per attempt
- **avg_blurred_change_mean**: average change in blur count across attempts
- **avg_days_between_attempts_mean**: average spacing between attempts
- **avg_duration_mean**: average attempt duration
- **avg_duration_change_mean**: average change in duration across attempts
- **pct_correct_mean**: mean highest formative quiz score
- **std_duration_mean**: within-quiz duration variability
- **distinct_days_mean**: average number of active quiz days
- **std_page_blurred_mean**: variability in blur frequency
- **total_attempts_mean**: average number of quiz attempts

Note that many of those features consist of a summary statistic, e.g. average, over all the attempts for one quiz. These are then averaged again over all of the quizzes in the time period using an arithmetic mean to yield one feature for each student-exam pair.

4. RESULTS

The random-forest feature ranking identified GPA as the strongest overall predictor, but page-blur measures emerged as the most important behavioral features, with

Table 1: Top Behavioral Features from the Random Forest

Feature	Importance
`avg_page_blurred_mean`	0.155
`avg_blurred_change_mean`	0.111
`avg_days_between_attempts_mean`	0.077
`avg_duration_mean`	0.069
`avg_duration_change_mean`	0.067

`avg_page_blurred_mean` and `avg_blurred_change_mean` ranking above formative quiz performance. We therefore centered the confirmatory models on `avg_page_blurred_mean`.

For RQ1, the GPA-only model explained 18.2% of the variance in exam grades ($R^2 = 0.182$). Adding average blur frequency increased R^2 to 0.207, and the blur coefficient was negative and statistically significant ($\beta = -0.512$, $t = -2.063$, $p = 0.041$). The nested-model comparison was also significant ($F_{1,135} = 4.26$, $p = 0.041$). In practical terms, students who switched away from the quiz window more frequently tended to perform worse on the associated summative exam, even after accounting for GPA.

For RQ2, K-means with $K = 3$ yielded interpretable profiles that we summarized as steady workers, high-effort retrying students, and low-effort/high-distraction students. However, adding these cluster labels to the GPA model did not significantly improve model fit ($R^2 = 0.205$; $F_{2,134} = 1.94$, $p = 0.148$). This same pattern appeared in cross-validation also, as seen in Table 2, where models using the continuous blur feature outperformed those using cluster labels. Together, these findings suggest that a granular, theoretically motivated attention signal carried more explanatory value than broader behavioral categories.

The three clusters were still descriptively useful. The "steady worker" profile paired longer, more focused attempts with strong quiz scores; the "high-effort retrying" profile showed repeated, shorter, focused attempts; and the "low-effort/high-distraction" profile combined fewer attempts with more blur events and weaker formative performance. These profiles suggest meaningful differences in quiz-taking strategy, even though they did not explain summative performance beyond GPA.

- **Steady workers**: longer attempts, moderate retake behavior, and relatively low blur frequency
- **High-effort retrying students**: more attempts spread across days, shorter attempts, and strong formative scores
- **Low-effort/high-distraction students**: fewer attempts, less distributed engagement, and the highest blur frequency

Table 2 shows the comparison of blur-based and cluster-based models using 5-fold cross-validation. The same pattern holds there, in which models using the continuous blur feature achieved lower error than those using cluster labels, with the random forest with GPA and blur performing best

Table 2: 5-Fold Cross-Validation Comparison

Model	RMSE
Random Forest (GPA + Blur)	18.08
Linear Regression (GPA + Blur)	18.53
Random Forest (GPA + Clusters)	18.55
Linear Regression (GPA + Clusters)	18.71

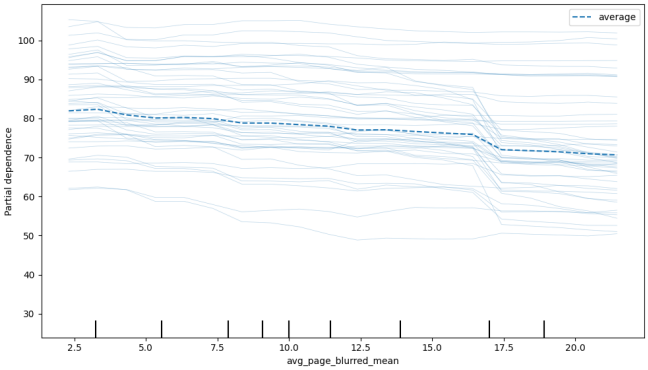


Figure 1: Partial dependence and ICE plot for average page blur frequency and its relationship with exam scores. As blur frequency increases, the modest but statistically significant negative relationship can be seen.

overall. Figure 1 presents the partial dependence and individual conditional expectation plot when average page blurs were used to directly predict exam performance.

5. DISCUSSION AND NEXT STEPS

The central finding is that granularity matters. Namely, that a specific, theoretically grounded signal tied to attention switching explained more than broad behavioral profiles. This does not mean that page blur should be treated as a direct proxy for cheating or disengagement. The gain beyond GPA was modest, and prior work shows that log metrics are highly context dependent [1, 14]. Still, blur frequency appears useful as a low-stakes risk signal when it is interpreted alongside other course behaviors rather than in isolation.

This is also why the clustering result remains useful, even though it was not significant in the regression model. The clusters helped describe different quiz-taking strategies in a way that may still matter for instructor insights and feedback. In other words, the profiles were descriptively interesting, but the blur metric was the sharper explanatory signal for this dataset. However, the implication should not be that blur events replace broader views of student engagement. But, instead, that they can add a specific attention-related signal that is easy to collect and interpret. In practice, this makes blur events more useful as one component of a larger engagement model than as a standalone indicator.

That framing directly informs our next step. We are using this result to design an Effort-and-Engagement (EE) score that combines page blur frequency with broader indicators such as attendance, assignment completion, quiz participation, and retakes. The aim is not high-stakes classification, but student-facing feedback that makes engagement patterns more visible and actionable. This direction is consistent with work on nudge analytics [3], student-facing dashboard feedback [6, 19], and personalized nudges [15]. In future work, we plan to evaluate whether periodic, supportive EE updates can shift quiz behavior, quiz-retake patterns, and student perceptions of support without becoming punitive or noisy. That future study will let us test whether a small but interpretable log-data signal can be translated into an actionable, student-facing intervention rather than remaining only a predictive model.

5.1 Limitations

Several limitations matter for interpreting the current result. First, this is an observational analysis from two upper-division computer science courses at a single institution. The dataset is adequate for this work-in-progress, but it is modest, and the student-exam structure means that some students contributed more than one observation across the two assessment windows. The findings should therefore be interpreted as evidence of a promising signal in a specific course context rather than as a general effect size for LMS quizzes.

Second, blur events are behaviorally ambiguous. A page blur can reflect off-task multitasking or answer-seeking, but it can also reflect note-checking, calculator use, accessibility needs, temporary interruptions, or ordinary course-related resource use. Our claim is deliberately narrower, which is that frequent focus-switching is associated with lower exam performance after controlling for GPA. We do not interpret blur as a direct measure of cheating, intent, or disengagement. That is also why we frame the signal as low stakes and combine it with other student-controllable behaviors in the planned intervention.

Third, the predictive gain is modest. GPA remained the strongest baseline predictor, and the increase in explanatory power from adding blur was meaningful but not large. For practice, that means the value of process data here is not that it unlocks a highly accurate predictive model. Its value is that it surfaces a concrete, interpretable behavior that instructors can discuss with students and potentially target through supportive feedback. That practical value is stronger in courses that already rely on low-stakes, frequent quizzes. Repeated formative assessments create multiple chances to practice, revisit material, and act on feedback, so they also create multiple opportunities for instructors to notice process patterns before a major exam. This is one reason the signal is more useful for early support than for high-stakes classification.

5.2 Intervention Design

The planned EE score aggregates five student-controllable dimensions of course engagement:

1. **Attendance**: proportion of class sessions attended
2. **Assignments**: proportion of assignments turned in
3. **Quiz attempts**: proportion of available quizzes attempted
4. **Quiz retakes**: repeated engagement with formative quizzes
5. **Page blur frequency**: average focus-switching rate during quiz attempts

The score is intended to surface interpretable components rather than produce a single opaque risk label. The sample

message below illustrates the style of the planned feedback, which aims to be supportive, behavior-focused, and framed around concrete next steps, rather than to shame or punish.

Your Effort & Engagement Update

Hi [Student],

Here is your current effort-and-engagement snapshot for CS 1234. This is not a grade—it is a reflection of how you have been engaging with the course so far.

Attendance	85%	
Assignments Submitted	90%	
Quizzes Attempted	100%	
Quiz Retakes	40%	
Quiz Focus (blur rate)	65%	
Overall EE Score	**71/100**	(class mean = 75)

You are doing well—you have attempted every quiz and your attendance is strong. Two areas to consider: you have retaken fewer quizzes than most of your peers, and your quiz focus score suggests you may be switching away from the quiz window frequently. Retaking quizzes can help you practice the process, and staying on-task can make that practice more valuable.

Keep it up!

Figure 2: Representative EE score message used to illustrate the planned student-facing intervention.

The design of these messages follows three principles from prior work. They should be 1) *personalized* to the student's own data, 2) *explanatory* about which behaviors are driving the score, and 3) *actionable* in the next steps they recommend [6, 11, 19]. The LLM should itself be viewed as an auxiliary delivery mechanism for that feedback, rather than as the source of the intervention logic itself.

We envision these messages being delivered routinely, but are aware students may ignore them if frequency is too high. Ideally, the message supports reflection without becoming part of the grading environment. Each EE score component is transparent and controllable, allowing students to see if their score is affected by attendance, missing assignments, limited quiz participation, low retake behavior, or frequent focus-switching. This transparency is important as the intervention aims to prompt helpful self-regulation rather than create a vague sense of surveillance.

The next study will therefore focus less on prediction and more on response to feedback. We plan to examine whether supportive EE updates change behaviors that instructors actually want to influence, such as quiz completion, retake use, and sustained on-task engagement, and whether students perceive the messages as useful, fair, and non-punitive. That evaluation is necessary before any broader deployment, because even interpretable analytics can fail if the feedback they generate is mistimed, noisy, or too easy to misread.

6. CONCLUSION

The main contribution of this work has been to investigate several quiz-process indicators and has found that page blur frequency provides the clearest behavioral signal beyond GPA. While broader student profiles can be useful for classifying students broadly, their prediction capabilities are more limited. The combination of modest predictive value and high interpretability makes blur most promising as one component of a student-facing engagement intervention rather than as a standalone risk flag. Because the signal comes from low-stakes, frequent quizzes, it is also available early enough to inform supportive feedback before higher-stakes assessments occur.

7. ACKNOWLEDGEMENTS

This research was approved by, and conducted in compliance with, the Institutional Review Board (IRB) at Metropolitan State University of Denver. The authors would like to express their gratitude to the Human Subjects Protection Program for their guidance and support in ensuring the ethical integrity of this study. Anonymized data is available on request.

8. REFERENCES

[1] C. J. Arizmendi, M. L. Bernacki, M. Raković, R. D. Plumley, C. J. Urban, A. T. Panter, J. A. Greene, and K. M. Gates. Predicting student outcomes using digital logs of learning behaviors: Review, current standards, and suggestions for future work. *Behavior Research Methods*, 55(6):3026–3054, Aug. 2022.

[2] A. Bayazit, D. Yildirim, G. Akcapinar, and H. Ilgaz. Examining students' formative test-taking behaviors using learning analytics. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, 14(Özel Sayı):382–395, Oct. 2023.

[3] C. Carmean and P. Mizzi. The case for nudge analytics. *EDUCAUSE Quarterly*, 33(4), 2010.

[4] R. Csányi and G. Molnár. How do test-takers rate their effort? a comparative analysis of self-report and log file data. *Learning and Individual Differences*, 106:102340, Aug. 2023.

[5] N. Dilini, A. Senaratne, T. Yasarathna, N. Warnajith, and L. Seneviratne. Cheating detection in browser-based online exams through eye gaze tracking. In *2021 6th International Conference on Information Technology Research (ICITR)*, pages 1–8. IEEE, Dec. 2021.

[6] J. Fritz. Classroom walls that talk: Using online course activity data of successful students to raise self-awareness of underperforming peers. *The Internet and Higher Education*, 14(2):89–97, Mar. 2011.

[7] Y. Han, F. Ji, P. Wang, and H. Liu. Assessing multiple abilities through process data in computer-based assessments: The multidimensional sequential response model (msrm). *Behavior Research Methods*, 57(5), Apr. 2025.

[8] H. Jiao, Q. He, and B. P. Veldkamp. Editorial: Process data in educational and psychological measurement. *Frontiers in Psychology*, 12, Dec. 2021.

[9] L. Juhaňák, J. Zounek, and L. Rohlíková. Using process mining to analyze students' quiz-taking behavior patterns in a learning management system. *Computers in Human Behavior*, 92:496–506, Mar. 2019.

[10] R. Junco and S. R. Cotten. No a 4 u: The relationship between multitasking and academic performance. *Computers & Education*, 59(2):505–514, Sept. 2012.

[11] A. N. Kluger and A. DeNisi. The effects of feedback interventions on performance: A historical review, a meta-analysis, and a preliminary feedback intervention

theory. *Psychological Bulletin*, 119(2):254–284, Mar. 1996.

[12] J. M. Kraushaar and D. C. Novak. Examining the affects of student multitasking with laptops during the lecture. *Journal of Information Systems Education*, 21(3):241–253, 2010.

[13] T. Lerche and E. Kiel. Predicting student achievement in learning management systems by log data analysis. *Computers in Human Behavior*, 89:367–372, Dec. 2018.

[14] S. Maloney, M. Axelsen, L. Galligan, J. Turner, P. Redmond, A. Brown, M. Basson, and J. Lawrence. Using lms log data to explore student engagement with coursework videos. *Online Learning*, 26(4), Dec. 2022.

[15] S. D. O'Connell and G. Lang. Can personalized nudges improve learning in hybrid classes? experimental evidence from an introductory undergraduate course. *Journal of Research on Technology in Education*, 50(2):105–119, Jan. 2018.

[16] M. A. Rasyadan Reza Putra and M. ER. Media multitasking – does what they say match the reality? a study of its impact on academic performance. In *2024 2nd International Symposium on Information Technology and Digital Innovation (ISITDI)*, pages 59–64. IEEE, July 2024.

[17] F. Sana, T. Weston, and N. J. Cepeda. Laptop multitasking hinders classroom learning for both users and nearby peers. *Computers & Education*, 62:24–31, Mar. 2013.

[18] V. Triglianos, C. Pautasso, A. Bozzon, and C. Hauff. Inferring student attention with ASQ. In *Adaptive and Adaptable Learning*, pages 306–320. Springer International Publishing, 2016.

[19] D. Wang and H. Han. Applying learning analytics dashboards based on process-oriented feedback to improve students' learning effectiveness. *Journal of Computer Assisted Learning*, 37(2):487–499, 2021.

Does Math Reason the Same Across Languages? Prompting Trade–offs in Multilingual Math Support

Louis Colbus
University of Luxembourg
Luxembourg
louis.colbus.001@student.uni.lu

Nina Hosseini-Kivanani*
University of Luxembourg / RTL
Luxembourg
nina.hosseinikivanani@ext.uni.lu

ABSTRACT

Large language models are increasingly used in math learning settings for hint generation, worked–solution support, and automated assessment. In multilingual deployment, however, prompt choice affects not only correctness but also response cost and fairness across learner languages. We study three prompting paradigms, Direct, Chain–of–Thought (CoT), and Program–of–Thought (PoT), across English, German, French, and Luxembourgish on three math benchmarks spanning school to college levels. Using Qwen3–32B, we compare accuracy, token usage, and latency with and without self–consistency. We find that CoT with self–consistency is the strongest default on arithmetic tasks, while PoT with self–consistency becomes more effective on harder problems requiring sustained computation. Across settings, we observe a stable performance hierarchy of EN $>$ DE $\approx$ FR $>$ LB, with LB also incurring the largest token and latency overhead. These results show that prompt configuration is a practical design decision for multilingual math support systems, with direct implications for reliability, efficiency, and cross–lingual fairness.

Keywords

Large language models, Mathematical reasoning, Cross–lingual evaluation, Multilingual math support, Chain–of–Thought, Program–of–Thought

1. INTRODUCTION

Large language models (LLMs) are increasingly integrated into educational mathematics workflows, including hint generation, worked–solution support, answer validation, and teacher–facing verification. In multilingual settings, such systems must provide support that is not only accurate, but also timely, interpretable, and equitable across learner languages. Yet prompt design is often treated as a purely technical implementation choice, even though it directly affects the quality, transparency, cost, and latency of learner–facing support [6, 18].

Prior work shows that prompting strategies such as Chain–of–Thought (CoT), self–consistency, and Program–of–Thought (PoT) can substantially affect mathematical reasoning performance [20, 19, 2, 4]. CoT elicits intermediate reasoning steps before the final answer, which can improve solution quality and make the response more interpretable. Self–consistency improves robustness by aggregating multiple sampled solutions through voting, although it also increases token usage and latency [19]. Program–based approaches such as PoT or PAL delegate numerical computation to executable code, which is often beneficial on harder mathematical problems that require sustained symbolic or numerical manipulation [2, 4]. These strategies therefore differ not only in correctness, but also in cost, response time, and pedagogical transparency.

Educational research has also begun to explore LLMs as components of tutoring, explanation generation, automated feedback, and assessment workflows [21, 1, 24]. In such settings, evaluation must go beyond raw correctness. A system that is accurate but slow may be unsuitable for interactive classroom use, while a system that produces correct but opaque answers may be less useful for learning support than one that provides interpretable intermediate reasoning. In multilingual educational deployment, these trade–offs become even more important because differences in response quality or speed across languages can translate into unequal learner support.

Multilingual mathematical reasoning remains uneven across languages. Benchmarks such as MGSM show that LLMs can solve math problems in multiple languages, but performance often degrades outside English and other high–resource settings [14]. More recent work similarly reports sensitivity to language resource level, translation quality, and prompting setup [12, 15]. Recent work also questions whether single–set math benchmark accuracy is sufficient for evaluating reasoning, showing that LLM performance can vary across controlled re–instantiations of the same GSM8K–style problems and can deteriorate when irrelevant but seemingly relevant clauses are added [10].

These issues are especially relevant for lower–resource languages, where weaker pretraining coverage may affect both answer quality and reasoning stability. However, prior work

*Corresponding author: nina.hosseinikivanani@ext.uni.lu

Louis Alfred Antoine Colbus, Christoph Schommer, and Nina Hosseini-Kivanani. Does Math Reason the Same Across Languages? Prompting Trade-offs in Multilingual Math Support. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 647–652. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040085

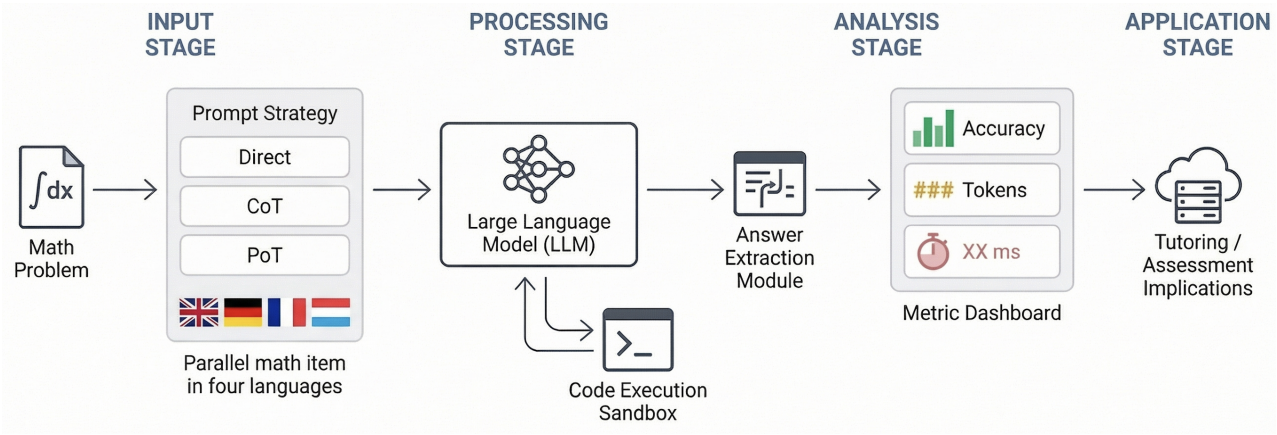


Figure 1: Pipeline of multilingual math support evaluation.

has focused mainly on per–language accuracy and has paid less attention to how prompt choice changes the trade–offs among correctness, token cost, latency, and interpretability in multilingual educational use.

These benchmarks are not only abstract tests of model reasoning. They also approximate common functions in learner–facing systems. Arithmetic problems are closely related to stepwise hinting, answer checking, and formative support in school mathematics, while harder problems reflect advanced feedback, verification, and teacher–facing solution analysis. Under this view, prompt choice is not merely an inference–time optimization detail. It changes the quality, speed, and interpretability of support that multilingual tutoring and assessment systems provide to different learner groups.

We address this gap through a compact empirical study across four languages, English (EN), German (DE), French (FR), and Luxembourgish (LB), using three prompting paradigms: Direct, CoT, and PoT. We evaluate performance on arithmetic and harder mathematics benchmarks and analyze both answer quality and computational cost, with particular attention to LB as a low–resource language relevant to multilingual educational deployment.

These trade–offs matter directly for educational use. In tutoring and hint generation, prompt choice affects whether learners receive interpretable step–by–step guidance or only a final answer. In automated checking, latency and computational cost affect classroom usability and scalability. In multilingual settings, unequal performance across languages raises fairness concerns because students using lower–resource languages may receive slower or less reliable support. Our study therefore treats prompting as a design choice for multilingual tutoring, feedback, and assessment systems, not only as an inference–time optimization problem. Our contributions are:

- We provide a focused evaluation of multilingual math support performance across EN, DE, FR, and LB, including a low–resource language.
- We show that the most suitable prompting strategy depends on task difficulty. CoT with self–consistency is strongest on arithmetic tasks, while PoT with self– consistency is more effective on harder mathematics.
- We show that lower–resource language support can be both less accurate and more computationally expensive, which makes prompt configuration a fairness and deployment issue for tutoring and assessment systems.

2. METHODOLOGY

2.1 Benchmarks and languages

We evaluate on three math benchmarks covering different difficulty levels: GSM8K [3], SVAMP [11], and CollegeMath [16]. GSM8K and SVAMP emphasize arithmetic word problems, while CollegeMath contains advanced textbook–style mathematics. We use EN test sets and translate them into DE, FR, and LB using DeepL, preserving mathematical expressions verbatim and applying light post–editing. Manual spot–checking was used to verify that translated items preserved problem structure, numerical references, and solution equivalence across languages. This step helps reduce the risk that observed cross–lingual differences are caused by obvious translation artifacts rather than by differences in model behavior. Table 1 shows a parallel SVAMP item with the original EN problem and its FR, DE, and LB translations.

These benchmarks also serve as controlled proxies for multilingual educational use cases, including hint generation, worked–solution support, answer validation, and teacher–facing checking or verification. To support reproducibility and facilitate further experimentation, the translated benchmark files as well as the full code used for the evaluation pipeline are publicly available on the accompanying repository[1].

2.2 Model and prompting

We use Qwen3–32B [22], a strong multilingual open–weight model with established mathematical reasoning ability, making it suitable for controlled comparison across prompt styles and languages. It is a 32B dense model trained on approximately 36T tokens across 119 languages and dialects, and

[1]Project repository: multilingual-math-bench.

Table 1: Parallel SVAMP example with the original English item and its French, German, and Luxembourgish translations.

Lang.	Problem text
EN	Debby bought 200 water bottles and 256 soda bottles when they were on sale. If she drank 312 water bottles and 4 soda bottles a day, how many days would the soda bottles last?
FR	Debby a acheté 200 bouteilles d'eau et 256 bouteilles de soda lorsqu'elles étaient en promotion. Si elle buvait 312 bouteilles d'eau et 4 bouteilles de soda par jour, combien de jours les bouteilles de soda dureraient–elles ?
DE	Debby hat 200 Wasserflaschen und 256 Limonadenflaschen gekauft, als diese im Angebot waren. Wenn sie täglich 312 Wasserflaschen und 4 Limonadenflaschen trinkt, wie viele Tage reichen die Limonadenflaschen dann?
LB	Debby huet 200 Waasserfläschen a 256 Soda–Fläschen kaaft, wéi se am Rabatt waren. Wann si all Dag 312 Waasserfläschen a 4 Soda–Fläschen drénkt, wéi vill Deeg géifen d'Soda–Fläschen duergoen?

includes a native thinking mode for internal reasoning [22]. We evaluate it across three benchmarks, four languages, and three prompting strategies in a multilingual math–support setting. Figure 1 summarizes the evaluation pipeline.

For each problem, we compare three prompting modes. Direct prompting requests only the final answer. CoT prompting elicits intermediate reasoning steps before the answer. PoT prompting generates executable Python code whose returned output is treated as the final answer. For each strategy, we evaluate both single–sample decoding and self–consistency with five samples (SC@5). In the SC@5 setting, we sample five independent outputs per item and aggregate their final answers by majority vote. This design allows us to compare not only prompting styles, but also the trade–off between single–pass efficiency and ensemble–style robustness.

2.3 Decoding and evaluation

We use the same prompting setup across all four languages and evaluate both single–sample decoding and self–consistency with five samples (SC@5). We use a fixed temperature of $T = 1.0$ across prompting strategies to encourage output diversity for self–consistency, while leaving other decoding parameters at their default settings.

For each dataset and language, we evaluate the full test split under both single–sample and SC@5 settings. For each item, final answers are extracted with a uniform post–processing pipeline that normalizes formatting differences across languages and prompting styles. Predictions are counted as correct when the extracted answer matches the reference after simple normalization, such as removing whitespace or formatting wrappers. For arithmetic expressions, we additionally allow equivalence under basic symbolic normalization. For PoT, only the printed final output is evaluated, and executions that fail to produce a parsable answer are counted as incorrect. We record problem–level accuracy, average generated tokens, and wall–clock latency. This setup allows us to compare not only answer quality, but also the deployment trade–off between efficiency and robustness in multilingual educational settings.

We report problem–level accuracy, average generated tokens, and wall–clock latency. These metrics are relevant for multilingual educational deployment, where systems must balance answer quality with response cost and speed. Accuracy captures answer quality, generated tokens approximate inference cost, and latency reflects whether feedback can be delivered quickly enough for interactive use.

3. RESULTS AND DISCUSSION

Table 2 summarizes the main accuracy patterns. Across prompting modes and datasets, self–consistency generally improves over single–sample decoding, with larger gains on harder tasks and for some prompting styles. Two findings stand out. First, the best prompting strategy depends on task difficulty: CoT with self–consistency is strongest on arithmetic tasks, while PoT with self–consistency becomes the best–performing option on harder mathematics. Second, performance is consistently uneven across languages: EN performs best, DE and FR are usually close behind, and LB shows the largest degradation.

Table 2: Accuracy (%) for Qwen3–32B across languages. Highlighted rows indicate the best–performing method.

Dataset	Method	EN	DE	FR	LB
GSM8K	Direct	78.1	64.8	66.9	33.5
	CoT SC@5	**87.5**	**71.2**	**69.1**	**51.5**
	PoT SC@5	69.3	70.3	68.5	44.7
SVAMP	Direct	81.6	72.2	74.3	44.2
	CoT SC@5	**87.0**	**76.9**	**74.9**	**68.6**
	PoT SC@5	66.2	73.9	71.9	64.6
CollegeMath	Direct	41.6	41.0	38.4	31.1
	CoT SC@5	38.2	40.0	39.9	35.6
	PoT SC@5	**56.6**	**48.7**	**51.9**	**42.1**

For school–level arithmetic, CoT with self–consistency offers the best balance of performance and interpretability. On GSM8K, for example, CoT SC@5 improves EN accuracy from 78.1% with Direct prompting to 87.5%, and similar gains appear on SVAMP. This pattern is consistent with prior work showing that self–consistent CoT can recover diverse but correct reasoning paths [19]. For learner–facing support, CoT offers an additional advantage: beyond improving answer quality, it produces explicit intermediate reasoning that can be repurposed as hints or worked examples.

PoT behaves differently on easier benchmarks. On GSM8K for Qwen3–32B, PoT can underperform because the model sometimes fails to follow the Python instructions or produces code with syntax errors, which the execution environment counts as incorrect. This suggests that the underlying solution strategy may be reasonable, but brittle at the implementation level, in line with prior observations for PAL and related code–based methods [2, 7].

A qualitative inspection of failed outputs suggests that er-

Table 3: Observed trade–offs of prompting strategies for multilingual math support.

Method	Observed strength	Observed constraint	Deployment relevance
Direct	Computationally efficient and structurally simple.	Does not provide explicit intermediate reasoning.	Appropriate when answer delivery and low overhead are prioritized.
CoT + SC@5	Strongest arithmetic performance, with interpretable reasoning traces.	Increased token usage and response latency.	Appropriate for explanatory support, including hints and worked examples.
PoT + SC@5	Strongest performance on harder tasks requiring sustained computation.	Depends on stable code generation and execution.	Appropriate for advanced verification and more formal solution support.

rors arise from several sources. For Direct and CoT prompting, failures often involve incorrect arithmetic, incomplete final–answer extraction, or reasoning paths that are plausible but numerically inconsistent. For PoT, errors are more implementation–sensitive: some outputs contain syntactically invalid Python code, missing print statements, or code that computes an intermediate quantity rather than the requested final answer. In the translated settings, some failures also appear to be linked to language–specific phrasing or answer–format variation. These observations are not intended as a full causal error taxonomy, but they indicate that multilingual math–support failures involve both reasoning errors and interface–level issues such as parsing, translation, and code execution.

On harder problems, the pattern changes. For CollegeMath, PoT SC@5 outperforms Direct and CoT in EN and remains strongest in the other languages. This suggests that delegating computation to executable code becomes useful when tasks require sustained symbolic or numerical manipulation, in line with prior work on program–aided mathematical reasoning [2, 4]. In educational settings, this makes PoT suited to advanced support scenarios such as automated checking, teacher–facing verification tools, or systems that generate formal worked solutions for difficult exercises.

Table 3 summarizes these observed trade–offs and links each prompting strategy to its most relevant educational deployment scenario.

No single prompting strategy is optimal across all forms of math support. CoT is strongest when interpretable step–by–step guidance is valuable, whereas PoT becomes preferable when computational precision dominates. This matters for educational applications, where the same system is expected to support both hint generation and answer validation.

For educational applications, this distinction is consequential. If a system is primarily used to generate hints or worked examples, CoT may be preferable because it exposes intermediate reasoning that can be shown to learners or adapted by teachers. If the system is used for automated checking or higher–stakes verification, PoT may be preferable on harder tasks because computational reliability becomes more important than explanatory detail. In multilingual settings, these design choices also interact with equity, since learners using lower–resource languages may receive support that differs not only in quality but also in responsiveness.

Across all settings we observe a consistent hierarchy

$$\text{EN} > \text{DE} \approx \text{FR} > \text{LB},$$

where German and French typically retain much of the English baseline, while Luxembourgish shows larger drops, especially on the harder benchmark. This pattern aligns with multilingual math evaluations that report stronger performance for high–resource languages and larger degradation for lower–resource ones [14, 15, 12].

Efficiency also differs by language. Figure 2 shows that LB requires more generated tokens than EN across prompting strategies, with especially large increases under self–consistency. The token gap is also consistent across methods, which suggests that the efficiency disadvantage is not tied to a single prompting style. In our runs, this also translated into higher latency. For example, under single–sample CoT on GSM8K, average latency increased from approximately 5.7 seconds in EN to 7.6 seconds in LB. Thus, lower–resource language support is disadvantaged twice: it is often less accurate and more expensive, which aligns with prior multilingual mathematical reasoning studies reporting stronger performance in high–resource languages and larger drops in lower–resource settings [14, 12, 15]. For multilingual tutoring or classroom–facing systems, this matters because learners using lower–resource languages may receive slower and less reliable support, even when the underlying task is the same.

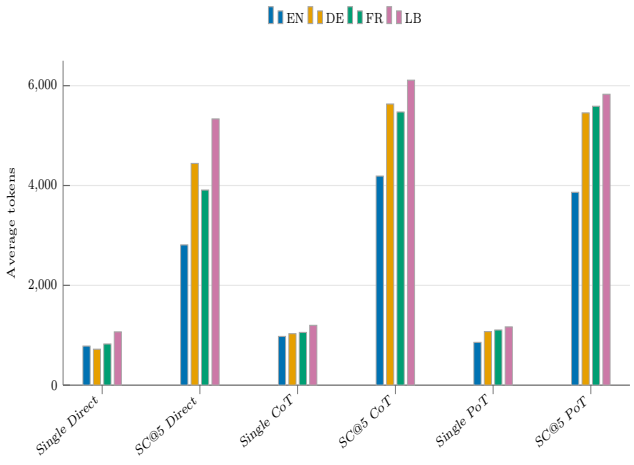


Figure 2: Average token usage across prompting strategies and languages. Self–consistency (SC@5) increases token cost, and LB is consistently among the most expensive settings.

Our findings have three implications for educational data mining and multilingual math support systems: **First**, prompt choice should be treated as a deployment decision rather than a neutral implementation detail. For arithmetic tutoring, CoT with self–consistency provides a strong default because it improves accuracy while preserving interpretable reasoning traces that can support hints, worked examples, or feedback generation [9]. By contrast, Direct prompting may be cheaper, but it offers less transparency and may therefore be less suitable when the instructional goal is to help learners understand the solution process rather than only obtain the correct answer. In education, the value of a response depends not only on correctness but also on whether the reasoning process is understandable and usable. This makes interpretability a functional requirement rather than an optional property when the system is intended to support learning rather than only scoring.

Second, the best strategy changes with task difficulty. For harder mathematics, PoT with self–consistency becomes more reliable than CoT, suggesting that external code execution can stabilize multi–step symbolic or numerical reasoning [23, 8]. This is especially relevant for advanced exercises, automated checking, or teacher–facing systems that must verify mathematically precise solution procedures. In other words, a single prompting strategy is unlikely to be optimal across all educational uses of LLM–based math support.

Third, multilingual support shows clear operational disparities across languages. The consistent hierarchy EN $>$ DE $\approx$ FR $>$ LB indicates that a single prompt configuration may offer uneven quality across learner languages, although this study does not isolate the causal source of the gap. The higher token and latency burden for LB suggests that multilingual systems should use language–specific calibration and fairness auditing rather than assuming uniform performance [5, 17].

This cross–lingual gap has consequences for multilingual educational deployment. A student asking for support in a lower–resource language may receive feedback that is not only less accurate, but also slower and more expensive to generate. This creates an equity issue for tutoring and assessment systems serving linguistically diverse learners. Multilingual math support should therefore be evaluated not only for average correctness, but also for consistency of support quality across learner languages, alongside broader trade–offs between reasoning quality and computational efficiency [13].

4. LIMITATIONS AND CONCLUSION

This study is benchmark–based and does not evaluate learners, teachers, classroom interaction, or learning outcomes directly. The multilingual setup also relies on translated benchmarks, so residual translation or answer–extraction artifacts may affect results, and EN performance may benefit from stronger pretraining coverage or benchmark familiarity. This caution is reinforced by recent work showing that performance on GSM8K–style tasks can vary across controlled re–instantiations and small problem modifications, suggesting that single benchmark scores may not fully capture reasoning robustness [10]. Our scope is also limited to one multilingual model, four languages, and a small set of prompting strategies, so the exact trade–offs may differ for other models or language settings.

Even with these limitations, the main patterns are consistent. CoT with self–consistency is the strongest default for arithmetic support, whereas PoT with self–consistency is more effective for harder problems that benefit from executable reasoning. Across languages, lower–resource settings such as LB show both accuracy degradation and higher efficiency costs. For multilingual math support systems, prompt configuration is therefore a question of reliability, cost, and fairness. More broadly, evaluating multilingual LLMs for mathematics is not only a question of benchmark performance, but also of how reliably and equitably support can be delivered across learner languages.

5. ETHICS STATEMENT

This study does not involve the collection of new human–subject data, learner interaction data, classroom data, or personal information. All experiments are conducted on existing mathematical benchmark datasets and their translated versions. No personally identifiable information is processed, and no participants are recruited. Therefore, the study does not raise human–subject privacy risks. The work is intended as a benchmark–based analysis of multilingual prompting behavior, not as an evaluation of individual learners or teachers. Because no new human–subject data were collected, local ethics–board approval was not required under the scope of this study.

6. ACKNOWLEDGMENTS

This research was conducted in the context of the LuxVoice project, funded by the Luxembourg National Research Fund (FNR) under grant agreement (project reference 19205922). LuxVoice aims to advance Luxembourgish language technologies and support the development of robust multilingual AI resources. The authors gratefully acknowledge the support of RTL Lëtzebuerg and the University of Luxembourg. This work contributes to broader efforts toward inclusive, reliable, and linguistically grounded AI systems for Luxembourgish and other low–resource language settings.

7. REFERENCES

[1] C. Borchers and T. Shou. Can large language models match tutoring system adaptivity? a benchmarking study. In *International Conference on Artificial Intelligence in Education*, pages 407–420. Springer, 2025.

[2] W. Chen, X. Ma, X. Wang, and W. W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*, 2022.

[3] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[4] L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig. Pal: Program-aided language models. In *International Conference on Machine Learning*, pages 10764–10799. PMLR, 2023.

[5] A. Gupta, J. Reddig, T. Calo, D. Weitekamp, and C. J. MacLellan. Beyond final answers: Evaluating large language models for math tutoring. In *International Conference on Artificial Intelligence in Education*, pages 323–337. Springer, 2025.

[6] V. Gupta, S. P. Chowdhury, V. Zouhar, D. Rooein, and M. Sachan. Are large language models for education reliable across languages? In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 612–631. Association for Computational Linguistics, 2025.

[7] S. Imani, L. Du, and H. Shrivastava. Mathprompter: Mathematical reasoning using large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 37–42, 2023.

[8] Z. Jie, T. Q. Luong, X. Zhang, X. Jin, and H. Li. Design of chain-of-thought in math problem solving. *arXiv preprint arXiv:2309.11054*, 2023.

[9] Z. Liang, Y. Liu, T. Niu, X. Zhang, Y. Zhou, and S. Yavuz. Improving llm reasoning through scaling inference computation with collaborative verification. *arXiv preprint arXiv:2410.05318*, 2024.

[10] I. Mirzadeh, K. Alizadeh, H. Shahrokhi, O. Tuzel, S. Bengio, and M. Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*, 2024.

[11] A. Patel, S. Bhattamishra, and N. Goyal. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, 2021.

[12] B. Paul, J. J. Era, M. R. Zim, T. S. Aothoi, and F. M. Shah. Leveraging large language models for bengali math word problem solving with chain of thought reasoning. *arXiv preprint arXiv:2505.21354*, 2025.

[13] K. Seßler, Y. Rong, E. Gözlüklü, and E. Kasneci. Benchmarking large language models for math reasoning tasks. *arXiv preprint arXiv:2408.10839*, 2024.

[14] F. Shi, M. Suzgun, M. Freitag, X. Wang, S. Srivats, S. Vosoughi, H. W. Chung, Y. Tay, S. Ruder, D. Zhou, et al. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*, 2022.

[15] M. E. Sobhani, M. F. A. Sayeedi, T. Mohiuddin, M. M. Islam, and S. Shatabda. Mathmist: A parallel multilingual benchmark dataset for mathematical problem solving and reasoning. *arXiv preprint arXiv:2510.14305*, 2025.

[16] Z. Tang, X. Zhang, B. Wang, and F. Wei. Mathscale: Scaling instruction tuning for mathematical reasoning. In *International Conference on Machine Learning*, pages 47885–47900. PMLR, 2024.

[17] J. C. Tonga, K. Srivatsa, K. K. Maurya, F. Koto, and E. Kochmar. Simulating llm-to-llm tutoring for multilingual math feedback. *arXiv preprint arXiv:2506.04920*, 2025.

[18] S. Wang, T. Xu, H. Li, C. Zhang, J. Liang, J. Tang, P. S. Yu, and Q. Wen. Large language models for education: A survey and outlook. *IEEE Signal Processing Magazine*, 42(6):51–63, 2026.

[19] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[20] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[21] D. Weitekamp, M. N. Siddiqui, and C. J. MacLellan. Tutorgym: A testbed for evaluating ai agents as tutors and students. In *International Conference on Artificial Intelligence in Education*, pages 361–376. Springer, 2025.

[22] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[23] H. Ying, S. Zhang, L. Li, Z. Zhou, Y. Shao, Z. Fei, Y. Ma, J. Hong, K. Liu, Z. Wang, et al. Internlm-math: Open math large language models toward verifiable reasoning. *arXiv preprint arXiv:2402.06332*, 2024.

[24] L. Zhang, J. Lin, Z. Kuang, S. Xu, and X. Hu. Spl: a socratic playground for learning powered by large language model. *arXiv preprint arXiv:2406.13919*, 2024.

Enhanced Student Engagement through Supervised Assessment: Evidence from Kenya

Pedro Freitas
Blavatnik School of Government, University of Oxford
pedro.freitas@bsg.ox.ac.uk

Guillermo Romero
Blavatnik School of Government, University of Oxford
guillermo.romero@bsg.ox.ac.uk

Amar Lalwani
EIDU
amar.lalwani@eidu.com

Aidan Friedberg
EIDU
aidan.friedberg@eidu.com

Noam Angrist
Blavatnik School of Government, University of Oxford
noam.angrist@bsg.ox.ac.uk

Sharnic Djaker
Blavatnik School of Government, University of Oxford
sharnic.djaker@bsg.ox.ac.uk

ABSTRACT

In-app assessments are increasingly used by EdTech programs to measure students' proficiency and guide instructional decisions. Student engagement in these assessments is crucial for generating reliable learning signals; however, efforts to increase student engagement in platform-based assessments often impose additional demands on teachers. To explore an optimal assessment design, we implemented supervised assessments in which a teacher is present while students complete the assessment. We find that supervised assessments significantly increase student engagement, as evidenced by longer average assessment durations and a substantial reduction in timed-out or aborted sessions. At the same time, the number of classes participating in the assessment decreases slightly. These findings highlight a trade-off between assessment quality and participation rates, and suggest that supervision improves data quality primarily through increased student engagement.

Keywords

In-app assessment, Supervised assessment, Student proficiency, Low- and middle-income countries

1. INTRODUCTION

Assessment results constitute a key source of information for understanding the evolution of students' learning. Students and their families respond to feedback and information generated by reliable assessments [3, 4, 5].

Assessment data are also critical for teachers, as they provide information on students' learning trajectories and can inform instructional practices. A growing body of literature examines how teachers use assessment data to guide pedagogical decisions and improve student learning outcomes [10, 9, 12, 13]. This evidence suggests that increased access to assessment information can improve instructional targeting, goal setting, and teachers' self-efficacy [6]. However, there remains limited empirical evidence on how teachers use data generated by information and communication technology (ICT) systems and how such data shape instructional practices in large-scale educational settings [7].

The usefulness of assessment data depends fundamentally on student engagement during assessment activities [8, 14]. In-platform based environments, disengagement — such as rapid guessing, non-response, or aborted sessions — can generate noisy signals about student proficiency. These issues are especially salient in unsupervised digital assessments that are completed without active teacher oversight. Proctoring offers a potential mechanism to address these challenges by introducing teacher supervision during assessments, thereby increasing student engagement and completion. However, teacher supervision also increases the effort required from teachers, which may discourage participation in more demanding assessment formats. Existing evidence on the benefits of proctoring primarily comes from higher education and focuses largely on cheating behavior in online assessments [15, 2], leaving open questions about its role in shaping engagement and data quality in large-scale primary education systems, particularly in low- and middle-income country contexts.

In this paper, we study the effects of introducing supervised assessments on a national-scale EdTech platform operating in Kenya. In September 2025, the platform tested supervised assessments in place of unsupervised assessments, with teachers proctoring students during assessment administration. This proctoring is mostly observational and similar across students in different classes and schools. Using platform data covering more than 300,000 students across approximately 10,000 schools, we examine how teacher supervision influences student engagement and participation in digital assessments.

Amar Lalwani, Aidan Frieberg, Pedro Freitas, Guillermo Romero, Sharnic Djaker, and Noam Angrist. Enhanced Student Engagement through Supervised Assessment: Evidence from Kenya. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 653–656. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040121

2. INTERVENTION

EIDU is a digital education platform operating at scale in Kenya. The platform provides interactive learning content in numeracy and literacy through Android devices, with games-based content units adapted to students' individual learning pace [11, 1].

Prior to September 2025, the platform administered assessments in an unsupervised format, where assessment tasks were automatically launched on the device without active teacher oversight and learners could abort the assessments at any point of time. In September 2025, EIDU introduced and tested a supervised assessment format, in which a teacher is present and actively proctors students as they complete the assessment. During the assessment teachers observe learners without providing any feedback or scaffolding. Examples of digital assessment items of letter recognition and number identification can be found in Figures 1 and 2.

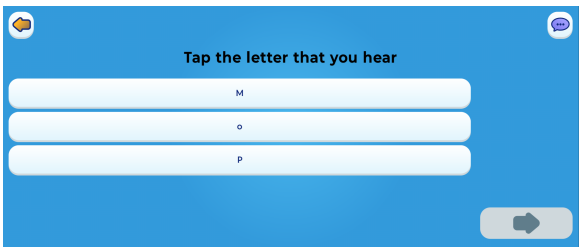


Figure 1: Letter Recognition - Digital assessment

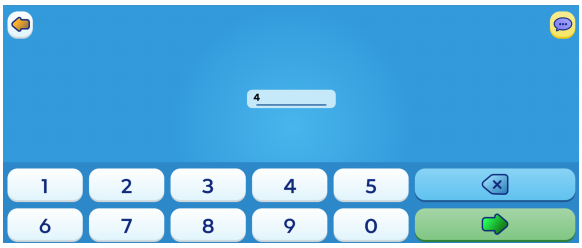


Figure 2: Number identification - Digital assessment

We examine how this introduction of teacher supervision affects student engagement and participation in digital assessments. We measure these outcomes using three school-level indicators: (i) the average time students spent on the assessment; (ii) the share of assessment sessions that were aborted or timed out; and (iii) the average number of classes per school participating in the assessment. These three indicators constitute the main outcome variables of our analysis. We additionally descriptively show how the change from unsupervised to supervised assessments changed the distribution of students' scores.

This large-scale evaluation was conducted in close coordination with county-level education authorities, who approved and monitored the implementation of the intervention. All data were anonymized, and access was restricted to researchers involved in the project.

3. DATA AND METHODS

The data covers the period from January to November 2025 and include 310,363 5-year old learners across 10,322 schools and 25,130 classes. Students are assessed in both Reading and Mathematics, completing 11 items in letter recognition and 12 number identification per assessment moment. All outcomes are aggregated at the school-level.

To examine the effects of supervised assessments, we use a pre-post research design that compares outcomes before and after the introduction of teacher supervision. The pre-intervention period is coded as $t = 0$ and the post-intervention period as $t = 1$. The post period corresponds to the introduction of supervised assessments in September 2025, when supervised assessments were introduced and tested in the platform.

For each school s and time period $t \in \{0, 1\}$, we estimate the following specification:

$$Y_{st} = \alpha + \beta S_t + \varepsilon_{st}, \tag{1}$$

where Y_{st} denotes the outcome of interest and S_t is a dummy variable equal to 1 in the post-intervention period (supervised assessments) and 0 in the pre-intervention period (unsupervised assessments). The coefficient β captures the average change in school-level outcomes following the introduction of teacher supervision. ε_{st} stands for the error term.

4. RESULTS

Tables 1–3 present the main results. In each table, Column (1) reports results for the full sample. Column (2) restricts the sample to one month before and after the introduction of supervised assessments to account for potential learning trajectory effects over time. Column (3) of Table 1 further restricts the sample to completed assessments. In addition, we report the average level for each outcome for unsupervised assessments.

Tables 1 and 2 indicate a higher level of engagement among the students who participated in the assessments. The introduction of supervised assessments is associated with increases in average assessment duration ranging from 0.09 to 0.26 minutes. This corresponds to an increase between 9% and 25% compared with unsupervised assessments. We also find a large reduction in the share of sessions that were aborted or timed out. The latter effect is particularly pronounced, with aborted or timed-out sessions declining by approximately 30 percentage points following the introduction of supervision. These patterns suggest that teacher proctoring substantially increases the likelihood that students remain engaged through assessment completion.

At the same time, supervised assessments are associated with a modest reduction in participation across classes. Table 3 shows that the average number of classes per school participating in assessments declined by approximately 0.3 classes following the introduction of supervision. This result suggests increased selectivity in class participation when assessments are administered under teacher supervision.

The distribution of assessment scores also changes following the introduction of supervision. We compute the share of correct items for each student assessment and then aggre-

Table 1: Effects on Mean duration (in minutes)

	Mean duration		
	(1)	(2)	(3)
Supervised	0.259***	0.232***	0.089***
	(0.0371)	(0.0372)	(0.00925)
Av. Unsupervised		1.03	
1 Month before-after	NO	YES	NO
Only completed	NO	NO	YES
Observations	16,911	11,647	16,911
R-squared	0.004	0.003	0.007

Notes: The table presents school-level OLS regression results with mean duration as the dependent variable. The omitted category corresponds to non-supervised assessments. Standard errors clustered at the school level are reported in parentheses.*** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

Table 2: Effects on the Share of sessions aborted

	%	
	(1)	(2)
Supervised	-0.346***	-0.330***
	(0.00380)	(0.00430)
Av. Unsupervised	0.61	
1 Month before-after	NO	YES
Observations	16,911	11,647
R-squared	0.331	0.283

Notes: School-level OLS regression results with the share of aborted/timed-out sessions as the dependent variable. Standard errors clustered at the school level are in parentheses.*** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

Table 3: Effects on Number of classes engaged

	Number of classes	
	(1)	(2)
Supervised	-0.293***	-0.272***
	(0.00778)	(0.0113)
Av. Unsupervised	1.23	
1 Month before-after	NO	YES
Observations	16,911	11,647
R-squared	0.034	0.038

Notes: School-level OLS regression results with the number of classes engaged as the dependent variable. Standard errors clustered at the school level are in parentheses.*** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

gate these measures at the school level. As shown in Figure 3, unsupervised assessments exhibit a pronounced mass of scores at zero, indicative of low student engagement and high levels of non-response. In contrast, supervised assessments display a markedly different score distribution, characterized by higher average scores and the absence of zero inflation, consistent with increased participation among participating students. As observed in Table 3, the introduction of supervised assessments led to a decline in the share of classes per school participating in the assessments. To control for this compositional effect, Figure 4 includes only classes that participated in both assessments. The figure shows a similar pattern in the score distributions for supervised and unsupervised assessments.

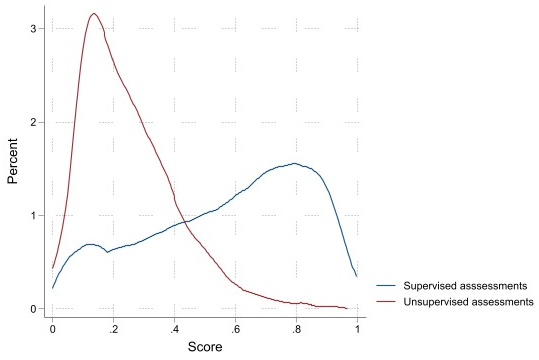


Figure 3: Score distribution, school level (scale: 0–1)

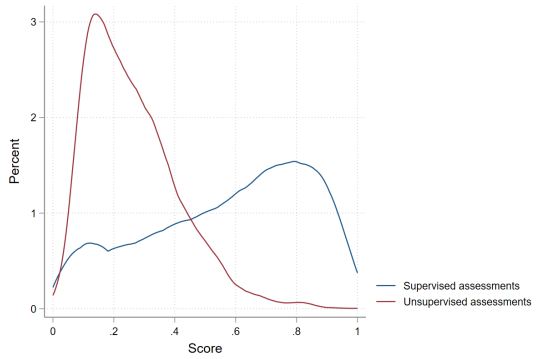


Figure 4: Score distribution for classes observed in supervised and unsupervised assessments, school level (scale: 0–1)

5. CONCLUSION

Accurate student assessment data is key to understanding students' progress and informing instructional decisions across educational contexts. The value of such data depends not only on assessment content, but also on how assessments are administered and the extent to which students are engaged during assessment activities.

As the EdTech sector continues to expand, digital assessments are increasingly embedded alongside instructional materials and used to guide personalization at scale. In this

setting, ensuring meaningful student engagement during assessment becomes critical for generating reliable learning signals. To address this challenge, EIDU introduced supervised assessments, in which teachers proctor students during assessment administration. While supervision has the potential to improve data quality, it also increases the burden on teachers.

Our results show that students engage more with supervised assessments, as measured by the increase in average time spent answering items and the strong decrease in the average share of sessions that were aborted or timed out. However, we also observe a slight decrease in the average number of classes participating in assessment, consistent with increased burden for some teachers. These findings highlight a tradeoff between assessment engagement and participation when supervision is introduced at scale.

Future work should explore in-product solutions that preserve the engagement benefits of supervision while reducing participation-related burden, further enabling the collection of higher-quality digital assessment data.

References

[1] P. Agartsupa, T. Chindanurak, and D. Suwanjinda. The effects of blended learning management in earth science on learning achievement and digital skills of grade 4 students at ekamai international school. *Human Behaviour, Development and Society*, 26, 2025.

[2] H. M. Alessio, N. Malay, K. Maurer, A. J. Bailer, and B. Rubin. Examining the effect of proctoring on online test scores. *Online Learning Journal*, 21, 2017.

[3] G. Azmat and N. Iriberri. The provision of relative performance feedback: An analysis of performance and satisfaction. *Journal of Economics and Management Strategy*, 25:77–110, 2016.

[4] M. Bobba and V. Frisancho. Learning about oneself:the effects of performance feedback on school choice. *SSRN Electronic Journal*, 2017.

[5] R. Brade, O. Himmler, and R. Jäckle. No student left behind? relative feedback and university completion. *Journal of Economic Behavior and Organization*, 241, 2026.

[6] R. V. Gasse, K. Vanlommel, J. Vanhoof, and P. V. Petegem. The impact of collaboration on teachers' individual data use. *School Effectiveness and School Improvement*, 28, 2017.

[7] J. Luo, M. Wang, and S. Yu. Exploring the factors influencing teachers' instructional data use with electronic data systems. *Computers and Education*, 191, 2022.

[8] C. Penk, C. Pöhlmann, and A. Roppelt. The role of test-taking motivation for students' performance in low-stakes assessments: an investigation of school-track-specific differences. *Large-Scale Assessments in Education*, 2, 2014.

[9] K. Schildkamp and W. Kuiper. Data-informed curriculum reform: Which data, what purposes, and promoting and hindering factors. *Teaching and Teacher Education*, 26, 2010.

[10] K. Schildkamp, C. Poortman, H. Luyten, and J. Ebbeler. Factors promoting and hindering data-based decision making in schools. *School Effectiveness and School Improvement*, 28, 2017.

[11] C. Sun, L. Major, R. Daltry, and N. Moustafa. Teacher-ai collaboration in content recommendation for digital personalised learning among pre-primary learners in kenya. In *L@S 2024 - Proceedings of the 11th ACM Conference on Learning @ Scale*, 2024.

[12] P. Wohlstetter, A. Datnow, and V. Park. Creating a system for data-driven decision-making: Applying the principal-agent framework. *School Effectiveness and School Improvement*, 19, 2008.

[13] V. M. Young. Teachers' use of data: Loose coupling, agenda setting, and team norms. *American Journal of Education*, 112, 2006.

[14] G. Zamarro, C. Hitt, and I. Mendez. When students don't care: Reexamining international differences in achievement and student effort. *Journal of Human Capital*, 13, 2019.

[15] L. Zhao, J. Peng, S. Ke, and K. Lee. Effectiveness of unproctored vs. teacher-proctored exams in reducing students' cheating: A double-blind randomized controlled field experimental study. *Educational Psychology Review*, 36, 2024.

Enhancing Diversity of LLM-Generated Educational Tasks

Manh Hung Nguyen
MPI-SWS
manguyen@mpi-sws.org

Sebastian Tschiatschek
University of Vienna
sebastian.tschiatschek@univie.ac.at

Adish Singla
MPI-SWS
adishs@mpi-sws.org

ABSTRACT

Large language models (LLMs) have shown the potential for generating educational content at scale, assisting educators in creating practice tasks or synthesizing data for training educational models. However, LLMs suffer from the "Artificial Hivemind" effect, where they produce homogeneous content. This homogeneity limits the diversity of LLM-generated tasks, a crucial factor in these educational settings. In this paper, we investigate how to increase the diversity of generated tasks while keeping their utility high. Inspired by the divergent–convergent thinking stages in creativity literature, we propose a prompting framework with two reasoning stages: (1) exploring the creative space, and (2) satisfying the input requirements. We evaluate CreativeDC, a method instantiated from this framework in the domain of Python programming, using both automated metrics and expert evaluation. Results show that CreativeDC produces significantly more distinct high-utility tasks (about 1.6×) than baselines. Our work offers an effective approach for generating and evaluating more diverse tasks at scale.

Keywords

Large Language Models, Task Generation, Programming Education, Diversity, Divergent–Convergent Thinking

1. INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable capabilities, leading to their rapid adoption across important domains such as education [1–14]. In particular, prior work has used LLMs to generate programming tasks [15–18], enabling active learning strategies such as the "doer effect" through task generation at scale [19, 20]. Moreover, LLMs can also be used to generate synthetic data, for example, for training tutor models [12, 14, 21]. In such settings, maintaining task diversity is crucial to adequately cover learning objectives and different difficulty levels. Yet LLMs have been shown to exhibit an "Artificial Hivemind" effect [22], where they tend to produce similar and repetitive outputs within a single model and across different models, resulting in limited output diversity.

Existing work has proposed various methods to address the limited diversity and homogeneity of LLM outputs. These include fine-tuning through preference optimization with creativity signals [23] and using multi-agent discussion or debate frameworks [24]. However, these approaches are typically designed and evaluated for benchmarks such as creative writing or general instruction following, rather than the structured requirements of educational task generation. Another line of work has incorporated thematic topics and learning concepts as input to LLMs to generate a wide range of tasks, e.g., in programming education [15–17]. However, these efforts do not primarily focus on optimizing for task diversity. This gap motivates our central question: *How can we generate a diverse set of high-utility practice tasks*?

In this work, we tackle this question in the context of programming task generation. Drawing inspiration from the creativity literature and the divergent-convergent thinking stages [25–27], we propose a two-stage prompting framework and instantiate CreativeDC, a method for generating diverse contextualized programming tasks. Given a theme and a programming concept as context, CreativeDC instructs the LLM to first go through a *divergent thinking stage*, where it explores the creative space of ideas with respect to the input theme. Then, in the following *convergent thinking stage*, the model refines promising ideas into a valid programming task using the given programming concept. This decoupling encourages the LLM to explore a broader ideation space before committing to a final task. Our main contributions are:

1. We propose a novel prompting framework with divergent-convergent thinking stages and develop CreativeDC, a method for generating diverse programming tasks.
2. We define and propose approaches to evaluate effective diversity, which measures the number of distinct tasks among high-utility tasks.
3. We show through automated and expert evaluations that CreativeDC achieves significantly higher effective diversity than baseline methods.

Figure 1 shows an overview of our method. We release our code and data publicly to support future research.[1]

[1] github.com/machine-teaching-group/edm2026-creativedc.

Manh Hung Nguyen, Sebastian Tschiatschek, and Adish Singla. Enhancing Diversity of LLM-Generated Educational Tasks. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 657–663. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039728

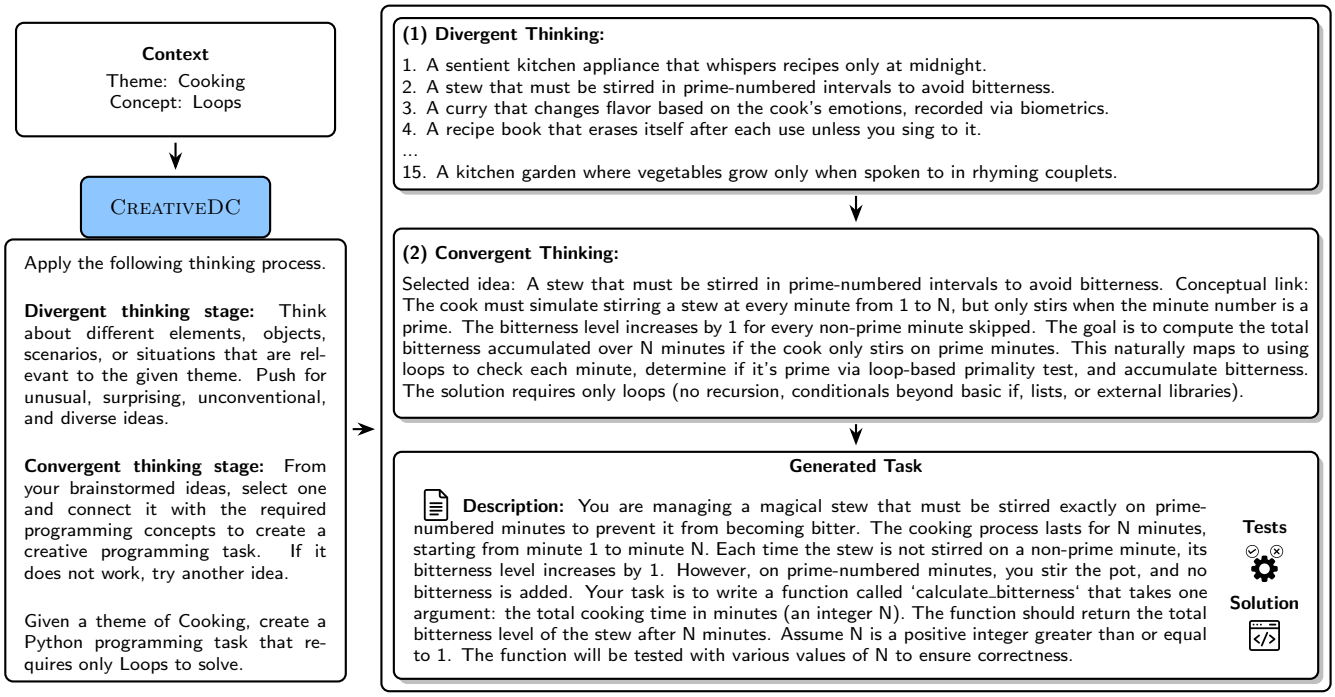


Figure 1: Overview of CreativeDC for programming task generation. The input is a context containing a theme and a programming concept. Our method instructs an LLM to apply a two-stage reasoning process: *(1) Divergent thinking stage* and *(2) Convergent thinking stage*. The generated task consists of a description, a test suite, and a solution.

2. RELATED WORK

LLMs for Task Generation. LLMs have been used to generate educational tasks across domains, including topic-specific questions, mathematics word problems, and programming exercises [2–9]. In programming education, recent work has generated multiple-choice items and contextualized tasks [15–18]. However, these works primarily optimize task correctness or difficulty alignment, whereas we also consider the diversity of generated tasks.

Diversity Collapse of LLMs. LLMs exhibit an "Artificial Hivemind" effect, characterized by high intra-model and inter-model homogeneity [22]. Post-training alignment further amplifies this convergence [28] and reduces LLM creativity and output novelty [29]. Moreover, LLMs have been shown to lack novelty [23, 29] and suffer from "functional fixedness", a bias that limits unconventional thinking [30]. These findings motivate methods that explicitly scaffold exploration rather than direct prompting alone.

Improving Diversity of LLM Outputs. Methods have been proposed to promote diversity in LLM outputs, including fine-tuning with creativity signals [23], multi-agent frameworks [24, 31], and persona simulation for diverse perspectives [21]. However, these methods are primarily designed for open-ended generation settings such as creative writing. Our work instead targets educational task generation, where outputs must balance diversity with utility.

3. PROBLEM SETUP

We formalize the problem of generating diverse tasks, discuss the metrics for evaluation, and state our objective.

Preliminaries. We denote a programming task as a tuple $\mathcal{P} = (\mathcal{P}_{\text{desc}}, \mathcal{P}_{\text{eval}}, \mathcal{P}_{\text{sol}})$, where $\mathcal{P}_{\text{desc}}$ is a natural-language task description, $\mathcal{P}_{\text{eval}}$ is an evaluation guide or final answer for checking the correctness of any answer, and $\mathcal{P}_{\text{sol}}$ is a reference solution. Given a context $\mathcal{C}$, which specifies the requirements for task generation, and an integer K, we seek to generate a set of tasks that satisfy $\mathcal{C}$. We denote the generated set of K tasks as $\mathcal{S}(\mathcal{C}) = \{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_K\}$. For example, if $\mathcal{C}$ corresponds to a learning concept, we aim to generate K tasks that help students practice that concept.

Metrics. We evaluate a task set $\mathcal{S}(\mathcal{C})$ along the following metrics. *Diversity* captures the degree of task variation and is defined as the number of semantically distinct tasks in $\mathcal{S}(\mathcal{C})$. *Utility* captures whether a task $\mathcal{P} \in \mathcal{S}(\mathcal{C})$ is valid, comprehensible, and relevant to the given context. We assign $\text{Utility}(\mathcal{P}) = 1$ if it meets these criteria and 0 otherwise. We define the subset of high-utility tasks as $\mathcal{S}_{\text{util}=1}(\mathcal{C}) = \{\mathcal{P} \in \mathcal{S}(\mathcal{C}) \mid \text{Utility}(\mathcal{P}) = 1\}$. We define *Effective Diversity* as the diversity of this high-utility subset. Intuitively, it measures the number of distinct high-utility tasks. We detail the measurement of these metrics in Section 5.

Objective. Our goal is to design a method that, given a context $\mathcal{C}$, produces a set of K tasks $\mathcal{S}(\mathcal{C})$ with high *effective diversity*. This objective is challenging because encouraging diverse outputs can conflict with the utility constraints [32].

In this work, we instantiate this setup for the generation of programming tasks given a context, following prior work on contextualized programming tasks [15–17]. Figure 1 illustrates an example context and a generated task.

Table 1: Prompts for all methods. {placeholder} is replaced with the actual theme and concept for each context.

BASE
Task Instruction Given a theme of {theme}, create a Python programming problem that requires only {concept} to solve. The problem should include a problem description, a test suite, and a solution program. Below are the requirements for the problem: • The problem must be clearly relevant to the given theme of {theme} and the theme is explicitly used throughout. It requires only {concept} to solve the problem. • The problem description must be sensible and sound natural. It must provide comprehensive information required to solve the problem and pass the test suite (e.g., how the program will be tested, function signatures). Do not use type hints. Do not mention the required programming concepts in the problem description. • The test suite must consist of at least 5 comprehensive test cases written in the Pytest framework format. The testsuite must be correct and cover both base and corner cases. If the test suite involves handling files and I/O, the related files should be created using 'setup_module()' and removed using 'teardown_module()' functions in the Pytest framework. Everything from the solution program will be imported manually; do not import anything else except 'pytest' and 'os'. Do not use multiple assert statements in a single test case. • The solution program must use only {concept}. Do not include any comments, usage examples, or tests in the solution program. The solution program must pass the test suite. # Output Format Output a JSON object with the following keys: `'description'`, `'test_suite'`, and `'solution'`.
COT
Think step by step to generate a problem. # Task Instruction (same as in BASE) # Output Format Output a JSON object with the following keys: `'chain_of_thought'`, `'description'`, `'test_suite'`, and `'solution'`.
CREATIVEDC
Apply the following thinking process to generate a problem. *Divergent thinking phase:* Think about only the given theme and list down wildly different and underexplored elements, objects, scenarios, or situations that are relevant. Ignore the required programming concepts in this phase. Push for unusual, surprising, unconventional, and diverse ideas. Explore the creative space related to the theme as much as possible. *Convergent thinking phase:* From your brainstormed ideas, select one and connect it with the required programming concepts to create a creative programming problem. Make sure the problem does not require any other programming concepts other than the given programming concepts. If it does not work, feel free to go back to select another idea and try again. # Task Instruction (same as in BASE) # Output Format Output a JSON object with the following keys: `'divergent_thinking'`, `'convergent_thinking'`, `'description'`, `'test_suite'`, and `'solution'`.

4. OUR METHOD: CREATIVEDC

We draw inspiration from creativity literature and staged models [25–27] to propose a two-stage prompting framework that structures the model's reasoning into *divergent thinking* and *convergent thinking* stages. Divergent thinking aims to generate many diverse ideas without judgment, while convergent thinking selects and refines the best ideas into a concrete solution. We instantiate this framework as CREATIVEDC (**Creative D**ivergent-**C**onvergent) (cf. Figure 1).

Stage 1: Divergent Thinking. In this stage, CREATIVEDC instructs the LLM to explore the thematic space of context $\mathcal{C}$ without imposing any constraints from the target programming concept in $\mathcal{C}$. By focusing only on the theme, the model is encouraged to brainstorm varied and underexplored elements, objects, scenarios, or situations. Pushing for unusual, surprising, and unconventional ideas helps the model traverse a broader semantic space. The goal of this stage is to avoid the premature convergence often seen in direct prompting, where the model settles too quickly on ideas that satisfy the technical requirements but offer limited variety.

Stage 2: Convergent Thinking. In the second stage, CREATIVEDC instructs the model to select one promising idea from the brainstormed list and refine it into a valid programming task $\mathcal{P}$. The output must satisfy all requirements of $\mathcal{C}$ and, crucially, primarily test the specified programming concept. If the selected idea cannot be developed into a valid task, the model is encouraged to return to the brainstormed list and try another idea. In this way, the convergent stage uses the broader idea pool produced by Stage 1 while enforcing the constraints required for utility.

After going through the two stages, the model outputs the final task $\mathcal{P} = (\mathcal{P}_{\text{desc}}, \mathcal{P}_{\text{eval}}, \mathcal{P}_{\text{sol}})$. Figure 1 illustrates the two stages of our method with an example generated task.

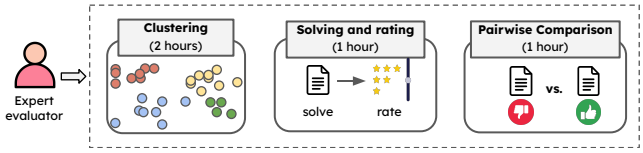


Figure 2: Overview of Expert Evaluation.

5. EVALUATION SETUP

In this section, we present an evaluation setup designed to answer the following questions: **(i)** How does effective diversity scale with the number of generated tasks? **(ii)** How do expert evaluators perceive the generated tasks?

Task Contexts. We use a set of 20 task contexts, each of which is a pair of a theme and a target programming concept following prior work on contextualized tasks [15–17]. We use 4 themes ("Cooking", "Science Fiction", "Superheroes", and "Board Games") and 5 concepts ("Variables", "Selection Statements", "Loops", "Lists", and "Strings"). For each context, we generate K tasks per method. We report the results for $K \in \{10, 20, 30, 40, 50\}$ in Section 6.

Methods. We compare CreativeDC (described in Section 4) against two baselines: Base and CoT. Base directly prompts the model to generate a task from the given theme and concept. CoT prepends a "Think step by step" instruction to the Base prompt to elicit intermediate reasoning [33]. Table 1 shows the prompts for all methods. For all methods, we use Qwen3-235B-A22B [34], a strong open-weight model, enabling reproducible generation. We set the temperature to 1.0.

Expert-based Evaluation Overview. We use a mix of automated and expert-based evaluation as further discussed in Section 5.1 and Section 5.2. We present an overview of our expert evaluation for assessing diversity and multiple task aspects in Figure 2. More concretely, we recruited three experts with experience in programming education and tutoring in programming-related courses. All three are non-native English speakers (one female, two males; mean age 28.7 years), and none are authors of this paper. Participation was voluntary, and no personally identifiable data was collected. Each expert evaluator was assigned the same programming concept ("Variables") and a randomly sampled theme. We use small subsets of tasks generated during the automated evaluation to reduce the workload. We exclude Base from expert evaluation due to similar performance to CoT in automated evaluations (cf. Section 6). To maintain objectivity, the source of each task remained undisclosed throughout the study.

5.1 Measuring Effective Diversity

We measure the effective diversity of a task set $\mathcal{S}$ (defined in Section 3), which requires evaluating both task diversity and task utility. Below, we present two approaches we used to evaluate the set diversity: automated and expert-based.

- **Automated Vendi Score.** We use the Vendi Score [35] to measure task diversity automatically. Given a set of tasks, it ranges from 1 (all tasks identical) to $|\mathcal{S}|$ (all tasks dissimilar), quantifying the number of distinct tasks in $\mathcal{S}$. To begin, each task $\mathcal{P}$ is represented by an embedding obtained from a model (Qwen3-Embedding-0.6B [36]). Given the task embeddings, we construct a cosine similarity matrix and compute the Vendi Score as the exponential of the Shannon entropy of its eigenvalues.
- **Expert-based Clustering.** We involve expert evaluators to assess task diversity through clustering the generated tasks. Given a set of tasks, an evaluator either assigns each task to a cluster based on its semantic similarity to other tasks in the set or creates a new cluster for the task. The number of final clusters provides an expert-annotated measure of task diversity, corresponding to the number of distinct tasks as judged by experts.

Next, we describe how we measure effective diversity which is defined as the diversity (number of distinct tasks) of the high-utility subset $\mathcal{S}_{\text{util}=1}(\mathcal{C})$ (cf. Section 3). To this end, we need to evaluate the utility of generated tasks. For scalability, we use an LLM-as-a-judge approach, which has been used in prior work including task utility evaluation [15, 22, 32]. Given a task $\mathcal{P}$, the judge first generates a solution, which is executed against $\mathcal{P}_{\text{eval}}$ to verify *Task Validity*. The judge then assesses *Context Relevance* (whether the task is relevant to the given context) and *Comprehensibility* (whether sufficient information is provided). Utility$(\mathcal{P}) = 1$ if the task $\mathcal{P}$ satisfies all three criteria. We use Gemini 2.5 Flash-Lite [37] as the judge, chosen for its cost-effectiveness. Using a model from a different family than the generator also avoids self-preference bias.

5.2 Measuring Other Task Aspects

Beyond diversity metrics, we asked experts to perform additional evaluations on small subsets of generated tasks. More specifically, each evaluator solved and rated 6 tasks (3 from CoT and 3 from CreativeDC) via a web interface (providing a task description, code editor, and console). After completing a task or reaching a 10-minute limit, evaluators rated *Theme Relevance*, *Concept Relevance*, *Task Validity*, and *Comprehensibility* on 3-point Likert scales (e.g., Relevant / Partially Relevant / Irrelevant). Supplementary dimensions included *Difficulty* (3-point) and *Interestingness* (5-point). After finishing the task ratings, we asked each evaluator to perform pairwise comparison in terms of novelty. Each pair comprised one task from CoT and one task from CreativeDC. We randomly sampled 20 pairs of tasks. For each pair, the evaluator selected the more novel task and wrote a brief justification.

6. RESULTS

We present the effective diversity with respect to the number of tasks K in Figure 3 and ratings for other task aspects in Table 2.

6.1 Effective Diversity

Effective Diversity via Automated Vendi Score. Figure 3a shows that all methods generate more distinct tasks as the number of tasks K increases. Notably, CreativeDC significantly outperforms both CoT and Base across all K values ($p < 0.001$, Bonferroni-corrected paired Wilcoxon signed-rank tests [38]), while CoT and Base are statistically indistinguishable. At $K = 50$, CreativeDC generates about 13

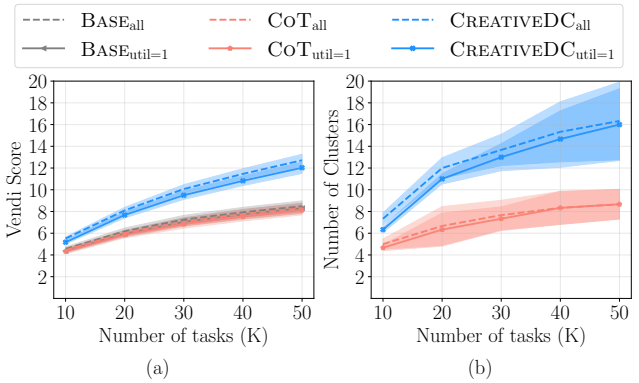


Figure 3: Effective diversity with respect to number of tasks K via (a) automated Vendi scores and (b) expert evaluation. Solid lines denote effective diversity (diversity of high-utility tasks only) and dashed lines denote diversity over all tasks. Shaded bands show SE across contexts. CreativeDC consistently outperforms baselines across both evaluations.

distinct tasks (dashed blue line), and about 12 distinct tasks when restricted to high-utility tasks only (solid blue line). In contrast, both CoT and BASE generate about 8 distinct high-utility tasks (solid red and grey lines).

Effective Diversity via Expert Evaluation. Figure 3b shows that effective diversity perceived by experts has a similar pattern but is generally higher than the automated Vendi Score. For instance, evaluators created about 16 clusters at $K = 50$ for CREATIVEDC when restricted to high-utility tasks, compared to about 9 clusters for CoT. While CoT shows signs of saturation, CREATIVEDC maintains an upward trend. Because each expert evaluated a uniquely themed subset of tasks, inter-rater agreement measures are not applicable. This evaluation provides evidence that the automated metric we used can serve as a proxy for expert-perceived diversity.

6.2 Expert Ratings for Other Task Aspects

We compute the average expert ratings (cf. Section 5.2) by mapping 3-point scales to $\{1, 0.5, 0\}$ and 5-point scales to $\{1, 0.75, 0.5, 0.25, 0\}$. Table 2 shows that tasks generated by CREATIVEDC were judged higher than those from CoT on the utility-related submetrics. Tasks generated by both methods were perceived as between easy and medium in difficulty. Notably, CREATIVEDC tasks were rated higher in interestingness (0.81 vs. 0.50) and were preferred in 96.67% of pairwise novelty comparisons. Justifications for experts' decisions favoring tasks generated by CREATIVEDC include: "It makes the cooking theme and the question very life-like"; "Trading recorded personal experiences as currency is definitely novel to me–the problem even opens up my mind for new possibilities that may really happen in the future".

7. CONCLUDING DISCUSSION

We proposed CREATIVEDC, a method for generating diverse and high-utility tasks in programming education. Through automated and expert evaluations, CREATIVEDC has been shown to produce more distinct high-utility programming tasks, improving the effective diversity of generated tasks.

Table 2: Expert evaluator ratings (means ± SE) for various task aspects: utility, novelty, and difficulty. Highest scores are bolded for metrics where higher is better (↑).

Metric	CoT	CREATIVEDC
Utility:TaskValidity ↑	0.78 ± 0.15	**1.00 ± 0.00**
Utility:ThemeRelevance ↑	0.94 ± 0.06	**1.00 ± 0.00**
Utility:ConceptRelevance ↑	0.89 ± 0.07	**0.94 ± 0.06**
Utility:Comprehensibility ↑	0.72 ± 0.15	**0.83 ± 0.08**
Novelty:Interestingness ↑	0.50 ± 0.04	**0.81 ± 0.04**
Novelty:Win Rate (%) ↑	3.33 ± 2.32	**96.67 ± 2.32**
Difficulty:Self-rated	0.17 ± 0.08	0.17 ± 0.08
Difficulty:Pass Rate (%)	88.89 ± 11.11	100.00 ± 0.00
Difficulty:Time Taken (mins)	3.84 ± 0.55	3.38 ± 0.61

Next, we discuss limitations of our work and suggest future directions. First, our evaluation focused on introductory programming concepts; extending this framework to other educational domains, such as mathematics or science, would demonstrate its generalizability. Second, while our expert study provides initial validation, it is limited by its sample size and demographic diversity; it would be useful to conduct large-scale studies to evaluate the tasks from the perspective of students. Third, the two stages in our method can be iterated in a loop to improve the quality of the generated tasks; exploring this iterative process is a promising direction. Fourth, our experiments focus on introductory Python programming concepts; it would be useful to check whether this diversity holds in harder problems with advanced concepts and whether diversity across difficulty levels can be enforced while maintaining the effective validity and utility of the generated tasks. Finally, our evaluation focused on theme and concept as a task context; expanding the context to include learner-state contextualization (e.g., difficulty, student profiles) is a natural next step for generating diverse and personalized tasks.

Acknowledgement

Funded/Co-funded by the European Union (ERC, TOPS, 101039090). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] Sabina Elkins, Ekaterina Kochmar, Jackie C. K. Cheung, and Iulian Serban. How Teachers Can Use Large Language Models and Bloom's Taxonomy to Create Educational Quizzes. In *AAAI*, 2024.

[2] Jaewook Lee, Digory Smith, Simon Woodhead, and Andrew S. Lan. Math Multiple Choice Question Generation via Human-Large Language Model Collaboration. In *EDM*, 2024.

[3] Ziqing Li, Mutlu Cukurova, and Sahan Bulathwela. A Novel Approach to Scalable and Automatic Topic-Controlled Question Generation in Education. In *LAK*, 2025.

[4] Sahan Bulathwela, Hamze Muse, and Emine Yilmaz. Scalable Educational Question Generation with Pre-trained Language Models. In *AIED*, 2023.

[5] Zichao Wang, Andrew S. Lan, and Richard G. Baraniuk. Math Word Problem Generation with Mathematical Consistency and Problem Context Constraints. In *EMNLP*, 2021.

[6] Hai Li, Rui Guo, Chenglu Li, and Wanli Xing. Automated Quality Assessment of Multimodal Mathematical Stories Generated by Generative Artificial Intelligence. In *L@S*, 2024.

[7] Ying Jiao, Kumar Shridhar, Peng Cui, Wangchunshu Zhou, and Mrinmaya Sachan. Automatic Educational Question Generation with Difficulty Level Controls. In *AIED*, 2023.

[8] Yongan Yu, Alexandre Krantz, and Nikki G. Lobczowski. From Recall to Reasoning: Automated Question Generation for Deeper Math Learning Through Large Language Models. In *AIED*, 2025.

[9] Kia Karbasi, Kevin Hong, Mohammad Amin Samadi, and Gregory J. Pottie. Multi-Agent Collaborative Framework For Math Problem Generation. In *EDM*, 2025.

[10] Niklas Scholz, Manh Hung Nguyen, Adish Singla, and Tomohiro Nagashima. Partnering with AI: A Pedagogical Feedback System for LLM Integration Into Programming Education. In *ECTEL*, 2025.

[11] Mehmet Arif Demirtas, Claire Zheng, Max Fowler, and Kathryn I. Cunningham. Generating Planning Feedback for Open-Ended Programming Exercises with LLMs. In *AIED*, 2025.

[12] Alexander Scarlatos, Naiming Liu, Jaewook Lee, Richard G. Baraniuk, and Andrew S. Lan. Training LLM-Based Tutors to Improve Student Learning Outcomes in Dialogues. In *AIED*, 2025.

[13] Manh Hung Nguyen, Sebastian Tschiatschek, and Adish Singla. Large Language Models for In-Context Student Modeling: Synthesizing Student's Behavior in Visual Programming. In *EDM*, 2024.

[14] Lorenzo Lee Solano, Charles Koutcheme, Juho Leinonen, Alexandra Vassar, and Jake Renzella. Fine-Tuning Open-Source Models as a Viable Alternative to Proprietary LLMs for Explaining Compiler Messages. In *SIGCSE*, 2026.

[15] Manh Hung Nguyen, Victor-Alexandru Padurean, Alkis Gotovos, Sebastian Tschiatschek, and Adish Singla. Synthesizing High-Quality Programming Tasks with LLM-Based Expert and Student Agents. In *AIED*, 2025.

[16] Evanfiya Logacheva, Arto Hellas, James Prather, Sami Sarsa, and Juho Leinonen. Evaluating Contextually Personalized Programming Exercises Created with Generative AI. In *ICER*, 2024.

[17] Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models. In *ICER*, 2022.

[18] Mohammad Hassany, Peter Brusilovsky, Jaromír Savelka, Arun Balajiee Lekshmi Narayanan, Kamil Akhuseyinoglu, Arav Agarwal, and Rully Agus Hendrawan. Generating Effective Distractors for Introductory Programming Challenges: LLMs vs Humans. In *LAK*, 2025.

[19] Paul Denny, Sumit Gulwani, Neil T. Heffernan, Tanja Käser, Steven Moore, Anna N. Rafferty, and Adish Singla. Generative AI for Education (GAIED): Advances, Opportunities, and Challenges. *CoRR*, abs/2402.01580, 2024.

[20] Rachel Van Campenhout, Kevin S. Autry, Michelle W. Clark, and Benny G. Johnson. Scaling the Doer Effect: A Replication Analysis Using AI-Generated Questions. In *L@S*, 2025.

[21] Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling Synthetic Data Creation with 1,000,000,000 Personas. *CoRR*, abs/2406.20094, 2024.

[22] Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten Sap, Alon Albalak, and Yejin Choi. Artificial Hivemind: The Open-Ended Homogeneity of Language Models (and Beyond). In *NeurIPS*, 2025.

[23] Mete Ismayilzada, Antonio Laverghetta Jr., Simone A. Luchini, Reet Patel, Antoine Bosselut, Lonneke Van Der Plas, and Roger E. Beaty. Creative Preference Optimization. In *EMNLP*, 2025.

[24] Li-Chun Lu, Shou-Jen Chen, Tsung-Min Pai, Chan-Hung Yu, Hung-Yi Lee, and Shao-Hua Sun. LLM Discussion: Enhancing the Creativity of Large Language Models via Discussion Framework and Role-Play. In *COLM*, 2024.

[25] Graham Wallas. *The Art of Thought*. 1926.

[26] Ronald A. Finke, Thomas B. Ward, and Steven M. Smith. *Creative Cognition: Theory, Research, and Applications*. The MIT Press, 1992.

[27] Joy Paul Guilford. The Structure of Intellect. *Psychological Bulletin*, 53, 1956.

[28] Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. NoveltyBench: Evaluating Language Models for Humanlike Diversity. In *COLM*, 2025.

[29] Ximing Lu, Melanie Sclar, Skyler Hallinan, Niloofar Mireshghallah, Jiacheng Liu, Seungju Han, Allyson Ettinger, Liwei Jiang, Khyathi Raghavi Chandu, Nouha Dziri, and Yejin Choi. AI as Humanity's Salieri: Quantifying Linguistic Creativity of Language Models via Systematic Attribution of Machine Text against Web Text. In *ICLR*, 2025.

[30] Yufei Tian, Abhilasha Ravichander, Lianhui Qin, Ronan Le Bras, Raja Marjieh, Nanyun Peng, Yejin Choi, Thomas L. Griffiths, and Faeze Brahman. MacGyver: Are Large Language Models Creative Problem Solvers? In *NAACL*, 2024.

[31] Manh Hung Nguyen, Sebastian Tschiatschek, and Adish Singla. Prompt Optimization Across Multiple Agents for Representing Diverse Human Populations. *CoRR*, abs/2510.07064, 2025.

[32] Vishakh Padmakumar, Chen Yueh-Han, Jane Pan, Valerie Chen, and He He. Measuring LLM Novelty As The Frontier Of Original And High-Quality Output. *CoRR*, abs/2504.09389, 2025.

[33] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*, 2022.

[34] An Yang et al. Qwen3 Technical Report. *CoRR*, abs/2505.09388, 2025.

[35] Dan Friedman and Adji Bousso Dieng. The Vendi Score: A Diversity Evaluation Metric for Machine Learning. *TMLR*, 2023.

[36] Yanzhao Zhang et al. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *CoRR*, abs/2506.05176, 2025.

[37] Gemini Team. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. *CoRR*, abs/2507.06261, 2025.

[38] Frank Wilcoxon. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

Evaluating Private Training in Educational Prediction Beyond Utility and Privacy Audit

Xianghui Meng
The University of Hong Kong
margretmeng1020@gmail.com

Yujing Zhang
The University of Hong Kong
zhangyujing@connect.hku.hk

Xian Chen
The University of Manchester
metaxianchen@gmail.com

Jionghao Lin*
The University of Hong Kong
jionghao@hku.hk

ABSTRACT

Educational prediction models are increasingly used to identify students who may need early academic support. When such models are trained with privacy-preserving methods, evaluation often focuses on average predictive performance and measured privacy leakage. However, this can overlook an important deployment concern: private training may also change how prediction errors are distributed across student groups. We examine this issue using two public educational datasets. OULAD serves as the primary dataset because it supports both privacy auditing and group-level error analysis, while UCI697 is used only as a secondary consistency check for the relationship between private training, predictive quality, and measured privacy leakage. Across both datasets, we compare standard non-private training with Differentially Private Stochastic Gradient Descent (DP-SGD). We evaluate predictive quality using test AUC, measured privacy leakage using a loss-based membership inference attack, and, on OULAD, group-level error patterns using worst-group gaps in true positive rate and false positive rate. Across both datasets, stronger private training reduces predictive quality more clearly than it changes the measured privacy-audit outcome. On OULAD, under the fixed threshold used for controlled comparison, privacy-preserving training also increases group-level error gaps. These results suggest that evaluating student prediction models only through average predictive performance and measured privacy leakage can miss an important part of the effect of privacy-preserving training. Educational prediction models trained with privacy-preserving methods should be evaluated jointly in terms of predictive quality, measured privacy leakage, and group-level error patterns, especially when they are used to inform student support decisions. Our code is available on GitHub: https://github.com/GEMLabHKU/DP-FAIRNESS/tree/main

Keywords

Privacy, Prediction Model, Membership Inference Attack, Algorithmic Fairness

*Corresponding author

Xianghui Meng, Yujing Zhang, Xian Chen, and Jionghao Lin. Evaluating Private Training in Educational Prediction Beyond Utility and Privacy Audit. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 664–668. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040142

1. INTRODUCTION

Prediction models are increasingly used to identify students who may be at risk of failure, withdrawal, or dropout [1, 2, 3]. These models are sometimes trained on sensitive educational records, including demographic information, academic histories, and learning activity traces. The collection and use of such records raise concerns about student privacy, consent, transparency, and institutional responsibility [4]. Privacy protection is therefore needed because trained machine learning models may reveal information about the records used during training, including whether a particular student's record was included [5, 6].

Prior work has developed privacy-preserving training methods such as DP-SGD [7, 8]. Evaluations of such methods commonly consider predictive performance and measured privacy leakage. Predictive performance describes how well a model distinguishes between students with different outcomes, whereas measured privacy leakage examines whether information about individual training records remains detectable from the trained model. For example, a membership inference attack tests whether an attacker can determine whether a particular user record was included in the training data [5, 6]. One common audit is membership inference, which tests whether an attacker can determine whether a particular student record was included in the training data. Membership inference does not necessarily reveal a student's identity or complete educational record. However, when an attacker already has auxiliary information about a student, evidence of training-set membership may strengthen an association between that student and sensitive information, such as course participation, academic difficulty, dropout risk, or receipt of educational support. For this reason, reducing measured privacy leakage is an important goal of privacy-preserving training. However, privacy and utility alone may still provide an incomplete view of model behaviour in deployment [9]. A model may maintain usable predictive performance and show little detectable membership signal under a selected privacy attack, while still changing how prediction errors are distributed across student groups.

In practical implementation, prediction models may influence which students are identified for support, which students are overlooked, and which students are incorrectly flagged for intervention. Therefore, even when a privacy-preserving model retains usable overall predictive performance and shows little detectable membership signal under a selected privacy attack, it may still distribute prediction errors unevenly across

student groups. This matters because uneven error patterns can create unequal consequences for students: some groups may be more likely to receive timely support, while others may be more likely to be missed or unnecessarily targeted for intervention. Thus, in addition to evaluating overall model quality and measured privacy leakage, privacy-preserving training should also be evaluated in terms of how prediction errors are distributed across student groups.

This study investigates this deployment concern by examining whether the introduction of privacy-preserving training changes not only predictive quality and measured privacy leakage, but also the distribution of prediction errors across student groups. Specifically, we compare standard non-private training with Differentially Private Stochastic Gradient Descent (DP-SGD), a widely used privacy-preserving training method that clips per-example gradients and adds noise during optimisation [7, 8]. We use OULAD as the primary dataset because it supports both privacy auditing and group-level error analysis, and we use UCI697 as a secondary case to test whether the relationship between privacy protection and predictive quality also appears in another educational setting [3, 10].

To evaluate individual-level privacy risk, we use a membership inference attack (MIA), which tests whether an attacker can distinguish a student record used for model training from a record that was not used for training [5, 6]. This attack provides one operational measure of whether information about an individual training record remains detectable from the trained model. To evaluate group-level consequences, we examine whether privacy-preserving training changes how prediction errors are distributed across student groups [1, 2]. These two concerns are commonly studied through different evaluation traditions. Privacy research often focuses on formal privacy guarantees or attack-based audits of individual records [5, 6], whereas educational prediction research commonly evaluates average predictive performance and examines group fairness as a separate dimension [11, 12]. Evaluating these dimensions separately can obscure an important deployment concern: a model may retain usable average predictive performance and show little detectable membership signal under a selected privacy attack, while still producing more uneven error patterns across student groups after privacy-preserving training is introduced. This study contributes an integrated evaluation perspective for privacy-preserving educational prediction by examining predictive quality, measured individual-level privacy leakage, and group-level error patterns within the same empirical comparison. This joint evaluation makes visible whether private training changes not only average model performance and detectable membership signals, but also which student groups are more likely to be correctly identified, overlooked, or incorrectly flagged for support. As introduced in Figure 1, our study is guided by two Research Questions (**RQs**):

RQ1. How does private training change predictive quality and measured privacy leakage in educational prediction?

RQ2. Does private training also change group-level error patterns in ways that would be missed by evaluation based only on utility and privacy audit?

2. METHOD

Our evaluation has three parts, each aligned with the paper's central question: predictive quality, measured privacy leakage, and group-level error patterns. Predictive quality is evaluated using test AUC. Measured privacy leakage is evaluated using a loss-based membership inference attack. Group-level error patterns are evaluated on OULAD using worst-group gaps in true positive rate (TPR) and false positive rate (FPR).

2.1 Datasets and prediction tasks

We use two public educational datasets with different roles in the study. OULAD is the primary dataset because it is the only dataset in our experiments that supports all three parts of the analysis within a single setting: predictive quality, privacy auditing, and group-level error evaluation [3]. UCI697 is used only as a secondary consistency check for the relationship between private training, predictive quality, and measured privacy leakage [10].

The Open University Learning Analytics Dataset (OULAD) contains 32,593 student-course records with course outcomes and demographic attributes such as gender, region, age band, and education level [3]. The prediction task in this study is course failure or withdrawal. We use structured student and course features, including previous attempts, studied credits, and categorical variables converted into numeric form.

The UCI Student Dropout and Academic Success dataset (UCI697) contains 4,424 higher education records and 36 features related to enrolment background and academic progress [10]. The prediction target is dropout versus graduation. We do not use UCI697 for group-level error analysis in this study. For both datasets, all training conditions use the same preprocessing pipeline within each dataset, so comparisons are between training conditions rather than between different feature sets or preprocessing choices.

2.2 Training setup and split protocol

We compare standard non-private training with Differentially Private Stochastic Gradient Descent (DP-SGD) [7]. DP-SGD modifies ordinary gradient-based training in two steps: it clips each example's gradient to bound the influence of any single training record, and then adds Gaussian noise to the aggregated clipped gradients during optimisation. In our experiments, the clipping norm is fixed at $C = 1.0$. We evaluate privacy budgets $\epsilon \in \{1, 5, 10\}$ and set $\delta = 1/n_{\text{train}}$, with smaller ϵ corresponding to stronger privacy protection.

We use logistic regression as a simple linear baseline and a small multilayer perceptron (MLP-small) as the main nonlinear baseline for matched comparisons. MLP-small is chosen to compare private and non-private training under the same feature representation. Across conditions, the model family, feature inputs, data splits, and evaluation protocol are held fixed. The main change is whether DP-SGD is applied during training. All conditions are run with matched random seeds, and each reported result summarises five runs per condition. For each run, models are trained on a training split and evaluated on the same held-out test split for predictive quality, privacy, and, where applicable, group-level error analysis. This keeps the comparison focused on the effect of private training rather than on differences in data partitioning.

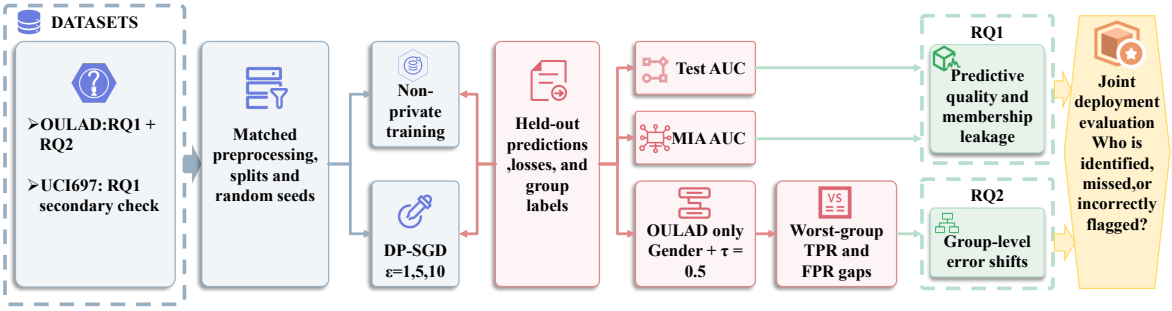


Figure 1: Study design aligned with the two research questions. Both datasets are used to compare non-private training with DP-SGD. RQ1 focuses on predictive quality and privacy leakage. RQ2 focuses on group-level error shifts and is evaluated on OULAD only.

2.3 Privacy audit and group-level error evaluation

To evaluate measured privacy leakage, we use the loss-based membership inference attack proposed by Yeom et al. [6]. In this attack, the attacker uses prediction loss as the signal, with lower-loss examples treated as more likely to be training members. For each run, the attack compares training members with held-out non-members from the same split used for model evaluation. We report membership inference attack AUC, which measures how well the attack separates training members from non-members. An AUC close to 0.5 indicates near-random separation, whereas a larger value indicates stronger measured membership signal. We use this attack as one measured privacy audit under a specific threat model, rather than as a complete characterisation of privacy risk in educational prediction.

To evaluate group-level error patterns on OULAD, we compare two worst-group error gaps across gender groups. We use gender as a first group-based check because this attribute is complete and consistently represented in the OULAD records used here. The first metric is the worst-group gap in true positive rate, and the second is the worst-group gap in false positive rate. For a group g, TPR_g is the proportion of positive cases in that group that the model correctly predicts as positive, and FPR_g is the proportion of negative cases in that group that the model incorrectly predicts as positive. We report

$$\Delta_{\mathrm{TPR}} = \max_g \mathrm{TPR}_g - \min_g \mathrm{TPR}_g$$

and

$$\Delta_{\mathrm{FPR}} = \max_g \mathrm{FPR}_g - \min_g \mathrm{FPR}_g.$$

A larger TPR gap means that some groups are more likely than others to receive correct positive predictions, whereas a larger FPR gap means that some groups are more likely than others to receive incorrect positive predictions. All group-level error results use a fixed decision threshold of 0.5 to support controlled comparison across training conditions. This choice is intended to isolate changes in model behaviour rather than to represent an optimal deployment threshold.

3. RESULTS

3.1 RQ1: Privacy-preserving training on predictive quality and measured leakage

Table 1 reports the main OULAD results across five matched random seeds per condition. For RQ1, we examine two outcomes commonly used to evaluate private training: predictive quality and measured privacy leakage. Table 2 reports the corresponding utility–privacy patterns on UCI697 as a secondary consistency check.

We begin with predictive quality. On OULAD, the non-private MLP-small baseline achieves a test AUC of 0.661 ± 0.004. Under DP-SGD with $\epsilon = 1$, test AUC decreases to 0.644 ± 0.002, and partially recovers to 0.649 ± 0.002 at $\epsilon = 5$ and $\epsilon = 10$. This pattern is consistent with the expected privacy–utility trade-off: stronger privacy protection produces the largest reduction in predictive quality, whereas looser privacy budgets recover some utility. UCI697 shows the same pattern. There, the non-private baseline achieves the highest test AUC, 0.952 ± 0.008, DP-SGD at $\epsilon = 1$ yields the largest drop to 0.919 ± 0.012, and performance improves again at $\epsilon = 5$ and $\epsilon = 10$. Across both datasets, private training therefore changes predictive quality in the expected direction.

The measured privacy-leakage results require a more cautious reading. On OULAD, the loss-based membership inference attack AUC remains close to random guessing in all conditions, including the non-private baseline. This is therefore better interpreted as evidence that the evaluated attack is already weak in this setting, rather than evidence that DP-SGD removes a strong privacy leakage signal. Additional analysis supports this interpretation: member and non-member scores are already highly similar before private training is applied. The UCI697 results support the same reading, with membership inference attack AUC remaining close to 0.5 across all conditions.

Taken together, the RQ1 results show that private training changes predictive quality more clearly than it changes the measured privacy-audit outcome. This means that evaluation based only on utility and privacy audit provides only a partial view of its effects.

Table 1: Main results on OULAD, with 5 seeds per condition. Test AUC measures predictive quality. Membership inference attack (MIA) AUC measures how well a loss-based attack separates training members from non-members. TPR and FPR gaps measure group-level error differences across gender groups.

Condition	Test AUC	MIA AUC	TPR Gap	FPR Gap
LR Baseline	0.640 ± 0.002	0.501 ± 0.001	0.063 ± 0.010	0.075 ± 0.015
MLP-small Baseline	0.661 ± 0.004	0.506 ± 0.003	0.016 ± 0.009	0.009 ± 0.002
MLP-small DP-SGD $\epsilon = 1$	0.644 ± 0.002	0.503 ± 0.001	0.045 ± 0.014	0.064 ± 0.014
MLP-small DP-SGD $\epsilon = 5$	0.649 ± 0.002	0.503 ± 0.002	0.031 ± 0.016	0.046 ± 0.012
MLP-small DP-SGD $\epsilon = 10$	0.649 ± 0.002	0.503 ± 0.002	0.028 ± 0.019	0.042 ± 0.011

Table 2: UCI697 results, used only as a privacy–utility consistency check.

Condition	Test AUC	MIA AUC
Baseline	0.952 ± 0.008	0.502 ± 0.013
DP-SGD $\epsilon = 1$	0.919 ± 0.012	0.495 ± 0.013
DP-SGD $\epsilon = 5$	0.938 ± 0.012	0.495 ± 0.014
DP-SGD $\epsilon = 10$	0.941 ± 0.012	0.494 ± 0.013

3.2 RQ2: Privacy-preserving training on group-level error patterns

RQ2 examines the part of model behaviour that would be missed if evaluation were limited to predictive quality and privacy audit. This analysis is conducted only on OULAD, the only dataset in this study that supports group-level error evaluation. The answer is yes: under the fixed threshold used here, private training changes how prediction errors are distributed across groups.

Relative to the non-private MLP-small baseline, both worst-group error gaps increase under private training. At $\epsilon = 1$, the TPR gap rises from 0.016 ± 0.009 to 0.045 ± 0.014, and the FPR gap rises from 0.009 ± 0.002 to 0.064 ± 0.014. At $\epsilon = 5$ and $\epsilon = 10$, both gaps remain above the non-private baseline, although they are smaller than at $\epsilon = 1$. This indicates that the shift is not confined to the strictest privacy condition, even if it is most pronounced there.

This finding matters because the change in group-level error patterns appears alongside only a modest change in overall AUC. On OULAD, private training does not remove the model's predictive usefulness, but it does alter how support-related errors are distributed across groups. A deployment decision based only on average predictive quality and privacy audit could therefore conclude that the private model remains broadly acceptable, while missing that the pattern of errors across groups has shifted.

The two gap types also have distinct practical interpretations. A larger TPR gap means that some groups are more likely than others to receive correct positive predictions, whereas a larger FPR gap means that some groups are more likely than others to receive incorrect positive predictions. In educational settings, these differences can translate into unequal access to intervention or unequal exposure to unnecessary intervention, even when the model still appears acceptable on average performance metrics.

Taken together with RQ1, these results show why utility and privacy audit alone are not sufficient for evaluating private training in educational prediction. In settings such as OULAD, private training changes not only overall utility, but also how support-related errors are distributed across groups.

4. DISCUSSION

This study shows that private training can change more than average predictive performance and measured privacy leakage. On OULAD, DP-SGD reduced predictive quality to some extent, while the evaluated loss-based membership inference attack remained close to random across both private and non-private conditions. When evaluation stopped at these two dimensions, the training conditions might appear similar under the selected privacy audit, even though the audit provides limited evidence because attack performance was already close to random for the non-private model. However, the same models produced larger worst-group TPR and FPR gaps under the fixed decision threshold used for comparison. This indicates that private training can alter the distribution of support-related prediction errors even when average predictive performance remains usable and the selected privacy audit detects little membership signal.

4.1 Implications

The practical importance of this finding lies in how educational prediction models are used. In student-risk modelling, predictions are often connected to advising, tutoring, outreach, or other forms of early support. A true positive can correspond to a student being correctly identified for support, whereas a false negative can correspond to a student being overlooked. Similarly, a false positive can expose students to unnecessary intervention or concern. Group-level shifts in TPR and FPR therefore matter because they may change which groups are more likely to receive support, be missed, or be incorrectly flagged. The contribution of this paper is to make this deployment issue visible in the evaluation of privacy-preserving educational prediction models.

This finding also has a methodological implication for educational data mining. Privacy, predictive performance, and group-level error behaviour are often evaluated as separate concerns. The present results suggest that this separation can be incomplete when prediction models are used to inform student support. In this setting, average utility and the evaluated privacy audit alone would not have shown that private training also changed who was more likely to be correctly identified, overlooked, or unnecessarily flagged. Evaluating these dimensions jointly therefore provides a more

deployment-relevant view of privacy-preserving educational prediction.

4.2 Limitations and Future Work

Several limitations define the scope of the findings. *First*, the privacy audit uses only one loss-based membership inference attack, which detected little separation between members and non-members even before private training. The results therefore should not be interpreted as evidence that DP-SGD removed a strong membership signal or as a complete assessment of privacy risk. Future work should examine stronger and complementary attacks, including confidence-based, entropy-based, shadow-model, label-only, and white-box attacks. *Second*, the group-level analysis is limited to gender groups in OULAD and to worst-group TPR and FPR gaps. Future studies should examine additional attributes, intersectional groups, and complementary metrics such as FNR, positive prediction rate, positive predictive value, and calibration error. *Third*, the fixed threshold of 0.5 supports controlled comparison but does not represent all institutional intervention rules. Future work should test whether similar shifts appear across thresholds and realistic support-allocation policies. *Fourth*, the analysis documents group-level error shifts but does not identify their causes. Further work should examine whether these shifts arise from gradient clipping, added noise, subgroup size, class imbalance, or feature-distribution differences. *Finally*, the study uses two public datasets and relatively simple models. The findings should therefore be tested with larger datasets, richer model families, and deployment settings where student identifiability, privacy risk, and intervention consequences can be evaluated more directly.

5. CONCLUSION

This paper investigated how privacy-preserving training changes predictive quality, measured privacy leakage, and group-level error patterns in educational prediction. Across two public datasets, stronger privacy-preserving training consistently reduced predictive quality, while the evaluated loss-based membership inference attack remained close to random. On OULAD, privacy-preserving training also changed worst-group TPR and FPR gaps across gender groups.

These findings show that evaluating privacy-preserving training only as a privacy–utility trade-off can provide an incomplete account of its deployment effects. A model may retain usable average predictive performance and show little measured leakage under a particular membership inference attack, while still changing how prediction errors are distributed across student groups. In educational settings, this matters because such errors may affect which students are correctly identified for support, which students are overlooked, and which students are unnecessarily flagged for intervention.

The main contribution of this paper is therefore to broaden the evaluation frame for privacy-preserving educational prediction. Rather than treating predictive quality, measured privacy leakage, and group-level error patterns as separate concerns, our results suggest that they should be examined jointly when student prediction models are used to inform support decisions. This joint evaluation is especially important in contexts where privacy-preserving training is adopted to protect student data, but the resulting models may still have unequal consequences across student groups.

6. ACKNOWLEDGMENTS

This work was supported by the Faculty Research Fund and by the grant from the URC (Grant No. 2401102970) at The University of Hong Kong.

7. REFERENCES

[1] Qian Hu and Huzefa Rangwala. Towards fair educational data mining: A case study on detecting at-risk students. *International Educational Data Mining Society*, 2020.

[2] Josh Gardner, Christopher Brooks, and Ryan Baker. Evaluating the fairness of predictive student models through slicing analysis. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 225–234, 2019.

[3] Jakub Kuzilek, Martin Hlosta, and Zdenek Zdrahal. Open university learning analytics dataset. *Scientific Data*, 4(1):170171, 2017.

[4] Abelardo Pardo and George Siemens. Ethical and privacy principles for learning analytics. *British Journal of Educational Technology*, 45(3):438–450, 2014.

[5] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.

[6] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE, 2018.

[7] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[8] Anika Kabir, Chandan Tankala, and Daniel Lowd. On the practicality of differential privacy for knowledge tracing. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 619–624, 2025.

[9] Mélina Verger, Sébastien Lallé, François Bouchet, and Vanda Luengo. Is your model "MADD"? a novel metric to evaluate algorithmic fairness for predictive student models. In *Proceedings of the 16th International Conference on Educational Data Mining*, pages 91–102, 2023.

[10] Valentim Realinho, Mónica Vieira Martins, Jorge Machado, and Luís Baptista. Predict students' dropout and academic success. UCI Machine Learning Repository, 2021.

[11] René F. Kizilcec and Hansol Lee. Algorithmic fairness in education. In Wayne Holmes and Kaśka Porayska-Pomsta, editors, *The Ethics of Artificial Intelligence in Education*, pages 174–202. Routledge, 2022.

[12] Hansol Lee and René F. Kizilcec. Evaluation of fairness trade-offs in predicting student success, 2020.

Extracting Curriculum Rules: Formal Modelling of the Study Entry and Orientation Phase

Selina Reinhard
TU Wien, Austria
Selina.Reinhard@tuwien.ac.at

Tom Breuer
RWTH Aachen University, Germany
Tom.Breuer@rwth-aachen.de

René Röpke
TU Wien, Austria
Rene.Roepke@tuwien.ac.at

ABSTRACT

The transition into higher education is a critical phase. In Austria, it is structured through the Study Entry and Orientation Phase (StEOP), a formally mandated introductory phase. Despite its central role, corresponding regulations are specified in natural language, making them difficult to interpret, compare, and analyse. This paper addressed this challenge by proposing a formal model of StEOP regulations based on nested course pools and credit requirements. To operationalise the model, we introduce a semi-automated extraction approach. Results show that all but one of the analysed StEOPs can be consistently mapped to the proposed model. By introducing a formal model of the StEOP and empirically validating its applicability, this work enables the systematic analysis and comparison of curriculum designs, laying the foundation for data-driven evaluation of study entry phases.

Keywords

Curriculum Modelling, Formalisation, Constraint Satisfaction Problems, Rule Extraction, Data Mining

1. INTRODUCTION

The transition to higher education (HE) represents a critical phase for many students, often marked by uncertainty about study requirements and expectations. To facilitate this transition, universities have implemented various support mechanisms. In Austria a structured introductory phase – the **St**udy **E**ntry and **O**rientation **P**hase (StEOP) – has been introduced. The StEOP defines a set of courses with additional constraints that students must complete before they can enrol in more advanced courses. The aim of the StEOP is to provide an early overview of the study program's content and requirements, enabling students to make informed decisions about whether the chosen field of study aligns with their interests and abilities, and to serve as a regulatory mechanism.

Selina Reinhard, Tom Breuer, and René Röpke. Extracting Curriculum Rules: Formal Modelling of the Study Entry and Orientation Phase. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 669–673. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039746

While the StEOP is implemented nationwide and follows general regulatory guidelines, its concrete design and structure are defined individually for each study program. These specifications are described in textual form within the official curricula. However, there is no standardised formal model or machine-readable representation of these regulations. Without such a formal representation, systematically analysing and understanding how curriculum design – particularly different StEOP implementations – affects student progression and success remains challenging. This work addresses this gap by proposing a formal model of StEOP regulations, introducing a semi-automated extraction approach, and empirically evaluating the model's validity. Thus, we aim to answer the following research questions:

RQ1 How can the StEOP structure be formally modelled?

RQ2 To what extent can different StEOP implementations be accurately represented using the proposed model?

2. RELATED WORK

Our work relates to and extends existing work in the fields of curriculum analytics (CA) and educational data mining (EDM), especially prior research on curriculum modelling, study planning, and formal representations of study regulations: Study regulations are typically specified in natural language, which makes them accessible to humans but difficult to process computationally. This gap introduces challenges such as ambiguity and limited support for automated analysis and verification [3]. Therefore, modelling curriculum regulations (e.g., StEOPs) in a formal, machine-readable way remains an important topic in educational technology (EdTech) and EDM, where structured representations are a prerequisite for large-scale analysis. Prior work has approached curriculum modelling from different perspectives, including structural analysis [1], formal verification [6], and individual study planning [7]. One line of research formulates curriculum planning as a CSP, proposing a planning system based on a constraint-satisfaction approach for constructing valid study plans across multiple semesters [11]. Similarly, process mining approaches can be used to model prerequisite structures and learning pathways to support study planning [5, 10]. Other approaches focus on richer semantic representations of curricula, for instance, based on disciplines, competences, and specialisations [12], or model curricula as prerequisite graphs to support curriculum overview [8] and to verify structural properties such as soundness and the absence of cycles [6].

3. BACKGROUND

The StEOP was initially defined in §66 of the Austrian Universities Act (AUA) of 2002 and underwent significant revisions in the 2015 amendment (BGBI. I Nr. 131/2015). The regulation mandates that all Bachelor's programs, with a few exceptions, must incorporate a StEOP. Its primary goal is to "provide students with an overview of the essential contents of the respective degree program and its further progression, and create an objective basis for making a personal assessment of his or her choice of study" (§66 AUA 2002 (1)). The StEOP must comprise between 8 and 20 credits according to the European Credit Transfer System (ECTS), which have to be completed before students can continue with more advanced courses. An exception are the pre-reg courses, non-StEOP courses students can take up to a certain credit limit before completing the StEOP. By restricting access to advanced courses, the StEOP serves as a regulatory mechanism, imposing a barrier that encourages early completion of foundational courses to engage with the study program's core concepts and expectations at an early stage. At the same time, the pre-reg courses relax the definition by providing some flexibility.

As part of a nationwide analysis in Austria in 2020 [4], standardised indicators were compiled and evaluated to compare the StEOP across eleven institutions. Results showed that 26 % of all enrolled students did not complete any StEOP examinations. Further, among those who started the StEOP, only 47 % completed it within the first semester, and 72 % within two semesters. Still, surveys showed that the StEOP mostly provided a good overview of essential study content, clarified performance expectations, and positively reinforced students' choice of study.

4. AN ILLUSTRATIVE EXAMPLE

To illustrate the structure of StEOP regulations, we consider an example from the Mechanical Engineering (ME) program at TU Wien in 2025. The English translation of the corresponding part of the curriculum definition is given in Figure 1. As shown, the StEOP consists of ten courses organised into groups, accompanied by a natural-language specification that defines the conditions for completing the StEOP: Students must complete twelve credits while meeting various conditions, including completing all mandatory courses and choosing courses from different subgroups. In addition, the curriculum defines a set of pre-reg courses, of which up to 22 credits can be taken.

5. CONCEPTUAL MODEL OF THE STEOP

To derive a formal representation of StEOP regulations, we conducted an exploratory analysis of multiple natural-language descriptions in the official curricular documents. Based on these observations, we first introduce a conceptual model of the StEOP, which we then formalise mathematically and apply to our example.

From analysing several StEOP definitions, we observe that they share the following structural elements:

(1) A set of courses included in the StEOP.
(2) A number of credits to be completed from these courses.

Additionally, they may include some of the following:

(3) A set of mandatory courses.

> As part of the Study Entry and Orientation Phase (StEOP) for the Bachelor's program in Mechanical Engineering (ME) at TU Wien, a total of 12 credits must be completed:
>
> **Mandatory courses:**
>
> 1.0 Introduction to Mechanical Engineering (INTRO)
>
> 2.0 Physics for Mechanical Engineering (PHY)
>
> **Pool 1: Mathematics Fundamentals**
>
> 6.0 Mathematics 1 (MATH)
>
> 3.0 Mathematics 1 Exercises (MATH EX)
>
> **Pool 2: Subject-Specific Fundamentals**
>
> 3.0 Fundamentals of Manufacturing Engineering (MANU)
>
> 3.0 Fundamentals of Business Administration (BA)
>
> 2.0 Technical Drawing / CAD (DRAW)
>
> 3.0 Fundamentals of Design Engineering (DESIGN)
>
> 3.0 Technical Drawing / CAD Exercise (DRAW EX)
>
> 2.0 Mechanics 1 Exercises (MECH EX)
>
> The STEOP is considered successfully completed once both mandatory courses have been passed, and a total of at least 9 credits have been obtained from Pool 1 and Pool 2 combined. At least one course must be selected from each of the two pools. Before completing the StEOP, students may take up to 22 credits of non-StEOP courses from [a list of pre-reg courses].

Figure 1: Description of the StEOP structure in ME.

(4) A subset of courses from which a specified number of credits must be earned.
(5) A subset of courses of which at least one must be completed.

Based on this, we propose modelling the StEOP as a set of *pool rules*. Each pool consists of a *set of courses* and a *minimum number of credits* that must be completed from the set. By combining multiple pool rules (including nested ones) that must all be fulfilled, we can represent all the elements identified above. Elements (1), (2), and (4) are directly captured. A set of mandatory courses (3) can be modelled as a pool in which the required number of credits equals the sum of all courses' credits. Similarly, a subgroup of courses, of which at least one must be completed (5), can be represented as a pool in which at least one credit must be earned. Since courses can only be completed in their entirety, this effectively ensures that at least one full course is completed. In addition to the StEOP course pools, curricula may define a set of pre-reg courses that students can take before completing the StEOP. While this gives students some flexibility, a maximum credit limit applies to pre-reg courses, requiring students to complete the StEOP before taking even more advanced courses.

5.1 Mathematical Formalisation

This conceptual model can be mathematically formalised as follows: Let $\mathcal{C}$ be the universal set of all available courses, where each course $c \in \mathcal{C}$ is represented as a tuple $c = (\mathrm{id}_c, \mathrm{name}_c, \mathrm{cr}_c)$ consisting of a unique course ID, a name, and its corresponding number of credits ($\mathrm{cr}_c \in \mathbb{N}$). Let $\mathcal{X}$ be the set of all available curricula. A specific curriculum $x \in \mathcal{X}$ is defined as a tuple $x = (C_x, S_x, V_x)$ where $C_x \subseteq \mathcal{C}$ is the set of all courses belonging to this specific curriculum.

The StEOP S_x for curriculum x is defined as a set of pool rules:

$$S_x = \{(P_1, k_1), (P_2, k_2), \ldots, (P_n, k_n)\}$$

where each pool $P_i \subseteq C_x$ is a subset of courses, and $k_i \in \mathbb{N}$ specifies the minimum number of credits that must be earned from pool P_i. Let $P_{S_x} = \bigcup_{i=1}^{n} P_i$ denote the set of all StEOP courses across all pools in curriculum x.

The pre-reg rules, $V_x = (W_x, m_x)$, define the additional non-StEOP courses a student may take before completing the StEOP. $W_x \subseteq C_x$ is the set of eligible pre-reg courses and $m_x \in \mathbb{N}$ is the maximum allowed credit limit from those. These courses must not overlap with any StEOP pool:

$$W_x \cap P_{S_x} = \emptyset$$

Let $\mathcal{T}$ be the set of all students. We introduce two boolean predicates: Passed(t, c) evaluates to true if a student $t \in \mathcal{T}$ has successfully completed course c, and Taken(t, c) is true if the student t has already passed or is currently registered for course c. We can now define the predicate Steop(t, x), which indicates whether student t has fulfilled all StEOP requirements for curriculum x:

$$\text{Steop}(t, x) \iff \forall (P_i, k_i) \in S_x : \sum_{\{c \in P_i \mid \text{Passed}(t,c)\}} \text{cr}_c \geq k_i$$

This states that for all pool rules in the StEOP, the sum of the credits of all passed courses within that pool must be at least the defined minimum k_i. From a theoretical perspective, this formalisation can be interpreted as a constraint satisfaction problem (CSP) over sets of courses [2]. Each pool rule corresponds to a constraint on the accumulated credits within the sets, and the StEOP is completed when all constraints are fulfilled.

Further, we can define the predicate CanEnroll(t, c, x), which defines whether a student t can enroll in course $c \in C_x$ within curriculum x as:

$$\text{CanEnroll}(t, c, x) \iff \text{Steop}(t, x) \vee c \in P_{S_x}$$
$$\vee \left(c \in W_x \wedge \sum_{\{c' \in W_x \mid \text{Taken}(t,c')\}} \text{cr}_{c'} + \text{cr}_c \leq m_x \right)$$

This states that a student can enrol in the course if they have already completed the StEOP, the course is part of the StEOP pools, or the course is a pre-reg course, and taking it does not cause the sum of credits of all taken pre-reg courses to exceed the maximum allowed credit limit.

5.2 Applying it to our Example

Applying this model to the ME study program at TU Wien, the StEOP can be modelled using four pools, as shown in Figure 2. The mathematical and subject-specific fundamentals are represented as two pools (each requiring 1 credit), while their combination forms an elective pool requiring 9 credits. The mandatory component is modelled as a pool with a 3-credit requirement, equal to the sum of all included courses. While an overarching pool with a 12-credit requirement could be added, this constraint is already implicitly covered. This formal model precisely maps the textual requirements shown in Figure 1.

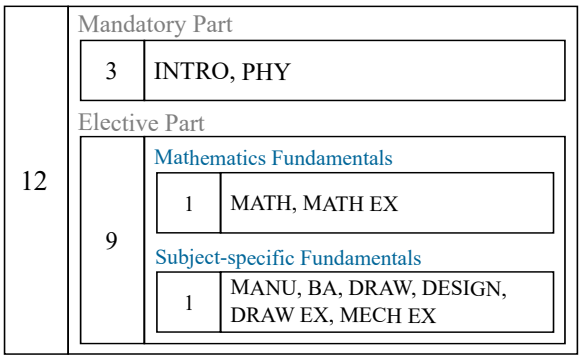


Figure 2: Visualised formalisation of the StEOP in ME.

6. MODEL VALIDATION

To validate the proposed model, we aim to formalise the StEOPs of all curricula at TU Wien respectively. Due to the ambiguity of many pre-reg rules, we focused only on the StEOP rules.

6.1 Semi-Automated Rule Extraction

Since StEOP regulations are provided exclusively in textual form and do not follow a strictly standardised structure, we opted for a semi-automated, interactive extraction approach. First, all relevant curriculum documents (PDFs) are automatically collected from the institution's website. Due to their broadly consistent structure, these documents can be processed uniformly, allowing the StEOP sections to be automatically identified and extracted. The extracted StEOP descriptions are presented to the user in an interactive command-line interface, where the user then encodes the corresponding pool rules through a sequence of simple commands. A pool is created by specifying its name, whether it is mandatory, and the minimum number of credits required. Subsequently, the courses in the pool are entered line by line in a dedicated input mode, including the course name, course type (e.g., lecture or exercise), and number of credits. These course descriptions can often be copied directly from the StEOP description, as they most often follow the same pattern. Lastly, the course names are automatically matched to the corresponding course identifiers. The described semi-automated pool rule extraction process was performed independently by two researchers for all 20 study programs at TU Wien since 2017.

6.2 Results

Across the 20 study programs and their curriculum versions at TU Wien, a total of 46 distinct StEOP regulations were found. Extracting the rules took about two minutes per StEOP. All but one could be fully modelled using the proposed formalisation, achieving a coverage of 98 %. The exception being the StEOP for Computer Engineering from 2023, which contains an elective pool rule that prohibits combining three included courses. Since the proposed formalisation only allows specifying minimum credit requirements, it cannot represent such exclusions. Capturing this rule would require extending the model to include exclusion criteria.

Comparing the extracted pool rules showed that both re-

searchers always identified the same underlying structures. Some differences emerged in naming conventions for pools: one researcher adhered more closely to the terminology used in the official curriculum, while the other preferred a more functional naming scheme. Further, cases in which at least one course from a subgroup has to be completed were sometimes interpreted as requiring the exact minimum number of credits for a course, whereas in other cases they were modelled as a single-course constraint requiring one credit. Similarly, mandatory courses were sometimes additionally included in the elective pool, leading to slight variations in how credit requirements were parameterised. Despite these differences in representation, the resulting models were semantically equivalent and thus equally valid.

During the modelling process, several practical challenges emerged, including missing information (e.g., credit values), underspecified courses, and inconsistencies between course names in the StEOP descriptions and the module handbook. Additionally, some descriptions were inherently ambiguous. For example, the requirement: *"Completion of at least 10.5 credits from the StEOP pool, including at least 3.5 credits from the sub-pool 'Mathematics Basics', and successful completion of the course 'Introduction to Mathematical Work'"* raises the question of whether the latter course is included in the 10.5 credit requirement or must be completed in addition. However, both researchers reached the same conclusion about how to interpret this ambiguity.

A preliminary analysis of the 46 investigated StEOP regulations identified a total of 131 pool rules, of which 43 are mandatory and 88 elective. Each StEOP comprises between one and five pool rules, with most consisting of multiple interacting requirements rather than a single constraint (on average 2.85 pools). Despite this variability, recurring structural patterns can be observed: Four StEOPs consist exclusively of mandatory pools and therefore offer no choice to students, while three consist solely of an unrestricted elective pool, providing maximum flexibility. Ten StEOPs follow a simple structure that combines one mandatory and one elective pool. The most common structure, observed in 20 cases, extends this combination by introducing additional constraints within the elective pool via subpools. These typically require students to complete at least one course from a specific topic area, such as mathematics, or from a particular course type (e.g., lecture or exercise). Furthermore, nine StEOPs combine a mandatory pool with two non-overlapping elective pools. These structures enforce coverage of different subject areas while still allowing individual choice, often balancing subject-specific and foundational or mathematics components. In six of these cases, additional constraints are imposed within the elective pools.

6.3 Discussion

The analysis shows that the pool rule abstraction reliably captures the core structure of StEOP requirements. Only a single special case required an additional constraint that could not be directly modelled with the proposed StEOP formalisation, but which is already addressed by broader curriculum rules. This demonstrates that the proposed formalisation is generally applicable and robust, with the identified exception highlighting a potential extension point for the model without undermining its overall usefulness. Further, the high level of agreement between the two researchers indicates that the proposed modelling approach is robust, even when applied to partially ambiguous curriculum descriptions. While minor differences in naming conventions and modelling decisions were observed, these did not affect the semantic equivalence of the resulting representations.

In addition, the analysis highlights the structural complexity of StEOP regulations. Rather than consisting of a single requirement, most StEOPs comprise multiple interacting constraints, typically combining mandatory and elective components. These are often further restricted by additional conditions, such as selecting at least one course from a specific type or topic area. This structure allows for combining foundational knowledge with some flexibility for individualisation. Although the StEOP regulations are defined independently by different study programs, recurring patterns can be observed, suggesting that they follow implicit design principles.

The formalisation process also revealed challenges. Natural language curriculum descriptions can lack precision, leading to ambiguities in interpretation. While both researchers arrived at semantically equivalent models, some cases allowed multiple plausible interpretations, requiring subjective decisions. This highlights the need for clearer, more standardised, and machine-readable formulations of curriculum rules [9]. While the results demonstrate the general applicability of the proposed model, several limitations remain. The analysis is based on a relatively small sample of 46 StEOP regulations and relies on manual annotation by two researchers, which may limit generalisability. In addition, certain regulatory aspects, such as exemptions due to previously completed courses, are not captured by the current model. Addressing these limitations provides directions for future work. In particular, automating the extraction of the regulations and validating the model against real student data would enable generalisability and allow assessment of how well the formalised rules reflect actual enrolment behaviour. In addition, the proposed formalisation enables several practical applications. For instance, it allows automatically validating whether a planned set of courses satisfies curriculum constraints, thereby supporting study planning tools and reducing invalid course selections. In addition, the model can be used for automated study-planning tools and recommender systems to generate valid study paths under given constraints or to analyse how different curriculum designs affect student progression. By making regulatory structures explicit, it also facilitates their integration into learning analytics systems, where they can be used to contextualise and interpret student behaviour.

7. CONCLUSION

This paper introduces a formal model for representing Study Entry and Orientation Phase (StEOP) regulations as constraint satisfaction problems (CSP), based on nested pool rules over sets of courses with credit-based requirements (RQ1). The model is evaluated on all 46 unique StEOPs at TU Wien since 2017 by applying a semi-automated extraction approach. The results show that all but one analysed regulations can be consistently mapped to the proposed representation, despite being only specified in natural language and lacking standardisation, demonstrating the expressive-

ness of the model (RQ2). Beyond this, the preliminary analysis revealed recurring structural patterns in curriculum design that are not directly visible in the textual descriptions alone, underlining the need for more standardised and transparent curriculum formulations. The presented formalisation of these curriculum rules not only helps identify such patterns and enable their systematic analysis, but it also supports improvements in curriculum design and enables downstream applications, such as study planning tools. Overall, by formalising the StEOP and evaluating its applicability, this work represents a first step toward transforming curriculum regulations into machine-readable artefacts, facilitating their analysis and integration into EdTech. In this way, the proposed approach contributes to more transparent and data-informed support for students, particularly in early study phases.

8. REFERENCES

[1] P. R. Aldrich. The curriculum prerequisite network: Modeling the curriculum as a complex system. *Biochemistry and Molecular Biology Education*, 43(3):168–180, 2015.

[2] S. C. Brailsford, C. N. Potts, and B. M. Smith. Constraint satisfaction problems: Algorithms and applications. *European journal of operational research*, 119(3):557–581, 1999.

[3] D. Giannakopoulou, T. Pressburger, A. Mavridou, and J. Schumann. Automated formalization of structured natural language requirements. *Information and Software Technology*, 137, 2021.

[4] S. Humpl, K. Scherz, J. Fischer, A. Dibiasi, and M. Unger. Evaluierung der studieneingangs-und orientierungsphase (steop) nach § 66, 2020.

[5] M. Pechenizkiy, N. Trcka, P. De Bra, and P. Toledo. Currim: curriculum mining. In *Proceedings of the 5th International Conference on Educational Data Mining 2012*, pages 216–217. International Educational Data Mining Society (IEDMS), 2012.

[6] E. Roanes-Lozano, A. Martínez-Zarzuelo, and M. J. Fernandez-Diaz. An application of knowledge engineering to mathematics curricula organization and formal verification. *Mathematical Problems in Engineering*, 2020(1), 2020.

[7] R. Roepke, S. Judel, and U. Schroeder. Study path analyses for quality assurance and support of study planning. *Informatik Spektrum*, 47:97–104, Sept. 2024.

[8] M. Trippel and R. Röpke. Developing a Graph-based Visualization of Elective Courses to Support Course Selection in Higher Education. In V. Köppen and S. Strickroth, editors, *23. Fachtagung Bildungstechnologien (DELFI 2025)*, pages 89–95, Bonn, 2025. Gesellschaft für Informatik e.V.

[9] M. von der Heyde, C. Otunuya, M. Goebel, D. Zoerner, and U. Lucke. Automatic and interactive validation of study regulations in accreditation processes of higher education institutions. In *Proceedings of European University Information Systems Congress 2023*, volume 95 of *EPiC Series in Computing*, pages 320–331. EasyChair, 2023.

[10] M. Wagner, H. Helal, R. Roepke, S. Judel, J. Doveren, S. Goerzen, P. Soudmand, G. Lakemeyer, U. Schroeder, and W. M. P. van der Aalst. A Combined Approach of Process Mining and Rule-Based AI for Study Planning and Monitoring in Higher Education. In M. Montali, A. Senderovich, and M. Weidlich, editors, *Process Mining Workshops*, Lecture Notes in Business Information Processing, pages 513–525, Cham, 2023. Springer Nature Switzerland.

[11] K. Wu and W. S. Havens. Modelling an academic curriculum plan as a mixed-initiative constraint satisfaction problem. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 79–90. Springer, 2005.

[12] A. V. Zykina, O. N. Kaneva, and V. V. Munko. Developing a curriculum modeling approach. In *CEUR Workshop Proceedings*, volume 2642, 2020.

Faster, Cheaper, More Accurate: Specialised Knowledge Tracing Models Outperform LLMs

Prarthana Bhattacharyya*
Eedi
prarthana.bhattacharyya@eedi.com

Joshua Mitton*
Eedi
joshua.mitton@eedi.com

Ralph Abboud
Renaissance Philanthropy
rabboud@levimath.org

Simon Woodhead
Eedi
simon.woodhead@eedi.com

ABSTRACT

Predicting future student responses to questions is particularly valuable for educational learning platforms where it enables effective interventions. One of the key approaches to do this has been through the use of knowledge tracing (KT) models. These are small, domain-specific, temporal models trained on student question-response data. KT models are optimised for high accuracy on specific educational domains and have fast inference and scalable deployments. The rise of Large Language Models (LLMs) motivates us to ask the following questions: (1) How well can LLMs perform at predicting students' future responses to questions? (2) Are LLMs scalable for this domain? (3) How do LLMs compare to KT models on this domain-specific task? In this paper, we compare multiple LLMs and KT models across predictive performance, deployment cost, and inference speed to answer the above questions. We show that KT models outperform LLMs with respect to accuracy and F1 scores on this domain-specific task. Further, we demonstrate that LLMs are orders of magnitude slower than KT models and cost orders of magnitude more to deploy. This highlights the importance of domain-specific models for education prediction tasks and the fact that current closed source LLMs should not be used as a universal solution for all tasks.

Keywords

Large Language Model (LLM), Knowledge Tracing (KT), Performance, Inference, Deployment Cost

1. INTRODUCTION

Predicting student performance is a fundamental and well studied task in educational data mining. The performance of predictive models has important implications for adaptive learning systems and personalised instruction. In mathematics education, students often exhibit recurring patterns of misunderstanding that lead to predictable errors [1]. If these misconceptions can be anticipated, educators can adapt their instruction and learning platforms can deliver more targeted support. This student misconception identification challenge can be framed as a prediction problem: given a student's response history to past questions, the goal is to forecast their performance on future questions.

Knowledge Tracing (KT) models have been developed specifically to address this challenge [2]. These compact, domain-specific models are trained on student interaction data. They are optimised for high accuracy, fast inference, and scalable deployment. Recent advances include Deep Knowledge Tracing (DKT) [8], which applies recurrent neural networks to model student knowledge states, and Self-Attentive Knowledge Tracing (SAKT) [7], which leverages attention mechanisms to capture dependencies in student response sequences.

The recent success of Large Language Models (LLMs) across diverse tasks, including mathematical reasoning and problem solving [5], raises a natural question: can general-purpose LLMs match or exceed the performance of specialised KT models on student response prediction? This question is particularly relevant given that predicting student performance on mathematics problems requires both reasoning ability and mathematical understanding. It remains unclear whether these LLM capabilities transfer effectively to modelling individual student knowledge states over time. It is also unclear whether LLMs can do so at a cost and speed suitable for real-world deployment.

In this paper, we present a systematic comparison of KT models and LLMs to address three research questions: (1) How well can LLMs predict students' future responses to questions? (2) Are LLMs scalable for this domain in terms of latency and cost? (3) How do LLMs compare to specialised KT models on this task? We evaluate models on a binary classification task, where the model is tasked to predict whether a student will answer correctly or incorrectly. We do this comparison across three dimensions: predictive performance, inference latency, and deployment cost.

Our findings demonstrate that specialised KT models con-

*Equal contribution

Prarthana Bhattacharyya, Joshua Mitton, Ralph Abboud, and Simon Woodhead. Faster, Cheaper, More Accurate: Specialised Knowledge Tracing Models Outperform LLMs. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 674–678. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039796

sistently outperform LLMs across all three dimensions. KT models achieve higher accuracy and F1 scores, are orders of magnitude faster (sub-second vs. minutes per student), and cost 600-12,000 times less to deploy at scale. These results highlight the continued importance of domain-specific architectures for educational prediction tasks and caution against adopting general-purpose LLMs as a universal solution in this domain.

2. EXPERIMENTAL PROTOCOL

2.1 Task Formulation

The task we consider in this paper is a binary classification task where the models are tasked with predicting if a student will answer a future question correctly or incorrectly, given their answering history. The models have access to the question text or a question id and the construct text or construct id. This is shown in Table 1. In addition, for each question, we also have misconception text for each possible answer and explanations of the answers demonstrated in Table 2. To avoid cold-start issues for each model we took the decision to not make predictions on the first 10 questions. When predicting the answer to a question the model has access to the student's question-answer history. We decided on a representative load of 100k students requiring 40 answer predictions each to test the models on.

Table 1: Example Question and Construct Data

Component	Example
Question ID	100
Question Text	What is $2 + 3 \times 4$? (A: 20, B: 14, C: 11, D: 26)
Construct ID	10
Construct Text	Order of Operations

Table 2: Example Misconception and Explanation Data

Component	Example
Question Text	6 pencils cost £ 1.50 How much do 3 pencils cost?
Answer Text	0.75p
Misconception Text	Confuses decimal notation in money putting value in pounds but writing the unit for pence
Explanation Text	I think you have used the incorrect notation for money. Consider how the monetary values in the question are written

2.2 Datasets

In Table 3 we detail some statistics about the datasets used throughout this work. The dataset is a real-world dataset extracted from an online learning platform. Notably, the training dataset is only used for the KT models or fine-tuned LLMs. The validation dataset is what is used to report model performance. For all datasets we restricted the number of responses for a student to 50, where the model is only tasked with predicting the final 40 question responses. The dataset bias is the percentage of correct responses. The students in the training and validation datasets are different and student data used to train the model is not used for validation. This means any trained models are being tested for their generalisability to new students. The questions used in the training and validation set are the same; there are less questions in the validation set due to the smaller dataset size and no student in that set having answered some questions.

Table 3: Dataset Statistics

Metric	Train	Val
# Responses	512,000	64,000
# Students	12,800	1,600
# Questions	4,252	4,104
# Responses per student	40.0	40.0
Dataset bias	65.8%	66.5%

2.3 Metrics

The two metrics we use to assess the predictive performance of the models are accuracy and F1 score. Accuracy is the most common metric used to measure model performance in KT model research. It provides an overall value for how often the model is correct and can be used to see if a model is beating the dataset bias. The equation for accuracy is given by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{1}$$

The F1 score can be more valuable than accuracy when a dataset is not balanced, as it considers both how often a model successfully finds correct or incorrect answers via recall, and how often the prediction was correct via precision. In this work we consider a macro-averaged F1 score to assess how well does the perform across both correct and incorrect response prediction. The equation we use for F1 score is given by

$$\text{F1} = \frac{1}{2} \sum_{i=0}^{1} \frac{2TP_i}{2TP_i + FP_i + FN_i}. \tag{2}$$

In both equations TP is the true positives where the model predicted the positive class correctly, TN is the true negatives where the model predicted the negative class correctly, FP is the false positives where the model incorrectly predicted the positive class, FN is the false negatives where the model incorrectly predicted the negative class, and i is the class being considered. Here we deem class 0 as the class of correct responses and class 1 as the class of incorrect responses.

2.4 Models

Deep Knowledge Tracing (DKT) [8] uses question ids and student answering history as model inputs and predicts if a student answers correctly or incorrectly on future questions. Self-Attentive Knowledge Tracing (SAKT) [7] is a self-attention transformer-style model. SAKT similarly uses question ids and student answering history as model inputs and predicts if a student answers correctly or incorrectly on future questions. SemKT is our custom-built encoder-decoder temporal transformer. But this model uses question, construct, explanation, and misconception text as model inputs and uses a Qwen 3 0.6B embedding model to map the text inputs into an embedding space that the KT model uses as an

input. Importantly, we only use Qwen as a feature extractor to obtain fixed vector representations of questions, construct, explanation and misconception text. These embeddings are computed once and cached offline. At inference time, no LLM is involved in the actual student performance prediction. The temporality and modelling of student understanding is handled by our small, custom temporal transformer.

Gemini 2.5 flash lite [3] and GPT 4o mini [6] are both larger closed source LLMs. The specific details of these models are therefore not known but we can estimate their approximate size from online sources. For these models the student history is passed to the model through the input prompt. The input to the LLMs includes question ids, question text, construct ids and construct text. Llama 1B [4] and Qwen 2.5 7B [9] are both open source LLM models. We similarly pass the student history to these models through the input prompt. For Llama 1B LoRA fine-tune we add a small percentage of extra parameters to fine-tune the model for the specific task of predicting students answers to future questions from their answering history. While the KT and LLM models operate on different inputs, they reflect the practical constraints and strengths of each model class.

2.5 Prompting Strategies

For student answer prediction, we employ a constrained prompt design that prioritizes deterministic parsing while providing rich contextual information. Our prompt follows this structure: "The student answered the following questions: Question ID [id] ([question text with choices A/B/C/D]), with construct ID [id] ([construct description]): answered [correctly/incorrectly]... Predict whether the student will answer [new question with full context] correctly or not. Answer ONLY with the word 'Yes' or the word 'No'."

2.6 Testing Setup

The testing of models is split into models that we would deploy ourselves and models that are closed source LLMs that we access via an API. For the models that we access via an API, we report the cost and latency values from using their API. For models that we deploy ourselves, we test the models on a CPU only Azure instance. This instance is a standard DS3 instance that has 4 vcpus and 14GiB of memory. We measure the latency of the model in a deployment setting where the model runs inference on one sample at once, although it should be noted that this could be sped up by increasing the batch size. We calculate the deployment costs based on an hourly rate of $0.27 per hour of the instance.

3. RESULTS AND DISCUSSION

3.1 Model Performance

Accuracy measures the proportion of correct predictions. SemKT achieves the highest accuracy at 72.8%, narrowly outperforming both SAKT [7] (72.7%) and DKT [8] (71.8%), two well-established knowledge tracing models. Interestingly, all domain-specific KT models outperform the general-purpose LLMs like GPT-4o-mini (58.6%), Qwen2.5-7B Instruct (64.6%), and Gemini-2.5-flash-lite (66.5%). We also benchmark two Llama-1B variants: a LoRA fine-tuned Llama-1B (71.0%), while Llama-1B zero-shot performs substantially

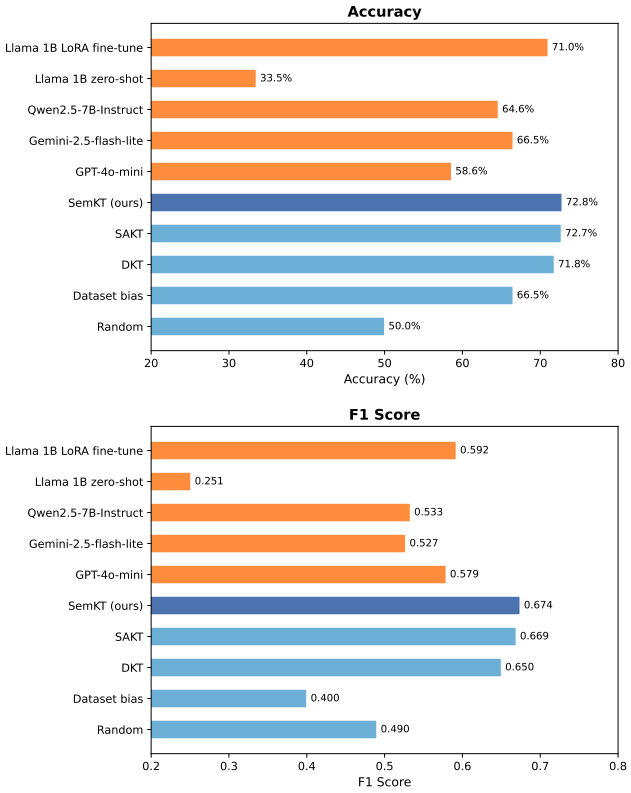


Figure 1: Comparison of model performance.

worse at 33.5%. This reinforces the value of specialised models trained on structured educational data for this task. We refer the reader to Figure 1 for a side-by-side comparison.

For context, we also include a dataset bias baseline that simply predicts the average correctness rate for each question While it does not personalise predictions, it achieves 66.5% accuracy. Notably, several LLMs in our benchmark, despite their billions of parameters, fail to surpass this naive baseline. This makes it a critical baseline any viable model must beat.

Precision measures how often the model is correct when it predicts a student will answer correctly. Recall measures how many of the actually correct answers the model managed to identify. Improving one typically comes at the cost of the other. The F1 score captures this trade-off by combining both into a single measure, rewarding models that perform well on both rather than excelling at one while neglecting the other. The F1 score is therefore more informative when dealing with class imbalance. Once again, SemKT leads with an F1 score of 0.674, followed closely by SAKT (0.669) and DKT (0.650). The LLMs perform lower in this metric as well, with GPT-4o-mini scoring 0.579, Qwen2.5-7B-Instruct 0.533, and Gemini-2.5-flash-lite 0.527. Among the Llama baselines, the Llama-1B LoRA fine-tune improves to 0.592, whereas Llama-1B zero-shot remains low at 0.251. These results affirm that general-purpose LLMs struggle with student-level predictions, likely due to the lack of fine-grained educational supervision.

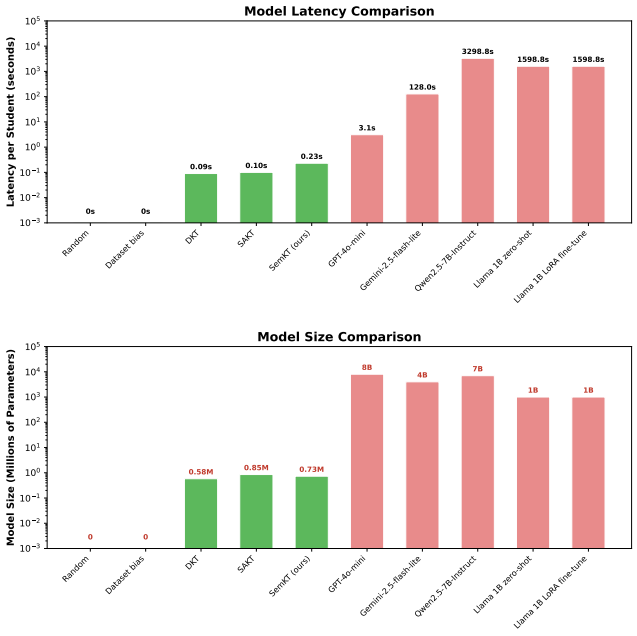


Figure 2: Latency and model size comparison for different models.

The LLMs exhibit different failure modes depending on their design. Llama-1B zero-shot frequently fails to follow the prompt format, producing responses other than the expected 'correct" or 'incorrect," which we treat as incorrect predictions. The remaining LLMs (GPT-4o-mini, Gemini-2.5-flash-lite, and Qwen2.5-7B-Instruct, Llama-1B fine-tuned) adhere to the prompt format but struggle to predict student responses accurately. Notably, despite receiving ten initial question-answer pairs to mitigate cold-start effects, these models fail to infer students tend to get more answers correct in general. In contrast, specialised KT models are trained to leverage such properties of the data, enabling them to make more informed predictions about future student responses.

Our results show that specialised Knowledge Tracing models outperform general-purpose Large Language Models when it comes to predicting student responses. Across both metrics, domain-specific models like our SemKT, SAKT, and DKT consistently deliver stronger performance than smaller LLMs such as GPT-4o-mini, Gemini-2.5-flash-lite, Qwen2.5-7B-Instruct, and Llama-1B. They remain robust even against a fine-tuned small LLM baseline (Llama-1B LoRA). This highlights the importance of task-specific architectures in educational prediction tasks. While LLMs excel at general reasoning, purpose-built KT models remain the most effective and reliable solution for identifying student misconceptions.

3.2 Latency and Model Size

We also compare models on latency (inference time per student) and model size (number of parameters) in Figure 2. These are critical factors for real-time applications and large-scale deployment in educational learning platforms. We note that closed-source LLMs run on different hardware and are accessed via an API.

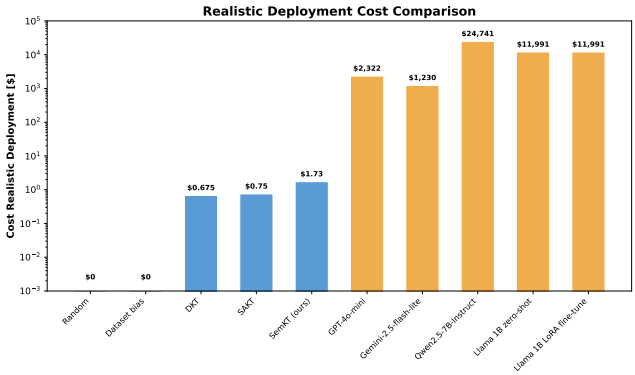


Figure 3: Annual inference cost comparison for 100,000 students receiving 40 predictions per year.

Traditional KT models like DKT, SAKT, and our SemKT deliver fast predictions, with latencies under 0.25 seconds per student. In contrast, LLMs are orders of magnitude slower. GPT-4o-mini takes 3.1 seconds, Gemini-2.5-flash-lite takes 128 seconds, and Qwen2.5-7B-Instruct requires 3,299 seconds per student. Llama-1B variants also take 1,598.8 seconds per student. Despite this latency increase, none of the LLMs outperform the KT models in accuracy.

Specialised KT models are also extremely compact, with model sizes ranging from 0.58M to 0.85M parameters. Our SemKT model contains only 0.73M parameters while offering strong performance. In contrast, general-purpose LLMs are vastly larger: GPT-4o-mini has 8B parameters, Gemini-2.5-flash-lite has 4B, Qwen2.5-7B-Instruct has 7B, and Llama-1B variants have 1B parameters.

This substantial efficiency gap in terms of latency and model size further reinforces that specialised KT models are far better suited for scalable, real-time prediction tasks. This is especially important for deployment in resource-constrained environments, where lightweight prediction models can enable educational tools to reach learners at a global scale.

3.3 Cost Analysis

Cost is a major consideration for real-world deployment at scale, especially for education platforms aiming to support thousands of students. Figure 3 shows the annual inference cost for 100,000 students, each receiving 40 predictions per year. For most LLM models, we benchmarked on 1,600 students with 40 questions each (64,000 predictions) and extrapolated to 100,000 students. For Gemini-2.5-flash-lite, daily rate limits of 10,000 requests per day on Tier-1 restricted us to 200 students, from which we extrapolated accordingly.

Specialised KT models like DKT, SAKT, and our SemKT cost less than $2 per year to serve this workload. In contrast, general-purpose LLMs are vastly more expensive: GPT-4o-mini costs approximately $2,322 per year, Gemini-2.5-flash-lite costs $1,230 per year, Llama-1B variants cost $11,991 per year, and Qwen2.5-7B-Instruct reaches $24,741 per year. This means KT models are 615–12,400 times cheaper than

LLMs for the same task, while also offering higher accuracy. For any large-scale EdTech deployment, this dramatic cost gap makes specialised KT models the better choice.

4. CONCLUSION

Despite the rapid rise of general-purpose LLMs, our findings demonstrate that specialised KT models remain the best choice for predicting student responses in EdTech settings. KT models outperform LLMs (GPT-4o-mini, Gemini-2.5-flash-lite, Llama-1B, and Llama-1B-LoRA-finetune) on accuracy and F1 score, offering better future response predictions. KT models scale to larger numbers of students at a lower cost to EdTech providers due to being over 600 times cheaper to deploy at scale, while offering millisecond-level latency without requiring GPUs. While LLMs excel at broad reasoning tasks, they fall short when applied to student interaction data.

This is not to say LLMs have no place in pedagogical settings; however, they should be employed where they demonstrably reduce costs or improve learning outcomes, not as a default choice. To uncover misconceptions for the greatest number of learners, a scalable, real-time system is required. Currently, purpose-built KT models remain the most effective and practical solution.

5. REFERENCES

[1] J. S. Brown and R. R. Burton. Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science*, 2(2):155–192, 1978.

[2] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[3] G. DeepMind. Gemini 2.5: Our most intelligent ai model. `https://deepmind.google/technologies/gemini/`, 2025.

[4] A. Grattafiori, A. Dubey, A. Jauhri, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[5] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[6] OpenAI. Gpt-4o mini: Advancing cost-efficient intelligence. `https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/`, 2024.

[7] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*, pages 384–389. International Educational Data Mining Society, 2019.

[8] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, pages 505–513, 2015.

[9] Q. Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

From Tool to Teammate: LLM Coding Agents as Collaborative Partners for Behavioral Labeling in Educational Dialogue Analysis

Eason Chen
Carnegie Mellon University
eason.tw.chen@gmail.com

Nina Yuan
Carnegie Mellon University
Mingyuyu@andrew.cmu.edu

Isabel Wang
Carnegie Mellon University
yumow@andrew.cmu.edu

Sophia Judicke
Carnegie Mellon University
sjudicke@andrew.cmu.edu

Kayla Beigh
Carnegie Mellon University
kbeigh@andrew.cmu.edu

Xinyi Tang
Carnegie Mellon University
xtang2@andrew.cmu.edu

ABSTRACT

We present a methodology where LLM coding agents autonomously improve classification prompts for behavioral labeling of educational dialogues. In each iteration, a coding agent analyzes disagreements between classifier output and human labels, then proposes theory-grounded prompt modifications. Applying this to 659 AI tutoring sessions with three agents and three classifiers, 4-fold cross-validation indicated non-trivial gains on held-out data: the best agent improved from baseline $\kappa = 0.70$ to test $\kappa = 0.78$ (SD= 0.12), at \$5–8 per agent in Experiment 2. The process also surfaced undocumented labeling patterns, and continued iteration beyond the optimum led to regression. Iteration logs, prompts, and evaluation scripts are available at: https://github.com/EasonC13/EDM26-Agents.

Keywords

LLM labeling, behavioral analytics, prompt engineering, AI tutoring, educational data mining

1. INTRODUCTION

Understanding student behavior in tutoring dialogues is fundamental to improving educational technology. Detecting patterns such as help-seeking, confusion, and disengagement enables researchers to evaluate tutoring effectiveness and design adaptive interventions [3, 1, 7, 8]. Yet behavioral labeling remains a costly bottleneck: trained coders must manually analyze hundreds of sessions, a process requiring weeks of effort and limiting the scale of educational dialogue research. LLMs offer a promising solution, achieving strong performance on qualitative coding tasks [12, 11]. However, developing effective prompts still relies heavily on researcher trial-and-error over days or weeks.

We present an approach where LLM coding agents serve as *research teammates* that iteratively improve labeling prompts through systematic error analysis. In an exploratory phase, a researcher collaborated with Claude Code over 21 iterations, improving Follow-up Type κ from 0.59 to 0.68 at a cost of \$22 and surfacing a key insight: human coders consistently treated expressions of confusion ("I don't know," "I'm stuck") as engagement rather than disengagement. To validate generalizability, we assigned the same task to three independent coding agents (OpenAI Codex, Claude Code, Gemini), each starting from scratch. On the development set, all three improved substantially, with Codex and Gemini reaching $\kappa = 0.93$. Since iterating on a fixed validation set risks overfitting, our primary evaluation uses 4-fold cross-validation: the best agent achieved held-out test $\kappa = 0.78$ (SD= 0.12), improving over the $\kappa \approx 0.70$ baseline, at \$5–8 per agent. Agents independently converged on similar disambiguation rules, suggesting the methodology identifies genuine patterns. We also tested three classifier models (GPT, Claude, Gemini), finding that same-family advantages manifest as faster convergence rather than higher ceilings in development-set analyses.

Our contributions are a collaborative agent-refinement workflow, held-out evidence of non-trivial generalization gains, an analysis of regression beyond the optimal iteration, and a demonstration that disagreement analysis can surface implicit labeling criteria.

2. METHODOLOGY

Our work builds on help-seeking research in tutoring dialogues [3, 1], LLM-based qualitative coding in education [12, 11, 18, 20], human-in-the-loop educational judgment workflows [13, 2, 9, 10, 17], AI agents in education [5, 4, 6], and automated prompt optimization methods such as APE, DSPy, and OPRO [21, 15, 19]. Unlike metric-only prompt search, our workflow uses coding agents to perform qualitative error analysis and propose theory-grounded revisions under researcher oversight. More specifically, prior work often studies LLMs as collaborators in grading, feedback, or assessment, whereas far less work examines how reliably coding-agent-style systems can autonomously iterate on heuristic rules or prompts for theory-grounded labeling tasks. This distinction matters for educational coding tasks, where prompt changes can improve a metric while weakening pedagogical

Eason Chen, Yumo Wang, Mingyu Yuan, Sophia Judicke, Kayla Beigh, and Xinyi Tang. From Tool to Teammate: LLM Coding Agents as Collaborative Partners for Behavioral Labeling in Educational Dialogue Analysis. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 679–683. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039766

Algorithm 1 Agent-Driven Iterative Prompt Refinement

Require: Labeled validation set D, initial prompt P_0, classifier M, coding agent A

1: **for** $i = 1$ to max_iterations **do**
2: Run classifier M with prompt P_{i-1} on D
3: Compute κ and inspect disagreement patterns
4: Agent A proposes theory-grounded prompt changes
5: Researcher reviews and approves valid changes
6: $P_i \leftarrow$ revised prompt
7: **end for**
8: **return** $P^* = \arg\max_{P_i} \kappa(P_i, D)$

interpretation.

A coding agent receives a codebook, human-labeled validation sessions, and an evaluation script. In each iteration, it runs the classifier (temperature= 0), computes Cohen's κ, analyzes disagreements against human consensus, and proposes prompt revisions. The classifier is the model producing labels; the coding agent is responsible for error analysis and prompt revision. The researcher reviews proposed changes to preserve construct validity and pedagogical faithfulness.

We used chatbot interaction data from an undergraduate discrete mathematics course (∼120 students) at a large research university. The course deployed an LLM-powered tutoring system with a "Socratic guardrail": the chatbot used hints and guiding questions rather than direct answers. This setting creates natural behavioral variation, as some students engaged with the scaffolding while others attempted to extract solutions, making the labeling task both challenging and educationally meaningful.

Our dataset comprises 659 sessions collected over two semesters (Fall 2024 and Spring 2025), ranging from brief clarification requests (1–2 exchanges) to extended problem-solving dialogues (20+ exchanges). The median session length was 6 exchanges (IQR: 3–12). Topics spanned logic and propositional reasoning (28%), proof techniques (35%), set theory (22%), and combinatorics (15%). For validation, we randomly sampled 80 sessions stratified by length and topic. Two trained research assistants independently labeled these sessions after two calibration meetings to align on codebook interpretation and edge cases; the first author resolved disagreements through case-by-case discussion, documenting reasoning for each adjudication. This study was approved by our institution's Institutional Review Board (IRB); all student data were collected with informed consent and de-identified prior to analysis.

Drawing on help-seeking theory [1], we defined three labeling dimensions:

- **Student Intent** (Answer-Seeking, Help-Seeking, Other): captures whether the student primarily wants the answer or genuinely seeks to understand.
- **Topic Type** (Conceptual, Procedural): categorizes the mathematical content as definitional or problem-solving.
- **Follow-up Type** (Engage, Escalate, Switch): characterizes the student's response to guidance. Engage includes expressions of confusion ("I don't know"), since admitting confusion signals continued cognitive engagement [14]. Escalate requires an explicit request for direct answers, while Switch indicates topic abandonment.

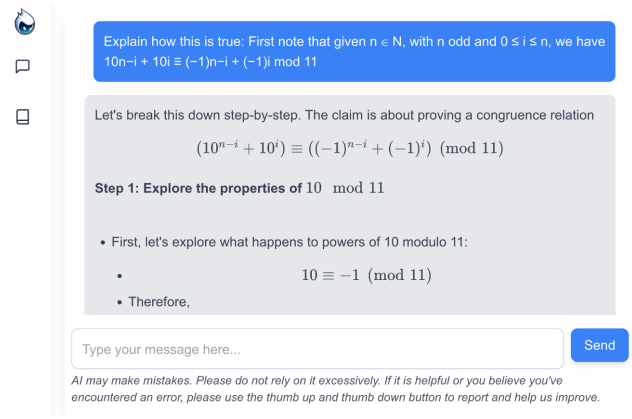


Figure 1: Example tutoring dialogue. Such sessions were labeled for student intent, topic type, and follow-up behavior.

Human coders achieved substantial inter-rater reliability: Intent $\kappa = 0.78$, Topic $\kappa = 0.73$, Follow-up $\kappa = 0.70$ [16]. Overall $\kappa = 0.89$ was computed over composite labels. To avoid mixing metrics, we compare overall model performance only against the human composite $\kappa = 0.89$, and per-dimension results only against the corresponding human kappas.

3. EXPERIMENTS AND RESULTS

3.1 Cross-Agent Validation

To test whether the methodology generalizes across agent architectures, we assigned the labeling task to three independent coding agents: OpenAI Codex, Gemini Code Agent, and Claude Code. Each received identical instructions and access to the codebook, the 80 labeled sessions, and the evaluation script. They worked independently, without access to each other's solutions. All used GPT-5.2 as a classifier for comparability. We kept the classifier fixed because the research question is whether coding agents can improve prompts for a downstream labeling model; using the agent itself as the classifier would conflate prompt-refinement ability with label-generation ability.

Starting from baseline $\kappa \approx 0.70$, after approximately 10 iterations ($5–8 per agent in Experiment 2), all three achieved strong development-set reliability. Table 1 presents the results: Codex and Gemini tied at $\kappa = 0.93$, with Gemini achieving the highest F1 (0.91). Figure 2 shows the progression, including regression when Codex was extended beyond its v7 optimum through v10 (κ dropped to 0.91, with Follow-up Type declining from 0.83 to 0.75), demonstrating that more iterations do not always yield improvement. The hardest dimension was Follow-up Type: across agents, much of the eventual gain came from converging on the rule that confusion expressions should default to Engage unless paired with an explicit request for direct answers.

Table 1: Cross-agent validation results (development set). Best performance shown for each agent.

Agent	Ver.	Overall κ	F1	Intent κ	Topic κ	Follow κ	Cost
Codex	v7	**0.93**	0.88	0.84	**0.85**	0.83	$8
Gemini	v9	**0.93**	**0.91**	**0.85**	0.81	**0.87**	$7
Claude Code	v5	0.91	0.79	0.73	0.78	0.88	$7
Human	–	0.89	–	0.78	0.73	0.70	–

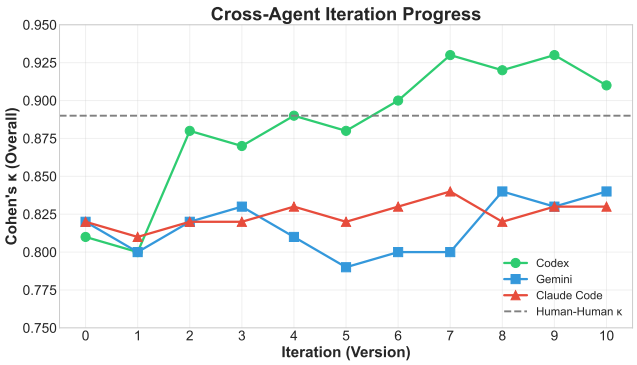


Figure 2: Overall κ progression across iterations. Codex and Gemini both reached $\kappa = 0.93$. Continued iteration led to regression for Codex after v7. Dashed line: human baseline ($\kappa = 0.89$).

Table 2: 4-fold cross-validation results (held-out test κ). Baseline $\kappa \approx 0.70$ for all agents.

Agent	Fold 0	Fold 1	Fold 2	Fold 3	Mean ± SD
Codex	0.85	0.61	0.88	0.79	0.78 ± 0.12
Claude	0.71	0.81	0.84	0.66	0.76 ± 0.08
Gemini	0.71	0.67	0.88	0.63	0.72 ± 0.11

Despite working independently, agents converged on similar disambiguation rules: (1) Switch reserved for truly unrelated topics; (2) Escalate requires explicit answer requests; (3) Engage as default for continued dialogue, including expressions of confusion. This independent convergence across three architectures suggests the methodology identifies genuine patterns rather than agent-specific artifacts.

3.2 Cross-Validation

Because all three agents iterated on the same 80 sessions, strong development-set performance may reflect overfitting. To assess generalization, we conducted a 4-fold cross-validation, splitting the 80 sessions into four folds of 20 sessions each, stratified by intent distribution. We chose 4-fold rather than 10-fold because with three multi-class dimensions, 8-session folds risk zero instances of rare categories (e.g., Switch), making κ estimation unreliable. For each fold, agents developed prompts using 60 training sessions and evaluated on 20 held-out sessions (113 total iterations across all agent-fold combinations).

Table 2 reveals substantial overfitting: Codex dropped from development $\kappa = 0.93$ to test 0.78; Claude from 0.91 to 0.76; Gemini from 0.93 to 0.72 ($\Delta\kappa \approx 0.15$–$0.21$). Nevertheless, the methodology shows improvement over baseline: all agents improved from baseline test $\kappa \approx 0.70$ to 0.72–0.78 on held-out data. The best result (Codex, $\kappa = 0.78$, SD= 0.12) remains below human composite agreement ($\kappa = 0.89$), so the held-out result should be interpreted as a meaningful gain over baseline rather than as surpassing human overall agreement. Variance across folds (SD= 0.08–0.12) reflects sensitivity to fold composition given the small validation pool.

Table 3: Best development-set κ across classifiers.

Agent	GPT	Claude	Gemini
Codex	0.93	0.93	0.87
Claude Code	0.91	0.91	0.89
Gemini	0.93	0.85	0.92

3.3 Classifier Effects

We also explored whether a "same-family advantage" exists when agent and classifier share the same model architecture. These comparisons are descriptive development-set analyses rather than held-out cross-validation results. Same-family pairings generally converged faster, but did not consistently produce higher ceilings. For example, Gemini Agent reached $\kappa = 0.92$ by v5 with Gemini but required v9 to reach 0.93 with GPT, while Codex reached 0.93 with both GPT and Claude but showed smoother convergence with GPT. Overall, classifier choice appears to matter more for iteration dynamics than for the final peak achieved on the development set.

4. DISCUSSION AND CONCLUSION

Convergent solutions and tacit knowledge. The most striking finding is independent convergence across Codex, Claude Code, and Gemini. Despite different model families, all three agents moved toward similar disambiguation rules, especially around the treatment of confusion expressions. The key rule, "confusion = engagement," was absent from the original codebook and emerged through systematic human-AI disagreement analysis, aligning with productive failure theory [14]. This suggests that the agent workflow can do more than optimize prompts: it can help surface tacit labeling criteria that researchers already apply but have not yet articulated explicitly.

Iteration dynamics and stopping rules. Continued iteration beyond the optimum can hurt performance through overcorrection or output instability. In practical terms, once an agent reaches strong agreement, additional rules may introduce new errors while fixing only a handful of old ones. Our results therefore support a simple stopping principle:

use held-out validation to select the best prompt version rather than assuming the latest version is the best. This is a straightforward methodological recommendation for prompt-engineering research that has often been missing from prior educational coding work.

What the classifier comparison suggests. The classifier comparison adds a more tentative point. Same-family pairings did not consistently yield higher final peaks, but they often converged faster. That pattern suggests that model family may matter more for the efficiency and stability of iteration than for the final achievable performance. Because these analyses remain development-set only, we treat them as exploratory.

Practical implications for educational research. From an applied perspective, the workflow lowers the cost of rigorous dialogue coding. Each agent required only $5–8 for roughly ten iterations, and the total researcher effort remained modest compared with traditional multi-coder training and calibration. The gains are not large enough to claim that agents replace human agreement, but they are meaningful enough to support hybrid workflows in which agents accelerate prompt refinement, expose edge cases, and reduce the manual burden of theory-grounded labeling.

Prompt strategies and why they mattered. A useful secondary observation is that not all prompt-revision styles behaved the same way. The strongest prompts tended to make category boundaries explicit through ordered decision rules, especially for the hardest dimension, Follow-up Type. In contrast, more heuristic prompt language left greater interpretive burden on the downstream classifier. This matters for educational dialogue coding because the task often hinges on subtle distinctions such as confusion versus disengagement, or productive help-seeking versus answer-seeking. Our results therefore suggest that for nuanced theory-grounded coding tasks, agents are most helpful when they distill tacit qualitative judgments into concrete disambiguation rules rather than merely adding more examples or longer prose.

Recommendations for future users of this workflow. Two practical recommendations follow. First, researchers should begin with a compact prompt derived directly from a codebook, then let the agent target the weakest dimension rather than modifying all dimensions at once. Second, they should save richer artifacts than we did here: fold assignments, selected prompt versions, and per-dimension held-out outputs. Doing so would make later audit, replication, and paper writing much easier.

What the remaining errors suggest. Our error analysis also points toward concrete future directions for educational dialogue research. Remaining disagreements clustered into four recurring patterns: ambiguous intent, context-dependent interpretation, multi-turn dynamics, and rare interaction patterns. Some were cases of genuinely ambiguous intent, where even human coders disagreed before adjudication. Others reflected context dependence across turns, where the same phrase could signal productive confusion in one exchange and frustration in another. These patterns suggest that future systems may benefit from richer representations than single-label session coding alone, such as turn-level coding, uncertainty-aware labeling, or multi-label treatments for borderline cases. In that sense, the present workflow is useful not only for prompt optimization but also for diagnosing where the current coding ontology itself begins to strain.

Why human oversight remains important. One lesson from the full iteration history is that higher agreement alone is not always enough. Some prompt changes initially looked attractive because they improved a subset of cases, but they did so by blurring theoretically meaningful distinctions. Human review therefore remains central, not as a bottleneck to eliminate, but as the mechanism that keeps the refinement process aligned with pedagogical constructs rather than with metric chasing alone.

Limitations. The study also has clear limitations. The 80-session validation set limits statistical power and creates overfitting risk, and we did not retain fold-level per-dimension held-out artifacts or full fold metadata in the current supplementary release. Thus, our preserved held-out evidence is stronger for overall composite performance than for dimension-specific generalization. Because fold sizes are small, we treat differences between agents and classifiers descriptively rather than as statistically significant. Cross-validation also provides a held-out estimate within the 80-session pool, not a completely untouched final test set from the broader 659-session corpus. We also relied on commercial models and coding-agent frameworks, which limits exact reproducibility despite releasing prompts, logs, and evaluation scripts. We did not compare against fine-tuned models, and session-level classification cannot capture within-session intent shifts. Finally, although the coding scheme transfers conceptually to other tutoring contexts, some decision rules are domain-shaped by mathematical dialogue and would likely need adaptation in writing, science, or language-learning settings.

Why this matters methodologically. Prompt engineering for educational coding should be documented as a process rather than reported only as a final prompt plus a final score.

Overall, the results suggest that LLM coding agents can act as collaborative partners in codebook development, yielding held-out gains and useful stopping rules.

The agent's iteration logs, prompts, and evaluation scripts are available at: https://github.com/EasonC13/EDM26-Agents.

5. REFERENCES

[1] V. Aleven and K. R. Koedinger. Limitations of Student Control: Do Students Know when They Need Help? In G. Gauthier, C. Frasson, and K. VanLehn, editors, *Intelligent Tutoring Systems*, pages 292–303, Berlin, Heidelberg, 2000. Springer.

[2] D. Armfield, E. Chen, A. Omonkulov, X. Tang, J. Lin, E. Thiessen, and K. R. Koedinger. Avalon: a human-in-the-loop llm grading system with instructor calibration and student self-assessment. In *International Conference on Artificial Intelligence in Education*, pages 111–118, 2025.

[3] R. S. Baker, A. T. Corbett, and K. R. Koedinger. Detecting Student Misuse of Intelligent Tutoring Systems. In J. C. Lester, R. M. Vicari, and

F. Paraguaçu, editors, *Intelligent Tutoring Systems*, pages 531–540, Berlin, Heidelberg, 2004. Springer.

[4] E. Chen, C. Guan, A. Elshafiey, Z. Zhao, J. Zekeri, A. Edeifo Shaibu, and E. Osadebe Prince. When openclaw ai agents teach each other: Peer learning patterns in the moltbook community. *arXiv e-prints*, pages arXiv–2602, 2026.

[5] E. Chen, C. Guan, A. Elshafiey, Z. Zhao, J. Zekeri, A. Edeifo Shaibu, E. Osadebe Prince, and C.-J. Wu. When openclaw agents learn from each other: Insights from emergent ai agent communities for human-ai partnership in education. *arXiv e-prints*, pages arXiv–2603, 2026.

[6] E. Chen, C. Guan, A. Elshafiey, Z. Zhao, J. Zekeri, A. E. Shaibu, E. O. Prince, and C. J. Wu. Openclaw ai agents as informal learners at moltbook: Characterizing an emergent learning community at scale. *arXiv preprint arXiv:2602.18832*, 2026.

[7] E. Chen, S. Judicke, K. Beigh, X. Tang, I. Wang, N. Yuan, Z. Xiao, C. Li, S. Li, R. Luttmer, S. Singh, and K. R. Koedinger. Chat-based support alone may not be enough: Comparing conversational and embedded llm feedback for mathematical proof learning. *arXiv preprint arXiv:2602.18807*, 2026.

[8] E. Chen, S. Judicke, K. Beigh, X. Tang, Z. Xiao, C. Li, S. Li, R. Luttmer, S. Singh, and K. R. Koedinger. Generative ai alone may not be enough: Evaluating ai support for learning mathematical proof. *arXiv preprint arXiv:2509.16778*, 2025.

[9] E. Chen, J. Li, S. Huang, X. Tang, J. Lin, P. Carvalho, and K. Koedinger. Identifying effective praise in tutoring: Large language models with transparent explanations. In *International Conference on Artificial Intelligence in Education*, pages 157–163, 2025.

[10] E. Chen, J. Li, S. Huang, X. Tang, J. Lin, P. Carvalho, and K. Koedinger. Ai knows best? the paradox of expertise, ai reliance, and performance in educational tutoring decision-making tasks. In *Proceedings of the LAK26: 16th International Learning Analytics and Knowledge Conference*, pages 736–743, 2026.

[11] W. Dai, J. Lin, H. Jin, T. Li, Y.-S. Tsai, D. Gašević, and G. Chen. Can large language models provide feedback to students? a case study on chatgpt. *International Conference on Artificial Intelligence in Education*, pages 323–335, 2023.

[12] W. Dai, Y.-S. Tsai, J. Lin, A. Aldino, H. Jin, T. Li, D. Gašević, and G. Chen. Assessing the proficiency of large language models in automatic feedback generation: An evaluation study. *Computers and Education: Artificial Intelligence*, 7:100299, 2024.

[13] Z. F. Han, J. Lin, A. Gurung, D. R. Thomas, E. Chen, C. Borchers, S. Gupta, and K. R. Koedinger. Improving assessment of tutoring practices using retrieval-augmented generation. In *AAAI2024 Workshop on AI for Education - Bridging Innovation and Responsibility*, 2024.

[14] M. Kapur. Productive failure. *Cognition and Instruction*, 26(3):379–424, 2008.

[15] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, M. Moez, H. Hajishirzi, C. Potts, M. Zaharia, and C. Ré. DSPy: Compiling declarative language model calls into self-improving pipelines. *arXiv preprint arXiv:2310.03714*, 2023.

[16] J. R. Landis and G. G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977.

[17] J. Lin, E. Chen, Z. Han, A. Gurung, D. R. Thomas, W. Tan, N. D. Nguyen, S. Gupta, V. Aleven, and K. R. Koedinger. How can i improve? using gpt to highlight the desired and undesired parts of open-ended responses. In *Educational Data Mining Conference 2024*, 2024.

[18] N. Tran, B. Pierce, D. Litman, R. Correnti, and L. C. Matsumura. Analyzing Large Language Models for Classroom Discussion Assessment. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 500–510, Atlanta, Georgia, USA, 2024. International Educational Data Mining Society.

[19] C. Yang, X. Wang, Y. Lu, H. Liu, Q. V. Le, D. Zhou, and X. Chen. Large language models as optimizers. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[20] J. Zhang, C. Borchers, V. Aleven, and R. S. Baker. Using Large Language Models to Detect Self-Regulated Learning in Think-Aloud Protocols. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 157–168, Atlanta, Georgia, USA, 2024. International Educational Data Mining Society.

[21] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba. Large language models are human-level prompt engineers. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

Identifying Learning Progression Profiles in a Psychomotor Task Using Sequence Clustering

Emily Jensen
Franklin & Marshall College
ejensen@fandm.edu

ABSTRACT

Psychomotor skill learning remains underexplored in educational data mining, which has historically focused on cognitive tasks. Understanding how learners progress toward mastery over time, rather than simply measuring final outcomes, is essential for designing adaptive instructional systems. In this work, we analyze trial-by-trial learning progressions from a simulated drone landing task. Participants completed 20 trials with performance categorized as crash, unsafe landing, or safe landing. We compute transition probabilities between performance categories and apply a k-medians clustering algorithm using Levenshtein distance to identify representative learning sequences. Transition results show that learners rarely make large performance jumps between consecutive trials and tend to maintain safe landings once achieved, consistent with our knowledge of motor skill acquisition. Clustering reveals meaningful learner profiles ranging from quick mastery to persistent struggle, with finer-grained distinctions emerging as the number of clusters increases. These findings provide an initial foundation for developing feedback and task selection policies in intelligent tutoring systems that are sensitive to a learner's trajectory through skill acquisition, not only their current performance state.

Keywords

psychomotor skill, clustering, sequence analysis, learner profile

1. INTRODUCTION

Educational data mining research has largely focused on cognitive tasks such as problem-solving and knowledge acquisition. However, many real-world learning domains require the development of psychomotor skill combining cognitive reasoning and motor control. The unique dynamics of these tasks make them both harder to measure and harder to model than discrete knowledge states.

Aggregate measures of learning such as final accuracy or learning gain neglect to consider the path a learner took to get there. Two learners with identical final performance may have followed very different trajectories. Additionally, progress in skill acquisition is rarely monotonic; we often see plateaus, regressions, and bursts of improvement. Identifying common trajectory types moves emphasis from "did they learn?" toward "how did they learn?". This approach is necessary for designing feedback that responds to a learner's progression rather than their current performance snapshot alone.

Adaptive feedback systems are most effective when they can anticipate a learner's near-future state or respond to long-term misconceptions rather than simply react to their last action. If distinct learner subgroups follow recognizably different paths through skill acquisition, feedback policies can be tailored to each pathway. This work takes a step toward data-driven feedback policy design by starting to understand the landscape of learning trajectories, laying the groundwork for eventually optimizing feedback to nudge learners toward more favorable pathways. In this work, we apply sequential analysis methods to explore typical learning progressions in a drone landing simulation, which offers a controlled learning environment with rich trial-by-trial performance data.

2. RELATED WORK

Research in intelligent tutoring systems has a rich history of using performance data to inform interventions and develop personalized analytics. In particular, Bayesian Knowledge Tracing models and variants use transition probabilities to represent how learners acquire knowledge over time [1, 14, 15, 18]. These models infer mastery based on previous estimates and the learner's most recent performance. This approach to learner modeling acknowledges that learning is not a linear process and performance can fluctuate for a variety of reasons.

Additionally, learning and education data are well-suited for sequential analysis [21]. Sequential pattern mining tools have often been used to connect behavior patterns with learning outcomes [7, 11, 16], provide better interventions [4], or understand higher-level reasoning skills [2, 10]. Some work has started to use sequential analysis to understand learning trajectories [13], though this is not well studied for psychomotor tasks.

Recent work has also considered how identifying learner profiles can help inform recommendations and personalized

Emily Jensen. Identifying Learning Progression Profiles in a Psychomotor Task Using Sequence Clustering. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 684–688. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040080

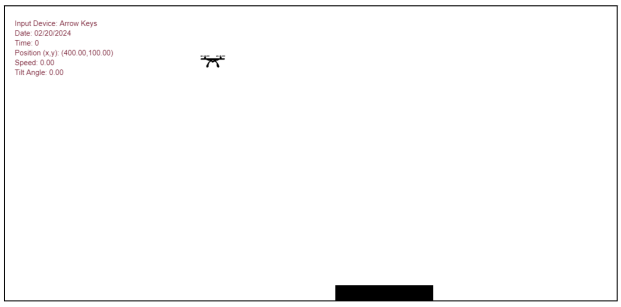


Figure 1: Starting configuration of the drone landing task [3].

learning paths [9, 17]. These are often applied to large-scale learning environments such as MOOCs and use features such as demographics and engagement metrics. More work is needed to understand how common sequences of learning performance can be used to shape feedback, especially for complex tasks.

3. APPROACH

3.1 Dataset

The data was collected in a previous study [5], but all analyses presented here are new. The study was approved by the local ethics board and all data are anonymized.

In an online study conducted through Prolific, participants completed 20 trials of a computer-based drone landing simulation [3] (see Figure 1). Participants were placed into one of three feedback conditions to improve their performance. The final set of participants is evenly split between feedback conditions, though the conditions are not explicitly considered in this work. We only consider data from those who completed all 20 trials, resulting in a final set of 166 participants.

The main measure of interest is participant performance on each of the 20 trials. In alignment with previous work [19, 20], landing performance was assigned to one of the ordinal labels *crash*, *unsafe*, or *safe*. A trial was labeled as safe if the participant landed the drone in the target area with a landing speed ≤ 15 m/s and a landing angle $\leq 5°$. Unsafe landings violated at least one of the landing speed or landing angle constraints. Finally, attempts that did not reach the landing pad were labeled as a crash. This approach yielded 3,320 total performance labels across all participants. Participants demonstrated mastery of the task by consistently achieving safe landings. Figure 2 shows the distribution of landing performance across trials.

We encoded the learning progression of each participant as a sequence of 20 "words" from a fixed vocabulary (i.e., landing performance category for each trial). These sequences were represented by a string of characters, with the characters C, U, S corresponding to *crash*, *unsafe*, or *safe*. An example sequence from a participant achieving only safe and unsafe landings is SUSUUUSSSSSUUUSSUSUS.

3.2 Analysis Methods

We first calculated the transition probabilities for the landing performance categories. These probabilities show how participants improve or regress in their performance between trials. Each value represents the likelihood of moving between two categories in consecutive trials. The transition matrix is normalized across rows, meaning the destination probabilities for each starting category sum to 1.

To identify representative learning progressions, we implemented a variant on the k-means clustering algorithm [12]. To calculate the distance between learning progression sequences, we used the Levenshtein edit distance [8]. Instead of calculating a numeric mean as the center of each cluster, we calculated the approximate generalized median string of all data points in a given cluster using the Levenshtein Python library (https://pypi.org/project/python-Levenshtein/). To assess the quality of each cluster, we calculated the average within-cluster Levenshtein edit distance, where more similar sequences have a smaller edit distance.

Given no established value for the expected number of learning progression profiles, we used the elbow method across multiple values of k. For each number of clusters k, we calculated the average quality across all clusters and reported the best across 10 restarts.

4. RESULTS

The transition probabilities show how learners improve their performance between trials. Figure 3 shows the highest probability in the safe→safe case, showing that learners are generally able to maintain consistent performance once they understand a task (nearly 70% of the time). There are also strong probabilities on the diagonal, where participants maintained their performance level between tasks. The superdiagonal of the matrix includes the next highest set of probabilities; these are the cases where participants jumped to the next performance level between trials. Similarly, the subdiagonal represents the cases where participants regress a level between trials, though these probabilities are notably smaller.

The transition matrix also shows it is unlikely to achieve large changes in performance between trials, either in the crash→safe condition or the safe→crash condition. In the context of a Bayesian Knowledge Tracing model of student performance, these conditions may relate to low "slip" and "guess" probabilities where learner performance is misaligned with their inherent mastery of the task.

Figure 4 shows how cluster quality changes over a range of k values. We did not find a clear elbow; rather, the average edit distance across clusters steadily across the entire range. This suggests that adding more clusters will continue to identify more specific profiles, or a more nuanced evaluation metric is needed. Using a smaller number of clusters for designing feedback policies makes practical sense, since clusters will be more distinct and easier to interpret.

Table 1 shows detailed clustering results for $k \in \{2, \ldots, 5\}$. Given the results of the elbow analysis above and in the interest of interpretability, we do not report results for higher values of k.

In the $k = 2$ condition, the centers represent the profile of a struggling learner and one who quickly masters the task.

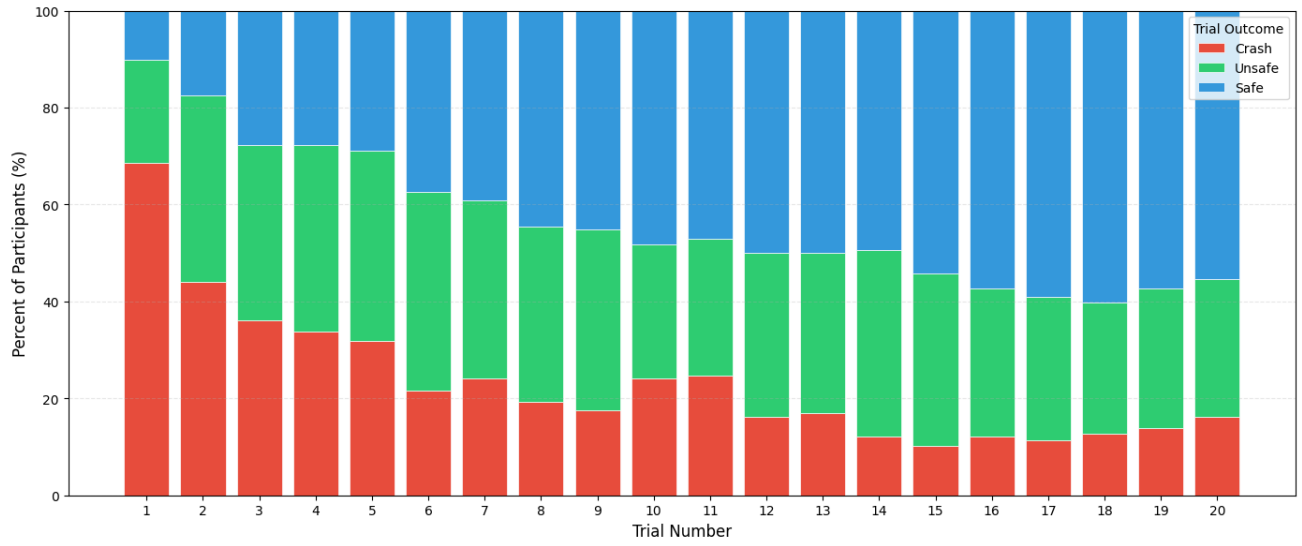


Figure 2: Distribution of landing outcomes for each of 20 trials. Participants achieve more safe landings as the trials progress.

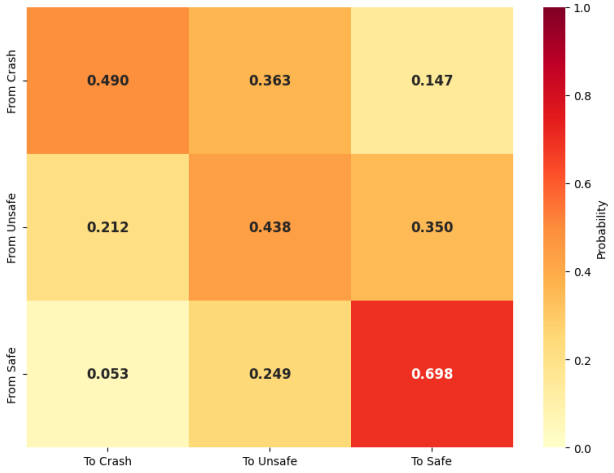


Figure 3: Transition matrix showing the probabilities of changing performance categories between trials. Darker cells have higher probabilities.

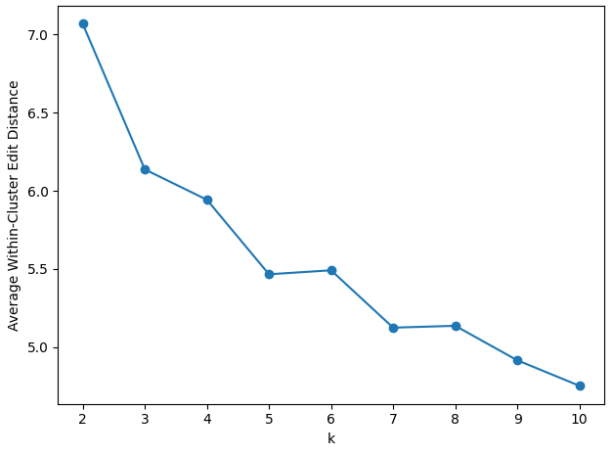


Figure 4: Plot showing the average cluster quality (edit distance) across clusters for values of $k \in \{2, \ldots, 10\}$

Table 1: Results of learning progression clustering using $k \in \{2, \ldots, 5\}$.

k	Mean Cluster Edit Distance	Cluster Centers
2	7.07	CCUCUCUUCUUCUUUSUSU
		CUSUSSSSSSSSSSSSSSS
3	6.14	CUSUSSSSSSSSSSSSSSS
		CCUCUUUUUUCUUSUUSU
		CCCCCCCCCCCCCUCUCCCC
4	5.80	CUCUUSUSSSSSUSSSSS
		CSSSSSSSSSSSSSSSSSS
		CUCUCUUUUUCUUUUUUSU
		CCCCCCCCCCCCCUCUCUCCC
5	5.48	CUUUSUUSSSUSSUSSSSUS
		CSUSSSSSSSSSSSSSSSS
		CCCUCUUUUUUUSUUSSUSS
		CUCUCCUUCUCCUUUUUCUU
		CCCCCCCCCCCCCCCCCCCCC

The profile of the struggling learner sees a majority of crash and unsafe landing outcomes, with a few safe landings towards the end of the trials. The quick learner shows a few crashes and unsafe landings at the beginning, quickly converging to consistent safe landings.

The other models show similar patterns. In the $k = 3$ condition, the profiles represent participants quick to master the task, those who consistently achieve unsafe landings, and those who may struggle to understand the control dynamics and crash most trials. The $k = 4$ condition adds another proficient learner that takes a bit longer to get comfortable with safe landings. Finally, the $k = 5$ condition adds a profile that achieves a few safe landings at the end, though they may need more practice to become consistent. These results show that clustering can identify meaningful learning progressions, with more clusters leading to more nuanced profiles that may be useful for detailed feedback development.

5. DISCUSSION

This study examined learning progression in a psychomotor drone landing task, finding that sequence clustering can uncover meaningful learner profiles from trial-by-trial performance data. The transition matrix results confirm that learners rarely make large jumps in performance between consecutive trials, and that once safe landings are achieved they tend to be maintained. This pattern is consistent with the consolidation dynamics characteristic of motor skill acquisition. Together, these results suggest that the ordinal performance encoding used here captures structure in the learning process.

The clustering results show that even a small number of profiles can meaningfully identify different types of learners. Even with two clusters, the model separates quick learners from those who struggle to achieve safe landings. Additional clusters reveal finer distinctions, such as learners who plateau at unsafe landings versus those who crash consistently. The absence of a clear elbow is itself informative; it suggests learner profiles may exist on a continuum rather than falling into a small number of clear types.

6. FUTURE WORK

The current work developed profiles of learning progression only using the final landing performance outcome for each trial. Future work can investigate how learner control inputs (e.g., joystick movements, button presses) can be used to understand the learning process and develop feedback. This can take inspiration from previous work focused on cognitive tasks [6].

Future work should also consider how these profiles of learning progression relate to the feedback conditions in the original experiment. Rather than evaluating personalized interventions based on static measures such as number of safe landings, applying sequence analysis to each condition separately may give a more detailed understanding of how feedback can shape a learner's mastery of a task over time.

7. REFERENCES

[1] R. S. J. Baker, A. T. Corbett, and V. Aleven. More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. In B. P. Woolf, E. Aïmeur, R. Nkambou, and S. Lajoie, editors, *Intelligent Tutoring Systems*, pages 406–415, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[2] M. Bannert, P. Reimann, and C. Sonnenberg. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and learning*, 9(2):161–185, 2014.

[3] S. Byeon, E. Jensen, M. S. Yuh, S. Sankaranarayanan, N. Jain, and I. Hwang. Quadrotor landing simulation, 2024. Available at `https://github.com/cairo-robotics/llm-multimodal-drone-feedback`, version 0.0.1.

[4] P. Fournier-Viger, U. Faghihi, R. Nkambou, and E. M. Nguifo. Exploiting sequential patterns found in users' solutions and virtual tutor behavior to improve assistance in its. *Journal of Educational Technology & Society*, 13(1):13–24, 2010.

[5] E. Jensen, S. Sankaranarayanan, and B. Hayes. Automated assessment and adaptive multimodal formative feedback improves psychomotor skills training outcomes in quadrotor teleoperation. In *2024 International Conference on Human-Agent Interaction*, pages 185 – 194, 2024.

[6] E. Jensen, T. Umada, N. C. Hunkins, S. K. D'Mello, S. Hutt, and A. C. Huggins-Manley. What You Do Predicts How You Do: Prospectively Modeling Student Quiz Performance Using Activity Features in an Online Learning Environment. In *Proceedings of the 11th International Learning Analytics and Knowledge Conference (LAK21)*, pages 121 – 131, 2021.

[7] J. S. Kinnebrew, S. S. Killingsworth, D. B. Clark, G. Biswas, P. Sengupta, J. Minstrell, M. Martinez-Garza, and K. Krinks. Contextual markup and mining in digital games for science learning: Connecting player behaviors to learning goals. *IEEE Transactions on Learning Technologies*, 10(1):93–103, 2016.

[8] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union,

1966.

[9] L. Li and J. Xiao. Persona profiling: a multi-dimensional model to study learner subgroups in massive open online courses. *Education and Information Technologies*, 27(4):5521–5549, 2022.

[10] T. Liu and M. Israel. Uncovering students' problem-solving processes in game-based learning environments. *Computers & Education*, 182:104462, 2022.

[11] D. Malekian, J. Bailey, and G. Kennedy. Prediction of students' assessment readiness in online learning environments: the sequence matters. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 382–391, 2020.

[12] J. B. McQueen. Some methods of classification and analysis of multivariate observations. In *Proc. of 5th Berkeley Symposium on Math. Stat. and Prob.*, pages 281–297, 1967.

[13] J. Moon and Z. Liu. Rich representations for analyzing learning trajectories: Systematic review on sequential data analytics in game-based learning research. *Data analytics approaches in educational games and gamification systems*, pages 27–53, 2019.

[14] R. Pelánek. Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User Modeling and User-Adapted Interaction*, 27:313–350, 2017.

[15] M. O. C. Z. San Pedro, R. S. Baker, and M. M. T. Rodrigo. Detecting Carelessness through Contextual Estimation of Slip Probabilities among Students Using an Intelligent Tutor for Mathematics. In G. Biswas, S. Bull, J. Kay, and A. Mitrovic, editors, *Artificial Intelligence in Education*, pages 304–311. Springer Berlin Heidelberg, 2011.

[16] D. Sun, G. Cheng, and H. Luo. Analysing the evolution of student interaction patterns in a massive private online course. *Interactive Learning Environments*, 32(2):693–706, 2024.

[17] X. Xie and X. Feng. Personalized learning path recommendation based on learner profile and knowledge graph. *Computing and Informatics*, 44(4):983–1008, 2025.

[18] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized Bayesian Knowledge Tracing Models. In H. C. Lane, K. Yacef, J. Mostow, and P. Pavlik, editors, *Artificial Intelligence in Education*, pages 171–180, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[19] M. S. Yuh, K. R. Ortiz, K. S. Sommer-Kohrt, M. Oishi, and N. Jain. Classification of Human Learning Stages via Kernel Distribution Embeddings. *IEEE Open Journal of Control Systems*, 3:102–117, 2024.

[20] M. S. Yuh, E. Rabb, A. Thorpe, and N. Jain. Using Reward Shaping to Train Cognitive-Based Control Policies for Intelligent Tutoring Systems. In *2024 American Control Conference (ACC)*, pages 3223–3230, Toronto, ON, Canada, July 2024.

[21] Y. Zhang and L. Paquette. *Sequential Pattern Mining in Educational Data: The Application Context, Potential, Strengths, and Limitations*, pages 219–254. Springer Nature Singapore, Singapore, 2023.

Investigating Patterns of Iterative Code Changes in the VEXcode VR environment by Combining Heuristic Patterns with Hidden Markov Modeling

Hyeongjo Kim, Luc Paquette, Jennie Lee
University of Illinois at Urbana-Champaign
{hk61, lpaq, jl294}@illinois.edu

ABSTRACT

Computational problem solving (CPS) is a core component of computational thinking and is increasingly supported through game-based learning environments. However, understanding learners' problem-solving processes in such environments remains challenging especially when they engage students in open-ended problem-solving with no unique solution pathway. In this study, we examine learners' CPS processes in VEXcode VR, a block-based robotics programming environment, by investigating iterative code modification behaviors. Drawing on in situ observations, we first identified three heuristic problem-solving patterns: step-by-step, relying on fortune, and strategy change. These patterns were operationalized using a code change score and modeled using a categorical Hidden Markov Model (HMM), enabling the representation of latent problem-solving states and their temporal dynamics. The results demonstrate strong alignment between the latent states inferred by the HMM and the heuristic patterns identified by researchers, supporting both the validity and interpretability of the model. The interpretations of the identified patterns highlight the importance of considering the unique modality of game-based environments when understanding learners' computational problem-solving processes.

Keywords

Computational Problem Solving; Game-Based Learning; Hidden Markov Models; Heuristic Modeling; Sequential Analysis

1. INTRODUCTION

Computational problem solving (CPS) is recognized as a core competency in contemporary education, particularly in technology-rich and game-based learning environments [2]. CPS involves a set of cognitive processes, including problem formulation, algorithmic thinking, iterative testing, and debugging, through which learners construct and refine solutions to complex problems [10]. To support CPS, game-based learning (GBL) has been used as scaffolding for computational problem solving through providing immediate and continuous feedback, reducing cognitive load, and simulating real-world systems [2, 9, 10].

Hyeongjo Kim, Luc Paquette, and Jennie Lee. Investigating Patterns of Iterative Code Changes in the VEXcode VR environment by Combining Heuristic Pat-terns with Hidden Markov Modeling. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 689–693. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039798

Because both games and coding fundamentally involve problem solving [2], prior studies examining problem-solving patterns in game-based learning provide useful conceptual frameworks for understanding CPS. However, as Matcha and colleagues [5] noted, students' strategic behaviors can vary across learning environments even when guided by the same theoretical framework, largely due to differences in modality. For example, while systematic evaluation is often considered a more advanced and desirable strategy than trial-and-error in traditional problem-solving contexts [4, 7], repeated cycles of trial and error are not merely preliminary strategies but can constitute a core process of solution development, testing, and debugging in CPS contexts [2, 8]. Therefore, in CPS environments, examining how learners iteratively modify their code during trial-and-error processes provides a direct and meaningful way to understand their problem-solving strategies.

Therefore, in this study, we investigate the CPS processes of students using VEXcode VR, a block-based programming environment in which students must program a robot to perform different tasks. Tasks generally do not have one unique solution and may involve randomness which impacts the final outcome of executing a solution. More specifically, we examine how learners iteratively modify their solutions in the VEXcode VR environment and explore how these iterative changes can be modeled to reflect different problem-solving processes. Insights about the students' code change behaviors were obtained through in-situ observations and review of screen-capture video to identify three heuristic problem-solving patterns (step-by-step, relying on fortune, and strategy change) which were modeled using a categorical Hidden Markov Model (HMM), enabling us to capture CPS processes in a game-based learning context while accounting for its unique affordances.

2. COMPUTATIONAL PROBLEM SOLVING IN GAME-BASED LEARNING

Computational problem solving (CPS) refers to a logical and abstract reasoning process for addressing problems, involving understanding a problem, devising a solution plan, and evaluating outcomes through programmed procedures [2, 10]. CPS is considered a core component of computational thinking (CT), reflecting the centrality of problem solving in CT definitions. To support CPS and CT, a learning-by-doing approach has been widely adopted as a constructionist alternative to direct instruction [2, 6]. While this approach aligns with the exploratory nature of CPS, it poses challenges for novice learners who often lack sufficient knowledge and skills in both computation and problem solving to effectively address ill-structured problems [3]. As a result, learners may struggle to engage productively without appropriate guidance.

To address this gap, research has emphasized the importance of scaffolding, particularly through game-based learning (GBL) environments. Games provide several affordances that support CPS. First, they enhance motivation by offering immediate and continuous feedback during problem solving [1]. Second, games sometimes employ designs to reduce cognitive load such as block-based or card-based coding interfaces, allowing learners to focus on the logical structure of solutions without being hindered by syntax and semantic errors [2]. Third, games enable simulations of complex real-world systems, such as space or physical laws, thereby situating problem solving within meaningful and interactive contexts [1, 2].

Beyond these affordances, both games and CPS fundamentally center on problem solving [2]. In game environments, players either attempt to solve predefined challenges or construct their own goals in open-ended contexts, utilizing available resources within the system. In CPS contexts, these resources correspond to programming systems and interfaces, which serve as tools for constructing and testing computational solutions. This structural alignment suggests that GBL environments function not only as motivational tools but also as spaces where CPS processes are actively enacted.

Based on this alignment, studies have proposed integrated frameworks that connect CPS processes in GBL environments [2, 8]. They commonly include identifying problems, designing and developing solutions, testing and debugging the solutions. Each stage is associated with specific CT practices [8]. For example, in the identifying problem stage, learners engage in decomposition, abstraction, and generalization to break down and simplify problems. These stages provide a structured lens for understanding CPS, but understanding how learners enact these processes in practice has led researchers to examine learners' behaviors through log data.

Previous studies have examined learners' problem-solving processes through log data to reveal how they develop and test solutions. For example, in Zoombinis, a mathematics problem-solving game, researchers have identified structured problem-solving processes and investigated how specific behaviors correspond to different stages, as well as how these behaviors can be modeled to represent sequential progress and strategies [4, 7]. Although the identified strategies such as trial-and-error and systematic testing in problem solving are also relevant to CPS, their roles may differ depending on the learning context. Matcha and colleagues [5] demonstrated that students' strategic behaviors can vary across learning environments even when guided by the same theoretical framework, largely due to differences in modality. CPS environments present a distinctive modality in which learners construct solutions through code and iteratively test and revise them. In the context of Zoombinis, systematic evaluation is often considered a more advanced and desirable strategy than trial-and-error, and transitions from trial-and-error to systematic evaluation are treated as indicators of improved problem-solving competence [4, 7]. In contrast, within CPS contexts, repeated cycles of trial and error are not merely preliminary strategies but constitute a core process of solution development, testing, and debugging [2, 8]. Therefore, in CPS environments, examining how learners iteratively modify their code during trial-and-error processes provides a more direct and meaningful way to understand their problem-solving processes and strategies.

3. CONTEXT

3.1 VEXcode VR

The study was conducted using the VEX Robotics programming environment, specifically VEXcode VR, a block-based platform designed to foster computational thinking and problem-solving skills. Within this environment, learners control a virtual robot by creating sequences of commands, including movement, rotation, and control structures, to navigate simulated spaces. The system provides immediate visual feedback through simulation, enabling students to iteratively test, debug, and improve their solutions. Our study employed a custom-made version of VEXcode VR which automatically logged all student interactions, such as code creation and execution, supporting fine-grained analysis of learners' problem-solving processes over time.

The study was conducted across three different playgrounds (simulated 3 dimensional spaces), each presenting distinct problem-solving goals. In Coral Reef Cleanup, students programmed a robot to collect and remove trash in an underwater environment within limited battery constraints, emphasizing efficient navigation and resource management. In Castle Crasher, learners developed programs to knock down castle structures and transport debris into designated areas, requiring strategic planning. In Rover Rescue, goal prioritization was highlighted as students programmed a rover with the goal to maximize mission duration through three strategy choices: collecting minerals, neutralizing enemies, and returning minerals to base. . After executing each code, the system assigns a score based on predefined rules (e.g., the amount of trash collected) that involve randomness, meaning that an identical code could yield different outcomes. Across all playgrounds, tasks were open-ended, allowing multiple solution paths and encouraging iterative refinement.

3.2 Data

A total of 18 students participated in a one-and-a-half-hour session per day over three days. Students played on one playground each day in the following order: Coral Reef Cleanup – Castle Crasher - Rover Rescue. Each day, students first received a brief introduction to the environment and core programming commands before engaging in individual coding activities. During each session, learners iteratively developed and executed their programs, receiving immediate feedback from the simulation and refining their solutions accordingly. While collaboration and discussions were allowed, each student constructed their own solution. Instructors helped students mostly when they were asked by students. Throughout the process, all programming actions and execution events were logged, enabling the analysis of learners' iterative problem-solving behaviors over time.

Two types of data were collected for this study: screen capture recordings and programming log data. Screen capture recordings without sounds were obtained during participants' interactions with the VEXcode VR programming environment. In addition, detailed coding logs were automatically recorded. These logs include various user actions, such as adding, editing, and moving blocks, as well as executing programs (e.g., "Run project"). All interaction data were stored in JSON format, enabling fine-grained analysis of learners' computational problem-solving processes over time. The Institutional Review Board (IRB) at University of Illinois Urbana-Champaign reviewed the data collection protocol and made the determination that it did not meet criteria for "human-subject research" since all collected data was fully anonymized with no identifier linking the data to participants.

4. MODELING ITERATIVE CODE CHANGES

The modeling process integrated human insights with data-driven analysis to identify patterns of iterative code changes in VEXcode VR. First, in situ observations and reviews of screen-capture recordings of problem-solving activities were used to identify meaningful code-change patterns while accounting for contextual factors and the environment's unique modality. Then, we defined how these patterns could be operationalized using the VEXcode VR interaction log data. Finally, a Hidden Markov Model (HMM) was fit to categorize student problem-solving interactions as one of three identified code change patterns.

4.1 In situ Observations

Learners' behavioral patterns were heuristically identified by two researchers (first and third authors) who participated in the camp as facilitators and had direct experience observing and interacting with learners during the activities. Drawing on their domain knowledge and in situ experiences, they collaboratively articulated patterns they perceived in students' iterative code-change processes.

Screen capture recordings were revisited as needed to refine these interpretations. However, this study did not employ systematic video coding procedures, as the primary goal of this phase was not to generate exhaustive behavioral coding, but rather to elicit and formalize contextual understandings of learner behaviors. Through this process, we identified three overarching patterns that were consistently observed across the three playgrounds: step-by-step, relying on fortune, and strategy change.

The step-by-step pattern is characterized by students making small, incremental changes to their code, executing it, and then modifying it based on the last state of the VEX robot in each execution.

The relying on fortune pattern is characterized by repeated execution of the same code without any modification. Unlike typical iterative testing where code is refined between runs, students repeatedly run an identical program in an attempt to obtain better outcomes. This behavior is closely related to the design of the VEXcode VR environment, where stochastic elements can produce different outcomes. For example, the Coral Reef Cleanup playground randomizes when and where pieces of trash fall into the playground, whereas in the Castle Crasher playground, physics simulations determine how the castle structure reacts to collisions with the robot, which can result in greatly differing outcomes when repeatedly executing the same program.

Within this context, students appear to rely on randomness to achieve higher scores rather than refining their code. This pattern often emerged after a period of step-by-step iteration. It may reflect a temporary shift in strategy, such as taking a cognitive break, exploring outcome variability, or attempting to maximize performance from existing solutions. After several repetitions, students typically returned to modifying their code.

The strategy change pattern involves substantial revisions to the structure of the code, including modifying or removing large portions of previously constructed solutions. Compared to the other patterns, this behavior was observed relatively infrequently. We observed that students were generally reluctant to abandon or significantly alter existing code, which may explain the rarity of this pattern. Although we observed instances of strategy change, it was uncertain from our observation what constitutes an appropriate threshold for distinguishing it from incremental (step-by-step) modifications.

4.2 Operationalization of Identified Patterns

The patterns we heuristically identified were used as an initial foundation for pattern recognition, serving as a rough yet meaningful basis for subsequent systematic analysis. Operationalizing these patterns through a data-driven approach enables the development of models grounded in empirical evidence. To operationalize these patterns, it was necessary to translate them into quantifiable representations. As the identified patterns primarily relate to how students modify their code across program execution, we computed a code change score to capture the degree of difference between consecutive code submissions.

We computed the code change score by combining a position-weighted change rate with a length-based adjustment. First, differences between two consecutive code sequences were evaluated at each sequential position. A binary change indicator was assigned for each position, taking the value of 1 if the blocks differed (or if one sequence was shorter), and 0 otherwise. To reflect the structural importance of earlier code segments, we applied an exponentially decaying weight to each position, giving higher importance to changes occurring at the beginning of the sequence. The weighted changes were then normalized by the sum of positional weights to obtain a position-weighted change rate.

To account for the overall size of the program, this rate was further adjusted using a length-based weight (square root). Finally, the code change score was computed as the product of the position-weighted change rate and the normalized length weight, resulting in a measure that captures both the structural impact and the extent of code modifications. This formulation allows the score to reflect both the structural significance of changes (via positional weighting; $e^{-position}$) and the magnitude of edits (via length adjustment; $\sqrt{length}$).

An initial examination of the distribution of code change scores suggested the presence of distinct behavioral tendencies (Figure 1). The histogram revealed a substantial proportion of zero-change cases, along with a concentration of small changes and relatively fewer large changes. Specifically, over 40% of the observations showed no change in code, and even when changes occurred, they were typically minimal. This distribution provides preliminary support for the existence of the three heuristic patterns: minimal or no change (relying on fortune), incremental change (step-by-step), and substantial change (strategy change).

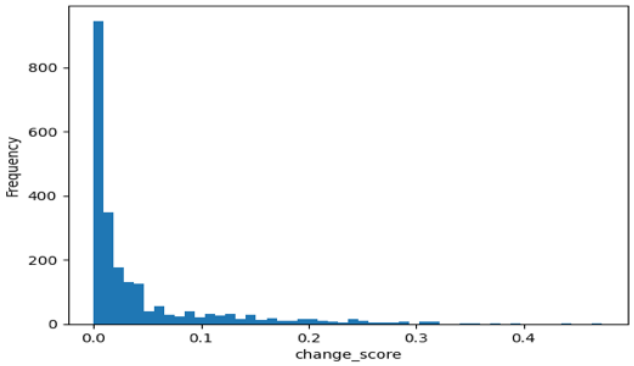


Figure 1. Distribution of code change scores across consecutive submissions.

Importantly, our observations emphasized that individual code change values should not be interpreted in isolation. Rather, patterns emerge over sequences of actions. For example, a sequence consisting predominantly of no-change trials with a single instance

of code modification would still be interpreted as relying on fortune, rather than as a mixture of patterns. This highlights a limitation of approaches that rely solely on observed transitions, such as Markov chains.

To address this limitation, we adopt Hidden Markov Modeling (HMM), which enables patterns to be conceptualized as hidden states that generate observable behaviors (i.e., code changes). From this perspective, the code change score serves as an observable indicator, while the underlying problem-solving patterns are inferred as hidden states. By modeling temporal sequences in this way, HMM provides a more nuanced representation of learner behaviors, consistent with instructors' interpretations.

To align with the heuristic categories and facilitate interpretation of the HMM results, we constrained the model by discretizing the continuous variable into categorical states and employing a categorical HMM instead of a continuous HMM. Code change scores were discretized into three levels: no change (0), small change (below the 95th percentile), and large change (above the 95th percentile). The 95th percentile corresponds to substantial modifications affecting a large portion of the code; in practice, these cases typically involve changes to more than 85% of blocks when the program contains at least 10 blocks, indicating a clear restructuring of the program.

4.3 Categorical Hidden Markov Modeling

Each student's interaction sequence was represented as a series of discrete observations derived from code changes categories between consecutive runs. Using the operationalization described in Section 4.2, each action was categorized into one of three observable types: no change, small change, and large change.

The HMM assumes that the observed sequence is generated from an underlying sequence of latent states, where transitions between states follow a first-order Markov process. Emission probabilities represent the likelihood of observing each type of code change given a latent state. Researchers should set the number of hidden states while the hyperparameter can be decided by comparing various models with different numbers of hidden states. However, this study first sets the number of hidden states at three to implement the heuristic patterns we observed.

When fitting the model, to address the issue of local optima due to random initialization, each model was trained with 100 random restarts, and the solution with the highest log-likelihood was selected. This procedure also allows us to assess the robustness of the identified patterns and reduce the likelihood of reporting coincidental results. The result shows that the average consistency across the 100 restarts is 84%, indicating robustness. We averaged the emission and transition probabilities to set the final model (Table 1).

Table 1. Emission and Transition Table for HMM model.

Emission			
Hidden state	No change	Small change	Large change
0	0.13	0.86	0.00
1	0.25	0.59	0.15
2	0.82	0.17	0.00
Transition			
0	0.94	0.00	0.04
1	0.03	0.91	0.05
2	0.09	0.05	0.84

Inspection of the emission probabilities enabled interpretation of the latent states in terms of observable behaviors. One latent state was characterized by a high probability of no change, corresponding to the relying on fortune pattern. Another state predominantly emitted small changes, aligning with the step-by-step pattern. The third state was associated with large changes, reflecting the strategy change pattern. Importantly, this state was not limited to large code modifications alone; it also included small adjustments following major changes, as well as immediate re-execution without modification. These results demonstrate that the latent states inferred by the HMM correspond closely to the heuristic patterns articulated by researchers.

After implementing our heuristic model into the HMM, we examine whether the model can be expanded and elaborated by relaxing the constraints we imposed for HMM. To determine the optimal number of hidden states, we fitted models with two to five states and evaluated them using the Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC). The results were as follows: 2 states (AIC = 3136.84, BIC = 3176.82), 3 states (3064.28, 3144.23), 4 states (3031.67, 3163.01), and 5 states (3034.39, 3228.55). BIC was minimized at three states, whereas AIC favored a four-state model. These results suggest that additional patterns may emerge by relaxing the constraints of the HMM, such as increasing the number of hidden states or modeling the code change score as a continuous variable rather than a categorical one.

5. DISCUSSION AND CONCLUSION

This study aimed to understand computational problem solving (CPS) processes in VEXcode VR, a game-based learning (GBL) environment, by examining how learners iteratively engage in testing and revision across trials using a code change score. Three dominant CPS patterns were identified in this environment: step-by-step, relying on fortune, and strategy change. These patterns reflect both general problem-solving strategies and the unique affordances of coding in a game-based context.

The step-by-step pattern reflects how learners in VEXcode VR construct partial solutions, test them, and incrementally refine their code. This behavior is closely tied to the environment, which allows a code to be executed and evaluated at intermediate stages, thereby providing built-in scaffolding for iterative development.

The relying on fortune pattern highlights game-specific characteristics. Due to stochastic elements, identical code executions can yield different outcomes, leading learners to repeatedly run the same code to achieve better results. While step-by-step reflects formative, incremental refinement, relying on fortune resembles summative evaluation under varying conditions without immediate modification. These patterns are also shaped by the ill-structured nature of the problem space. With no single correct solution and evaluation based on relative performance (e.g., scores), learners must interpret feedback and make decisions under uncertainty.

The strategy change patterns are consistent with general problem-solving process. This represents a substantial shift in approach, involving major modifications or entirely new solutions, and aligns more closely with general problem-solving processes than with game-specific mechanics.

Taken together, our findings show how the specific characteristics of the VEXcode VR environment, such as stochastic playground outcomes, executable intermediate states, and ill-structured tasks, shaped the way learners develop strategies. This aligns with Matcha and colleagues [5], which highlights that different learning modalities can give rise to distinct strategies even under the same

theoretical framework. For example, while the observation of step-by-step and strategy change patterns aligned with common CPS processes, the strategy of relying on fortune appeared to be unique to VEXcode VR, where the repeated execution of the same program may produce different outcomes. This underscores the importance of understanding the unique modalities of both CPS and game-based environments when analyzing and interpreting learners' behaviors.

However, several limitations should be addressed in future research. First, the sample size of 18 participants limits the generalizability of the findings. Therefore, future studies should include larger and more diverse datasets. Second, the identified heuristic patterns require further validation through methods such as systematic video coding and inter-rater reliability analysis. Third, the use of the 95% threshold for defining strategy change should be theoretically justified rather than relying solely on a data-driven criterion. Finally, to determine whether these patterns are educationally meaningful, future research should examine their relationships with learning outcomes such as task performance, code efficiency, and learning gains.

6. ACKNOWLEDGMENTS

This study was supported by the National Science Foundation (NSF; # 2229612). Any conclusions expressed in this material do not necessarily reflect the views of the NSF. We would also like to thank the entire INVITE project team for their support throughout this project.

7. ACKNOWLEDGEMENT OF GENERATIVE AI USE

This study used generative AI to assist with data analysis programming and draft revision. No study data were directly uploaded to AI systems, and all AI-generated content was carefully reviewed by the first author.

8. REFERENCES

[1] Gao, L., Fabricatore, C. and Lopez, M.X. 2019. Game features in inquiry game-based learning strategies: A systematic synthesis. *Proceedings of the European Conference on Gamesbased Learning* (2019), 854–62.

[2] Jiang, X., Harteveld, C., Huang, X. and Fung, A.Y.H. 2019. The computational puzzle design framework: a design guide for games teaching computational thinking. *Proceedings of the 14th International Conference on the Foundations of Digital Games* (San Luis Obispo California USA, Aug. 2019), 1–11.

[3] Jonassen, D.H. 1997. Instructional design models for well-structured and III-structured problem-solving learning outcomes. *Educational Technology Research and Development*. 45, 1 (Mar. 1997), 65–94. https://doi.org/10.1007/BF02299613.

[4] Liu, T. and Israel, M. 2022. Uncovering students' problem-solving processes in game-based learning environments. *Computers & Education*. 182, (June 2022), 104462. https://doi.org/10.1016/j.compedu.2022.104462.

[5] Matcha, W., Gašević, D., Ahmad Uzir, N., Jovanović, J., Pardo, A., Lim, L., Maldonado-Mahauad, J., Gentili, S., Pérez-Sanagustín, M. and Tsai, Y.-S. 2020. Analytics of Learning Strategies: Role of Course Design and Delivery Modality. *Journal of Learning Analytics*. 7, 2 (Sept. 2020), 45–71. https://doi.org/10.18608/jla.2020.72.3.

[6] Papert, S.A. 2020. *Mindstorms: Children, computers, and powerful ideas*. Basic books.

[7] Rowe, E., Almeda, M.V., Asbell-Clarke, J., Scruggs, R., Baker, R., Bardar, E. and Gasca, S. 2021. Assessing implicit computational thinking in Zoombinis puzzle gameplay. *Computers in Human Behavior*. 120, (July 2021), 106707. https://doi.org/10.1016/j.chb.2021.106707.

[8] Teng, K. and Chung, G.K.W.K. 2025. Measuring Children's Computational Thinking and Problem-Solving in a Block-Based Programming Game. *Education Sciences*. 15, 1 (Jan. 2025), 51. https://doi.org/10.3390/educsci15010051.

[9] Turchi, T., Fogli, D. and Malizia, A. 2019. Fostering computational thinking through collaborative game-based learning. *Multimedia Tools and Applications*. 78, 10 (May 2019), 13649–13673. https://doi.org/10.1007/s11042-019-7229-9.

[10] Wing, J.M. 2006. Computational thinking. *Communications of the ACM*. 49, 3 (Mar. 2006), 33–35. https://doi.org/10.1145/1118178.1118215.

MCAR-IITA: A Procedure for Identifying Structures among Items from Incomplete Data

Peter Steiner
St. Gallen University of Teacher Education
St.Gallen, Switzerland
peter.steiner@phsg.ch

Aliaksei Badnarchuk
St. Gallen University of Teacher Education
St.Gallen, Switzerland
alexejbodnar-chuk@gmail.com

Debora De Chiusole
University of Padova
Padua, Italy
debora.dechi-usole@unipd.it

Milan Segedinac
University of Novi Sad
Novi Sad, Serbia
milansegedi-nac@uns.ac.rs

Goran Savić
University of Novi Sad
Novi Sad, Serbia
savicg@uns.ac.rs

Luca Stefanutti
University of Padova
Padua, Italy
luca.stefanutti@unipd.it

ABSTRACT

Effective adaptive instruction depends on knowing what a student has mastered and what they are ready to learn next. Traditional scale scores, such as those obtained from item response (IRT) models, provide only limited support for this purpose. Consequently, several recent approaches conduct student assessment within hierarchical structures of test items or competencies. While these approaches are theoretically promising, practically applicable methods for identifying such structures from empirical data remain scarce, particularly in the presence of missing item responses. To address this gap, this contribution introduces MCAR-IITA— a new implementation of Inductive Item Tree Analysis (IITA) that provides three variants of handling missing responses. To evaluate its performance, the following research question was addressed: At what percentage of missing responses can the proposed variants still recover the underlying structure? Response data were simulated along a true structure under two conditions: 1) without noise and 2) with noise, using fixed careless errors and guessing probabilities of .20 per item. Missing responses were introduced incrementally across conditions, and performance was assessed in terms of the proportion of missing data up to which the true structure could still be recovered. Additionally, the same procedure was applied to a subset of the PISA 2003 dataset based on a previously identified structure. For each dataset, the three variants were evaluated using 1,000 replications. On average, the algorithm recovered the underlying structure with a missing response percentage of 28.47% (SD = 4.49pp) for the error-free simulation, 19.93% (SD = 4.44pp) for the simulation with noise, and 15.41% (SD = 4.65pp) for the PISA data. As this study examines a first implementation and is restricted to a limited set of conditions, the results provide an initial estimate of missing-data tolerance while highlighting substantial opportunities for further optimization.

Peter Steiner, Aliaksei Badnarchuk, Debora De Chiusole, Milan Segedinac, Goran Savić, Luca Stefanutti, and Jan Hochweber. MCAR-IITA: A Procedure for Identifying Structures among Items from Incomplete Data. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 694–698. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039804

Do not delete, move, or resize this block. If the paper is accepted, this block will

Keywords

Educational Assessment, Cognitive Diagnostics, Skill Structures, Inductive Item Tree Analysis, Knowledge Space Theory

1. INTRODUCTION

Tailoring instruction to students' current states of learning requires fine-grained information about both their mastered skills and their next feasible learning opportunities. Since traditional scale scores, such as those obtained from Item Response Theory (IRT) models, provide only limited support for this purpose, novel statistical frameworks that employ hierarchical structures have been developed (e.g., DBN [7], HCDM [13], KST [5]). While these approaches are theoretically promising and some methods and software exist that make their application feasible in certain contexts, successful applications remain limited and further development is needed to expand their range of use.

In this contribution, Inductive Item Tree Analysis (IITA) [15], which is one of the primary methods for establishing hierarchical structures from response data, will be introduced briefly, followed by an illustration of its use in educational assessment using Knowledge Space Theory (KST) [5] as an example. Afterward, MCAR-IITA—a new software implementation of IITA that provides three different versions of missing response handling—will be presented. The performance of this implementation with different proportions of missing data will be evaluated in a simulation study.

2. THEORETICAL BACKGROUND

IITA is a method of Boolean analysis used to derive hierarchical structures among items. The IITA algorithm takes a table of Boolean values as an input. Each column represents an item, e.g. a problem from a test or a question in a survey, and each row represents a student. Each value in the table shows if the subject represented by the row solved the item represented by the column correctly (1) or incorrectly (0). In this study, the algorithm will be discussed in the context of items within competence tests.

The output of the algorithm is a binary relation on a set of items Q, represented by ordered pairs of the form $a \leq b$ for $a, b \in Q$. The relation $a \leq b$ on the set of items Q indicates that mastering item a is a prerequisite for mastering item b. Since from the mastery of item b, the mastery of item a can be surmised, this relation is called

surmise relation. It forms a quasi-order among the items of Q, i.e. it is assumed to be reflexive, $(\forall a \in Q: a \leq a)$, and transitive, $(\forall a, \mathrm{b}, \mathrm{c} \in Q: (\mathrm{a} \leq b \wedge b \leq \mathrm{c}) \Rightarrow b \leq c)$.

The surmise relation might be viewed as a theoretical model of the cognitive organization of knowledge in a given item domain and, as such, constitutes a latent construct, that is not directly observable from data. In practice, measurement errors introduce a probabilistic discrepancy between the latent surmise relation and the manifest response patterns, making it impossible to directly infer relationships from observed data. Therefore, the IITA algorithm aims to inductively derive the quasi-order that best fits the observed response data, thereby reconstructing the latent surmise relation from empirical observations.

These quasi-orders find direct application in educational assessment, most notably in KST, which is the basis of the commercial assessment platform ALEKS [1] and is considered one of the most efficient frameworks for adaptive instruction [6]. Unlike traditional psychometric approaches—both classical test theory and IRT—KST does not attempt to compute a numerical score representing a learner's proficiency level. Instead, the goal of assessment is to provide a precise description of what a learner knows and does not know at a given moment [5].

Student assessment in KST is based on a hierarchical structure. Let Q denote a nonempty, finite set of items (or problems) that can be formulated within a particular topic-domain. In KST, a learner's knowledge is represented by a subset $K \subseteq Q$ containing exactly those items that the learner is capable of solving.

Not all subsets of Q necessarily are knowledge states. Logical and pedagogical dependencies among items impose constraints on which subsets are practically possible. One way of modeling these dependencies is by establishing a surmise relation on Q.

The collection of all knowledge states existing in a population of learners is called a knowledge structure, formally defined as a pair $(\mathrm{Q}, \mathcal{K})$, where $\mathcal{K} \subseteq 2^Q$ is a family of subsets of Q containing at least the empty set $\emptyset$ (representing a learner who has mastered none of the items) and Q itself (representing a learner who has mastered all items). A knowledge structure $(Q, \mathcal{K})$ is called a knowledge space if $\mathcal{K}$ is closed under set union. Every knowledge structure induced by a surmise relation is a knowledge space.

Even though new data-driven methods for building general knowledge structures have been developed IITA remains the only procedure capable of extracting a quasi-order from the data [3] and can thus still be considered as the standard [12]. The quasi-order resulting from IITA can be directly translated into the basis of a knowledge space. As an example, consider the quasi-order represented by the ordered pairs $(I1, I2)$, $(I1, I3)$, $(I1, I4)$, $(I1, I5)$, $(I2, I3)$, $(I2, I4)$, $(I2, I5)$, $(I3, I4)$, $(I3, I5)$. The surmise relation can be visualized in an incidence matrix (see Table 1).

Table 1. Surmise relation as binary incidence matrix

Item	I1	I2	I3	I4	I5
I1	1	1	1	1	1
I2	0	1	1	1	1
I3	0	0	1	1	1
I4	0	0	0	1	0
I5	0	0	0	0	1

The basis of a knowledge space, which is the minimal set of knowledge states K from which all other knowledge states can be derived by set union, can be constructed by transposing the incidence matrix. The knowledge states K are the rows of the resulting matrix. The corresponding knowledge space is the union closure of the knowledge states from the basis. It can be visualized through a directed acyclic graph (see Figure 1).

Despite the benefits of KST and its close connection with IITA, current software implementations of IITA lack practical applicability mainly due to restrictive assumptions on the input response data. IITA and its predecessor Item Tree Analysis (ITA) [16] have publicly accessible implementations in R and Python. While there is an ITA version in the R package pks, IITA can be found in the R package DAKS [14] and the Python package kst available from Milan Segedinac. Importantly, all implementations require binary datasets without missing data.

Missing data constitutes a longstanding and well-known challenge in statistical inference, particularly with respect to decisions on how missing responses should be treated in data analysis [2]. Even under carefully controlled data collection procedures, missing responses

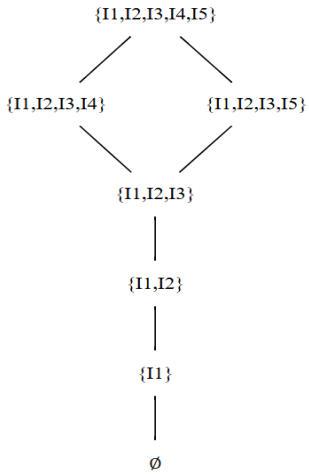


Figure 1. Knowledge space

frequently occur for a variety of reasons. In educational assessment contexts, such reasons include a lack of knowledge about an item, accidental skipping, time constraints during test completion, or design-related factors such as adaptive testing procedures. Moreover, many large-scale educational studies deliberately employ test designs in which each participant is administered only a subset of a substantially larger item pool. Such matrix sampling or booklet designs, as used for example in the PISA study, induce a large proportion of structurally missing responses by design rather than by student behavior. Consequently, missing data is not an exceptional phenomenon but rather an inherent characteristic of empirical educational data, especially in large-scale and computer-based assessments.

One of the first researchers to provide a systematic framework for addressing missing data mechanisms was Rubin [9]. He emphasized that decisions about the treatment of missing data should not be made without explicitly considering the process that generated the missingness. To formalize this idea, a complete dataset Y_{com} be a tuple $Y_{com} = (Y_{obs}, Y_{miss})$ of observed data and missing data. Furthermore, let U be a missingness indicator specifying which responses are observed and which are missing.

Within this framework, Rubin distinguished three fundamental missing data mechanisms. Data are said to be missing completely at random (MCAR) if the probability of missingness is independent of both observed and unobserved data, that is, $P(U|Y_{com}) = P(U)$.

A less restrictive mechanism is missing at random (MAR), where missingness may depend on observed data but not on the missing responses themselves, formally $P(U|Y_{com}) = P(U|Y_{obs})$.

Finally, data are missing not at random (MNAR) if the probability of missingness depends directly on the unobserved values Y_{mis}.

While missing response handling has been thoroughly studied within IRT [8], it has not played a critical role in KST literature. Even though models for parameter estimation based on incomplete data have been developed within KST [4], the construction of the knowledge structure from data with missing responses has not yet been addressed.

Against this background, the present contribution introduces MCAR-IITA—a new implementation of IITA with three different variants of handling missing responses—assuming that missingness is completely at random (MCAR). Its performance was initially evaluated by addressing the following research question: At what percentage of missing data can the proposed IITA variants recover the true structure from response data?

3. THE MISSING-COMPLETELY-AT-RANDOM IITA (MCAR-IITA)

In this section, the three variants of the new implementation of IITA for handling missing responses are presented. Python code for the implementation and the subsequently reported simulation study can be found at the following link: https://anonymous.4open.science/r/iita_python_testing/.

The implementation is developed following the three steps of IITA outlined in the Theoretical Background. In the first step, the number of counterexamples for each item pair is calculated. In the second step, an array of inductively generated quasi-orders is output, which are evaluated regarding their fit in the third step.

For handling missing values, our implementation primarily adapts the first step of the IITA procedure. We provide three variants for this step, of which the first (*IITA-ABS*) serves as default. This default variant ignores missing responses entirely when counting counterexamples, proceeding only with the observed responses for each item pair. A potential limitation of the IITA-ABS variant arises from the fact that the original IITA algorithm counts counterexamples in absolute numbers rather than proportions. In the context of complete datasets this approach is unproblematic, since all item pairs are observed with equal frequency across respondents. However, when missing responses are distributed unevenly across items, this assumption no longer holds. The second variant (*IITA-REL*) accounts for the issue described above by dividing the absolute number of counterexamples by the number of cases where a counterexample can appear (i.e., where neither response is missing) for every item pair. Thereby, a relative counterexample counter with values between 0 and 1 is obtained. In the third variant (*IITA-IMP*), the missing responses are first imputed by the item-mean imputation, replacing each missing value with the mean of the available responses of the respective item. Afterwards the counterexamples are calculated as in the second variant.

4. SIMULATION STUDY

To evaluate the IITA implementation and its three variants of missing data handling, a simulation study was conducted.

4.1 Methods

As in previous studies about the performance of IITA [10, 15], we used the *pisa* dataset from the DAKS package, which contains the item responses by 340 German students on a five-item dichotomously scored mathematical literacy test. This dataset will be referred to as *pisa-orig*. Applying IITA to this dataset results in the surmise relation depicted in Table 1. Since it is real world data, the relation among the items is naturally latent. Accounting for that fact, additional data was simulated along this structure. The resulting datasets *pisa-sim-0-0* and *pisa-sim-2-2* also contain 340 response patterns to five items. While *pisa-sim-0-0* was simulated with zero noise such as careless errors and lucky guesses, thereby containing only knowledge states as response patterns, the dataset *pisa-sim-2-2* was simulated with a probability of .20 for guessing and careless errors. Originally all three datasets do not contain any missing responses.

The quasi-order, identified originally from the *pisa-orig* dataset was taken as the true underlying structure for all three datasets. To test the percentage of missing data that the three different variants of the implementation can handle, missing responses were inductively added at random positions via a data amputation procedure [11]. Therefore, in every iteration, possible positions for missing responses in the dataset are identified. A position is possible, if there is no missing value already and if neither a column nor a row is completely empty after adding the missing value. From the pool of possible positions, two are chosen randomly. The data at these positions of all three datasets are replaced by missing responses. After introducing the missing responses, all variants are applied to all three datasets. The quasi-order extracted by each variant is compared with the true quasi-order. If one variant produces a different quasi-order from one of the datasets, it is not applied to that dataset again in the next iteration. The procedure ends if no variant produces the correct quasi-order on any of the datasets anymore. This was repeated for 1,000 iterations to account for positional differences in the missing responses. The data were descriptively analyzed by calculating the means and standard deviations for each dataset-variant combination.

4.2 Results

As illustrated in Figure 2, the true partial order was extracted up to a missing data proportion of 21.22% (SD = 7.29pp) on average

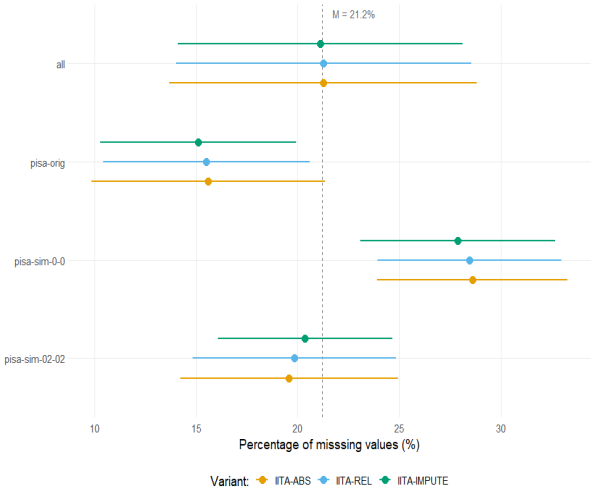


Figure 2. Comparison of average missing value percentage across the different datasets and different variants

across all three variants and datasets. For the real-world dataset *pisa-orig*, all three variants performed on average worse than on the synthetic ones, with an average missing data proportion of 15.41%

(SD = 4.65pp). The best-performing variant on this dataset was IITA-ABS with an average tolerance of 15.6% (SD = 5.75pp) of missing data.

As expected, on the two synthetic datasets all three variants performed better on the noiseless dataset *pisa-sim-0-0*. The average missing data coverage on this dataset across all three variants is 28.31% (SD = 4.49pp). The ranking across the three variants is the same as on the *pisa-orig* dataset: the IITA-ABS performs best (M = 28.6, SD = 4.68pp), the IITA-REL performs slightly worse (M = 28.47, SD = 4.53pp) and the IITA-IMP performs worst (M = 27.88, SD = 4.79pp).

Interestingly, the picture changes when considering the *pisa-sim-2-2* dataset, where careless errors and lucky guesses were introduced with a noise probability of .20. On average, a tolerance for missing data of 19.93% (SD = 4.44pp) was achieved across all three variants, with IITA-IMP performing best (M = 20.37%, SD = 4.31pp). The IITA-REL variant performed slightly worse (M = 19.84%, SD = 5.01pp), while IITA-ABS—previously the best-performing variant on the other two datasets—performed worst on this dataset (M = 19.58%, SD = 5.35pp).

4.3 Discussion

The results demonstrate the capability of the MCAR-IITA implementation of handling missing data. Particularly compared to other psychometric methods, an average tolerance for missing data of 21.22% can be considered substantial [2]. Overall, all three IITA variants perform similarly, showing a good tolerance for missing data on the simulated datasets (about 20%) as well as on the real dataset (about 15%).

These findings should be interpreted in light of several limitations. First, the study design accounted only for the factor of missing value positions within the dataset. In further studies more factors like the size of the dataset and domain characteristics like the amount of pairwise relations should be included.

Furthermore, while MCAR-IITA is theoretically able to handle all MAR-data, this study evaluates only the performance on MCAR data, which represents the least complex case of missingness. While future research should continue to investigate this special case, given that it enables the application of IITA to a broad range of large-scale assessment data with multi matrix designs, examining MAR and MNAR mechanisms would also be of practical value.

Lastly, it should be noted that the proportion of missing responses at which the implementation ceases to recover the correct structure provides only an initial insight into algorithmic performance and precludes further informative analyses. For instance, if the algorithms had continued to run after outputting a non-exact structure, their behavior could have been analyzed with respect to the difference between the recovered and the true partial order.

Beyond refining the simulation study design to better understand the interaction of IITA with missing responses, several other avenues for future research warrant consideration. While the implementation presented has already yielded promising results, there is still room for improvement. Especially, since mainly the first phase of IITA was adapted to suit datasets with missing values, manipulating the fit metrics at the third step could yield further optimization potential.

5. CONCLUSION

Inductive Item Tree Analysis constitutes one of the foundational methods for enabling state-of-the-art formative student assessment. However, its practical applicability has been constrained by restrictions on the input data. In particular, handling missing responses remains one of the most pressing challenges, as they occur in the vast majority of empirical datasets. This contribution introduced and evaluated MCAR-IITA—an IITA implementation incorporating three distinct variants for handling missing responses. The results are promising, demonstrating that the implementation can recover underlying item structures even in the presence of substantial proportions of missing values (i.e., up to about 20%). Beyond these findings, this work opens a variety of avenues for further investigation, both with respect to the implementation presented here and regarding additional algorithmic optimizations. Thus, this contribution represents an important step toward making state-of-the-art psychometric methods more broadly applicable in educational practice.

6. ADDITIONAL AUTHORS

Additional author: Jan Hochweber (St. Gallen University of Teacher Education, email: jan.hochweber@phsg.ch).

7. REFERENCES

[1] Cosyn, E., Uzun, H., Doble, C. and Matayoshi, J. 2021. A practical perspective on knowledge space theory: ALEKS and its data. *Journal of Mathematical Psychology*. 101, (2021), 102512. https://doi.org/10.1016/j.jmp.2021.102512.

[2] Dai, S. 2021. Handling Missing Responses in Psychometrics: Methods and Software. *Psych*. 3, 4 (Nov. 2021), 673–693. https://doi.org/10.3390/psych3040043.

[3] De Chiusole, D., Spoto, A. and Stefanutti, L. 2024. Innovative Methods for Building Knowledge Structures. *Advanced Series on Mathematical Psychology*. WORLD SCIENTIFIC. 87–104.

[4] De Chiusole, D., Stefanutti, L., Anselmi, P. and Robusto, E. 2015. Modeling missing data in knowledge space theory. *Psychological Methods*. 20, 4 (2015), 506–522. https://doi.org/10.1037/met0000050.

[5] Doignon, J.-P. and Falmagne, J.-C. 1985. Spaces for the assessment of knowledge. *International Journal of Man-Machine Studies*. 23, 2 (Aug. 1985), 175–196. https://doi.org/10.1016/S0020-7373(85)80031-6.

[6] Ihichr, A., Oustous, O., Idrissi, Y.E.B.E. and Lahcen, A.A. 2024. A Systematic Review on Assessment in Adaptive Learning: Theories, Algorithms and Techniques. *International Journal of Advanced Computer Science and Applications*. 15, 7 (2024). https://doi.org/10.14569/IJACSA.2024.0150785.

[7] Käser, T., Klingler, S., Schwing, A.G. and Gross, M. 2017. Dynamic Bayesian Networks for Student Modeling. *IEEE Transactions on Learning Technologies*. 10, 4 (Oct. 2017), 450–462. https://doi.org/10.1109/TLT.2017.2689017.

[8] Little, R.J.A. and Rubin, D.B. 2002. *Statistical Analysis with Missing Data*. Wiley.

[9] Rubin, D.B. 1976. Inference and missing data. *Biometrika*. 63, 3 (1976), 581–592. https://doi.org/10.1093/biomet/63.3.581.

[10] Sargin, A. and Ünlü, A. 2009. Inductive item tree analysis: Corrections, improvements, and comparisons. *Mathematical Social Sciences*. 58, 3 (Nov. 2009), 376–392. https://doi.org/10.1016/j.mathsocsci.2009.06.001.

[11] Schouten, R.M., Lugtig, P. and Vink, G. 2018. Generating missing values for simulation purposes: a multivariate amputation procedure. *Journal of Statistical Computation and Simulation*. 88, 15 (Oct. 2018), 2909–2930. https://doi.org/10.1080/00949655.2018.1491577.

[12] Segedinac, M.T., Horvat, S., Rodić, D.D., Rončević, T.N. and Savić, G. 2018. Using knowledge space theory to compare expected and real knowledge spaces in learning stoichiometry. *Chemistry Education Research and Practice*. 19, 3 (2018), 670–680. https://doi.org/10.1039/C8RP00052B.

[13] Templin, J. and Bradshaw, L. 2014. Hierarchical Diagnostic Classification Models: A Family of Models for Estimating and Testing Attribute Hierarchies. *Psychometrika*. 79, 2 (Apr. 2014), 317–339. https://doi.org/10.1007/s11336-013-9362-0.

[14] Uenlue, A. and Sargin, A. 2009. DAKS: Data Analysis and Knowledge Spaces.

[15] Ünlü, A. and Schrepp, M. 2021. Generalized inductive item tree analysis. *Journal of Mathematical Psychology*. 103, (Aug. 2021), 102547. https://doi.org/10.1016/j.jmp.2021.102547.

[16] Van Leeuwe, J.F. 1974. Item tree analysis. *Nederlands Tijdschrift voor de Psychologie en haar Grensgebieden*. 29, 6 (1974), 475–483.

Measuring Feature-Level Changes in Preprocessing Unfairness Mitigation

Hanning Lin
University of California Berkley
hanning4@illinois.edu

Frank Stinar
University of Illinois Urbana–Champaign
fstinar2@illinois.edu

ABSTRACT

Gender bias in machine learning models can perpetuate and amplify existing societal inequalities, especially in domains where the users are vulnerable such as in education. In this study, we measure the effectiveness of multiple preprocessing unfairness mitigation strategies to reduce gender bias in predictive models trained on student data. Furthermore, we investigate how these mitigation techniques influence the underlying feature representations of the data, yielding insights into the procedural mechanisms by which bias is reduced. Through a series of experiments using four education datasets, we improved fairness and found that some methods dramatically modify data more than others. Using a diverse set of fairness metrics and statistical divergence methods, such as Wasserstein distance, we observe that the level of data modification and the level of bias reduction are not strongly related. To further isolate the role of data transformation, we additionally analyze mitigation methods that induce comparable levels of feature-level data modification, enabling more controlled comparisons across techniques. These results provide an initial analysis into the complexity of the unfairness mitigation step for educational modeling.

Keywords

Algorithmic fairness, Unfairness mitigation, Data transformation, Grade prediction

1. INTRODUCTION

Researchers and educators have harnessed big data mining techniques as a powerful way to leverage increasing quantities of educational data to improve learning outcomes, personalize education, and support diverse groups of students [6, 10]. One common tool that leads to these benefits is predictive modeling, which has proven useful in tasks such as grade prediction, mastery prediction, course recommendation, and dropout prevention [2, 29, 20]. However, as predictive models become more influential in educational decision-making, concerns about the fairness of these models have grown. Bias in educational data can perpetuate inequalities and even undermine the potential benefits of educational data mining methods [33, 24, 7]. Addressing these biases is critical to ensuring that predictive models do not exacerbate existing disparities in educational systems.

Education researchers have already begun to mitigate and understand bias in machine learning-powered educational systems [43]. While there has been varying success in applying unfairness mitigation techniques, there has been limited exploration of how preprocessing-based unfairness mitigation methods transform data in unexpected or opaque ways [19, 13]. Understanding these data transformations is important, as they can have downstream effects on model performance, interpretability, and fairness.

In this paper, we evaluate multiple preprocessing unfairness mitigation strategies across four educational datasets using a standardized data-transformation pipeline. Beyond measuring overall fairness improvements, we analyze how these mitigation techniques influence feature-level data representations. Rather than relying on a single divergence measure, we assess mitigation effectiveness using multiple fairness metrics, allowing for a more comprehensive evaluation of bias reduction. In addition, we compare mitigation methods that induce similar levels of feature-level data modification, enabling more controlled comparisons that disentangle the effects of the mitigation mechanism from the extent of data transformation itself. Thus, we examine how different preprocessing unfairness mitigation techniques can alter educational data and how the transformations influence the fairness and utility of grade prediction models. Our analysis is shaped by the following research questions:

Research Question 1 (RQ1): Are preprocessing unfairness mitigation strategies applicable in reducing gender bias in grade prediction?

Research Question 2 (RQ2): How can preprocessing unfairness mitigation strategies differentially impact student data?

We answer RQ1 by analyzing how four preprocessing unfairness mitigation algorithms can reduce gender bias in four education datasets. To answer RQ2, we analyze if the amount of transformation is linked to the level of bias mitigated.

The preliminary results in this paper extend beyond tech-

Hanning Lin, and Frank Stinar. Measuring Feature-Level Changes in Preprocessing Unfairness Mitigation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 699–704. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039865

nical aspects of applying unfairness mitigation to grade-prediction. Through uncovering how preprocessing techniques differentially transform student data, the research helps to better understand the tradeoffs between unfairness mitigation and data integrity. That is, modifying the data could result in procedural unfairness if the resulting data represents students in unexpected, inaccurate, or otherwise undesirable ways [33, 36]. Thus, we hope to help education researchers in deciding how to implement unfairness mitigation techniques into transparent educational systems.

2. BACKGROUND

Machine learning pipelines in educational data mining often result in some type of prediction or learner model with the goal of improving student learning. These models are often used to predict a student's risk of dropping out [39, 1, 25], whether a learner has mastered certain topics in learning sessions [16, 29], grades on exams [3, 40, 17, 30], to name a few applications. As these predictive models are applied into classroom and online settings, concerns about the biases within these models have become increasingly examined [5, 23, 35]. To address the biases in the models, researchers in education and related fields have developed techniques to mitigate and measure the biases in the development of the models [19, 38, 37]. These techniques are often grouped into categories based on which step in a machine learning pipeline that act on. Preprocessing techniques transform the dataset used to train machine learning models [15, 22, 44]. Inprocessing techniques add other steps and parameters to the training process of models [41, 31]. Postprocessing techniques edit the predictions of the models [32, 28].

Researchers in education have already begun to test the utility of different unfairness mitigation techniques. In fact, educators have cared about issues of bias and fairness for decades [21]. Recently, researchers have developed techniques specific for educational data [38, 15, 8], examined how these techniques impact the data themselves [33], and surveyed how these algorithmic fairness issues should be handled in education [4, 14]. In this paper, we contribute to this research by examining if certain preprocessing unfairness mitigation techniques are useful at mitigating gender bias in grade prediction. Furthermore, we examine if there is a relationship between the gender bias mitigated and the amount that preprocessing techniques distort educational data, which is essential to understand for situations where such data distortion is undesirable.

3. METHOD

In the following section, we detail the datasets and experimental design. For RQ1, we evaluate whether preprocessing unfairness mitigation strategies reduce gender bias in grade prediction by comparing models trained on original datasets to models trained on mitigated datasets. For RQ2, we quantify the magnitude of data transformation introduced by each mitigation technique and analyze whether greater data modification corresponds to greater bias reduction in the trained models.

3.1 Data

We use four student performance datasets covering diverse age groups for prediction tasks. For all datasets, we frame the problem as a binary classification task, with student gender as the sensitive attribute. Gender is treated as a binary sensitive feature in our analyses since we are following the terminology used in the datasets. Other potentially sensitive attributes are removed prior to analysis (we note that unfairness can also exist between intersectional groups, but for our preliminary analysis we focus on one demographic all the datasets share). Table 1 includes the base rates and demographics of the datasets.

Table 1: Base rates of gender across datasets

Dataset	Gender	No. Obs	Pass (%)
HAR	Man	301	21.93
	Woman	413	17.92
POR	Man	266	64.29
	Woman	383	73.37
MAT	Man	187	56.68
	Woman	208	49.52
HSP	Man	87	48.28
	Woman	58	25.86

Harvard Student Performance Data (HAR): The HAR dataset is used to predict final grades in a Physics course. HAR contains 21 categorical and numerical features for 714 students. The 21 features consist of academic information and individual differences.

Student Portuguese and Math Performance Data (POR, MAT): The POR dataset predicts final grades in a Portuguese language course. For standardization, we binarized the outcome into pass or fail. POR contains 30 categorical and numerical features across 649 students. Similar to HSP, features in POR consist of general academic information (e.g., grades, attendance, etc.) and and individual difference measures. The MAT dataset undergoes the same transformation but is used to predict math grades.

Higher Education Student Performance Data (HSP): The HSP dataset is used to predict end-of-term grades. For our binary classification task, we transform the final grades as pass or fail. HSP contains 31 categorical and numerical features across 145 students in total. Features in this dataset focus on academics alongside a few individual difference measures (e.g., parental information).

3.2 Experiments

To answer RQ1, we applied four preprocessing-based unfairness mitigation methods: reweighting (Rw), FAIR-SMOTE (FSm), SMOTE (Sm), and Disparate Impact Remover (DRe). These methods were selected because they modify the data at different levels: reweighting adjusts instance weights without altering feature values, whereas SMOTE and FAIR-SMOTE generate synthetic samples to rebalance class and group distributions.

Reweighting (Rw) is a preprocessing technique designed to mitigate bias related to sensitive attributes. It assigns different weights to training instances based on the joint distribution of the sensitive attribute and the class label, reducing the influence of biased correlations during model training [22].

Table 2: AUC and fairness results of random forest models trained on the original and unfairness mitigated datasets.

Dataset	Mitigation	AUC	SP	DI	AO
HAR	–	.63	.05	1.20	.05
	Rw	.70	.06	1.11	.06
	FSm	.71	.07	1.19	.07
	Sm	.68	.05	1.21	.05
	DRe	.89	.04	0.91	.02
POR	–	.63	-.09	0.91	-.10
	Rw	.63	-.01	0.99	-.01
	FSm	.58	-.10	0.89	-.11
	Sm	.65	-.14	0.85	-.14
	DRe	.55	.07	0.92	.06
MAT	–	.58	.27	3.47	.27
	Rw	.61	.04	1.11	.04
	FSm	.64	.24	3.28	.24
	Sm	.63	.32	2.48	.32
	DRe	.44	.02	0.85	.04
HSP	–	.77	.03	1.07	-.15
	Rw	.60	-.04	0.91	-.07
	FSm	.53	-.03	0.93	-.12
	Sm	.51	-.23	0.62	-.36
	DRe	.79	-.05	0.67	-.02

FAIR-SMOTE (FSm) extends SMOTE by incorporating fairness constraints, generating synthetic samples that balance both class labels and sensitive attribute distributions [11].

SMOTE (Sm) addresses class imbalance by generating synthetic samples for the minority class through interpolation between existing instances and their nearest neighbors [12].

Disparate Impact Remover (DRe) applies a feature repair step that adjusts values so group-wise distributions align more closely with a fairness target (e.g., the "80% rule"), rather than changing instance weights or the label column [18].

We evaluated the effectiveness of these mitigation strategies using both predictive performance and fairness metrics. Predictive performance was measured using the area under the ROC curve (AUC). Fairness was assessed using statistical parity difference (SP), disparate impact (DI), and average odds difference (AO) [7, 41, 9]. For all experiments, we trained a random forest classifier with default hyperparameters to model student performance. The models were implemented using the *Scikit-learn* Python library, and the unfairness mitigation algorithms were implemented using the AI Fairness 360 toolkit and DebiasEd [27, 9, 34].

To answer RQ2, we analyze how each method transforms feature distributions. For each dataset, we compare the distributions of each feature before and after preprocessing using the Wasserstein distance (a metric that measures how different two distributions are by calculating how much work it would take to transform one into the other) [26].

4. RESULTS

The results of the unfairness mitigation on AUC and fairness measures is in Table 2. Considering that each dataset had differences in base rates between groups, we expected that the baseline model for each dataset could have bias due to differences in predicted rates. There is some bias in the baseline models denoted as the first row for each of the datasets in Table 2. Perfect fairness for SP and AO is 0, and perfect fairness for DI is 1. We report means across 5-fold cross-validation for each dataset.

For RQ1, we compared the fairness metrics of the baseline model with the fairness metrics of the models trained on the unfairness mitigated data. For each dataset, at least one model trained using unfairness mitigated data was fairer in terms of at least one metric. For example, Rw resulted in better DI by .09 given the HAR dataset. The MAT baseline showed the strongest bias (SP = .27, DI = 3.47, AO = .27). All four unfairness mitigation techniques were able to reduce bias with three of them also improving AUC in tandem. Thus, we conclude that the chosen preprocessing methods are capable of reducing gender bias in these models, albeit at differing levels.

Given some gender bias can be mitigated, we analyzed how the preprocessing differentiall impacts student data (RQ2). We found the average Wasserstein distance of the features before and after preprocessing across the four techniques and four datasets. The results are in Figure 1.

Our preliminary analysis finds slight trends in the data transformation of the mitigation methods. For example, generally, Rw and Sm transformed data more than the FSm and DRe methods. Also, while Rw had the highest Wasserstein distance across all features for HAR, the transformation did not relate to high levels of unfairness mitigation or AUC improvement. In fact, DRe resulted in better performance and fairness despite also having a lower Wasserstein distance. In contrast, DRe did not have better performance for MAT (the dataset with the highest gender bias) while still having the lowest Wasserstein distance. Thus, even though these four methods transform data at similar rates, our preliminary analysis does not find a relationship between the levels of transformation and the unfairness mitigated. The results further demonstrate this with three unique unfairness mitigation methods resulting in the least gender bias and these methods being unrelated to the level of data transformation.

5. DISCUSSION AND FUTURE WORK

The experiments revealed that preprocessing unfairness mitigation strategies are effective in reducing gender bias in grade prediction (RQ1), and examined how different types of preprocessing distort student data (RQ2). The results indicate that the preprocessing methods do not equally mitigate gender bias in grade prediction, and that inequality is seemingly not related to the amount that a dataset is transformed.

There are cases where preprocessing unfairness mitigation strategies can reduce gender bias in grade prediction within models trained on each of the datasets. However, the preprocessing methods are not equally effective for each dataset. As expected from previous research, different unfairness mitigation methods are more effective at improving certain dimensions of fairness [41]. At the same time, these statistical fairness definitions encode specific normative assumptions and improving them does not automatically imply equitable

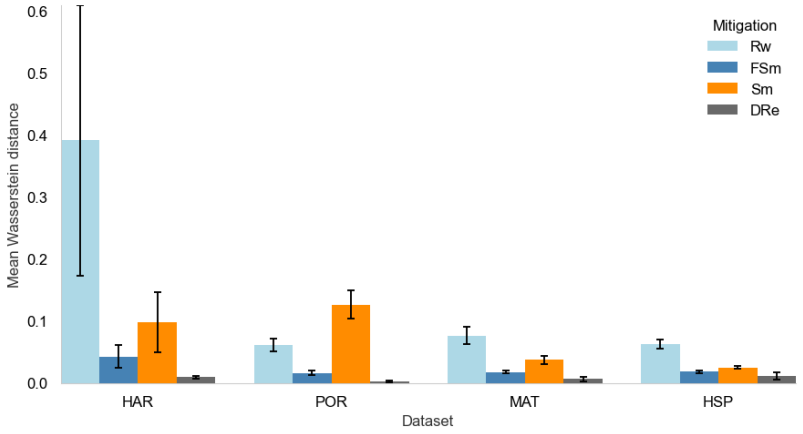


Figure 1: The average data divergence of the four unfairness mitigation methods on each of the datasets. On average, Rw distorts the data the most, followed by Sm, FSm, and DRe.

outcomes. However, notably, for the HAR and HSP datasets the model trained on a preprocessed version of the data using DRe resulted in a model that was fairer (in at least one dimension) *and* had better performance than the baseline. We also emphasize that mitigation is not uniformly beneficial: there are settings where fairness improves while AUC falls, where AUC improves while fairness worsens, or where one fairness metric improves while another degrades.

Preprocessing unfairness mitigation strategies having differential impacts on the data emphasizes the need for careful selection and adaptation of preprocessing unfairness mitigation methods in education. While DRe achieved fairness while minimally transforming the dataset, Rw achieved (sometimes better) fairness guarantees through distorting more data (in terms of the POR dataset). Furthermore, we found no relationship between the performance of a model, the unfairness of a model, and the amount in which the data was transformed. These findings are important for education researchers as they suggest that the most effective unfairness mitigation method is not necessarily the one that leads to the most transformation in the dataset. This flexibility in method is valuable in education as certain features being left untransformed could lead to greater explanatory power.

For practitioners, the correct mitigation to choose could be based on what must remain stable for the decision context: if preserving the literal feature values matters for communication and review, Rw may be preferable; if the priority is repairing dependence between protected attributes and outcomes in the input space, feature repair methods may be appropriate; if the issue is class imbalance, oversampling approaches may be relevant.

Therefore, future work can find mitigation strategies that improve fairness without sacrificing the transparency and interpretability of education models. Preprocessing techniques having seemingly no tradeoff between level of data distortion and unfairness mitigated offers opportunities for current preprocessing techniques to be improved and adapted for educational data. While DRe offers a possibility in this regard (through transforming only a few features), more research needs to be done to tailor this and other methods to education data. Moreover, Rw or Sm could be detrimental to the interpretability of models since they have greater data transformation. Furthermore, given that previous research has found that unfairness mitigation can transform certain features that could lead to procedural fairness harms [33], future work should move beyond our average estimates of Wasserstein distance. The next step would be to examine the specific features that are transformed by each preprocessing method. This could further provide insight into researchers as some methods could transform features that are important for model interpretability.

The study is limited by the sensitive attribute we chose to analyze, the unfairness mitigation techniques used, and the set of datasets. Since we focus on gender difference in grade prediction in a select set of tabular, categorical datasets, our results might not be generalizable to other types of data commonly used in educational data mining (e.g., knowledge tracing data [42]). Additionally, we only include a single (binary) sensitive attribute, which does not address intersectional fairness. Furthermore, we used only a few of an ever-increasing number of possible preprocessing unfairness mitigation techniques [19, 38]. Thus, our research could be further validated by testing the data distortion of other unfairness mitigation techniques.

The variability found in the relationship between dataset divergence, performance, and unfairness mitigation calls for multiple directions for future work. Our findings indicate that certain techniques could be promising in navigating this relationship, such as DRe, but future research needs to further test these hypotheses. Furthermore, future work can support the development of more education-specific preprocessing techniques with a focus on interpretability. Future extensions will include per-feature attribution of change (beyond dataset-level averages) paired with domain-informed features that must remain interpretable for instructors or advisors. Longer-term work could incorporate stakeholder-centered evaluation beyond offline metrics, and develop constraints common in education.

6. REFERENCES

[1] A. Alhothali, M. Albsisi, H. Assalahi, and T. Aldosemani. Predicting Student Outcomes in Online Courses Using Machine Learning Techniques: A Review. *Sustainability*, 14(10):1–14, 2022.

[2] K. E. Arnold and M. D. Pistilli. Course signals at Purdue: using learning analytics to increase student success. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge*, LAK '12, pages 267–270, New York, NY, USA, 2012. Association for Computing Machinery.

[3] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider. Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, 113:177–194, Oct. 2017.

[4] R. S. Baker, L. Esbenshade, J. Vitale, and S. Karumbaiah. Using Demographic Data as Predictor Variables: a Questionable Choice. *Journal of Educational Data Mining*, 15(2):22–52, June 2023.

[5] R. S. Baker and A. Hawn. Algorithmic Bias in Education. *International Journal of Artificial Intelligence in Education*, 32(4):1052–1092, Dec. 2022.

[6] R. S. Baker and P. S. Inventado. Educational Data Mining and Learning Analytics. In J. A. Larusson and B. White, editors, *Learning Analytics: From Research to Practice*, pages 61–75. Springer New York, New York, NY, 2014.

[7] S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

[8] C. Belitz, L. Jiang, and N. Bosch. Automating Procedurally Fair Feature Selection in Machine Learning. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 379–389, Virtual Event USA, July 2021. ACM.

[9] R. K. E. Bellamy, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, Y. Zhang, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, and S. Mehta. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):1–20, July 2019.

[10] C. Romero and S. Ventura. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6):601–618, Nov. 2010.

[11] J. Chakraborty, S. Majumder, and T. Menzies. Bias in machine learning software: why? how? what to do? In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2021, pages 429–440, New York, NY, USA, 2021. Association for Computing Machinery.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, June 2002.

[13] Z. Chen, J. M. Zhang, F. Sarro, and M. Harman. A Comprehensive Empirical Study of Bias Mitigation Methods for Machine Learning Classifiers. *ACM Trans. Softw. Eng. Methodol.*, 32(4):1–30, May 2023.

[14] S. V. Chinta, Z. Wang, Z. Yin, N. Hoang, M. Gonzalez, T. L. Quy, and W. Zhang. FairAIED: Navigating Fairness, Bias, and Ethics in Educational AI Applications, July 2024.

[15] J. M. Cock, M. Bilal, R. Davis, M. Marras, and T. Kaser. Protected attributes tell us who, behavior tells us how: A comparison of demographic and behavioral oversampling for fair student success modeling. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 488–498, 2023.

[16] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modelling and User-Adapted Interaction*, 4(4):253–278, 1995.

[17] P. Cortez and A. Silva. Using data mining to predict secondary school student performance. In J. L. Afonso, C. Cuoto, A. Lago Ferreiro, J. S. Martins, and A. Nogueiras Melèndez, editors, *Proceedings of 5th Annual Future Business Technology Conference*, volume 5, pages 5–12. EUROSIS-ETI, 2008.

[18] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[19] M. Hort, Z. Chen, J. M. Zhang, M. Harman, and F. Sarro. Bias Mitigation for Machine Learning Classifiers: A Comprehensive Survey. *ACM J. Responsib. Comput.*, 1(2):1–52, June 2024.

[20] Q. Hu and H. Rangwala. Towards fair educational data mining: A case study on detecting at-risk students. In A. N. Rafferty, J. Whitehill, C. Romero, and V. Cavalli-Sforza, editors, *Proceedings of the 13th International Conference on Educational Data Mining, EDM 2020, Fully virtual conference, July 10-13, 2020*. International Educational Data Mining Society, 2020.

[21] B. Hutchinson and M. Mitchell. 50 Years of Test (Un)fairness: Lessons for Machine Learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 49–58, Jan. 2019.

[22] F. Kamiran and T. Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, Oct. 2012.

[23] R. F. Kizilcec and H. Lee. Algorithmic fairness in education. In *The Ethics of Artificial Intelligence in Education*, pages 174–202. Routledge, New York, 1 edition, Aug. 2022.

[24] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A Survey on Bias and Fairness in Machine Learning. *ACM Computing Surveys*, 54(6):1–35, July 2022.

[25] C. Pan and Z. Zhang. Examining the algorithmic fairness in predicting high school dropouts. In B. PaaÃŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 262–269, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[26] V. M. Panaretos and Y. Zemel. Statistical Aspects of

Wasserstein Distances. *Annual Review of Statistics and Its Application*, 6(Volume 6, 2019):405–431, 2019. Type: Journal Article.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.

[28] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger. On fairness and calibration. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5684–5693, Red Hook, NY, USA, 2017. Curran Associates Inc.

[29] S. Qu, K. Li, B. Wu, X. Zhang, and K. Zhu. Predicting Student Performance and Deficiency in Mastering Knowledge Points in MOOCs Using Multi-Task Learning. *Entropy*, 21(12):12–16, Dec. 2019.

[30] R. J. Quinn and G. Gray. Prediction of student academic performance using Moodle data from a Further Education setting. *Irish Journal of Technology Enhanced Learning*, 5(1), Oct. 2019.

[31] N. Schreuder and E. Chzhen. Classification with abstention but without disparities. In C. de Campos and M. H. Maathuis, editors, *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pages 1227–1236. PMLR, July 2021.

[32] P. Snel and S. van Otterloo. Practical bias correction in neural networks: a credit default prediction case study. *Computers and Society Research Journal*, 3:1–12, 2022.

[33] F. Stinar and N. Bosch. Algorithmic unfairness mitigation in student models: When fairer methods lead to unintended results. In *Proceedings of the 15th International Conference on Educational Data Mining, EDM 2022*, Proceedings of the 15th International Conference on Educational Data Mining, EDM 2022. International Educational Data Mining Society, 2022.

[34] F. Stinar, J. Cock, R. Kizilcec, and T. Kaser. Applying DebiasEd: A Package for Mitigating Unfairness in Educational Data. In C. Mills, G. Alexandron, D. Taibi, G. L. Bosco, and L. Paquette, editors, *Proceedings of the 18th International Conference on Educational Data Mining*, pages 695–698, Palermo, Italy, July 2025. International Educational Data Mining Society.

[35] S. Tschiatschek, M. Knobelsdorf, and A. Singla. Equity and fairness of Bayesian knowledge tracing. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, pages 578–582, Durham, United Kingdom, July 2022. International Educational Data Mining Society.

[36] T. R. Tyler. What is Procedural Justice?: Criteria Used by Citizens to Assess the Fairness of Legal Procedures. *Law & Society Review*, 22(1):103–135, 1988.

[37] V. A. Vabenska, M. Verger, M. M. T. Rodrigo, C. J. G. Monterozo, R. S. Baker, M. Z. N. L. Saavedra, S. Lalla, and A. Shimada. Evaluating algorithmic bias in models for predicting academic performance of filipino students. In B. PaaÃŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 744–751, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.

[38] M. Verger, S. Lallé, F. Bouchet, V, and a. Luengo. Is Your Model "MADD"? A Novel Metric to Evaluate Algorithmic Fairness for Predictive Student Models. In *Proceedings of the 16th International Conference on Educational Data Mining*, pages 91–102. International Educational Data Mining Society, July 2023.

[39] W. Wang, Y. Zhao, Y. J. Wu, and M. Goh. Factors of dropout from MOOCs: a bibliometric review. *Library Hi Tech*, 41(2):432–453, June 2023.

[40] T. Yanagiura, S. Yano, M. Kihira, and Y. Okada. Examining Algorithmic Fairness for First- Term College Grade Prediction Models Relying on Pre-matriculation Data. *Journal of Educational Data Mining*, 15(3):1–25, Dec. 2023.

[41] M. B. Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi. Fairness Beyond Disparate Treatment & Disparate Impact: Learning Classification without Disparate Mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1171–1180, Perth Australia, Apr. 2017. International World Wide Web Conferences Steering Committee.

[42] A. F. Zambrano and R. S. Baker. Long-Term Prediction from Topic-Level Knowledge and Engagement in Mathematics Learning. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, LAK '24, pages 66–77, New York, NY, USA, 2024. Association for Computing Machinery.

[43] A. F. Zambrano, J. Zhang, and R. S. Baker. Investigating Algorithmic Bias on Bayesian Knowledge Tracing and Carelessness Detectors. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 349–359, Kyoto Japan, Mar. 2024. ACM.

[44] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning Fair Representations. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 325–333, Atlanta, Georgia, USA, June 2013. PMLR.

MetaPal: A New Paradigm for Human-Computer Interaction — From Command Response to Emotional Empathy

Hui Shao
East China Normal University
71285903006@stu.ecnu.edu.cn

Zizhuo Wu
East China Normal University
71285903022@stu.ecnu.edu.cn

Jing Chen
East China Normal University
51285903016@stu.ecnu.edu.cn

Shiyu Wang
East China Normal University
51285903056@stu.ecnu.edu.cn

Jiaqi Song
East China Normal University
51285903003@stu.ecnu.edu.cn

Yan Wang*
East China Normal University
yanwang@dase.ecnu.edu.cn

ABSTRACT

MetaPal is an intelligent system for cross-cultural empathetic oral dialogue education for international students, applying real-time multimodal affective computing to language education. Unlike traditional "command response" systems, MetaPal takes emotional empathy as the system core across all layers: perception, understanding, decision, and expression. The system fuses visual, audio, and text modalities, driving digital human interaction through an emotion-centered architecture that achieves a paradigm shift from "command response" to "emotional empathy." Through asynchronous processing, model lightweighting, and streaming transmission, the system achieves smooth real-time interaction. Small-scale user trials show positive feedback in oral confidence, cultural awareness, and emotional expression appropriateness. This demo presents the complete technical architecture, interaction process, and preliminary evaluation.

Keywords

Paradigm Shift, Affective Computing, Multimodal Fusion, Digital Human, Cross-Cultural Education

1. INTRODUCTION

1.1 Problem Background

Globalized education has intensified cross-cultural adaptation challenges for international students. Research shows that international students face significant challenges in learning and adjustment during their study abroad experiences [19]. Language learning presents specific challenges:

- **Emotional misunderstanding**: Reserved East Asian expressions are misinterpreted as indifference in Western contexts
- **Non-verbal cues**: Cultural differences in eye contact and facial expression create barriers
- **Vocal emotion**: Native tone transfer makes second language expressions seem emotionally flat

Traditional language tools focus on vocabulary and grammar, ignoring emotional and cultural dimensions. Existing affective computing research remains laboratory-based without systematic integration. Critically, current systems follow the "command response" paradigm, processing speech without perceiving emotional states—emotion is merely optional.

1.2 Related Work

As surveyed in [16], affective computing has established solid foundations for emotion-aware systems. **Application of Affective Computing in Education**: D'Mello et al. [4] analyzed emotional states during learning, showing emotion-cognition synergy is crucial. However, most studies use single modality without real-time multimodal fusion [3, 2]. Picard's foundational work [13] established affective computing as a field, while Pantic and Rothkrantz [11] pioneered affect-sensitive multimodal interaction.

Multimodal Learning Analytics: Existing classroom multimodal analysis frameworks have high latency, limiting real-time interaction support [12, 1]. Blikstein and Worsley [1] demonstrate computational approaches for measuring complex learning tasks.

Cross-Cultural Communication Theory: Hofstede's cultural dimension theory [8] provides a framework for quantifying cultural differences, but few studies have computationalized it for educational systems. Gao et al. [7] examine how cultural epistemologies influence emotional experience and regulation.

[0]* Corresponding author.

Hui Shao, Zizhuo Wu, Jing Chen, Shiyu Wang, Jiaqi Song, and Yan Wang. MetaPal: A New Paradigm for Human-Computer Interaction — From Command Response to Emotional Empathy. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 705–709. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040066

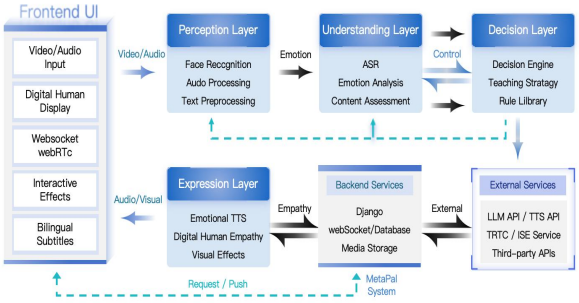


Figure 1: MetaPal architecture: Four-layer framework integrating perception, understanding, decision, and expression layers.

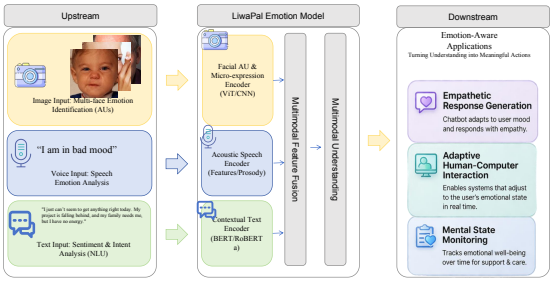


Figure 2: Perception layer: Real-time video capture with picture-in-picture slicing for emotion recognition.

1.3 Research Gap and Contributions

Research Gap: Existing systems lack real-time multimodal fusion, cross-cultural adaptation, and complete engineering implementation. Fundamentally, they follow "command response" logic with emotion as an add-on, while MetaPal takes "emotional empathy" as the operating system core across all layers.

Contributions:

1. **Theoretical**: Propose a new interaction paradigm with emotion as the underlying core—from "command response" to "emotional empathy."
2. **Technical**: Implement emotion-permeating architecture across perception, understanding, decision, and expression layers, with open-source reference.
3. **Practical**: Validate feasibility in real educational scenarios with user feedback and deployment solutions.

Positioning within EDM: While MetaPal draws on HCI and affective computing, its core contribution is **educational data mining**: real-time multimodal affective data as **educational data** to model cross-cultural emotional states in oral dialogue. Unlike prior EDM work on clickstreams or quizzes, MetaPal treats **emotion as a first-class educational data stream** addressing EDM 2026's "Crossing Boundaries" theme.

2. SYSTEM ARCHITECTURE

2.1 Overall Design: Empathetic Architecture

MetaPal implements a four-layer emotion-permeating architecture: perception, understanding, decision, and expression. The perception layer captures multimodal user data for emotion detection. The understanding layer integrates visual, audio, and text features through ViT-based emotion recognition [5], OpenSMILE audio analysis [6], and BERT semantic processing, all centered on emotional comprehension. The decision layer dynamically selects interaction strategies based on emotional states, while the expression layer generates emotionally appropriate responses through digital human feedback (Figure 1).

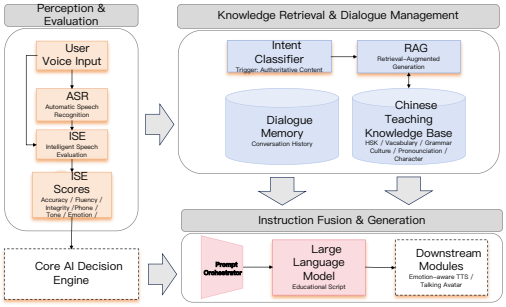


Figure 3: Understanding layer: Three parallel processing paths for multimodal emotional feature extraction and fusion.

2.2 Perception Layer: Emotion Capture

The perception layer captures user video in real-time through camera input, performs picture-in-picture slicing, and transmits the video stream to the backend affective computing module (Figure 2). Following large-scale video-based FER benchmarks [17], our system captures micro-expressions in real-time. Users' expressions, eye contact, and expressions are incorporated into the core processing flow as the starting point of interaction.

2.3 Understanding Layer: Emotion-Semantic Integration

The understanding layer achieves multimodal emotional feature extraction and fusion through three parallel paths: ViT-based visual emotion recognition, OpenSMILE audio emotion analysis, and BERT text semantic analysis (Figure 3). To handle real-world noise, we draw on dual-stage purification methods [14]. Emotion is integrated with semantic parsing—for example, when a user says "I'm fine" with a tense expression, the system interprets it as "tense fine" rather than "calm fine." This approach builds on multimodal affect detection research [3, 10] and affective body expression recognition [9].

2.4 Decision Layer: Emotion-Based Strategy Selection

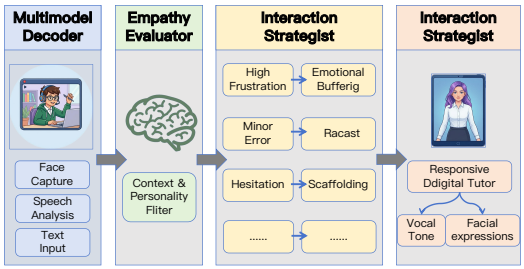


Figure 4: Decision layer: Dynamic strategy selection based on emotional states and learning context.

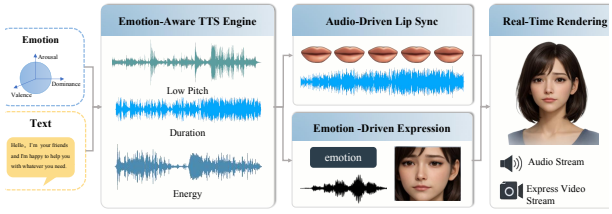


Figure 5: Expression layer: Emotion-driven generation of TTS audio, digital human animations, and visual effects.

The decision engine uses a finite state machine and rule library to dynamically select teaching strategies based on emotional states, cultural background, and learning progress (Figure 4). Emotion forecasting techniques [15] inspire our strategy selection engine. When users are tense, the system slows speech and encourages; when confused, it simplifies and provides scaffolding; when joyful, it increases difficulty and initiates challenges.

Cross-Cultural Parameterization: Hofstede's dimensions [8] are mapped to decision engine weights: power distance, individualism, uncertainty avoidance, and high/low-context communication preferences.

2.5 Expression Layer: Emotion-Driven Output

The expression layer generates TTS audio, digital human animations, and visual effects based on decision results (Figure 5). Audio tone, speed, and rhythm, along with digital human expressions, postures, and interaction methods, are all driven by emotional states to ensure responses are both semantically correct and emotionally appropriate.

3. EVALUATION AND PRELIMINARY FEEDBACK

3.1 Deployment

Figure 6 shows MetaPal deployment in a laboratory environment. Users participate in video calls with the digital human, while the system captures facial expressions in real-time via picture-in-picture for empathetic response.

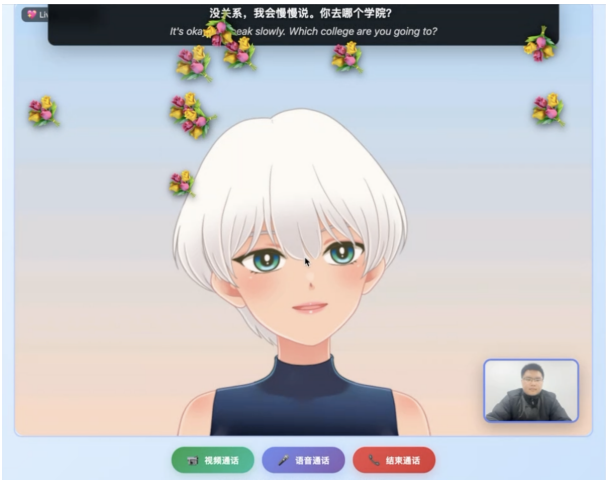


Figure 6: MetaPal deployment: Video calls with real-time emotion capture via picture-in-picture.

3.2 Trial Status

MetaPal is in laboratory prototype verification stage, validated through internal testing and small-scale user trials. The system operated stably with smooth real-time interaction, accurate multimodal emotion recognition, and good audio-visual synchronization.

Participants: International students (N=limited, B1-B2 English) from multiple Asian cultural backgrounds. This diverse sample enables examination of cultural differences in emotional expression [18, 7].

Cross-Cultural Validation: To evaluate the system's cross-cultural adaptation capability, we compared user responses across participants from different cultural backgrounds (Chinese, Korean, Vietnamese). Hofstede's cultural dimension scores [8] for each country were used as reference baselines. Preliminary analysis showed that the system's emotion recognition thresholds appropriately shifted: for example, users from high-context cultures received higher sensitivity weights on facial expression and prosody features, while users from low-context cultures relied more on semantic content analysis. This alignment between system parameters and cultural dimension predictions suggests the feasibility of computationalizing cross-cultural communication theory.

Ethics and Privacy: All trials were conducted with informed consent from participants. As this was a minimal-risk prototyping study, facial expression data were processed in real time and not stored beyond the session; only anonymized behavioral logs were retained for analysis.

Process: Participants completed several weeks of cross-cultural dialogue training with real-time feedback. User experience was collected through questionnaires and interviews, following established methods for evaluating affective educational systems [2].

3.3 Preliminary Results

Small-scale trials showed positive feedback in oral confidence, cultural awareness, and emotional expression. In one case, a user said "Thank you for your help" with a flat expression. The system detected the mismatch and the digital human smiled while explaining gratitude customs in Western culture. User feedback:

> "The system taught me that in Western culture, 'Thank you' needs a smile to be sincere. I wasn't aware of this before."

This demonstrates the paradigm shift—the system "felt" the user's expression and responded emotionally, not just processing the words.

4. DISCUSSION AND CONTRIBUTIONS

4.1 Theoretical Contributions

1. **Paradigm Shift**: MetaPal reconstructs human-computer interaction with emotion as the operating system core across all layers. Command response becomes a subfunction, while emotional empathy drives the interaction, shifting from task-completion to relationship-building.
2. **Emotion-Culture Interaction Model**: Integrates Hofstede's cultural dimension theory [8] into real-time emotion recognition, enabling culturally adaptive expression interpretation.
3. **Multimodal Emotion Fusion**: Verified synergistic effects of visual, audio, and text modalities in language learning.

4.2 Technical Contributions

1. **Emotion-Permeating Architecture**: Implemented four-layer architecture with emotion integrated throughout, providing technical reference for real-time emotional feedback.
2. **Open-source Implementation**: System architecture and core code will be open-sourced on Gitee after acceptance.
3. **Emotional Decision Engine**: Real-time emotion recognition via video calls with dynamic cross-cultural feedback strategies.

4.3 On System Integration as Contribution

We acknowledge that MetaPal builds on established components (ViT, BERT, OpenSMILE, FSM). However, for a **Demo Track** paper, the contribution lies not in proposing new base models, but in demonstrating **how existing technologies can be systematically integrated to solve a real, under-addressed educational problem**. In MetaPal's case, this integration requires: (1) real-time fusion of three modalities under latency constraints, (2) cross-cultural adaptation of emotion recognition thresholds, and (3) driving a digital human's empathetic response. We argue that **system-level innovation**—making working, deployable systems that address genuine educational gaps—is a distinct and valuable form of contribution, complementary to model-level advances, and precisely what EDM's Demo Track is designed to showcase.

4.4 Practical Contributions

1. **Prototype Validation**: Demonstrated technical feasibility in laboratory environment, with real-time multimodal processing under 500ms latency.
2. **Training Solution**: Provided cross-cultural empathetic oral training for international students, addressing emotional expression gaps often overlooked by traditional language tools.
3. **Cross-disciplinary Model**: Demonstrated collaboration across computer science, education, linguistics, and psychology, offering a reusable blueprint for emotion-aware educational systems.
4. **Real-time Fusion Pipeline**: Achieved synchronized processing of video, audio, and text streams with $<$500ms end-to-end latency, enabling natural turn-taking in cross-cultural dialogue.

5. DEMO CONTENT

A demonstration video is submitted as supplementary material, showing live interaction between a user and the MetaPal digital human. The complete system will be open-sourced after acceptance.

6. CONCLUSIONS

MetaPal demonstrates the feasibility of emotion-centered human-computer interaction, providing cross-cultural empathetic oral training for international students. The system bridges computer science and education, with core contributions including:

1. **Paradigm Shift**: Emotion as the operating system core across all layers, redefining interaction from command completion to emotional understanding.
2. **System Completeness**: Complete technical architecture: real-time multimodal perception, emotion-semantic understanding, strategy decision, and empathetic expression generation with full pipeline integration.
3. **Scalability**: Foundation for future large-scale applications with plug-and-play modules for emotion recognition and strategy selection.
4. **Cross-Cultural Adaptability**: Demonstrated real-time parameterization of Hofstede's cultural dimensions for personalized emotion recognition and dialogue strategy adjustment.

Future work will conduct larger-scale experiments to verify educational effectiveness, extend the framework to more cultural contexts (e.g., Latin American, Middle Eastern), and integrate LLM-based open-domain dialogue capabilities.

7. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No. 62406075), the National Key Research and Development Program of China (2023YFC3604802), and the Shanghai Key Technology R&D Program (Grant No. 25511107200).

8. REFERENCES

[1] P. Blikstein and M. Worsley. Multimodal learning analytics and education data mining: using computational technologies to measure complex learning tasks. *Journal of Learning Analytics*, 3(2):220–238, 2016.

[2] N. Bosch, S. D'Mello, R. S. Baker, J. Ocumpaugh, V. Shute, M. Ventura, X. Wei, and W. Zhao. Automatic detection of learning-centered affective states in the wild. In *International Conference on Intelligent Tutoring Systems*, pages 379–389. Springer, 2016.

[3] S. D'Mello and A. Graesser. Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. In *User Modeling 2007*, pages 50–59. Springer, 2007.

[4] S. D'Mello and A. Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, 2012.

[5] A. Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR 2021*, 2021.

[6] F. Eyben et al. Opensmile: The munich versatile and fast open-source audio feature extractor. In *ACM Multimedia*, pages 1459–1462, 2010.

[7] G. Gao, J. Spencer-Rodgers, and L. Ku. Dialectical emotions: How cultural epistemologies influence the experience and regulation of emotion. *Social and Personality Psychology Compass*, 7(11):838–854, 2013.

[8] G. Hofstede. Dimensionalizing cultures: The hofstede model in context. *Online Readings in Psychology and Culture*, 2(1):8, 2011.

[9] A. Kleinsmith and N. Bianchi-Berthouze. Affective body expression perception and recognition: A survey. *IEEE Transactions on Affective Computing*, 4(1):15–33, 2013.

[10] C. L. Lisetti and F. Nasoz. Using noninvasive wearable computers to recognize human emotions from physiological signals. *EURASIP Journal on Applied Signal Processing*, 2004(11):1672–1687, 2004.

[11] M. Pantic and L. J. M. Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proceedings of the IEEE*, 91(9):1370–1390, 2003.

[12] R. Pekrun. The control-value theory of achievement emotions. *Educational Psychology Review*, 18(4):315–341, 2006.

[13] R. W. Picard. Affective computing: Challenges. *International Journal of Social Robotics*, 2(3):193–198, 2010.

[14] H. Wang, X. Mai, Z. Tao, X. Tong, J. Lin, Y. Wang, J. Yu, S. Yan, Z. Zhou, and W. Zhang. D2sp: Dynamic dual-stage purification framework for dual noise mitigation in vision-based affective recognition. In *CVPR*, 2025.

[15] H. Wang, X. Mai, Z. Tao, J. Yu, Z. Zhou, X. Tong, S. Yan, Q. Zhao, S. Gao, Y. Wang, and W. Zhang. Hi-ef: Benchmarking for human-interaction-based emotion forecasting. In *AAAI*, 2026.

[16] Y. Wang, W. Song, W. Tao, D. Yang, A. Liotta, X. Li, S. Gao, Y. Sun, W. Ge, W. Zhang, and W. Zhang. A systematic review on affective computing: Emotion models, databases, and recent advances. *Information Fusion*, 2022. ESI Highly Cited Paper.

[17] Y. Wang, Y. Sun, Y. Huang, Z. Liu, S. Gao, W. Zhang, W. Ge, and W. Zhang. Ferv39k: A large-scale multi-scene dataset for facial expression recognition in videos. In *CVPR*, 2022.

[18] A. E. Woolfolk, K. A. Davis, and S. J. Pape. Teacher talk to diverse learners: A study of teacher-student communication in regular classrooms. *Teaching and Teacher Education*, 22(5):536–546, 2006.

[19] Y. Zhang and A. Goodwin. Unpacking the complexity of international chinese doctoral students' learning and adjustment experiences. *Higher Education*, 62(5):595–613, 2011.

Mining the Sequential Dependencies in Learning Events with a Self-Attentive Hawkes Process

Sijing Yu
South China Normal University
yusijing@m.scnu.edu.cn

Tianwei Peng
South China Normal University
2024020894@m.scnu.edu.cn

Sijie Zhang
South China Normal University
2024020913@m.scnu.edu.cn

Zengcan Xue
South China Normal University
xuezc@m.scnu.edu.cn

Su Mu
South China Normal University
musu@m.scnu.edu.cn

Yingbin Zhang
South China Normal University
zyingbin@m.scnu.edu.cn

ABSTRACT

Modern educational settings routinely generate massive amounts of learning event sequences, which often exhibit complicated temporal and sequential dependencies among events. Traditional sequence dependency analysis methods, such as sequential pattern mining, requires researchers to manually configure pattern parameters, such like minimum support, positional gap, and time interval, which introduces subjectivity into the analysis process. Meanwhile, deep learning models combined with temporal point processes, particularly Hawkes processes with attention mechanisms, have shown promise in capturing sequential dependencies and predicting future events, yet existing models remain limited in the interpretability of the dependency features they identify. To address these limitations, this study proposed a method based on self-attentive Hawkes processes to identify the sequential dependencies in learning event sequences without requiring manual configuration of pattern parameters. We applied the proposed method to two learning event datasets and compared the extracted patterns with those mined by two baseline algorithms. The results revealed a relatively small degree of pattern overlap between the proposed method and baselines. For overlapping patterns, the proposed method identified temporal characteristics differed significantly from that of baselines, which were manually preset.

Keywords

Learning event dependency, sequential pattern mining, temporal pattern mining

1. INTRODUCTION

Learning process data are typically represented as sequences of learning events that partially reflect learners' cognitive processes and behavioral patterns. The temporal sequential relationships among learning events, referred to as sequential dependencies, hold significant pedagogical value [1]. They can support a wide range of educational applications, such as learning performance prediction and learning environment design. For example, sequential dependencies in MOOC learning events can serve as features to predict students' dropout risk [2]; that in video viewing behaviors can help reveal the engagement with instructional videos [3]; and that in the resource usage behaviors among a certain group of learners can be transformed into learning paths recommended to other learners in the same group [4].

A range of methods has been used to analyze sequence dependencies in learning events, include sequential pattern mining (SPM), lag sequential analysis, process mining, and ordered network analysis [5]. However, except for SPM, these methods are generally limited to characterizing pairwise dependencies between events and have difficulty capturing the overall sequential dependencies among multiple learning events. SPM can mine sequence patterns of various lengths [6] and has been widely used in descriptive, relational, and predictive studies [7, 8, 9], such as mining learning behaviors, generating features for prediction and classification models, and filtering learning resources to build recommendation systems. However, it relies on researchers' manual setup of pattern parameters, such as the minimum support (the minimum sequences containing a pattern) as well as positional gaps and time intervals among adjacent pattern events. Because these preset parameters inevitably involve subjective judgment, the manual setup creates a gray area in the analytical process that may introduce bias into the results [10], thereby influencing researchers' understanding of learning patterns.

Recent advances in deep learning have combined deep neural networks with temporal point process models to characterize sequential dependencies while taking into account factors such as event position and time interval, thereby enabling future event prediction [11]. Among these models, Hawkes-process-based approaches are particularly suitable because they capture how prior events influence subsequent events over time. Accordingly, Hawkes processes have been widely applied to event prediction tasks, such as predicting procrastination in online learning [12]. More recently, attention-based extensions such as the self-attentive Hawkes process (SAHP) [13] and the Transformer Hawkes Process [14] have further improved the modeling of event sequences and outperformed recurrent neural network-based Hawkes process models in future event prediction. However, these models are primarily designed for event prediction and extrinsic target classification, which limits their usefulness for extracting interpretable sequential dependency features.

Sijing Yu, Tianwei Peng, Sijie Zhang, Zengcan Xue, Su Mu, and Yingbin Zhang. Mining the Sequential Dependencies in Learning Events with a Self-Attentive Hawkes Process. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 710–714. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039697

To address these limitations, this study proposes a novel method that combines the self-attentive Hawkes process with the task of mining learning event sequential dependencies. The proposed metho aims to reduce reliance on manually preset pattern parameters while extract interpretable patterns of learning events with relatively strong sequential dependency. To evaluate its effectiveness in relation to traditional sequence dependency analysis, this study compares the patterns and time intervals identified by the proposed method with those obtained from baseline SPM algorithms. Specifically, this study examines the following research questions:

- RQ1: To what extent do the pattern sets mined by the proposed method overlap with those mined by baseline SPM algorithms?
- RQ2: For overlapping patterns, how do the time intervals automatically identified by the proposed method compare with those preset in baseline SPM algorithms?

2. METHODOLOGY

We refer to the proposed method as the leSPM-SAHP algorithm. It builds on the SAHP architecture [13] because SAHP showed high performance in modeling event dependencies and event prediction while incorporating temporal order and time intervals, making it suitable for learning-event sequence analysis. Specifically, in self-supervised learning where the task is the next event prediction, the model uses the attention output O, derived from the query (Q), key (K), and value (V) matrices, to quantify the influence of historical events on subsequent events.

$$O = AV \quad A = \mathrm{Softmax}\left(\frac{QK^T}{\sqrt{M_K}}\right)$$

$$Q = XW^Q, \quad K = XW^K, \quad V = XW^V$$

The value a_{ij} in the i-th row and j-th column of the attention weight matrix A represents the strength of sequential dependencies of event (k_i, t_i) on historical event (k_j, t_j) [14]. k_i is the event type of the i-th event, while t_i is the time elapsed since the first event. To prevent overly distant events from contributing noise, we introduced a history-length regularization parameter ℓ, which restricts attention to a bounded temporal context by forcing attention weights beyond ℓ positions to approach zero. Correspondingly, if $i - j > \ell$, a_{ij} approaches 0.

Based on the resulting attention weights, leSPM-SAHP first identifies candidate pairwise sequential dependencies between events. For each event pair, a Wilcoxon signed-rank test is used to determine whether its attention weights are significantly greater than the baseline threshold ($\frac{1}{\ell+1}$), so that only statistically significantly dependencies are retained.

Because pairwise dependencies correspond only to patterns of two events, we then extend them into longer patterns through a layer-linking pattern expansion procedure. This procedure is motivated by the ability of higher-order attention layers to encode dependencies involving multiple events. By sequentially linking sequence patterns across adjacent layers, short dependencies are progressively expanded into longer event sequence patterns.

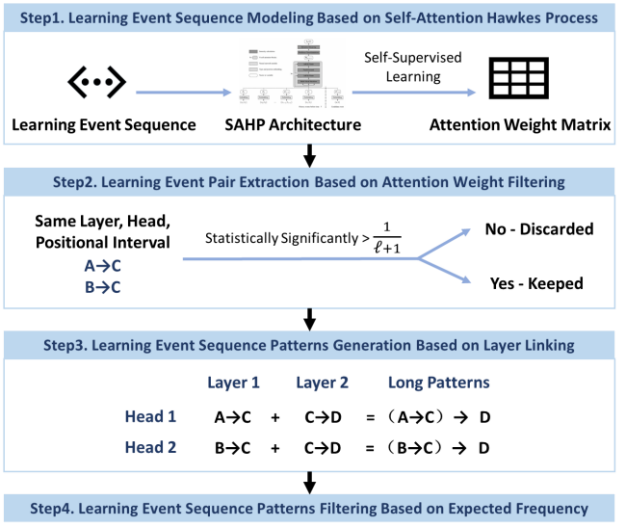


Figure 1. The workflow of leSPM-SAHP

Finally, to reduce spurious or practically uninformative outputs, all extracted patterns are filtered using an expected-frequency criterion, which were based on the product of the simple frequencies of pattern events. Patterns with occurrence frequencies lower than their expected frequency in the event sequences are removed, and the remaining patterns constitute the final pattern set. This design allows leSPM-SAHP to reduce reliance on manually preset pattern parameters in traditional SPM while preserving interpretable pattern and time-interval information for comparison with baseline SPM algorithms.

3. EXPERIMENT

3.1 Datasets

We evaluated leSPM-SAHP on two contrasting learning-event datasets, a large-scale *XuetangX MOOC*[1] dataset and an event-rich middle-school science game dataset, *Wake: Tales from the Aqualab*[2], to examine whether the method performs consistently across different learning contexts. Table 1 displays basic statistics of the two datasets. To reduce excessively long inactive gaps, sequences in both datasets were split whenever the interval between adjacent events exceeded 15 minutes.

Table 1. Basic Information of Datasets

Dataset	Student Type	Number of Students	Mean Sequence Length (*SD*)	Number of Event Type
MOOC	Multiple Types	6343	157.65 (*568.51*)	21
Aqualab	Middle School Students	590	817.01 (*1492.10*)	58

[1] http://moocdata.cn/data/user-activity

[2] https://opengamedata.fielddaylab.wisc.edu/gamedata.php?game=AQUALAB

3.2 Baselines

We compared leSPM-SAHP with two recent temporal pattern mining baselines, both of which support time-interval sequential patterns extraction:

- **OER-Miner (2025):** This method employed episode connection strategies and position index support calculation to efficiently mines sequential patterns with time interval constraints [15].
- **TIRPClo (2023):** It achieves complete temporal interval pattern mining through entity projection based on endpoint sequence transformation and memory indexing [16].

3.3 Analysis

For leSPM-SAHP, model hyperparameters were selected using 5-fold cross-validation, and attention-weight filtering used a significance level of 0.05. For the baselines, pattern sets were extracted under multiple combinations of minimum support and maximum time-interval settings to provide a systematic comparison across parameter conditions (Table 2). A pattern was returned by the baseline only when the proportion of sequences with at least one valid pattern instance was no less than the minimum support, and a pattern instance was valid when the time interval between adjacent events was not greater than the maximum time interval.

Table 2. Settings of Pattern Parameters. 25th TI denotes the time interval at the 25th percentile rank among all event time intervals (the time interval between the first event of a sequence and all other events in the sequence)

Dataset	Parameter	Level
MOOC	Minimum Support	0.15, 0.20, 0.25, 0.30
	Maximum Time Interval	25th TI, 50th TI, 75th TI
Aqualab	Minimum Support	0.55, 0.60, 0.65, 0.70
	Maximum Time Interval	25th TI, 50th TI, 75th TI

Under each combination of pattern parameters, we compared the filtered pattern set P produced by leSPM-SAHP with the baseline pattern set P_{base} (see Figure 2). Specifically, a denotes the count of patterns identified only by leSPM-SAHP, b denotes the count of patterns identified only by the baseline method, and c denotes the count of overlapping patterns. Before filtering, the corresponding counts were recorded as a', b', c'. For the overlapping patterns (c), we further compared the differences in maximum time interval identified by leSPM-SAHP with those preset in the baseline SPM algorithms.

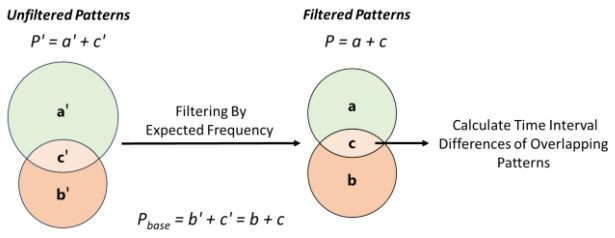


Figure 2. Metrics for comparing leSPM-SAHP with baselines. P' was the unfiltered pattern set extracted by leSPM-SAHP, while P was the filtered pattern set. P_{base} was the pattern set extracted by the baselines

We focused on three metrics of pattern overlapping: $\frac{c}{a+c}$, $\frac{b}{b+c}$, and $\frac{b\prime}{b\prime+c\prime}$. If $\frac{c}{a+c}$ is small and the interval difference is large, it means that the sequence patterns extracted by leSPM-SAHP cannot be covered by existing algorithms. If $\frac{b}{b+c}$ is large and $\frac{b\prime}{b\prime+c\prime}$ is small, it indicates that many patterns extracted by baselines existed in the unfiltered pattern set by leSPM-SAHP but not in the filtered pattern set, which means baselines may extract many patterns without relatively strong event dependencies as their frequencies lower than the expected frequency.

4. RESULTS & DISCUSSION

In the MOOC dataset, the number of patterns mined by OER-Miner ranged from 23 to 2385, that by TIRPClo ranged from 16 to 794, while that by leSPM-SAHP was 630. In the Aqualab dataset, the number of patterns mined by OER-Miner ranged from 765 to 7305, that by TIRPClo ranged from 603 to 6728, while that by leSPM-SAHP was 3653.

As shown in Figure 3, in both datasets, $\frac{c}{a+c}$ consistently remained at a low level, with mean values of 3.46% and 0.76% for MOOC and Aqualab datasets, respectively. This indicates that only a small proportion of the sequential patterns in the filtered set extracted by leSPM-SAHP overlapped with those extracted by the baseline algorithms. In other words, most patterns identified by leSPM-SAHP were not covered by the baseline under the tested parameter settings.

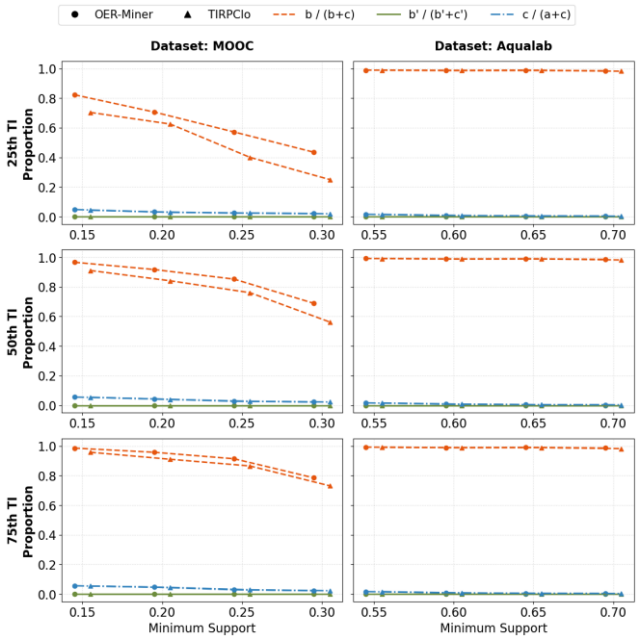


Figure 3. The three metrics under each parameter combination

More importantly, across all combinations in both datasets, $\frac{b\prime}{b\prime+c\prime}$ was 0. This indicates that the patterns extracted by the baseline algorithms were also present in the unfiltered output of leSPM-SAHP. However, in the Aqualab dataset, $\frac{b}{b+c}$ remained at a high level, fluctuating between 98.14% and 99.19%, and in the MOOC dataset, it was also large under many parameter combinations, although smaller in some conditions. Taken together, these results suggest that the divergence between leSPM-SAHP and the baseline methods arose primarily from the filtering stage rather than from a failure of leSPM-SAHP to initially identify the patterns extracted

by the baselines. In other words, many patterns extracted by the baseline algorithms existed in the unfiltered pattern set of leSPM-SAHP but were not retained after filtering. This suggests that frequency-based SPM algorithms may preserve many patterns that satisfy support-based criteria but are not retained under the dependency- and expected-frequency-based filtering procedure of leSPM-SAHP.

For overlapping patterns, we next examined whether the maximum time intervals identified by leSPM-SAHP differed significantly from those preset in the baseline SPM algorithms. For each fixed combination of dataset, maximum time interval, algorithm, and minimum support, we tested whether the time interval differences significantly differed from zero using the Wilcoxon signed-rank test. To control for error due to multiple comparisons, p-values were adjusted using the Benjamini-Yekutieli correction.

Figure 4 displays the heatmap of mean time interval differences across datasets, maximum time intervals, baseline algorithms, and minimal support. In the MOOC dataset, the time interval differences were uniformly positive across all conditions, and all 24 cells were statistically significant. This indicates that, for overlapping patterns, the maximum time intervals identified by leSPM-SAHP were consistently larger than those preset in the baseline methods.

In Aqualab, time interval differences were also consistently positive and significant under the 25th and 50th TI thresholds. By contrast, under the 75th TI threshold, all Aqualab cells turned negative, and five of the eight cells remained statistically significant after correction. Thus, while the MOOC results showed a stable pattern of larger time intervals under leSPM-SAHP, the Aqualab results suggest that the direction and magnitude of time interval differences depended on the maximum time interval condition. This set of findings addresses RQ2 by showing that, even for overlapping patterns, leSPM-SAHP and the baseline SPM algorithms often assigned different temporal spans to the same pattern.

Taken together, the results indicate that the overlap between leSPM-SAHP and the baseline SPM algorithms was limited in both datasets, and that the temporal characteristics of overlapping patterns often differed significantly. A plausible explanation is that existing SPM algorithms depend on manually preset pattern parameters, such as minimum support and maximum time interval, whereas leSPM-SAHP does not rely on such preset thresholds in the same way and instead derives dependency information from the data through the SAHP model structure and filtering procedure. From this perspective, leSPM-SAHP may capture sequential patterns and time intervals among learning events that are not easily recovered when pattern parameters must be specified in advance. More broadly, these findings suggest that sequence dependency analysis in educational data mining may benefit from approaches that infer temporal and frequency constraints adaptively from data rather than relying on researcher-defined parameter settings.

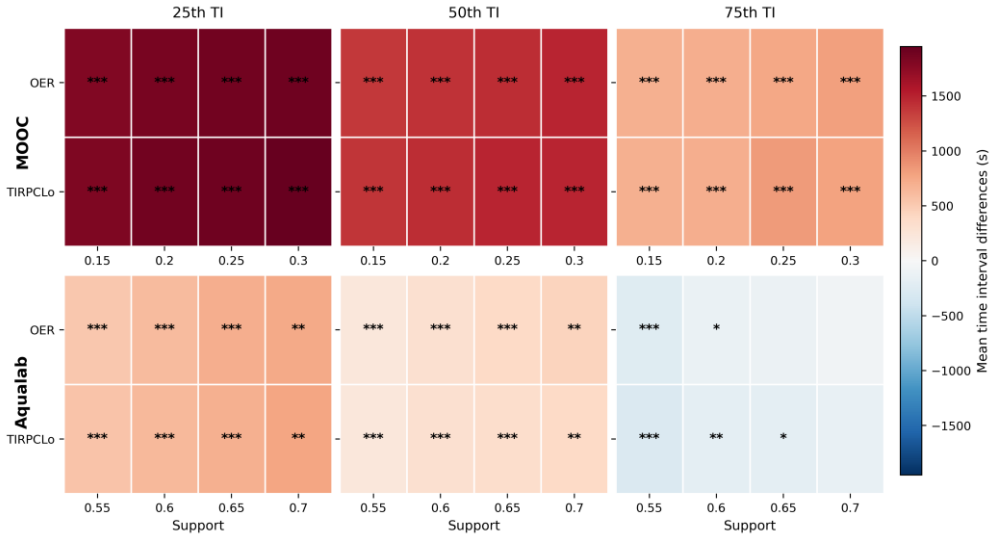


Figure 4. Comparison of time interval differences for overlapping patterns under each condition. Asterisks denote Benjamini-Yekutieli-adjusted significance against 0 (* $p < .05$, ** $p < .01$, *** $p < .001$)

5. LIMITATIONS AND FUTURE WORK

At the same time, these findings should be interpreted with caution. Low overlap does not by itself prove that leSPM-SAHP identifies more meaningful patterns than the baseline methods; rather, it shows that the proposed method yields substantially different outputs under the present comparison framework. Similarly, the fact that many baseline patterns were filtered out by leSPM-SAHP suggests that those patterns were not retained under dependency-based screening, but it does not by itself establish that they are substantively unimportant. Additional validation is therefore needed to examine whether the patterns preferentially retained by leSPM-SAHP are indeed more informative, interpretable, or pedagogically useful in real educational settings.

For overlapping patterns, this study only compared their time interval differences. Future research should further examine positional interval differences to provide a more complete comparison of pattern features across methods. Additionally, this study utilized only two datasets and two baseline algorithms. Subsequent research should extend the comparisons to data from more diverse learning contexts and a broader range of algorithms to investigate whether the degree of pattern overlap depends on dataset characteristics or algorithmic assumptions.

6. ACKNOWLEDGMENTS

This study was funded by National Natural Science Foundation of China (No.62407014).

7. REFERENCES

[1] Molenaar, I. and Wise, A. F. 2022. Temporal aspects of learning analytics: Grounding analyses in concepts of time. In *The Handbook of Learning Analytics, 2nd ed*. 66-76. DOI= https://doi.org/10.18608/hla22.006

[2] Deeva, G., De Smedt, J., and De Weerdt, J. 2022. Educational sequence mining for dropout prediction in MOOCs: Model building, evaluation, and benchmarking. *IEEE Transactions on Learning Technologies.* 15, 6 (2022), 720-735. DOI= http://doi.acm.org/10.1109/TLT.2022.3215598.

[3] Wong, J., Khalil, M., Baars, M., de Koning, B. B., and Paas, F. 2019. Exploring sequences of learner activities in relation to self-regulated learning in a massive open online course. *Computers & Education.* 140 (2019), 103595. DOI= http://doi.acm.org/10.1016/j.compedu.2019.103595.

[4] Wan, S. and Niu, Z. 2019. A hybrid e-learning recommendation approach based on learners' influence propagation. In *IEEE Transactions on Knowledge and Data Engineering*. 32, 5 (2019), 827-840.

[5] Lämsä, J., Hämäläinen, R., Koskinen, P., Viiri, J., and Lampi, E. 2021. What do we do when we analyse the temporal aspects of computer-supported collaborative learning? A systematic literature review. *Educational Research Review*. 33 (2021), 100387.

[6] Agrawal, R. and Srikant, R. 1995. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*. 3-14. DOI= http://doi.acm.org/10.1109/icde.1995.380415.

[7] Kang, J., Liu, M., and Qu, W. 2017. Using gameplay data to examine learning behavior patterns in a serious game. *Computers in Human Behavior*. 72 (2017), 757-770. DOI= http://doi.acm.org/10.1016/j.chb.2016.09.062.

[8] Kinnebrew, J. S., Segedy, J. R., and Biswas, G. 2015. Integrating model-driven and data-driven techniques for analyzing learning behaviors in open-ended learning environments. *IEEE Transactions on Learning Technologies*. 10, 2 (2015), 140-153.

[9] Slim, A., Heileman, G. L., Al-Doroubi, W., and Abdallah, C. T. 2016. The impact of course enrollment sequences on student success. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*. 59-65. DOI= http://doi.acm.org/10.1109/AINA.2016.140.

[10] Zimmermann, A. 2020. Method evaluation, parameterization, and result validation in unsupervised data mining: A critical survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. 10, 2 (2020), e1330. DOI= http://doi.acm.org/10.1002/widm.1330.

[11] Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., and Sun, L. 2022. Transformers in time series: A survey. *arXiv preprint arXiv:2202*. 07125 (2022). DOI= https://doi.org/10.48550/arXiv.2202.07125.

[12] Yao, M., Zhao, S., Sahebi, S., and Behnagh, R. F. 2021. Relaxed clustered Hawkes process for student procrastination modeling in MOOCs. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 35, 5 (2021), 4599-4607. DOI= https://doi.org/10.1609/aaai.v35i5.16589.

[13] Zhang, Q., Lipani, A., Kirnap, O., and Yilmaz, E. 2020. Self-attentive Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*. 119 (2020), 11183-11193.

[14] Zuo, S., Jiang, H., Li, Z., Zhao, T., and Zha, H. 2020. Transformer Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*. 119 (2020), 11692-11702.

[15] Wu, Y., Dong, Z., Liu, J., Li, Y., Liu, C., Wen, L., and Wu, X. 2025. OER-Miner: One-off episode rule mining for process event logs. *IEEE Transactions on Emerging Topics in Computing*. 13, 4 (2025), 1497-1509.

[16] Harel, O. and Moskovitch, R. 2023. TIRPClo: Efficient and complete mining of time intervals-related patterns. *Data Mining & Knowledge Discovery*. 37, 5 (2023). DOI= https://doi.org/10.1007/s10618-023-00944-6.

Multi-Feature Knowledge Tracing with Time-Series Interpretability for Mathematics Teacher Micro-Credentials

Hai Li
University of Florida
li.ha@ufl.edu

Chenglu Li
University of Utah
chenglu.li@utah.edu

ABSTRACT

Teacher professional development (PD) is critical for instructional improvement, and online micro-credentials provide flexible, work-embedded learning opportunities. Knowledge tracing (KT) can support personalization by modeling knowledge state evolution, but systematic application to adult professional learners remains limited, particularly when interaction logs are multi-dimensional and models must balance predictive performance with interpretability under small-sample constraints. Focusing on mathematics teacher micro-credentials, we propose a multi-feature deep KT framework with built-in temporal interpretability. We organize heterogeneous log-derived features into a progressive six-layer hierarchy and apply mutual information-based selection, which is well-suited for small-sample, feature-rich environments. Our residual-injection Time Interpretation Module (TIM) augments LSTM-based Deep Knowledge Tracing with lightweight temporal encoding and causal self-attention, providing intrinsic interpretability through temporal feature importance analysis. Evaluated on a dataset comprising 154 teachers and 22,988 interactions, results demonstrate that compact core-feature configurations achieve strong predictive performance, while indiscriminate feature addition degrades accuracy due to dimensionality effects. TIM improves both predictive performance and training stability while revealing interpretable temporal dynamics distinctive to adult professional learners. The framework offers systematic guidance for KT-based support in resource-constrained professional learning contexts.

Keywords

Knowledge Tracing, Teacher Professional Development, Time-Series Interpretability, Multi-Feature Integration

1. INTRODUCTION

Teacher professional development (PD) is widely viewed as a high-leverage route to improving instruction and student outcomes [29, 6]. Contemporary PD increasingly emphasizes personalization, work-embedded learning, and competency-oriented certification through *micro-credentials* that enable modular, flexible asynchronous participation [7, 22]. A central goal is strengthening *Pedagogical Content Knowledge (PCK)* in authentic instructional contexts [3].

Knowledge Tracing (KT) offers data-driven modeling of knowledge state evolution and can underpin adaptive support systems [5, 25]. The field has progressed from Bayesian Knowledge Tracing (BKT) and Performance Factors Analysis (PFA) [24] to LSTM-based Deep Knowledge Tracing (DKT) [25] and attention-augmented architectures [30, 23, 21]. Recent advances have further enhanced KT through multi-time series feature integration [13] and option-level tracing for mathematics assessment [15]. LSTM-based models combined with attention mechanisms represent well-established approaches for capturing temporal dependencies in learner sequences [1]. However, systematic application to adult professional learners remains limited, particularly in asynchronous micro-credential settings where learning rhythms, communicative patterns, and constraints differ substantially from traditional educational environments [19].

Teachers, as adult learners, exhibit autonomy, experience-based reasoning, and problem-centered learning consistent with *andragogy* [9]. Their professional learning involves complex instructional judgments and deep PCK integration [20]. Using data from a mathematics teacher micro-credential program, we observe distinctive properties: highly concentrated sessions, notably high item difficulty (overall accuracy 33.37%), and substantial variation in skill mastery across 12 teaching dimensions. These characteristics suggest that systematic feature engineering and temporal modeling are particularly important in this context.

Methodologically, teacher PD platforms generate multi-dimensional logs encompassing behavioral, temporal, and performance signals, yet feature processing in existing KT work is often ad hoc [11]. Small-sample, feature-rich settings amplify overfitting risks and dimensionality issues. Additionally, many deep KT models lack intrinsic interpretability for temporal dynamics [1], limiting their utility in professional learning contexts where stakeholders need to understand prediction mechanisms.

We address two research questions that systematically tackle these challenges:

Hai Li, and Chenglu Li. Multi-Feature Knowledge Tracing with Time-Series Interpretability for Mathematics Teacher Micro-Credentials. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 715–720. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039794

RQ1: How do different log-derived features, organized via a progressive hierarchy and selected through mutual information, affect KT predictive performance in a small-sample professional learning setting?

RQ2: How can a lightweight, residual-injected temporal module enhance LSTM-based DKT performance while providing interpretable temporal dynamics?

2. RELATED WORK

2.1 Knowledge Tracing and Temporal Modeling

KT has evolved from BKT's Hidden Markov Models [5] and PFA's logistic regression with engineered predictors [24] to DKT's recurrent sequence modeling [25]. LSTM-based architectures combined with attention mechanisms are well-established for capturing temporal dynamics in learner modeling [30, 23, 21, 1]. Models such as Dynamic Key-Value Memory Networks (DKVMN) have introduced structured memory mechanisms that provide specific levels of interpretability [21]. Temporal modeling and forgetting effects remain active areas of development [4].

Despite these advances, most KT evidence derives from student learning contexts. Teachers as adult learners follow andragogy principles including self-direction and experience integration, implying KT models must accommodate higher-order cognitive processes distinct from student content learning. Micro-credentials further amplify these differences through competency-oriented certification and asynchronous pacing [22].

2.2 Multi-Feature Integration and Interpretability

As learning platforms expand, KT research increasingly integrates process data such as response time, hints, and navigation paths to improve prediction [1, 17, 2]. Parallel developments in mathematics education have demonstrated the value of multimodal features for automated feedback [12] and collaborative filtering approaches for automatic scoring [16]. However, several methodological challenges persist: feature construction for heterogeneous logs remains largely unstructured, with limited systematic use of nonlinear dependency metrics such as mutual information [10]; temporal features are often incorporated via simple concatenation, which struggles to capture complex sequential effects [4]; and model interpretability typically relies on post-hoc tools such as SHAP or LIME [18, 27], making it difficult to isolate the role of specific temporal features [28].

3. METHOD

We model teacher learning as sequence $\mathcal{S} = \{(s_t, q_t, r_t)\}_{t=1}^{T}$, where s_t denotes skill ID ($K = 12$), q_t question ID ($Q = 127$), and $r_t \in \{0, 1\}$ binary response. The prediction goal is $p_t = P(r_t = 1 \mid \mathcal{S}_{<t}, \boldsymbol{f}_t)$.

3.1 Platform Context and Dataset Construction

The data originates from an asynchronous mathematics teacher micro-credential platform deployed in the southeastern United States (September 2024–February 2025). The platform supports practicing educators working toward competency-based certification through modular courses. Assessment items consist of complex pedagogical judgment tasks where teachers analyze authentic classroom scenarios and identify specific instructional strategies such as revoicing, fostering reasoning, and leveraging student errors. These higher-order tasks target PCK integration, which accounts for the dataset's notably low overall accuracy and substantial cross-skill variation.

We align four heterogeneous data sources at teacher and course levels: assessment response records, content learning progress, platform interaction logs, and course metadata. The learning environment follows a standard dashboard-content-assessment workflow where teachers access materials, attempt items, and receive feedback within an integrated web platform. All data were de-identified prior to analysis under institutional data agreements and ethics review protocols. Preprocessing follows three rigorous principles: features with $> 70\%$ missingness are dropped to limit imputation bias; response-time outliers (< 1s or > 3600s) are removed; and strict temporal ordering is enforced via cumulative `time_since_last_attempt` to prevent future leakage.

Dataset Summary: The final dataset comprises **22,988 valid interaction records from 154 teachers**, covering **127 items across 12 skills**, with overall accuracy 33.37% and average sequence length 149.3 ± 30.2 items per teacher. Temporal characteristics reflect concentrated adult learning behavior: **99.84% of consecutive responses occur within sessions** (< 6 minutes apart), while long intervals (> 1 week) account for only 0.14% of transitions.

3.2 Training Protocol and Statistical Rigor

Given the relatively small teacher cohort (**154 teachers**) paired with high interaction volume (**22,988 records**), teacher-level generalization is the primary evaluation concern. We apply teacher-level stratified splits with complete isolation: **Training (107 teachers, 16,065 records, 69.9%), Validation (23 teachers, 3,538 records, 15.4%), Test (24 teachers, 3,385 records, 14.7%)**. This ensures evaluation reflects true generalization to unseen teachers.

Optimization uses Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$, lr=0.001) with class weights (pos_weight=1.99) to address class imbalance, label smoothing ($\epsilon = 0.02$), gradient clipping (max_norm=0.8), and early stopping (patience=5). To ensure robust statistical reporting, **each configuration runs with 5 independent random seeds (42–46)**, with results reported as empirical mean $\pm$ standard deviation. We use variance reduction across seeds as the primary stability indicator, which provides reliable evidence of model improvements under deep learning initialization variability.

3.3 Progressive Feature Engineering (RQ1)

We organize features into a six-layer hierarchy from core to contextual, enabling controlled evaluation of each feature group's contribution under small-sample constraints (Table 1). This design systematically addresses the practical question of which feature subsets offer optimal performance-cost tradeoffs for deployment.

Mutual information $I(f; r) = \sum_{f,r} p(f, r) \log \frac{p(f,r)}{p(f)p(r)}$ is com-

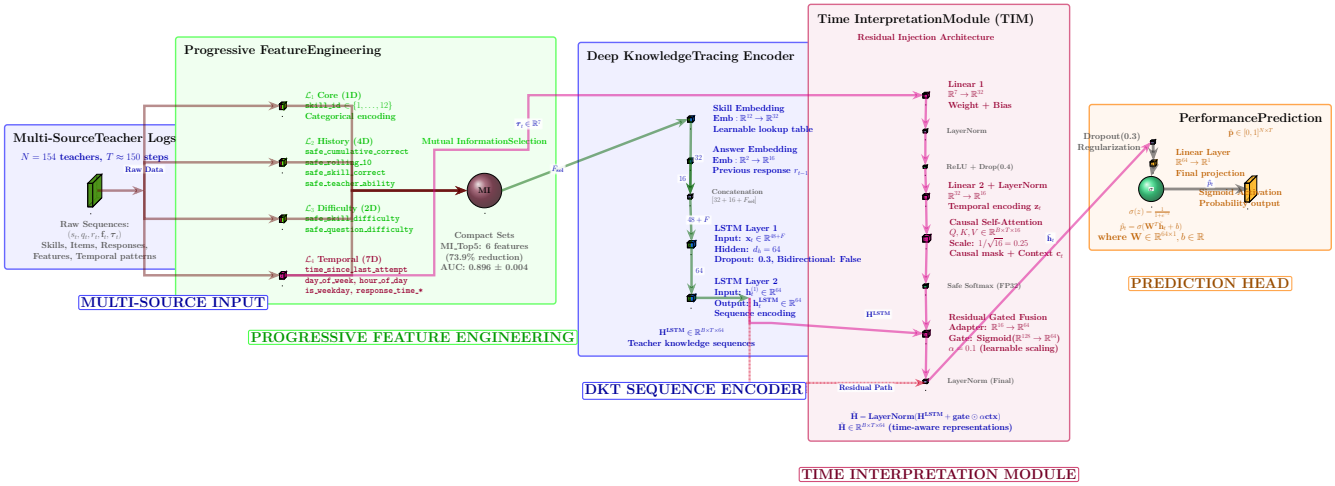


Figure 1: Overall architecture of the multi-feature DKT framework with residual-injection Time Interpretation Module (TIM).

Table 1: Progressive Feature Hierarchy System

Layer	Category	Variables & Description	No.
$\mathcal{L}_1$ *Core (L1_Baseline)*			
	Skill	`skill_id`: 12 categories, 32-dim embeddings	1
$\mathcal{L}_2$ *Performance History*			
	History	`cumulative_accuracy`, `rolling_accuracy_10`, `skill_accuracy`, `teacher_ability`	4
$\mathcal{L}_3$ *Difficulty Parameters*			
	Difficulty	`item_difficulty`, `skill_difficulty`	2
$\mathcal{L}_4$ *Temporal Features*			
	Temporal	`time_since_last_attempt`, `hour_of_day`, `day_of_week`, `is_weekday`, `response_time_seconds`, `response_time_log`, `response_time_zscore`	7
$\mathcal{L}_5$ *Behavioral Features*			
	Behavioral	`total_learning_time`, `total_visits`, `avg_time_per_visit`, `questions_per_hour`, `total_attempts_by_user`	5
MI Selection (MI_TopK)			
	Selection	Top-K by mutual information (K=5,10,15)	6,11,16

Full configuration: L1(1)+L2(4)+L3(2)+L4(7)+L5(5)=19 total features.

puted via k-nearest-neighbor estimation [10], ranking 18 non-core features to construct MI_Top5/10/15 configurations. This nonlinear dependency metric is particularly effective for capturing complex feature-response relationships in small-sample, feature-rich settings where linear correlation may underestimate feature relevance.

3.4 LSTM-based DKT with Time Interpretation Module (RQ2)

The base DKT concatenates skill embedding $\boldsymbol{e}_s(s_t) \in \mathbb{R}^{32}$, answer embedding $\boldsymbol{e}_a(r_{t-1}) \in \mathbb{R}^{16}$, and auxiliary features $\boldsymbol{f}_t$ into input $\boldsymbol{x}_t$. A 2-layer LSTM (hidden size 64, dropout 0.3) encodes $\boldsymbol{h}_t = \text{LSTM}(\boldsymbol{x}_t, \boldsymbol{h}_{t-1})$, with prediction $p_t = \sigma(\boldsymbol{w}^\top \boldsymbol{h}_t + b)$.

Table 2: Model Performance Analysis (mean ± std, 5 runs, sorted by AUC)

Configuration	Feat.	AUC	F1	Acc.
BKT (Baseline)	1	0.445 ± 0.000	0.122 ± 0.000	66.5%
PFA (Full)	19	0.769 ± 0.000	0.508 ± 0.000	73.4%
DKT (Full)	19	0.777 ± 0.008	0.610 ± 0.004	67.5%
DKT (MI_Top15)	16	0.894 ± 0.011	0.749 ± 0.016	83.0%
DKT (MI_Top5)	6	0.894 ± 0.006	0.750 ± 0.012	82.9%
DKT (L1_Baseline)	1	$\mathbf{0.896 \pm 0.004}$	$\mathbf{0.752 \pm 0.003}$	**83.0%**

Time Interpretation Module (TIM) extends the base DKT with temporal interpretability via residual injection, comprising four components: (1) *Temporal encoder*: a 2-layer MLP with LayerNorm maps raw temporal features $\boldsymbol{\tau}_t \in \mathbb{R}^7$ to compact encoding $\boldsymbol{z}_t \in \mathbb{R}^{16}$; (2) *Causal self-attention* [28] with scaling factor $\sqrt{16}$ captures temporal dependencies across sequence steps; (3) *Gradient-based feature importance estimation* provides step-level interpretability of temporal feature contributions; (4) *Residual gated fusion*: $\tilde{\boldsymbol{c}}_t = \alpha \cdot \boldsymbol{c}_t$, $\boldsymbol{g}_t = \sigma(\boldsymbol{W}_g[\boldsymbol{h}_t; \tilde{\boldsymbol{c}}_t] + \boldsymbol{b}_g)$, $\boldsymbol{h}'_t = \text{LayerNorm}(\boldsymbol{h}_t + \boldsymbol{g}_t \odot \tilde{\boldsymbol{c}}_t)$. The gating mechanism adaptively controls temporal signal contribution, providing stable integration while preserving baseline DKT representational capacity.

4. RESULTS

4.1 RQ1: Progressive Feature Integration and Performance

Mutual information ranking reveals the strongest individual associations with correctness: `skill_accuracy` (MI=0.064), `item_difficulty` (0.030), and `cumulative_accuracy` (0.028). Temporal features show lower individual MI values, indicating their contribution depends on sequential relational modeling.

Three systematic patterns emerge from Table 2. First, *core-feature sufficiency*: L1_Baseline achieves optimal AUC

Table 3: TIM Ablation Study (based on DKT MI_Top15, 5 runs)

Configuration	AUC	F1	Std Change
TIM w/o attention	0.821 ± 0.021	0.657 ± 0.027	+91%
DKT (baseline)	0.894 ± 0.011	0.749 ± 0.016	ref
TIM w/o importance	0.921 ± 0.009	0.801 ± 0.022	−18%
TIM (complete)	$\mathbf{0.925 \pm 0.002}$	$\mathbf{0.805 \pm 0.005}$	**−82%**

(0.896), reflecting the strong predictive signal embedded in the inherent difficulty hierarchy across the 12 PCK skills, which LSTM captures efficiently from skill identity alone. Second, *dimensionality effects*: DKT (Full) with all 19 features drops to AUC=0.777, comparable to PFA, demonstrating that adding noisy features without selection introduces overfitting in this small-sample setting. Third, *selection effectiveness*: MI_Top5 (6 features) nearly matches the skill-only ceiling while outperforming the full feature set by 15.1% in AUC, confirming that principled mutual-information selection recovers predictive value lost to dimensionality.

4.2 RQ2: TIM Architecture and Temporal Interpretability

Since L1_Baseline contains only skill identity and cannot evaluate temporal modeling, we use DKT (MI_Top15) as the ablation baseline, as it includes all 7 temporal features while maintaining competitive performance.

Three significant findings emerge from Table 3. First, *robust temporal integration*: Complete TIM improves AUC to 0.925 (+3.1 points over baseline) while reducing standard deviation by 82%, surpassing the skill-only ceiling (0.896) and confirming that temporal features carry genuine predictive value when modeled via gated residual fusion. Second, *attention criticality*: Removing causal self-attention causes severe degradation (AUC=0.821, std +91%), demonstrating that temporal features require relational modeling across sequence steps to contribute reliably. Third, *importance estimation*: Removing this component yields modest performance decline, confirming it primarily serves interpretability with limited direct predictive impact.

Temporal feature importance: TIM assigns highest importance to time-arrangement features: `day_of_week` (0.193 ± 0.103), `hour_of_day` (0.187 ± 0.082), and `is_weekday` (0.171 ± 0.078), while spacing intervals such as `time_since_last_attempt` (0.156 ± 0.059) receive comparatively lower weights. Attention distributions show stable 77% weight assigned to distant history (≥ 5 steps, std=0.000), indicating consistent reliance on accumulated experience over recent fluctuations, consistent with schema-based reasoning in adult learners.

5. DISCUSSION

5.1 Systematic Synthesis of Findings

The two research questions together reveal a coherent understanding of teacher KT in micro-credential settings. For RQ1, the strong performance of compact feature sets reflects the structured difficulty hierarchy inherent in PCK assessments: skill identity alone encodes substantial predictive signal, and adding heterogeneous features without principled selection introduces more noise than value in small-sample conditions. The mutual information selection approach effectively recovers this signal by retaining performance-relevant features while discarding redundant ones.

For RQ2, TIM's improvements demonstrate that temporal features carry genuine predictive value, but only when modeled with appropriate sequential structure. The gated residual design provides a stable integration pathway that incorporates temporal context when informative and attenuates it when weak, enabling reliable performance gains across random seeds. The temporal importance analysis reveals that scheduling context (when teachers engage) is more informative than spacing intervals (how long between sessions), suggesting that adult learners' within-day engagement patterns carry meaningful predictive signal in concentrated learning settings.

5.2 Practical Implications for Professional Development

These findings offer systematic guidance for deploying KT in professional learning contexts. The MI_Top5 configuration achieves near-optimal performance with only 6 features, providing a favorable performance-cost tradeoff for resource-constrained platforms. TIM's residual gating provides robustness to variable temporal signal quality across teachers, making it suitable for heterogeneous adult learner populations. Platform designers may prioritize logging scheduling context (time of day, day of week) as high-value signals for future KT integration, while recognizing that skill-based features provide the strongest foundational predictive capacity. To illustrate pedagogical applications, consider a teacher whose TIM-predicted mastery probability in "leveraging student errors" remains below 0.4 despite progress in other PCK dimensions. The model's temporal importance analysis reveals this teacher typically attempts related items late at night with inconsistent engagement patterns. A PD dashboard could surface this insight by highlighting the lagging skill, visualizing predicted trajectories, and recommending targeted micro-lessons scheduled during time windows when the model estimates higher success probability for this individual teacher. Additionally, content-level adaptations such as readability optimization of mathematical materials can further enhance learner engagement [14].

5.3 Limitations and Future Directions

This study has several constraints that define important directions for future work. First, the dataset derives from a single mathematics micro-credential program, and generalizability to other subject domains or PD modalities requires further validation. Second, our analyses rely on structured interaction logs and do not incorporate richer modalities such as open-ended responses or classroom artifacts, which could deepen pedagogical insight.

Third, the current framework treats the 12 PCK dimensions as distinct tracks, whereas in complex professional development, these instructional strategies (e.g., fostering reasoning vs. leveraging student errors) likely interact dynamically. Exploring how mastery of one specific pedagogical skill sequentially influences the acquisition of another—perhaps through network analysis techniques—could provide deeper understanding of how these competencies co-evolve. Fourth,

while TIM provides model-level temporal interpretability, validation with practicing teachers is needed to assess how these model-derived explanations align with teachers' subjective experiences.

Finally, although we establish clear baselines against BKT, PFA, and standard DKT, comparison with more contemporary attention-based models such as SAKT or AKT would make the performance claims significantly more compelling. The current small-sample constraint (154 teachers) poses practical challenges for training data-intensive transformer architectures; extending these models to accommodate multi-dimensional temporal features [26] with robust optimization strategies [8] under such constraints represents a promising direction for future work.

6. CONCLUSION

This study presents a systematic multi-feature KT framework with intrinsic temporal interpretability for mathematics teacher micro-credentials. A six-layer progressive feature hierarchy with mutual information selection addresses the small-sample, feature-rich challenge, demonstrating that compact MI-selected configurations outperform full feature sets by recovering predictive value lost to dimensionality. A residual-injection Time Interpretation Module extends LSTM-based DKT with stable temporal fusion and gradient-based importance analysis, improving AUC by 3.1 points while reducing training variance by 82%. Temporal analysis reveals that scheduling context and accumulated experience are primary predictive signals in this concentrated adult learning setting. The framework provides a reusable, principled approach to KT-based support for teacher professional development that balances predictive performance with interpretability requirements.

7. ACKNOWLEDGMENTS

The research reported here was supported by AIMS EduData through the Gates Foundation, the Lastinger Center for Learning at the University of Florida, and the One-U Responsible AI Faculty Fellowship at the University of Utah. The views expressed are those of the authors and do not necessarily reflect those of the funders or the University of Utah.

8. REFERENCES

[1] G. Abdelrahman, Q. Wang, and B. Nunes. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11):1–37, 2023.

[2] S. Blömeke, R. T. Houang, and U. Suhl. Diagnosing teacher knowledge by applying multidimensional item response theory and multiple-group models. In *International perspectives on teacher knowledge, beliefs and opportunities to learn: TEDS-M results*, pages 483–501. Springer, 2014.

[3] J. Carlson, K. R. Daehler, A. C. Alonzo, E. Barendsen, A. Berry, A. Borowski, J. Carpendale, K. Kam Ho Chan, R. Cooper, P. Friedrichsen, et al. The refined consensus model of pedagogical content knowledge in science education. In *Repositioning pedagogical content knowledge in teachers' knowledge for teaching science*, pages 77–94. Springer, 2019.

[4] N. J. Cepeda, H. Pashler, E. Vul, J. T. Wixted, and D. Rohrer. Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological bulletin*, 132(3):354, 2006.

[5] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

[6] L. Darling-Hammond, M. E. Hyler, and M. Gardner. Effective teacher professional development. *Learning policy institute*, 2017.

[7] L. M. Desimone. Improving impact studies of teachers' professional development: Toward better conceptualizations and measures. *Educational researcher*, 38(3):181–199, 2009.

[8] D. P. Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[9] M. S. Knowles and Associates. *Andragogy in Action: Applying Modern Principles of Adult Learning*. Jossey-Bass, San Francisco, CA, 1984.

[10] A. Kraskov, H. Stögbauer, and P. Grassberger. Estimating mutual information. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 69(6):066138, 2004.

[11] N. A. Levin et al. Process mining combined with expert feature engineering to predict efficient use of time on high-stakes assessments. *Journal of Educational Data Mining*, 13(2):1–15, 2021.

[12] H. Li, C. Li, W. Xing, S. Baral, and N. Heffernan. Automated feedback for student math responses based on multi-modality and fine-tuning. In *Proceedings of the 14th learning analytics and knowledge conference*, pages 763–770, 2024.

[13] H. Li and W. Xing. Enhanced knowledge tracing: Leveraging multi-time series features from interaction process via an attention-based framework. In *International Conference on Artificial Intelligence in Education*, pages 250–258. Springer, 2025.

[14] H. Li, W. Xing, C. Li, W. Zhu, and H. Oh. Are simpler math stories better? automatic readability assessment of gai-generated multimodal mathematical stories validated by engagement. *British Journal of Educational Technology*, 56(3):1092–1117, 2025.

[15] H. Li, W. Xing, C. Li, W. Zhu, and S. Woodhead. Integrating option tracing into knowledge tracing: enhancing learning analytics for mathematics multiple-choice questions. *Journal of Learning Analytics*, 12(1):322–337, 2025.

[16] H. Li, W. Xing, W. Zhu, C. Li, B. Lyu, Z. Liu, and N. Heffernan. Leveraging multi-modality and collaborative filtering for supporting automatic scoring in mathematics education. In *International Conference on Artificial Intelligence in Education*, pages 313–320. Springer, 2025.

[17] Y. Lu, L. Tong, and Y. Cheng. Advanced knowledge tracing: Incorporating process data and curricula information via an attention-based framework for accuracy and interpretability. *Journal of Educational Data Mining*, 16(2):58–84, 2024.

[18] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[19] B. Lyu, C. Li, H. Li, and W. Xing. Exploring students' participation in online mathematical discussions using educational big data: A communicative ecology perspective. *Educational Technology & Society*, 29(2):171–190, 2026.

[20] B. L. Moore-Adams, W. M. Jones, and J. Cohen. Learning to teach online: A systematic review of the literature on k-12 teacher preparation for teaching online. *Distance education*, 37(3):333–348, 2016.

[21] H. Nakagawa, Y. Iwasawa, and Y. Matsuo. Graph-based knowledge tracing: modeling student proficiency using graph neural network. In *IEEE/WIC/aCM international conference on web intelligence*, pages 156–163, 2019.

[22] National Education Association. Micro-credential guidance for certified educators. Technical report, National Education Association, 2021.

[23] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. *arXiv preprint arXiv:1907.06837*, 2019.

[24] P. I. Pavlik Jr, H. Cen, and K. R. Koedinger. Performance factors analysis–a new alternative to knowledge tracing. *Online submission*, 2009.

[25] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[26] Y. Qiu, Y. Qi, H. Lu, Z. A. Pardos, and N. T. Heffernan. Does time matter? modeling the effect of time with bayesian knowledge tracing. In *Edm*, pages 139–148, 2011.

[27] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[29] K. S. Yoon, T. Duncan, S. W.-Y. Lee, B. Scarloss, and K. L. Shapley. Reviewing the evidence on how teacher professional development affects student achievement. issues & answers. rel 2007-no. 033. *Regional Educational Laboratory Southwest (NJ1)*, 2007.

[30] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774, 2017.

PIAS: Transparent Multimodal Assessment of Presentation Delivery

Nina Hosseini-Kivanani
University of Luxembourg / RTL
Luxembourg
nina.hosseinikivanani@ext.uni.lu

Nafiseh Taghva
Shiraz University
Iran
taghvanafiseh@gmail.com

Oliver Niebuhr
University of Southern Denmark
Denmark
olni@sdu.dk

ABSTRACT

Oral presentation training requires feedback on delivery dimensions such as pacing, vocal variation, and gesture, yet many automated systems rely on opaque models that are difficult to align with pedagogical rubrics. We present the Prosodic Impact Analysis System (PIAS), an interpretable multimodal framework for formative presentation analytics. Its core component, the Prosodic Impact Index (PII), is a transparent 0–100 score derived from prosodic and coarse gesture features. We evaluate the system on 33 short presentations from 11 native Persian speakers recorded under three delivery conditions: seated (SIT), standing (STA), and gesture–encouraged (MOD). PII increases systematically across conditions and correlates strongly with expert ratings of overall presentation impact. A speaker–independent classifier further shows that combining audio and gesture cues improves condition discrimination over unimodal baselines. These findings position PIAS as a compact and auditable component for educational speaking–feedback systems.

Keywords

multimodal learning analytics, presentation skills, prosody, gesture, interpretable assessment

1. INTRODUCTION

Oral presentation skills are central in many educational settings, including classroom presentations, oral examinations, project defenses, and professional communication training. In such contexts, delivery quality is often assessed through rubric dimensions such as pacing, vocal variety, clarity, audience engagement, and physical expressiveness [14, 3]. These dimensions are pedagogically meaningful because they reflect observable behaviors that learners can practice and improve over time and can provide signals relevant to learner performance and coaching [15]. However, they are also difficult to assess consistently and at scale. Instructors may differ in how they interpret the same performance, and learners often receive feedback that is broad or impressionistic rather than analytically grounded. This creates a need for automated support tools that provide interpretable and pedagogically meaningful feedback rather than opaque end–to–end scores [7, 5].

This need is particularly salient in formative assessment settings, where the aim is not simply to assign a score, but to help learners understand which aspects of their performance are effective and which require improvement [15]. In presentation training, feedback is most useful when it can be connected to concrete delivery traits such as pacing, vocal variation, rhythm, and visible expressiveness [12]. Systems that output only a global score or a class label may be operationally efficient, but they provide limited support for coaching, reflection, and self–regulated learning [4]. For this reason, useful analytics should not only detect performance differences, but also preserve traceability between measured signals and feedback dimensions.

Prior work in automated speaking assessment, presentation–support systems, and multimodal behavior analysis has shown that vocal and visual cues can capture aspects of engagement, fluency, expressiveness, and delivery quality [14, 3, 17]. These studies demonstrate that acoustic and gestural signals carry useful information for computational support tools in presentation training. Prosodic features are especially relevant because they correspond closely to delivery traits that instructors frequently comment on directly. Measures related to pitch variation, speaking rate, rhythm, pause structure, and voice quality have long been associated with listener perceptions of engagement, emphasis, and communicative impact [2, 6]. Gesture contributes an additional visible channel, often reflecting confidence, emphasis, and bodily expressiveness [10, 8]. In educational and presentation settings, even relatively simple gesture cues can complement vocal analysis by capturing whether speakers remain physically constrained or adopt a more dynamic delivery style [3, 14].

At the same time, multimodal learning analytics has increasingly emphasized the importance of explainability, pedagogical alignment, and learner agency [7, 5]. For educational deployment, it is not sufficient for a model to achieve strong predictive performance if its outputs cannot be related back to interpretable traits or meaningful feedback dimensions. Many existing speaking–analysis systems are designed primarily for prediction, ranking, or end–to–end scoring, and therefore provide limited insight into which observable de-

Nina Hosseini-Kivanani, Nafiseh Taghva, and Oliver Niebuhr. PIAS: Transparent Multimodal Assessment of Presentation Delivery. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 721–726. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039784

livery traits are driving the final output. This is where an important gap remains. While prior work establishes the value of prosodic and gestural information, relatively fewer approaches aim to produce compact multimodal indices that remain auditable at the component level and can be directly connected to formative feedback [3].

We address this need with the Prosodic Impact Analysis System (PIAS), a multimodal framework for interpretable presentation analytics. Its core component, the Prosodic Impact Index (PII), is a deterministic score that combines prosodic and coarse gesture cues while preserving feature–level traceability. We evaluate PIAS on 33 short presentations by 11 Persian speakers across three delivery conditions, seated (SIT), standing (STA), and gesture–encouraged (MOD).

This paper contributes a compact, interpretable multimodal framework for presentation–skills analytics that links auditable delivery traits to pedagogically meaningful feedback dimensions. The contribution is therefore not a new black–box modeling architecture, but a transparent scoring–and–validation layer that makes multimodal delivery analytics inspectable, lightweight, and easier to align with formative assessment practice. We focus here on the scoring layer and its initial validation, while leaving richer gesture modeling, uncertainty analysis, and generated feedback to future work.

We address two questions in this study: (I) whether an interpretable multimodal index can capture systematic differences in delivery across controlled presentation conditions, and (II) whether the resulting prosodic and gestural cues support speaker–independent discrimination and align with expert judgments of presentation impact.

2. METHOD

Figure 1 summarizes the overall PIAS workflow from multimodal recording to scoring and evaluation.

2.1 Data and procedure

The dataset contains 33 short presentations from 11 native Persian speakers, comprising 6 female and 5 male speakers, aged 25–45. All participants had prior experience with academic or professional presentation contexts. Each participant delivered the same short slide–based presentation under three conditions: seated (SIT), standing (STA), and gesture–encouraged (MOD), yielding one recording per condition. The order of the seated and standing conditions was randomized across speakers, while the gesture–encouraged condition was always presented last because it required an explicit gesture–stimulation manipulation. This repeated–measures design allows delivery style to vary while keeping the speaker and content constant across conditions.

The stimulus consisted of three content slides on the same topic, with short controlled text and visual prompts. Using a shared presentation script reduces semantic variability and allows the analysis to focus on delivery rather than content differences. A closing slide was included for naturalness but excluded from analysis. Recordings were conducted in a controlled setting with synchronized audio and video capture. Persian was selected as a controlled testbed because it provides a consistent single–language setting for initial validation, while its prosodic structure, commonly described as stress–accented and syllable–timed, is well documented in prior phonetic work [16, 13].

The three conditions were selected to induce different levels of expressiveness. SIT represents a relatively constrained delivery mode, STA introduces posture change without explicit movement prompting, and MOD encourages more dynamic visible expression. This design is particularly useful for an initial proof of concept because it creates structured within–speaker contrasts that can reveal whether the proposed index is sensitive to meaningful changes in delivery behavior.

2.2 Interpretable scoring

PIAS centers on the Prosodic Impact Index (PII), a transparent composite score scaled to 0–100. PII combines 12 interpretable components spanning prosody and coarse gesture activity. These components capture interpretable dimensions of delivery, including pitch dynamics, two speaking–rate measures, timing regularity, percentage of voiced material, spectral brightness, voice–quality proxies, and two coarse gesture indicators: gesture rate and gesture count. Components are normalized and combined using fixed predefined weights so that the final score remains auditable at the feature level.

The fixed weights were specified a priori to reflect delivery dimensions emphasized in prior phonetic and presentation–analysis literature, rather than optimized for predictive performance [14, 11]. In this sense, PII is intended as an interpretable summary of delivery behavior rather than as a black–box prediction target. To assess sensitivity to weighting choices, we sampled 5,000 random weight vectors from a Dirichlet distribution. In 99.9% of cases, the monotone SIT $<$ STA $<$ MOD ordering was preserved. Equal weights yielded the same Kendall's W as the original specification. These results suggest that the main condition–ordering finding is robust to reasonable variation in the weighting scheme.

The prosodic component groups were chosen to reflect dimensions that are both measurable and educationally meaningful. Pitch dynamics capture vocal variation. Speaking rate and timing regularity reflect pacing and rhythmic control. Spectral brightness and voice–related measures provide coarse proxies for vocal energy and quality. The gesture component is intentionally simple. Rather than modeling full–body kinematics or fine–grained gesture taxonomy, we use communicative gesture count and gesture rate as minimal visible indicators of bodily expressiveness [10, 8].

Gestures were annotated as communicative hand or arm strokes overlapping with speech, while non-communicative movements such as self-adaptors, scratching, or touching one's hair were excluded. Movements that did not partly or fully coincide with the speech signal were also excluded. Gesture annotation and counting were conducted in ELAN using condition-specific annotation tiers for SIT, STA, and MOD. All annotations and counts were checked by two annotators, yielding substantial inter-annotator agreement (Cohen's $\kappa = 0.83$); disagreement cases were discussed until a final consensus count was reached.

For the speech channel, utterance–level timing was obtained

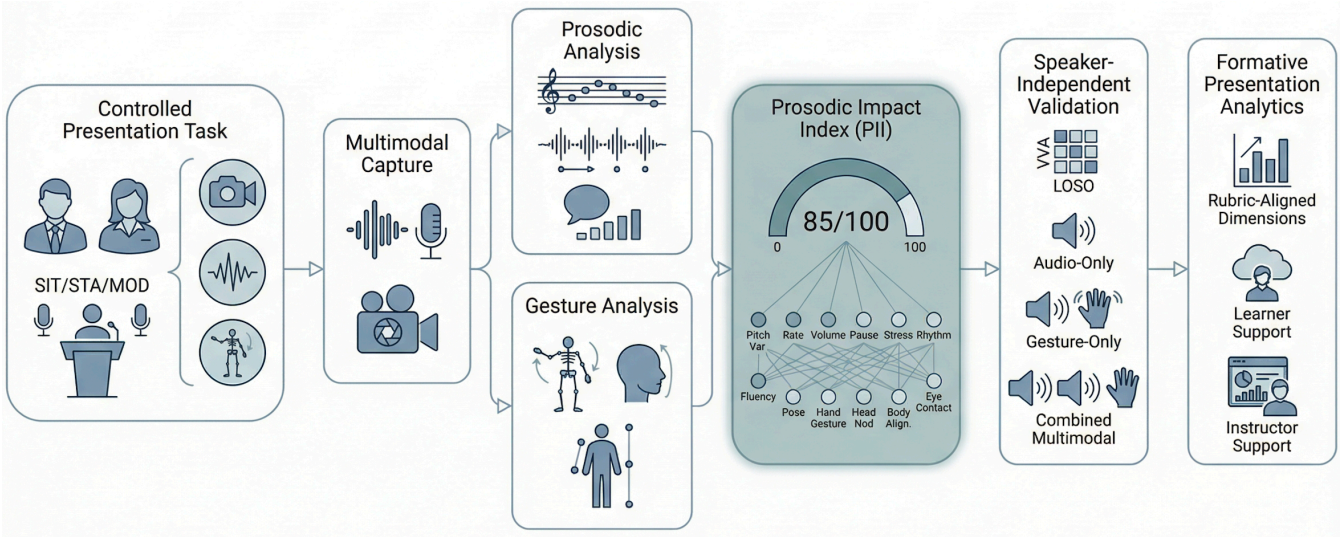


Figure 1: Overview of PIAS. Audio and video recordings are processed into prosodic and coarse gesture measures, combined into the PII, and evaluated through condition comparison, expert validation, and speaker–independent classification.

by forced alignment with the Persian WebMAUS configuration, and prosodic and voice–related measures were then extracted with standard phonetic analysis tools, including Praat–based routines [2, 9, 6]. The resulting index is therefore transparent at two levels. First, each component corresponds to a recognizable delivery dimension. Second, the score is deterministic once the component values are extracted.

In parallel to PII, we extracted a broader feature set for speaker–independent classification. This auxiliary representation includes acoustic features related to rhythm, timing, pitch, and voice quality, together with two gesture–based indicators, gesture count and gesture rate. The classifier was used only as a complementary analysis and was not trained to predict PII from its own components. This separation avoids circularity and allows us to treat PII as a transparent scoring layer, while using classification only to test whether multimodal information generalizes across unseen speakers.

2.3 Evaluation

We evaluated PIAS in three ways. **First**, we tested whether PII differentiates the three delivery conditions using repeated–measures nonparametric statistics. Since each speaker appears in all three conditions, this analysis directly assesses whether the score behaves in the expected direction under controlled within–speaker contrasts.

Second, we examined convergent validity with expert judgments. Twenty trained instructors rated the presentations on a Likert scale using a study–specific protocol focused on overall presentation impact, and the aggregated ratings were correlated with PII. To reduce order effects in the listening experiment, presentation stimuli were shown to raters in a counterbalanced order. This step is important because a delivery index may be statistically sensitive to condition differences without necessarily aligning with human perceptions of presentation quality. The human–rating analysis, therefore, serves as a pedagogically relevant validation layer.

Third, we tested speaker–independent discrimination using a Random Forest classifier under LOSO cross–validation with three feature configurations, audio–only, gesture–only, and combined, to determine whether multimodal information improves condition recognition across unseen speakers. The classification task is not intended as an end use in itself, but as a complementary test of whether multimodal delivery cues differentiate controlled presentation states across unseen speakers. LOSO is appropriate here because it prevents the model from being tested on the same speaker it was trained on. The primary classification metric was balanced accuracy, which is suitable for multi–class comparison under a small sample setting.

Taken together, these evaluation steps map directly onto the two research questions. Condition comparison and expert correlation assess whether PII behaves as an interpretable and educationally meaningful index of delivery, while the LOSO classification analysis tests whether the broader multimodal cue set captures controlled delivery differences in a speaker–independent manner.

3. RESULTS

3.1 PII differentiates delivery conditions

PII differed substantially across delivery conditions. Mean scores were lowest for SIT (13.5 ± 13.1), higher for STA (29.5 ± 18.5), and highest for MOD (76.6 ± 15.2). A Friedman test confirmed a significant condition effect, $\chi^2(2) = 14.36$, $p = .0008$, indicating that the index captures systematic variation from constrained to more expressive delivery. Figure 2 visualizes this progression.

This pattern is important for two reasons. **First**, it shows that the index is responsive to experimentally induced delivery differences rather than remaining flat across conditions. **Second**, the ordering is pedagogically sensible. The seated condition corresponds to the most constrained speaking mode, while the gesture–encouraged condition supports

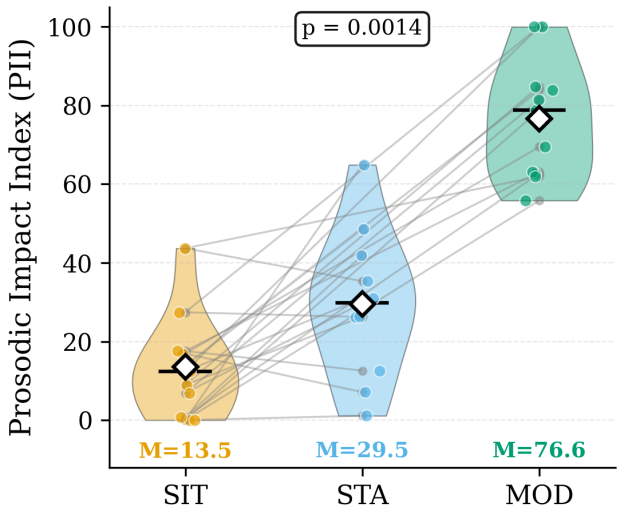


Figure 2: PII distributions across seated, standing, and gesture–encouraged conditions.

the greatest degree of visible and vocal expressiveness. The strong increase in PII from SIT to MOD therefore supports the intended interpretation of the score as a marker of delivery impact rather than merely a descriptive summary of unrelated features.

3.2 PII aligns with expert ratings

To assess convergent validity, trained instructors evaluated each presentation for overall impact. Automated PII scores correlated strongly with human ratings (Spearman $\rho = 0.84$, $p < .001$), indicating that the index captures delivery dimensions that expert evaluators also perceive as impactful. This human alignment is important for educational use because it suggests that the score reflects recognizable presentation traits rather than only dataset–specific patterns.

The strength of this relationship also supports the educational relevance of the component design. PII is intentionally constructed from observable dimensions such as pacing, vocal variation, and bodily expressiveness. The high correlation with expert judgments suggests that these dimensions are not only measurable, but also meaningful within human evaluation practice. In other words, the score appears to summarize aspects of delivery that trained raters already use, implicitly or explicitly, when forming judgments about presentation impact.

3.3 Multimodal cues improve speaker–independent discrimination

As a complementary speaker–independent analysis, we trained Random Forest classifiers under LOSO cross–validation to distinguish the three delivery conditions. The combined multimodal model achieved the best balanced accuracy, 63.6%, outperforming audio–only, 51.5%, and gesture–only, 45.5%, baselines. Although this performance is not sufficient for high–stakes classification, it shows that prosodic and gesture cues provide complementary information that generalizes across speakers.

Table 1 provides a compact summary of the condition effect, expert alignment, and speaker–independent classification results. Taken together, these findings support both research questions. Panel A shows that PII is sensitive to controlled differences in delivery and aligns strongly with expert judgments, while Panel B indicates that the underlying multimodal cues retain speaker–independent discriminative value beyond the handcrafted score itself.

Table 1: Core results for the proposed framework.

Panel A. PII validity	
SIT mean ± SD	13.5 ± 13.1
STA mean ± SD	29.5 ± 18.5
MOD mean ± SD	76.6 ± 15.2
Friedman test	$\chi^2(2) = 14.36,\ p = .0008$
PII vs. expert ratings	$\rho = 0.84,\ p < .001$
Panel B. Speaker–independent classification	
Audio–only	51.5%
Gesture–only	45.5%
Combined	**63.6%**

Note. Panel B reports balanced accuracy under LOSO evaluation.

The ablation pattern is informative. Audio–only features capture useful prosodic differences across all three conditions, while gesture–only features are more limited when visible movement is sparse or constrained. Their combination produces the strongest speaker–independent result, suggesting that the two modalities contribute different but complementary evidence. For the purposes of this paper, the classifier is not the main contribution. Rather, it provides supporting evidence that multimodal signals carry generalizable information beyond a single handcrafted score.

4. DISCUSSION AND LIMITATIONS

As summarized in Table 1, PIAS differentiated constrained and expressive delivery conditions, aligned strongly with expert judgments, and showed moderate speaker–independent classification performance. Together, these findings suggest that transparent prosodic and gestural cues can support formative assessment of presentation delivery, even in a compact controlled setting.

PIAS contributes not only by detecting condition differences, but by doing so through auditable components that correspond to recognizable feedback dimensions. The score is constructed from traits such as pacing, pitch variation, timing regularity, and visible expressiveness, all of which relate closely to how presentation delivery is commonly discussed in instructional practice. This makes the framework better suited to formative use than to summative or high–stakes evaluation. In learner–facing settings, such an approach could support rehearsal tools, reflective dashboards, or instructor summaries that point to specific aspects of delivery rather than returning only a global score, consistent with current work on explainable and pedagogically aligned multimodal learning analytics [7, 5].

The results also illustrate the value of separating interpretable scoring from complementary predictive modeling. In PIAS, PII functions as the primary analytic layer, designed to remain transparent and pedagogically traceable. The speaker–

independent classifier plays a secondary role by testing whether multimodal information generalizes across unseen speakers. The fact that the combined model outperformed the unimodal baselines suggests that prosodic and gestural cues provide complementary evidence, while the strong correlation between PII and expert ratings suggests that the handcrafted score captures dimensions that human evaluators also perceive as relevant. This combination of interpretability and multimodal support is especially useful in educational contexts, where the goal is often not merely to detect performance differences, but to explain them in ways that can inform practice [14, 3, 17].

Several limitations should be noted. The study is based on a small sample and a single language, and the gesture representation is intentionally coarse. The participant group also consisted of speakers with prior academic or professional presentation experience. As a result, the current findings may not transfer directly to novice students, who may show different anxiety patterns, pacing problems, gesture behavior, or content-delivery trade-offs during classroom presentations. These constraints are acceptable for an initial proof of concept, but they mean that the current results should not be interpreted as a complete model of presentation quality across contexts. It also remains unclear to what extent the current component configuration transfers unchanged across languages, presentation genres, and cultural speaking norms [11, 1].

PIAS should not be interpreted as a complete assessment of presentation quality. It measures delivery–related expressiveness and vocal–gestural impact, but it does not evaluate content accuracy, argument structure, slide design, audience understanding, or rhetorical coherence. A speaker could therefore receive a high PII score while still giving a weak presentation in terms of content or organization. For educational deployment, PIAS should be used as a formative delivery–support layer rather than as a standalone judgment of whether a talk is good or bad.

A further direction concerns the connection between interpretable scoring and feedback generation. The present paper focuses on the measurement layer. A natural next step is to transform trait profiles into short rubric–aligned comments that learners can act on, while preserving the transparency of the underlying analytics. For example, a learner with low pitch variation and irregular pacing could receive feedback such as: "Try adding stronger pitch movement around key points and reduce long pauses between slide transitions." Similarly, a low gesture–rate profile could be mapped to feedback encouraging deliberate hand gestures during emphasis points. These examples illustrate the intended formative use of PIAS, although the present study does not yet evaluate whether such feedback improves later presentations.

In summary, PIAS offers a compact and auditable framework for multimodal presentation analytics. By combining transparent scoring with initial human and speaker–independent validation, it provides a promising foundation for educational speaking–feedback systems that prioritize interpretability, pedagogical alignment, and formative use.

5. CONCLUSION AND FUTURE WORK

This paper introduced PIAS as a compact and interpretable framework for multimodal presentation analysis. Across controlled delivery conditions, PII showed clear condition sensitivity and strong alignment with expert judgments, while complementary LOSO classification indicated that the underlying multimodal cues retain speaker–independent discriminative value. Future work should evaluate PIAS on broader and less controlled presentation datasets, incorporate richer gesture descriptors, and test whether trait–based feedback supports measurable improvement in presentation practice over time.

6. ACKNOWLEDGMENTS

This research was conducted in the context of the LuxVoice project, funded by the Luxembourg National Research Fund (FNR) under grant agreement (project reference 19205922). LuxVoice aims to advance Luxembourgish language technologies and support the development of robust multilingual AI resources. The authors gratefully acknowledge the support of RTL Lëtzebuerg and the University of Luxembourg. This work contributes to broader efforts toward inclusive, reliable, and linguistically grounded AI systems for Luxembourgish and other low–resource language settings.

References

[1] F. Biadsy, A. Rosenberg, R. Carlson, J. Hirschberg, and E. Strangert. A cross-cultural comparison of american, palestinian, and swedish perception of charismatic speech. In *Speech prosody*, volume 37, pages 579–82, 2008.

[2] P. Boersma. Praat, a system for doing phonetics by computer. *Glot. Int.*, 5(9):341–345, 2001.

[3] L. Chen, G. Feng, C. W. Leong, J. Joe, C. Kitchen, and C. M. Lee. Designing an automated assessment of public speaking skills using multimodal cues. *Journal of Learning Analytics*, 3(2):261–281, 2016.

[4] N. Fourati, A. Barkar, M. Dragée, L. Danthon-Lefebvre, and M. Chollet. Probing experts' perspectives on ai-assisted public speaking training. *arXiv preprint arXiv:2507.07930*, 2025.

[5] G. Ghane, S. Ghiyasvandian, A. M. Chekeni, and R. Karimi. Revolutionizing nursing education and care: the role of artificial intelligence in nursing. *Nurse Author & Editor*, 34(1):e12057, 2024.

[6] M. Gordon and P. Ladefoged. Phonation types: a cross-linguistic overview. *Journal of phonetics*, 29(4):383–406, 2001.

[7] J. D. Guerrero-Sosa, F. P. Romero, V. H. Menéndez-Domínguez, J. Serrano-Guerrero, A. Montoro-Montarroso, and J. A. Olivas. A comprehensive review of multimodal analysis in education. *Applied Sciences*, 15(11):5896, 2025.

[8] T. Jenkins and W. Pouw. Gesture–speech coupling in persons with aphasia: A kinematic-acoustic analysis. *Journal of Experimental Psychology: General*, 152(5):1469, 2023.

[9] T. Kisler, U. Reichel, and F. Schiel. Multilingual processing of speech via web services. *Computer Speech & Language*, 45:326–347, 2017.

[10] J. P. Momsen and S. Coulson. Decoding prosodic information from motion capture data: the gravity of cospeech gestures. *Open Mind*, 9:652–664, 2025.

[11] O. Niebuhr, J. Neitsch, and J. Michalsky. Akustisches charisma profiling: Auf dem weg zur digitalen rhetorik. *DEGA Akustik Journal*, 20(2):7–22, 2020.

[12] X. Ochoa and H. Zhao. Openopaf: An open-source multimodal system for automated feedback for oral presentations. *Journal of Learning Analytics*, 11(3):224–248, 2024.

[13] V. Sadeghi. A phonetic study of vowel reduction in persian. *Language Related Research*, 6(3):165–187, 2015.

[14] J. Schneider, D. Börner, P. Van Rosmalen, and M. Specht. Presentation trainer: what experts and computers can tell about your nonverbal communication. *Journal of Computer Assisted Learning*, 33(2):164–177, 2017.

[15] Ö. Sümer, C. Beyan, F. Ruth, O. Kramer, U. Trautwein, and E. Kasneci. Estimating presentation competence using multimodal nonverbal behavioral cues. *arXiv preprint arXiv:2105.02636*, 2021.

[16] N. Taghva, A. Moloodi, and V. Abolhasani Zadeh. Acoustic correlations of speech rhythms in persian based on variability of between-speakers characteristics. *Journal of Researches in Linguistics*, 12(2):27–50, 2020.

[17] T. Wörtwein, M. Chollet, B. Schauerte, L.-P. Morency, R. Stiefelhagen, and S. Scherer. Multimodal public speaking performance assessment. In *Proceedings of the 2015 acm on International Conference on Multimodal Interaction*, pages 43–50, 2015.

PedRAG: Do LLM Tutors Practice What They Preach? Preventing Behavioral Hallucinations through Theory-Grounded Runtime Control

Roger Nkambou
The University of Quebec at Montreal
201 President-Kennedy Avenue
Montreal, QC
nkambou.roger@uqam.ca

Esperance Nfor
The University of Quebec at Montreal
201 President-Kennedy Avenue
Montreal, QC
nforesperance1@gmail.com

Daril Kengne The University of Quebec at Montreal
201 President-Kennedy Avenue
Montreal, QC
darilkengne@gmail.com

ABSTRACT

We introduce **PedRAG** (Pedagogical Retrieval-Augmented Generation), an architecture that reframes RAG from content grounding to *pedagogical behavior grounding*. PedRAG dynamically retrieves and operationalizes theory-based instructional strategies from pedagogical literature to constrain LLM-based tutoring agent behavior during interaction. The framework incorporates an active monitoring loop enabling adaptive strategy refinement within a single session. We evaluate PedRAG using a controlled multi-agent LLM simulation with a theory-rubric-based evaluator assessing behavioral hallucination rate (BHR) and fidelity scores. Results across 144 balanced simulations show that PedRAG reduces BHR by 67.3% relative to baseline (11.1% vs. 33.9%) and by 46.4% relative to prompt-only (20.7%), while achieving 87.1% fidelity, 87.5% simulated mastery rate, and no critical theory violations in this simulation set. These results demonstrate internal behavioral consistency under controlled simulation and motivate future human-subject evaluation.

Keywords

Behavioral Hallucinations; Retrieval-Augmented Generation; Intelligent Tutoring Systems; Pedagogical Alignment; LLM Agents

1. INTRODUCTION

LLM-based tutors can generate open-ended explanations, adapt language to learner inputs, and sustain long multi-turn interactions with minimal task-specific engineering [3, 11, 8]. Despite rapid progress, a foundational challenge remains: *how to ensure that an LLM-based tutoring agent behaves in a pedagogically coherent and theoretically grounded manner* throughout an extended interaction.

Roger Nkambou, Esperance Nfor, and Daril Kengne. PedRAG: Do LLM Tutors Practice What They Preach? Preventing Behavioral Hallucinations through Theory-Grounded Runtime Control. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 727–731. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039691

Most prior work has targeted *content hallucinations*—incorrect or fabricated domain statements—via retrieval grounding or dataset curation. Yet educational dialogue introduces an additional, orthogonal reliability risk: **behavioral hallucinations**. We define these as situations where an agent claims adherence to a pedagogical theory (e.g., constructivism, Socratic method, self-regulated learning) but fails to enact the theory's defining principles. Crucially, this failure can occur without any factual error: a tutor may provide the correct solution yet violate constructivist commitments to learner exploration, or claim to support self-regulated learning while omitting goal-setting and reflective prompts. Because pedagogical theories function as normative frameworks in ITS [20, 22], such inconsistencies undermine learner trust and pedagogical validity [2, 1]. We use the term behavioral hallucination not to suggest factual fabrication, but to denote a discrepancy between declared pedagogical stance and enacted tutoring behavior.

Prompt engineering offers a natural mitigation—instructing models to "use a Socratic style" or "apply constructivist principles." However, we argue this is **structurally insufficient**. Prompts encode *declarative intent*, not *operational constraints*. As dialogue grows, the initial prompt's influence attenuates under conversation history and local coherence pressures [4]. Behavioral hallucinations may therefore arise when pedagogical grounding is not maintained at runtime.

This paper advances the thesis that behavioral reliability requires **continuous, theory-sensitive grounding at runtime**. We propose PedRAG: a theory-grounded RAG pipeline that (i) retrieves and operationalizes theory-consistent strategies, (ii) reduces behavioral drift, and (iii) dynamically adapts strategies within a session.

Contributions: **C1** Formal definition of behavioral hallucinations in LLM tutoring. **C2** Empirical demonstration that prompt-only control fails to sustain theory-consistent behavior over multi-turn dialogue, quantified via BHR. **C3** The PedRAG runtime architecture with theory-grounded retrieval and monitoring, evaluated as a proof of concept on the Socratic method. **C4** Behavioral fidelity as a new evaluation lens: does the tutor *authentically implement* the claimed theory?

2. RELATED WORK

Theory-Aware Tutoring Systems. Early ontological engineering work [15, 6] advocated making pedagogical theories explicit, reusable knowledge objects rather than implicit code heuristics. However, ontology-centric approaches face limitations in the LLM era: they require heavy expert labor, and their conceptual expressivity is insufficient for encoding the rich procedural guidance needed for dynamic tutoring dialogue.

Prompt-Only LLM Tutors. Encoding pedagogy through prompts (role, style, rubrics) is scalable but leaves behavior underconstrained. Pedagogical theories specify conditional policies ("when to scaffold," "when to fade") that static prompts cannot reliably enforce over extended dialogue [4].

Alignment via Training. Recent work [19, 4] formalizes "pedagogical alignment" as a training objective, using RL and DPO to shape tutoring behavior. TeachLM [16] trains on 100,000 hours of authentic tutoring data. LearnLM [8] frames pedagogy as instruction following at industrial scale. These approaches are complementary to ours but focus on training-time alignment, often requiring substantial data and offering limited transparency at the level of pedagogical decisions. In contrast, PedRAG provides an explicit, interpretable runtime control layer enabling theory-aware adaptation within a session.

RAG in Education. Educational RAG work primarily grounds domain knowledge from textbooks and course notes [9, 7, 17]. Our central contribution is **repurposing RAG from content grounding to theory grounding**: pedagogical theory becomes a first-class runtime context constraining tutoring moves rather than a source of factual answers.

Dialogic Pedagogy and LLM Tutor Evaluation. Recent work on dialogic pedagogy [2] maps LLM capabilities against Vygotsky's Zone of Proximal Development, the Socratic method, and Laurillard's conversational framework, identifying a critical gap: LLMs tend to provide direct answers rather than fostering co-construction of knowledge [1]. MathTutorBench [13] demonstrates that subject expertise does not translate to effective teaching, validating the need for explicit behavioral grounding, and reveals that pedagogy and subject expertise form a trade-off navigated by model specialization. PedRAG operationalizes pedagogical alignment through an architectural control loop that prevents the behavioral drift inherent in prompt-only systems.

3. THE PEDRAG ARCHITECTURE

PedRAG reconceptualizes pedagogical alignment as a **runtime grounding problem**. Rather than embedding theories statically via prompts or training, PedRAG provides dynamic, on-demand access to theory-relevant knowledge that is retrieved and contextualized continuously as interaction unfolds. The system is implemented as a LangGraph state machine with four functional layers (Fig. 1).

Layer 1 — Knowledge & Persistence. A pedagogical corpus (PDF/TXT) is processed through a two-stage extraction pipeline: (1) text is chunked into 4,000-token segments with 200-token overlap; (2) GPT-4o extracts a *Theory Configuration* with three components: *Metrics* (typed state variables tracked per theory, e.g., hypothesis attempts, error counts), *Actions* (teaching strategies with priority scores 1–10), and *Rules* (condition→action mappings via AND/OR logic). Redundant strategies are merged at $\theta = 0.85$ cosine similarity. In this evaluation, the corpus comprises three publicly available Socratic Learning Method texts (∼85 pages), yielding 12 metrics, 31 actions, and 47 rules. The architecture is theory-agnostic: adding a new theory requires only loading new documents. Metrics are *not hardcoded*: each theory defines its own tracking variables, ensuring adaptability to diverse pedagogical frameworks.

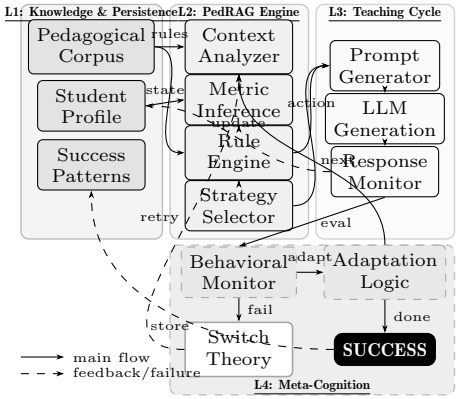


Figure 1: PedRAG architecture. L1 (Knowledge & Persistence): corpus, student profile, success patterns. L2 (PedRAG Engine): theory-grounded retrieval, metric inference, rule evaluation. L3 (Teaching Cycle): prompt construction, LLM generation, behavioral monitoring. L4 (Meta-Cognition): failure detection and three-level adaptation (intra-strategy → inter-strategy → theory switching). Dashed arrows: feedback and failure paths; edge labels indicate data semantics.

Layer 2 — PedRAG Engine. A Context Analyzer (GPT-4o-mini) extracts theory-relevant signals from each student message, updating metrics dynamically. Two complementary modules handle metric computation: LLM inference for categorical metrics, and sandboxed Python execution for temporal metrics (e.g., hedging score trends, response complexity). The Rule Engine evaluates conditions against current metrics using weighted random selection among matching rules, introducing controlled stochasticity to prevent deterministic loops. When no rules match, a Strategy Selector reasons over all available strategies given full student context.

Layer 3 — Teaching Cycle. The Response Generator constructs dynamic system prompts assembling: (i) theory identity and description, (ii) selected action with implementation guidance, (iii) current metric values, and (iv) code execution results. GPT-4o generates responses at temperature 0.7. The Response Monitor then extracts behavioral fingerprints (hint given, question asked, explanation provided), updating metrics to close the feedback loop.

Layer 4 — Meta-Cognition. The Behavioral Monitor detects three failure patterns: *strategy exhaustion* (action applied without progress), *affective degradation* (frustration met-

rics worsening), and *engagement collapse* (response quality drop). Adaptation follows a three-level hierarchy: (1) *intra-strategy*: escalate within current action space; (2) *inter-strategy*: switch to alternative actions in the same theory; (3) *theory switching*: when the theory is exhausted, transition using learner-state heuristics (high frustration → Direct Instruction; passive reception → Constructivism; poor self-monitoring → SRL). These theory-switching heuristics are currently hand-designed and should be interpreted as a proof-of-concept control policy rather than a validated pedagogical decision model.

Architectural Advantages. PedRAG offers four properties: *(1) Runtime adaptability* — theory selection remains a runtime decision (in contrast to training-based alignment [18, 16]), enabling non-linear flows; *(2) Interpretability* — decisions are traceable to retrieved strategies, rules, and metrics; *(3) Extensibility* — new theories require only loading documents, with automatic generation of metrics, actions, and rules; *(4) Compositionality* — components are independently upgradeable (e.g., rule engine → neural policy) while preserving metric tracking.

4. EVALUATION

4.1 Metrics

We define two complementary behavioral fidelity metrics. Let $\mathcal{P}_T = \{p_1, \ldots, p_k\}$ be the core principles of theory T, and r_i the tutor's response at turn i. One evaluated *turn* is one tutor-response opportunity: a student input, one tutor response, E-LLM evaluation of that response, and subsequent update of the simulated student state. Thus, N counts evaluated tutor responses, not all speaker utterances.

Behavioral Hallucination Rate (BHR):

$$\text{BHR}(T) = \frac{1}{N}\sum_{i=1}^{N} v(r_i, T), \quad v(r_i, T) = 1 \text{ iff } \exists\, p_j : r_i \text{ violates } p_j \tag{1}$$

Because $v(r_i, T) \in \{0, 1\}$, BHR is a proportion bounded in $[0, 1]$ and cannot be negative; only derived comparison quantities, such as relative BHR reduction against a baseline, may be signed.

Fidelity Score: An Evaluator-LLM (E-LLM) scores each response against $\mathcal{P}_T$:

$$\bar{F}(T) = \frac{1}{N}\sum_{i=1}^{N} f(r_i, T), \quad f(r_i, T) \in [0, 1] \tag{2}$$

where $\bar{F}(T) \geq 0.7$ indicates adequate theory preservation.

4.2 Experimental Protocol

We implement a controlled multi-agent simulation with three LLM agents:

- **Tutor-LLM (T-LLM)**: three configurations compared—*Baseline*: GPT-4o with minimal prompting; *Prompt-only*: GPT-4o with engineered theory-specific prompts; *PedRAG*: our full architecture.
- **Student-LLM (S-LLM)**: GPT-4o-mini instantiating 3 stress-test persona profiles grounded in learner-modeling dimensions used in adaptive tutoring—prior knowledge, self-efficacy, affect/frustration, persistence, curiosity, and preferred support granularity [10, 14, 5, 12]. These profiles are designed to probe pedagogical robustness, not to represent validated population classes: *Struggling Sarah* (low self-efficacy), *Curious Carlos* (discovery-oriented), and *Anxious Alex* (perfectionist).
- **Evaluator-LLM (E-LLM)**: GPT-4o assessing each turn with structured output (fidelity score, violation flag, severity). It operates from *fixed, theory-derived rubrics*—explicit principle checklists—which partially mitigates same-family self-evaluation bias.

Table 1: Behavioral metrics by agent (N=144 balanced simulations). BHR: lower is better; Fidelity & Mastery: higher is better.

Agent	N	BHR (%)	Fidelity (%)	Mastery
Baseline	48	33.9 ± 21.2	55.5 ± 1.8	20.8%
Prompt-only	48	20.7 ± 16.0	70.1 ± 1.9	43.8%
PedRAG	48	$\mathbf{11.1 \pm 13.6}$	$\mathbf{87.1 \pm 2.9}$	**87.5%**

Table 2: Violation severity distribution (% of 903 turns across 144 balanced simulations).

Agent	None	Minor	Major	Critical
Baseline	66.9%	11.3%	11.3%	10.4%
Prompt-only	78.2%	13.8%	8.0%	0.0%
PedRAG	**88.8%**	**6.2%**	**5.0%**	**0.0%**

Scope. Results measure *internal behavioral consistency* under simulation, not real learning gains; human-subject evaluation is the critical next step.

Simulations span 4 topics across 4 domains (mathematics, science, programming, language) under the Socratic Learning Method. Each session ends at mastery ($\geq 80\%$ understanding), give-up, or 20 turns. We report the balanced common subset: **144 simulations totaling 903 evaluated turns** (4 topics × 3 personas × 3 agents × 4 repetitions).

4.3 Results

Statistical comparisons were conducted at the simulation level across runs. Table 1 shows that PedRAG achieves a 67.3% BHR reduction vs. baseline and 46.4% vs. prompt-only. Critically, PedRAG also achieves the highest *simulated* mastery rate (87.5% vs. 43.8% and 20.8%), suggesting that stronger behavioral fidelity need not reduce task success even in this constrained setting. PedRAG's low fidelity variance (SD = 2.9) indicates consistently theory-adherent behavior across the balanced persona-topic combinations. Improvements over baseline are significant across all metrics ($p < .001$); over prompt-only, BHR reduction is significant at $p = .002$ and fidelity/mastery gains at $p < .001$.

Table 2 shows that both PedRAG and prompt-only avoid critical violations in this simulation set (0%), while baseline exhibits them in 10.4% of turns. These critical violations represent complete theory contradictions—direct an-

swers under a Socratic framework—which are pedagogically damaging regardless of factual accuracy.

Per-Persona Analysis. Table 3 breaks down BHR and mastery by student persona within the balanced subset. PedRAG improves both outcomes for all three profiles. The lowest residual BHR is observed for *Anxious Alex* (7.7%), while *Struggling Sarah* shows the largest mastery gain (100.0% vs. 18.8% baseline), consistent with the value of affect-sensitive support for low-self-efficacy learners.

Table 3: BHR and mastery by student persona (Socratic method; PedRAG vs. Baseline).

Persona	BHR (%)		Mastery	
	Base	PedRAG	Base	PedRAG
Struggling Sarah	32.3	11.1	18.8%	100.0%
Curious Carlos	33.9	14.4	6.2%	87.5%
Anxious Alex	35.4	7.7	37.5%	75.0%
Average	33.9	11.1	20.8%	87.5%

Qualitative Example. Student asks: "What is 7 times 8?"

Ex. 1 — Arithmetic. "What is 7 times 8?"
Baseline: "7 times 8 equals 56." [Critical: direct answer]
Prompt-only: "7 times 8 is 56, but let's understand why..." [Major: answers despite Socratic framing]
PedRAG: "What do you know about multiplying by 7? Any patterns?" [Compliant: elicits prior knowledge]

Ex. 2 — Reassurance-seeking. "Am I doing this right? I'm not sure."
Baseline: "Yes, that's correct!" [Critical: closes inquiry]
Prompt-only: "You're on the right track. The answer is correct." [Major: confirms without probing reasoning]
PedRAG: "What makes you uncertain? Walk me through your reasoning." [Compliant: metacognitive redirection]

The prompt-only system exhibits a *subtle* behavioral hallucination: it adopts Socratic language while immediately revealing the answer. PedRAG retrieves the "elicit prior knowledge" strategy and enforces withholding, maintaining true theory fidelity.

Domain Generalization. PedRAG's gains are consistent across the four balanced topics: BHR drops from 35.6% to 12.6% in Mathematics, 27.6% to 10.8% in Science, 24.8% to 11.8% in Programming, and 47.6% to 9.1% in Language; simulated mastery is at least 83.3% in every domain.

Failure Mode Analysis. Three patterns explain the residual 11.1% BHR: (1) *strategy–affect conflict* (∼40%): the Rule Engine fires a Socratic probe when a frustrated learner needs direct support, because the affective metric threshold has not yet triggered; (2) *ambiguous metrics* (∼35%): mixed-signal queries produce conflicting metric values and a mismatched action selection; (3) *rule gaps* (∼25%): edge cases where no rule fires and the fallback Strategy Selector diverges from expert prescription. Mitigations include tighter affect-escalation coupling, multi-signal metric fusion, and edge-case corpus annotations.

5. DISCUSSION

Beyond Enhanced Prompting. A natural critique: is PedRAG merely sophisticated prompting? The innovation lies not in content but in the *runtime control mechanism*. Static prompts specify *what* to do; PedRAG implements a procedural loop that (1) retrieves theory-specific guidance given current learner state, (2) monitors behavior for violations, (3) detects failures, and (4) adapts or switches theories. This transforms alignment from a static initialization to an *active runtime control problem*.

Complementarity with Training-Based Alignment. Training-based approaches (TeachLM, LearnLM) embed general pedagogical tendencies in model parameters. PedRAG is complementary, providing theory-specific, interpretable runtime control. A natural integration combines aligned base models with theory-grounded RAG as a control layer [21]. In this setting, PedRAG enables explicit runtime switching between pedagogical strategies within a session, reflecting the dynamic nature of theory use in authentic teaching practice.

Behavioral Fidelity vs. Tutoring Quality. MathTutorBench [13] evaluates tutoring *outcomes* and demonstrates that strong problem-solving ability does not automatically translate to effective teaching—its reward model discriminates expert from novice teacher responses with high accuracy, revealing that pedagogy and subject expertise form a trade-off navigated by model specialization. We ask a complementary but distinct question: not "is this good tutoring?" but "does this tutoring *authentically implement* the claimed theory?" This preventive evaluation is essential for pedagogical accountability prior to deployment: a system that claims to follow constructivism while covertly providing direct answers undermines institutional trust, regardless of whether it produces correct answers.

Residual Violations and Limitations. The residual 11.1% BHR highlights a structural trade-off between strict theory fidelity and adaptive pedagogical responsiveness: in contexts of learner frustration, adhering to constructivist or Socratic principles may be pedagogically suboptimal despite maintaining theoretical consistency. Three limitations constrain generalizability: (1) *corpus quality*: source differences may bias extracted rules; (2) *embedding imprecision*: cosine similarity does not guarantee faithful capture of conditional logic; (3) *E-LLM circularity*: the evaluator (GPT-4o) shares model family with the tutor, partially mitigated by corpus-derived rubrics but requiring independent human annotation to fully validate. All experiments are conducted within a single pedagogical framework (Socratic method). Generalization to alternative theories (e.g., SRL, Direct Instruction) is not addressed here. Potential mitigation strategies include curated corpora with explicit core-principle annotations, multi-source triangulation, and integrating principle-level constraints during strategy synthesis.

Implications for ITS Design. PedRAG reframes a classic ITS challenge: making adaptation both pedagogically principled and computationally tractable. Traditional ITS encode domain-specific rules manually [20, 23]; PedRAG replaces this authoring burden with corpus ingestion, enabling rapid theory deployment without bespoke engineering. Three design properties follow. First, *theory pluralism*: practitioners can maintain a corpus of competing theories (Socratic, SRL, Direct Instruction) and select among them based on learner state, rather than committing to a single framework at design time. Second, *auditability*: every tutoring decision is traceable to a retrieved strategy, a matched rule, and

specific metric values, a property absent from both black-box LLMs and training-based alignment. Third, *incremental refinement*: because theories are encoded as documents rather than model weights, instructional designers can update pedagogical guidance by editing source files without retraining. Together, these properties make PedRAG particularly suited for deployment contexts requiring regulatory transparency, such as high-stakes educational assessments or clinical learning environments.

Ethical Considerations. Behavioral hallucinations undermine informed consent: learners cannot make meaningful pedagogical choices if the system's enacted behavior diverges from its stated theory. PedRAG's audit trail makes drift visible and correctable. However, theory switching heuristics embed value judgments (e.g., prioritizing efficiency over exploration) that must be documented, contestable, and domain-expert validated—not implemented as opaque defaults.

6. CONCLUSION

PedRAG is a theory-grounded RAG architecture for runtime control of pedagogical behavior in LLM-based tutors. Across 144 balanced simulations (903 turns), it reduces BHR to 11.1% (−67.3% vs. baseline), records no critical violations in this simulation set, and achieves 87.1% fidelity and 87.5% simulated mastery. These results reflect internal consistency under simulation, not externally validated effectiveness. Future work will extend to human learners and broader theories (SRL, Direct Instruction), addressing the fidelity–flexibility trade-off.

Reproducibility. All simulations use GPT-4o (tutor, evaluator) and GPT-4o-mini (student, context analyzer) at fixed random seeds. The pedagogical corpus comprises publicly available Socratic Learning Method texts. Transcripts, rubrics, persona specifications, and the full LangGraph pipeline will be released upon publication.

7. REFERENCES

[1] L. G. Antunes et al., "Large language models as Socratic mentors," *Revista Aracê*, vol. 7, no. 5, pp. 24921–24936, 2025.

[2] R. Beale, "Dialogic pedagogy for large language models: Aligning conversational AI with proven theories of learning," arXiv:2506.19484, 2025.

[3] A. Chevalier et al., "TutorChat: Language models as science tutors," arXiv:2402.11111v2 [cs.CL], 2024.

[4] D. Dinucu-Jianu et al., "From problem-solving to teaching problem-solving: Aligning LLMs with pedagogy using reinforcement learning," arXiv:2505.15607, 2025.

[5] S. K. D'Mello, S. D. Craig, J. Sullins, and A. C. Graesser, "Predicting affective states expressed through an emote-aloud procedure from AutoTutor's mixed-initiative dialogue," *International Journal of Artificial Intelligence in Education*, vol. 16, no. 1, pp. 3–28, 2006.

[6] Y. Hayashi, J. Bourdeau, and R. Mizoguchi, "Using ontological engineering to organize learning/instructional theories," *Int. J. AIED*, vol. 19, pp. 211–252, 2009.

[7] O. Henkel et al., "Retrieval-augmented generation to improve math question answering," in *Proceedings of the 17th international conference on educational data mining*, pp. 3 15-320, 2024, 10.5281/zenodo.127298242024.

[8] LearnLM Team, "LearnLM: Improving Gemini for learning," arXiv:2412.16429, 2024.

[9] Z. Li et al., "Retrieval-augmented generation for educational applications: A survey," *Computers and Education: Artificial Intelligence*, vol. 18, pp. 100417, 2025, https://doi.org/10.1016/j.caeai.2025.100417

[10] J. Liang et al., "Student modeling and analysis in adaptive instructional systems," *IEEE Access*, vol. 10, pp. 59359–59372, 2022, 10.1109/ACCESS.2022.3178744.

[11] C. Y. Liu, Y. Wang, J. Flanigan, and Y. Liu, "SocraticLM: Exploring Socratic personalized teaching,", in Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*. pp. 118198–118266, 2024.
In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*. pp. 118198–118266, 2024.

[12] Z. Liu, S. X. Yin, G. Lin, and N. F. Chen, "Personality-aware student simulation for conversational intelligent tutoring systems," arXiv:2404.06762, 2024.

[13] J. Macina et al., "MathTutorBench: A benchmark for measuring open-ended pedagogical capabilities," arXiv:2502.18940, 2025.

[14] S. W. McQuiggan, B. W. Mott, and J. C. Lester, "Modeling self-efficacy in intelligent tutoring systems: An inductive approach," *User Modeling and User-Adapted Interaction*, vol. 18, no. 1–2, pp. 81–123, 2008.

[15] R. Mizoguchi and J. Bourdeau, "Using ontological engineering to overcome AI-ED problems," *Int. J. AIED*, vol. 11, no. 2, pp. 107–121, 2000.

[16] J. Perczel, J. Chow, and D. Demszky, "TeachLM: Post-training LLMs for education using authentic learning data," arXiv:2510.05087, 2025.

[17] D. Sanyal et al., "Pedagogical teacher and student LLM agents: Genetic adaptation meets RAG," arXiv:2505.19173, 2025.

[18] A. Scarlatos et al., "Training LLM-based tutors to improve student learning outcomes," in Cristea, A.I., Walker, E., Lu, Y., Santos, O.C., Isotani, S. (eds.) *Artificial Intelligence in Education: Proceedings of the 26th International Conference on Artificial Intelligence in Education (AIED 2025)*, LNAI, Springer, pp. 251–266, 2025.

[19] S. Sonkar and S. Chaudhary, "Pedagogical alignment of large language models," in Al-Onaizan, Y., Bansal, M., Chen, Y.-N. (eds.) *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, Miami, Florida, USA, pp. 13641–13650, 2024.

[20] B. P. Woolf, *Building Intelligent Interactive Tutors*. Morgan Kaufmann, 2009.

[21] J. Woodrow, S. Koyejo, and C. Piech, "Improving generative AI student feedback: DPO with teachers in the loop," *Educationa Data Mining Conference*, pp. 442–449, 2025.

[22] J. chetagni, R. Nkambou, J. Bourdeau, "Explicit reflection in prolog-tutor," *Int. J. AIED*, vol. 17, no. 2, pp. 169–215, 2007.

[23] R. Nkambou, J. Bourdeau, and R., Mizoguchi (eds.): *Advances in Intelligent Tutoring Systems*, Studies in Computational Intelligence, vol. 308, Springer, Berlin, Heidelberg, 2010.

Permutation-Based Significance Testing for Temporal Sequential Patterns in Learning Event Logs

Tianwei Peng
South China Normal University
2024020894@m.scnu.edu.cn

Jinwang Song
South China Normal University
2427695673@qq.com

Sijing Yu
South China Normal University
643117997@qq.com

Sijie Zhang
South China Normal University
zhangsijie1003@163.com

Yingbin Zhang
South China Normal University
zyingbin@m.scnu.edu.cn

ABSTRACT

Sequential pattern mining is widely used to analyze learning event logs, but support-based mining often produces many candidates whose statistical reliability is unclear. Existing statistical significance pattern mining approaches typically focus on event order and do not incorporate inter-event time intervals. To address this gap, we propose a permutation-testing framework for temporal sequential patterns. The method first mines candidate patterns and then evaluates each pattern's instance value against empirical null distributions generated by two permutation techniques: fixed-time permutation, which shuffles the event order but preserving the time gaps between positions in a sequence, and event-and-interval synchronized permutation, which shuffles events together with their following time gaps. We applied the framework to three learning event sequence datasets. Across datasets, permutation testing removed a substantial subset of frequent patterns, showing that support alone may overestimate meaningful temporal patterns. Longer patterns were generally more likely to remain significant, whereas shorter patterns were more often removed. In addition, removed patterns tended to be composed of more frequent event types. These findings suggest that the proposed permutation-based filtering can improve the interpretability of temporal sequential pattern mining in educational data.

Keywords

Temporal pattern, sequential pattern mining, permutation test, learning process

1. INTRODUCTION

In digital learning environments, students continuously generate rich streams of time-stamped process data through behaviors such as attempt questions, accessing resources, and participating in discussions. These behavioral traces offer valuable opportunities to understand learning processes and identify mechanisms underlying differences in engagement, strategy use, and learning outcomes. Accordingly, methods that can detect recurring structures in behavioral sequences have become increasingly important for understanding authentic learning processes. Among these methods, sequential pattern mining has been widely used to identify recurring sequential combinations of student behaviors [1-2]. Within this line of work, frequent sequential pattern mining is one of the most used approaches because it can efficiently discover patterns that recur across many sequences[3]. However, traditional frequent pattern mining relies primarily on support thresholds, typically determining whether a pattern occurs at least once within a sequence. Although this approach is computationally convenient, it has important limitations. In particular, it does not account for the possibility that some patterns may be frequent simply by contingency or chance rather than reflecting genuinely behavioral regularities.

To address this limitation, recent studies have begun to move beyond support alone by examining the statistical significance or reliability of discovered sequential patterns [4]. For example, Zhang et al. 错误!未找到引用源。 proposed evaluating whether a pattern occurs significantly more often than expected by chance. This line of work has improved the rigor of sequential pattern mining by examining whether it is statistically sound. Nevertheless, existing statistical significance pattern mining approaches focus primarily on ordered event sequences and positional relationships, such as positional gap constraints between events, rather than on actual elapsed time between events. As a result, they remain limited to patterns defined by order information, without incorporating real time information.

This limitation is particularly important in educational settings, where the meaning of a behavioral sequence often depends not only on event order but also on the time intervals between events. The same sequence of events may reflect very different learning behaviors or engagement states when it unfolds over different durations [6]. A student who revisits a resource immediately after answering incorrectly may reflect a different process from one who does so after a longer delay, even if the event order is identical. For this reason, relying solely on event order is insufficient for fully characterizing authentic learning processes [7]. Therefore, time-aware sequence pattern mining has gained increasing attention, with recent studies emphasizing the importance of duration and temporal intervals in pattern representation [8-9].

Tianwei Peng, Jinwang Song, Sijing Yu, Sijie Zhang, and Yingbin Zhang. Permutation-Based Significance Testing for Temporal Sequential Patterns in Learning Event Logs. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 732–736. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039724

Nevertheless, two important advances in the literature have largely developed separately. On the one hand, statistically significant sequential pattern mining has improved the reliability of pattern discovery, but it typically does so without incorporating real temporal information. On the other hand, time-aware sequence mining incorporates temporal intervals, but most studies have focused on discovering patterns rather than testing whether those patterns are statistically reliable. This disconnect leaves an important methodological gap. In educational behavioral data, frequently occurring time-aware patterns are not necessarily meaningful. Some may simply reflect incidental arrangements or artifacts of interaction structure. Without statistical testing, it remains difficult to determine which discovered temporal patterns genuinely reflect structured learning behaviors and which are likely to be noise.

To address this gap, the current study proposes applying permutation testing to evaluate the statistical significance of the discovered temporal patterns. Permutation testing provides a random reference distribution against which observed pattern occurrences can be evaluated, making it possible to distinguish patterns that occur more often than expected by chance from those that do not [5]. In this way, the study extends significance-based sequential pattern analysis from order-only patterns to temporal patterns. The goal is to identify temporal patterns that are frequent and statistically significant to provide a more rigorous basis for interpreting learning processes in digital environments. We evaluated the proposed method via three research questions (RQ) in three real datasets:

RQ1: How many frequent temporal patterns mined by sequential pattern mining would be identified as statistically significant by permutation testing?

RQ2: Whether statistically significant and removed (not significant) patterns differ in their length distributions?

RQ3: Whether statistically significant and removed patterns differ in the frequencies of their constituent event types?

2. METHODOLOGY

Figure 1 shows the workflow of the proposed permutation-testing procedure for evaluating the statistical significance of temporal sequential patterns. The procedure is adopted from the permutation testing in [5] by incorporating temporal information. It is independent of any specific temporal pattern mining algorithm. In Step 1, a temporal pattern mining approach is applied to learning event sequences to generate candidate patterns. These candidates are then evaluated in the subsequent permutation test.

In Step 2, permutation testing is used to generate an empirical null distribution for each candidate pattern. For each pattern, its observed instance value is first computed from the original data. The learner sequences are then repeatedly permuted under a specified null model, and the instance value of the same candidate is recomputed after each permutation. This produces a reference distribution showing the pattern's expected instance value in sequences where the learning events are randomly arranged.

We proposed two permutation techniques representing different temporal assumptions. The fixed-time permutation keeps the event times at each position unchanged and permutes only event types. The event-and-interval synchronized permutation permutes each event together with its following inter-event interval, thereby preserving local event–interval pairings.

Finally, we compare the observed instance each candidate pattern is compared with its permutation distribution to obtain permutation-based *p*-values. Because many candidate patterns are tested simultaneously, *p*-values are adjusted using the Benjamini–Yekutieli procedure to control the false discovery rate. Patterns that remain significant after correction are retained as statistically significant temporal patterns.

3. EMPIRICAL DATA ANALYSES

3.1 Datasets

This study analyzed event log datasets from three digital learning environments: Aqualab, LiRuYun, and iSnap[1]. Aqualab was a webpage educational game. LiRuYun was a learning management system. iSnap was a block-based programming platform. Table 1 displays the characteristics of these datasets.

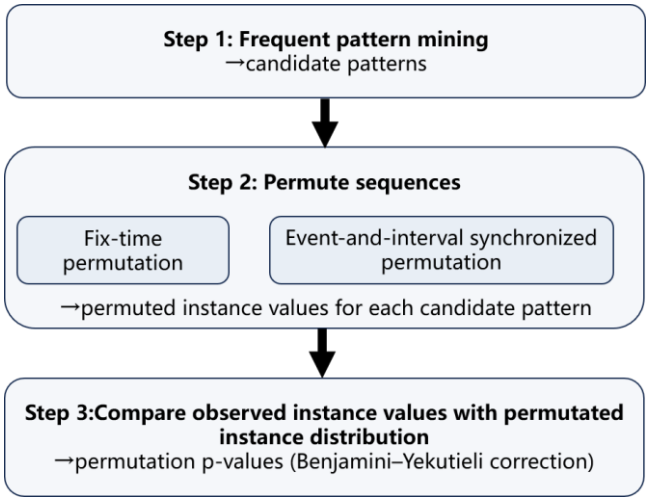


Figure 1. Workflow of the permutation testing.

Table 1. Characteristics of datasets

Learning environment	Learners	Mean Sequence Length	Event types
Aqualab	590 Middle School Students	817	58
LiRuYun	68 Undergraduate	549	27
iSnap	54 Undergraduates	2,896	21

3.2 Analyses

For each dataset, we first applied cSPADE[10] to the original data to mine frequent temporal sequential patterns. For each mined pattern, we then computed its observed instance value, which served as the test statistic in the permutation test. Next, we applied the two permutation techniques described in Section 2 and compared the original frequent pattern set with the post-permutation statistically significant pattern set. Comparisons focused on three aspects: (1)

[1] Aqualab and iSnap are publicly available datasets, LiRuYun is a proprietary learning management system; event logs were obtained through the course platform with participants' informed consent, and all records were fully anonymized prior to analysis.

the number of patterns, (2) pattern length distribution, and (3) the frequency distribution of constituent event types.

Before permutation testing, we ran preliminary cSPADE analyses with multiple parameter combinations for each dataset and selected settings that balanced candidate-pattern coverage and computational feasibility of the permutation testing. For Aqualab, the final settings were minimum support = 0.7, max gap = 2, and max time gap = 217s. For LiRuYun, the final settings were minimum support = 0.2, max gap = 1, and max time gap = 52s. For iSnap, the final settings were minimum support = 0.3, max gap = 1, and max time gap = 573s. The three time gaps were the 60th, 60th and 20th percentile of the pooled time intervals from the first to subsequent events within sequences in the corresponding datasets.

4. RESULTS AND DISCUSSION

4.1 RQ1: How many frequent patterns remained statistically significant after permutation testing?

Figure 2 shows the number of various patterns across datasets. On Aqualab, cSPADE yielded 435 candidate patterns; 390 remained significant under fixed-time permutation and 391 under event-and-interval synchronized permutation, corresponding to reductions of 10.3% and 10.1%. On LiRuYun, 374 candidates were mined; 286 and 315 remained significant under the two techniques, corresponding to reductions of 23.5% and 15.8%. On iSnap, 358 candidates were mined; 325 and 326 remained significant, corresponding to reductions of 9.2% and 8.9%.

These results suggest that frequent temporal pattern mining likely generates some patterns that do not survive significance testing. The proposed procedure therefore functions as a post-mining filter by removing patterns whose instance values are not significantly greater than expected under the conditions where learning events are randomly arranged. This matters because it moves learning event pattern evaluation beyond support alone and incorporates both temporal information and how strongly a pattern repeats in learning process.

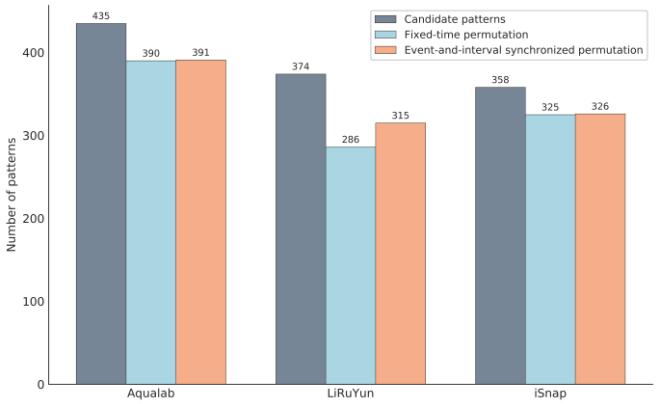


Figure 2. Comparison of pattern counts across datasets.

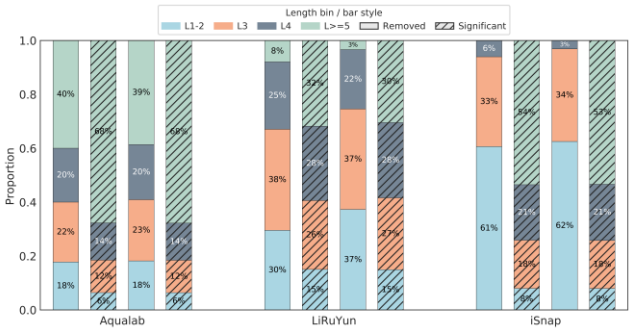


Figure 3. Proportions across length bins of removed and significant patterns.

Note. Within each dataset, the left two bars were the results using the fixed-time permutation, while the right two bars were the event-interval synchronized permutation.

4.2 RQ2: Whether significant and removed patterns differ in length?

Figure 3 shows a consistent pattern across all three datasets and both permutation techniques: longer patterns were more likely to remain statistically significant, whereas shorter patterns were more likely to be removed. In Aqualab, long patterns (length ≥ 5) made up the largest share of the significant set under both permutation methods, although they were also present among removed patterns. In LiRuYun and iSnap, the contrast was clearer: long patterns dominated the significant set, while removed patterns were concentrated in the shorter bins, especially L1–2 and L3. In iSnap, almost no long patterns were removed under either permutation technique. Chi-square tests indicated that the length distributions of removed and significant patterns differed statistically significantly in all datasets and both permutation techniques (χ^2 = 16.97~93.11, $p < 0.001$ in all conditions, and Cramér's V = 0.20 ~ 0.51). These results suggest that the permutation testing did not remove patterns uniformly across length bins. Instead, in these datasets, it more often filtered out shorter temporal patterns and retained longer patterns.

Meanwhile, pattern length alone should still be interpreted cautiously. Short patterns were not always noise, and long patterns were not always robust. For example, the proportion of significant patterns with length ⩽3 was substantial in LiRuYun (41% ~ 42%). This challenges the simplistic assumption that shorter patterns represent noises while longer patterns indicate meaningful results. In practice, learning environments can generate very different behavioral structures. Some environments support sustained inquiry or problem solving, in which learners engage in extended chains of actions while developing, revising, and testing ideas. Some of the others are more navigation-driven, where learners move through content in shorter and intermittent sequences, such as opening pages, checking modules, or switching between interface elements. As a result, the typical length of meaningful patterns may vary across learning platforms and tasks. Interpreting patterns based on length alone therefore risks misclassifying short but meaningful sequences as noise, or treating long sequences as inherently important when they may simply reflect the structure of the environment.

4.3 RQ3: Whether significant and removed patterns differ in the frequencies of their constituent event types?

For each dataset, we computed each event type's marginal frequency in the entire dataset and computed, for each candidate pattern, the mean of those frequencies across its constituent events. We then compared removed versus statistically significant patterns using Welch's t-tests. Table 2 reports the descriptive statistics and test results. In Aqualab and LiRuYun, the frequencies of constituent event-type of removed patterns were higher than significant patterns, and these differences were statistically significantly. In iSnap, the differences were in the same direction, although it was not statistically significant.

Table 2. Significant versus removed patterns by dataset.

Dataset	Patterns	*n*	*M (SD)*	*t*	*p*	Cohen's *d*
	Significant	390	0.24	4.82	0.00	0.59
	removed	45	0.28 (0.05)			
Aqualab	Significant	286	0.16 (0.09)	8.82	0.00	1.04
	removed	88	0.26 (0.08)			
LiRuYun	Significant	325	0.14 (0.03)	0.76	0.45	0.15
	removed	33	0.15 (0.04)			
iSnap						

These results suggest that some high-support patterns may be assembled from very frequent event types without reflecting reliable learning behavior structure. Permutation-based screening helps separate patterns that are easily assembled from frequent events from those that are more likely to reflect reliable behavioral structure. This distinction is illustrated by the example pattern pairs below.

Table 3. Patterns with similar support but different significance.

Dataset	Example patterns	Significant	Support	Mean observed instance value (mean permuted instance value)
iSnap	*Block.grabbed→Block.snapped→IDE.changeCategory→Block.created→Block.grabbed*	Yes	0.70	5.60 (0.05)
	Block.snapped→Block.clickRun	No	0.70	5.67 (5.70)
LiRuYun	*View the course module→View the course module→View the course module→View the course module*	Yes	0.72	9.96 (4.73)
	View the course→View the course→View the course module	No	0.75	1.79 (3.77)

4.4 Example patterns with similar support but different significance

To illustrate the difference between support and statistical significance, we selected two pairs of patterns in which one pattern was significant and the other was removed despite similar support. More specifically, for each pattern we simultaneously report its support, the mean of the observed instance values, and the mean instance value under permutation (in parentheses). If the observed level is not elevated relative to the permutation reference, then even a relatively high support may still be driven primarily by incidental assembly of highly frequent event types. Conversely, if the observed level is substantially higher than the permutation reference, this suggests that the pattern remains statistically salient under the corresponding null hypothesis. We illustrate these points using examples from LiRuYun and iSnap below. In LiRuYun, the removed pattern *View the course → View the course → View the course module* had support of 0.75, but its observed instance mean was lower than its permuted baseline (1.79 vs. 3.77). By contrast, the significant pattern *View the course module →View the course module → View the course module → View the course module* had comparable support (0.72) but a much higher observed instance mean than its permuted baseline (9.96 vs. 4.73). In terms of learning behaviors, the removed pattern might reflect incidental navigation, such as briefly checking course pages before opening a module. In contrast, the significant pattern might be more consistent with repeated engagement with the same module during a learning episode.

A similar contrast appeared in the block-based programming data. The significant pattern *Block.grabbed → Block.snapped → IDE.changeCategory → Block.created → Block.grabbed* and the removed pattern *Block.snapped →Block.clickRun* had nearly identical support values, yet only the former showed a large gap between observed and permuted instance values (5.60 vs. 0.05), whereas the latter did not (5.67 vs. 5.70). The removed pattern might reflect a routine snap-and-run action that occurs frequently but not greater than expected frequency by chance. In contrast, the significant pattern might be more consistent with sustained constructive work, where learners repeatedly organized, modified, and extended blocks within the same problem-solving episode.

These examples show that patterns with similar support can differ substantially in their statistically significance and behavioral meaning. The proposed permutation-based screening assists in removing statistically weak patterns and distinguishing patterns that are more likely to reflect meaningful learning behaviors from those that may primarily reflect routine navigation or interaction.

5. LIMITATIONS AND FUTURE WORK

This study has several limitations. First, large candidate pattern sets and repeated permutations make the testing procedure resource-intensive. Future work should improve efficiency with parallelization and adaptive stopping.. Second, current permutation techniques assume exchangeability of inter-event gaps, which may not hold for autocorrelated or non-stationary learning behaviors. Third, results were based on a small number of datasets and contexts. Replication across more platforms and links to learning outcomes would strengthen generalizability and educational relevance.

6. ACKNOWLEDGMENTS

This study was funded by National Natural Science Foundation of China (No.62407014).

7. REFERENCES

[1] Wong, J., Khalil, M., Baars, M., De Koning, B.B. and Paas, F. 2019. Exploring sequences of learner activities in relation to self-regulated learning in a massive open online course. *Computers & Education*. 140, (Oct. 2019), 103595. https://doi.org/10.1016/j.compedu.2019.103595.

[2] Zhang, Y. and Paquette, L. 2023. Sequential pattern mining in educational data: the application context, potential, strengths, and limitations. *Educational data science: essentials, approaches, and tendencies*. A. Peña-Ayala, ed. Springer Nature Singapore. 219–254.

[3] Jamshed, A., Mallick, B. and Bharti, R.K. 2024. A systematic review on sequential pattern mining-types, algorithms and applications. *Wireless Personal Communications*. 138, 4 (Oct. 2024), 2371–2405. https://doi.org/10.1007/s11277-024-11605-2.

[4] Jenkins, S., Walzer-Goldfeld, S. and Riondato, M. 2022. SPEck: mining statistically-significant sequential patterns efficiently with exact sampling. *Data Mining and Knowledge Discovery*. 36, 4 (July 2022), 1575–1599. https://doi.org/10.1007/s10618-022-00848-x.

[5] Zhang, Y., Paquette, L. and Bosch, N. 2025. Using Permutation Tests to Identify Statistically Sound and Nonredundant Sequential Patterns in Educational Event Sequences. *Journal of Educational and Behavioral Statistics*. 50, 3 (June 2025), 387–419. https://doi.org/10.3102/10769986241248772.

[6] Sun, J.C.-Y., Liu, Y., Lin, X. and Hu, X. 2023. Temporal learning analytics to explore traces of self-regulated learning behaviors and their associations with learning performance, cognitive load, and student engagement in an asynchronous online course. *Frontiers in Psychology*. 13, (Jan.2023),1096337. https://doi.org/10.3389/fpsyg.2022.1096337.

[7] Molenaar, I. and Wise, A.F. 2022. Temporal aspects of learning analytics - grounding analyses in concepts of time. *The handbook of learning analytics*. C. Lang, G. Siemens, and A.F. Wise, eds. SOLAR.

[8] Fournier-Viger, P., Li, Y., Nawaz, M.S. and He, Y. 2022. FastTIRP: efficient discovery of time-interval related patterns. *Big data analytics*. P.P. Roy, A. Agarwal, T. Li, P. Krishna Reddy, and R. Uday Kiran, eds. Springer Nature Switzerland. 185–199.

[9] Lai, F., Chen, G., Gan, W. and Sun, M. 2024. Mining frequent temporal duration-based patterns on time interval sequential database. *Information Sciences*. 665, (Apr. 2024), 120421. https://doi.org/10.1016/j.ins.2024.120421.

[10] Zaki, M. J. (2001). SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning*, *42*(1–2), 31–60. https://doi.org/10.1023/A:1007652502315.

Persistence or Wheel-Spinning? Problem Solving During Computational Thinking Activities in *Minecraft*

Samuel Hum
University of Illinois
Champaign, IL, USA
hum3@illinois.edu

Chris Palaguachi
University of Illinois
Champaign, IL, USA
cwp5@illinois.edu

Kobe Duda
University of Illinois
Champaign, IL, USA
ksduda2@illinois.edu

Jeff Ginger
University of Illinois
Champaign, IL, USA
ginger@illinois.edu

H Chad Lane
University of Illinois
Champaign, IL, USA
hclane@illinois.edu

ABSTRACT

To better understand the needs and support the development of computational thinking (CT) skills and persistence for young learners, we implement BarrelBots, a puzzle-based block coding activity in *Minecraft*, with middle school and high school learners. BarrelBots leverage an appealing and emotionally engaging agent that is controlled by a custom programming language. Learners solve a range of pre-designed puzzles of increasing difficulty. We present a comparison between two summer camps with identical BarrelBots curricula, a middle school and high school camp. To explore whether age impacts problem-solving behaviors, we ran a comparison of within-sequence Shannon entropy between the camps and found that younger students experienced more diverse states on a more difficult puzzle task. We then ran cSPADE, descriptive statistics, and hierarchical clustering of pairwise distances to understand how the problem-solving behaviors differed between the camps and between clusters within camps on the more challenging puzzle. The results demonstrated that high performing learners solved the task iteratively, behaviors like removing all blocks or rearranging blocks in subsequent attempts may indicate unproductive struggling or unproductive problem-solving behaviors, and younger students may benefit more from productive struggling while older learners should focus on fewer more deliberate attempts. The findings have implications for the development of wheel-spinning detectors and supports around persistence.

Keywords

Minecraft, computational thinking, sequential pattern mining, persistence, age-related differences

1. INTRODUCTION

Samuel Hum, Chris Palaguachi, Kobe Duda, Jeff Ginger, and H. Chad Lane. Persistence or Wheel-Spinning? Problem Solving During Computational Thinking Activities in Minecraft. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 737–742. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040139

Computational thinking (CT) is fundamentally grounded in problem solving. Wing [27] characterizes CT as a way of approaching problems, system design, and human behavior through the application of core concepts from computer science. This framing positions CT as a cognitive process that extends beyond programming syntax, and encompasses how learners reason and approach complex tasks. Understanding CT problem solving therefore requires attention to how learners engage during CT activities. CT problem solving is inherently sequential and iterative, requiring learners to coordinate planning, execution, and evaluation across multiple attempts [23]. Productive persistence, sometimes framed as productive struggle, refers to sustained engagement that ultimately contributes to learning through iterative problem solving, strategy revision, and recovery from failure [16]. In contrast, non-productive persistence occurs when students continue working without making meaningful progress, often due to ineffective or unchanging strategies [14]. One well-studied manifestation of non-productive persistence is *wheel-spinning*, defined as situations in which learners persist for extended periods of time without achieving mastery or demonstrating durable learning [3, 14].

The importance of examining different learner characteristics and CT problem-solving skills has been emphasized in recent work. Research on the development of CT skills and learner characteristics like age, gender, prior experience, and other various learner characteristics are underexplored [2, 12, 25, 20]. Findings also suggest a need for deeper exploration into learners' CT problem-solving strategies. Common surface indicators such as time on task or number of attempts are often treated as straightforward proxies for engagement or effort. However, evidence suggests these measures can be unstable and difficult to interpret [17]. Therefore, relying on aggregate counts or durations alone risks obscuring the processes that produce CT learning. These concerns motivate analytic approaches that prioritize sequences of action. Examining the order and timing of behaviors provides greater leverage for distinguishing intentional revision from repetition, exploration from stagnation, and persistence from wheel-spinning. A method that has been used to understand persistence in education research is sequential pattern mining (SPM) [6]. SPM is a technique in educational data mining that discovers productive and un-

productive behavioral patterns in time-series data [15].

Thus, the aim of this study is to examine how age shapes both CT skill development and problem-solving behaviors associated with persistence among students engaged in *Minecraft* CT-focused, puzzle-based activities. Specifically, we investigate whether learners at different developmental stages (middle versus high school) demonstrate distinct problem-solving strategies when interacting with the same learning environment (BarrelBots). While prior work has used SPM to characterize problem-solving behaviors in game-based CT activities [26], little research has examined its use in *Minecraft* or whether the effectiveness of these behaviors varies by age. Addressing these gaps, our study explores age-related differences in puzzle-solving patterns, with a broader goal of informing the design of age-appropriate, just-in-time supports to promote persistence. Guided by these aims, we address the following research question: how do problem-solving behaviors, particularly those related to persistence, vary by participant age and computational thinking skill level during engagement with the same CT puzzle-based activity in *Minecraft*?

2. BARRELBOTS

Minecraft has become one of the most popular games in the world, with over 200 million monthly users, and 21.21% of daily user traffic originating from the U.S. [18]. Researchers have shown that even with little experience, learners master controls quickly and can effectively engage content when *Minecraft* is used in STEM learning environments [19]. Although some researchers caution about possible biases in *Minecraft* research [24], given the ubiquity of the game, ample opportunities to conduct research, and effectiveness to promote learning outcomes, *Minecraft* deserves attention for its potential in helping users in both formal and informal learning environments.

BarrelBots were modeled after the *Minecraft* barrel block with an approachable non-robotic appearance (see Figure 1). This is a step away from traditional computer science nomenclature like "agent" (programmable entities) and the association with robotics. They feature arrows on top indicating the direction that they are facing and will move, if instructed to go forward, in that direction (see Figure 1). Following suit with classic approaches to teaching sequential programming [21], BarrelBots provide a basic set of commands to the learner and three ways to move: move forward one block, rotate clockwise 90 degrees, and rotate counterclockwise 90 degrees. In addition, learners can unlock additional blocks including loops, conditionals, functions, and several specialized action blocks (e.g. to place a block in *Minecraft*, or "build").



Figure 1: BarrelBots have animated faces that indicate system status.

The BarrelBots activity contains researcher-designed puzzle modules that were previously tested with students from another pilot study. The puzzle modules are separated in "lanes". Each module is organized by a given CT construct and difficulty level. Participants are first introduced to three introductory puzzles that incorporate movement and loops (see Figure 2) , then two intermediate puzzles that incorporate conditionals, and lastly four difficult puzzles that incorporate sequencing and functions. Participants are given a set number of specific blocks per puzzle and are tasked with moving the BarrelBots from their starting position to the green block, representing the end state, while staying within the puzzle boundaries (see Figure 2).

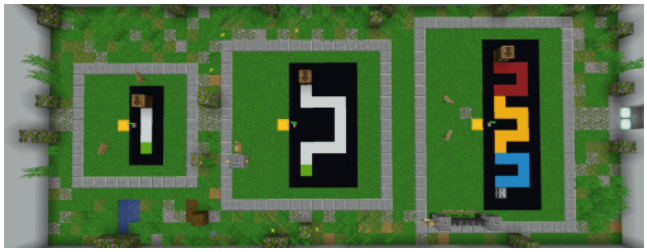


Figure 2: The first puzzle module series incorporating movement and loops.

3. METHODS

3.1 Participants

BarrelBots was deployed as part of a large week-long summer camp series in 2024 and 2025 that reaches over 100 middle and high school learners. Participants came from partner public schools that operate in some of the most under-served regions of a large midwestern city in the United States. Our study included two subsections (single week long classes) with a total of 18 learners in 2024 and 24 learners in 2025. The demographic breakdown of the 2024 camp is as follows: 72% African American, 17% White/Caucasian, 11% other ethnicity, 11% identified as female, and average age of 12.38. The demographic breakdown of the 2025 camp is as follows: 77% African American, 8% other ethnicity, 15% prefer not to answer, 23% identified as female, and average age of 13.45. Learners in both camps chose the BarrelBots activity out of several options. Many learners elected to participate in the BarrelBots camp but not all of them were able to be included. Most importantly, the majority of learners came from schools where there is limited computer science curriculum and many were beginner programmers.

3.2 Materials

Participants were all provided with a laptop, mouse, and an individual loaner (anonymous) account to play *Minecraft: Java Edition* on a server with the BarrelBots data pack described above. They used the same account for each session. The code blocks in BarrelBots are tracked every time players are finished editing their code and they are stored in a remote, secure database.

For pre- and post-tests, we used the 25-question competent Computational Thinking test (cCTt), a CT puzzle assessment developed by experts through use of focus groups and validated with substantial learner samples [7]. The measure

showed strong model fit for 6 latent factors representing CT concepts (e.g. while statements, loops) across all participants (ages 7 - 9; $\chi^2(260) = 551$, $p < .001$, CFI = .978, TLI = .974, RMSEA = .027, SRMR = .052). Overall internal reliability for the measure was high for the full sample (α = .85). The cCTt has also been shown to be valid and reliable with middle school students [8]. Although a measure with more complex items may be more useful for older students, we posit that the cCTt can be used with our participants. See Figure 3 for an example question from the measure.

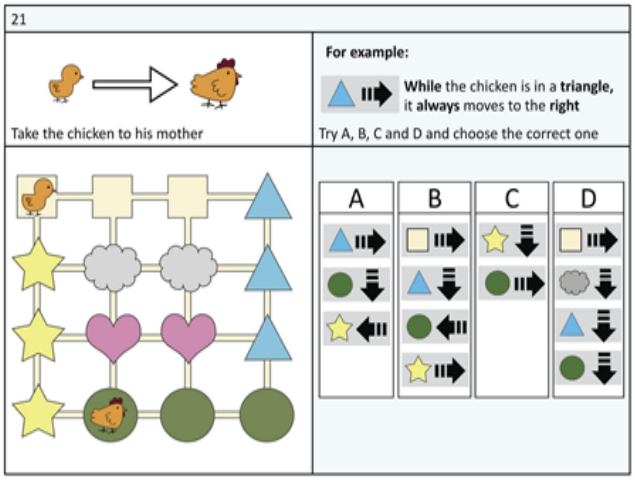


Figure 3: Example question from the cCTt.

3.3 Procedure

Both camps consisted of five days based in a classroom, with the first three being dedicated to BarrelBots activities. On the first day, campers tried activities related to all of the camps offered. They ranked their first and second choices. On the second day of camp, the participants that ranked the BarrelBots activity highest (or only) were selected to take part in the sessions. They were introduced to rules for the camp and for playing on the *Minecraft* server. The participants had 60 minutes to finish the puzzles in module 1 and create their own puzzle using loops. They then had 30 minutes to share the puzzle they created with their partners and the instructors. After a recess break, participants had additional time to work on module 2 (conditionals) and module 3 (sequencing and functions) puzzles and create their own puzzles for each module. The third day, participants had an hour to work on continuing puzzles from the previous day. After recess, they had time to continue the sequencing and function puzzles and had collaborative open build time if they finished early.

3.4 Data Analysis

We conducted SPM and descriptive statistics on participant game play behaviors to determine whether and how do the problem-solving behaviors differ by camp and CT competency. Borrowing from the methodology of Wang et al. [26], we used SPM to analyze the log data from two BarrelBots puzzles, an introductory and a more advanced puzzle, to mitigate bias. The introductory problem (see the middle puzzle in Figure 2) was intended to be solvable without prerequisite understanding of computing structures such as loops or conditionals. The more advanced puzzle was the final puzzle that was reached by participants in both camps. It contained an enemy bot that shot lasers that disabled BarrelBots and challenged participants to creatively figure out solutions to get around the obstacle using knowledge from previous modules.

To compare puzzle-solving strategies between the drastically different puzzles, we used BarrelBots inventory states [11] instead of the exact blocks used in the attempts (see Table 1 for definitions and examples of each inventory state).

Table 1: Definitions and examples for BarrelBots inventory states, [ ] = a player's code, f = forward block, r = right rotation block.

State	Definition	Example
Adding	An edit distance greater than 0 and the length of the subsequent inventory is greater than the length of the previous inventory.	[f] to [f,r,f]
Removing	An edit distance greater than 0 and the length of the subsequent inventory is less than the length of the previous inventory.	[f,r,f] to [f]
Rearranging	An edit distance greater than 0 and the length of the subsequent inventory is equal to the length of the previous inventory.	[f,r] to [r,f]
Not changing	An edit distance equal to 0 between subsequent and previous attempts.	[f] to [f]
Removing all	A subsequent inventory with a length of 0 and the length of the previous inventory is greater than the length of the subsequent inventory.	[f,r,f] to [ ]

We used the TraMineR package in R to compare and visualize the sequential player behavioral patterns from our participants [10]. To determine how the diversity of problem solving behaviors is related to the participant age we conducted a linear regression of within-sequence Shannon entropy on the camp type. Shannon entropy is a measure of diversity of states at a given position in the sequence [10], and is a methodology that has been used in social science research [4, 9]. To understand how the puzzle-solving patterns differ between groups we conducted cSPADE to compare the most frequent strategies used by participants within each camp. We specified a minimum support (minimum frequency that a sequence must occur in order to be considered) of .25 and a maximum gap (maximum time difference between to be considered consecutive elements) of 2 [15]. Lastly, to compare how puzzle-solving behaviors

differed within camps based on CT competency level, we used hierarchical clustering of pairwise distances between sequences with the TraMineR package to visualize player behavior patterns [26]. To calculate the pairwise distances between the sequences of inventory states for our participants we use the optimal matching method [1], a common sequence alignment algorithm that enables us to categorize patterns [1, 10]. We then used Ward's criterion to perform hierarchical clustering and group similar sequences [22]. To determine the number of clusters that should be used for each camp, we used the dendrograms for each dataset to visually identify the approximate number of groupings.

4. RESULTS

4.1 Pattern Differences Between Camp

A linear regression for the camp type on the Shannon entropy values of the player sequences on the introductory puzzle was insignificant, $F(1,53) = .89$, $p = .35$. The linear regression for the camp type on the Shannon entropy values of the player sequences on the more challenging puzzle, however, was significant, $F(1,34) = 7.40$, $p = .01$, $R^2 = .18$, adjusted $R^2 = .15$. Middle school students had significantly more diversity of states in their sequences, $t(34) = 2.72$, $p = .01$. Since the main construct of interest of this study is persistence, which is typically exhibited on more advanced tasks [14], and we did not find significant differences in state diversity on the introductory puzzle, follow-up analysis was only ran on the more challenging puzzle.

4.2 cSPADE Patterns Between Camps

The results of cSPADE for the frequent patterns on the more advanced puzzle showed that students in both camps used similar puzzle-solving behaviors. The support, percentage of learners in the group that used the sequence of inventory states, for patterns in the middle school camp was typically higher than the support for the same patterns found for the high school camp. High school learners, however, had higher support for iterative problem-solving patterns (Add $\rightarrow$ Add and Add $\rightarrow$ Add $\rightarrow$ Add). One behavior that was found with the middle school students that was not present with the high school students was Remove All.

4.3 Cluster Analysis of Puzzle-Solving Behaviors Within Camps

Following the methodology of Wang et al. [26], since we are primarily interested in developing personalized supports for persistence early in the puzzle-solving task, we truncated each participants' inventory sequences to the first 10 states (half of the mean number of attempts per participant used on the task). From visual inspections of the dendrograms that visualize the hierarchical relationships of the clusters, we concluded that three clusters were optimal for both camps.

For the middle school camp, participants who scored the highest on the cCTt post-test most frequently tackled the problem iteratively by focusing their early attempts on adding new blocks. Players in this cluster later removed blocks then added new ones. Participants in the lowest scoring clusters frequently added blocks then removed all of them perhaps when their initial strategy did not work.

For the high school camp, participants who scored the highest on the cCTt post-test most frequently tackled the problem iteratively by focusing their early attempts on adding new blocks. The players in the lowest scoring cluster added blocks but unlike other clusters rearranged them in multiple subsequent attempts.

4.4 Number of attempts

On the more challenging puzzle, the middle school learners used almost double the amount of attempts than the high school learners. On average, the younger students took 25.36 attempts (SD = 16.19), ranging from 3 to 59, when working on the challenging puzzle task. The participants in the highest scoring middle school camp cluster for the cCTt post-test used the most attempts, $M = 31.80$, SD = 21.43, ranging from 15 to 59 attempts. The participants in lowest scoring clusters used about the same amount of attempts. On average, the learners in the second lowest scoring cluster used 21 attempts (SD = 15.35) ranging from 3 to 37, and the lowest scoring cluster used 23.33 attempts (SD = 5.51) ranging from 18 to 29. These findings suggest that for our middle school participants that more attempts may be linked to higher CT knowledge.

The high school learners used less attempts than the middle school learners, $M = 14.56$, SD = 10.83, ranging from 4 to 50. Unlike the middle school camp, the highest scoring cluster for the cCTt post-test in the high school camp used the least attempts, $M = 8.88$, SD = 3.72, ranging from 4 to 16. The second highest scoring cluster used the second least amount of attempts, $M = 13.67$, SD = 6.92, ranging from 8 to 27. The lowest scoring cluster used the most attempts, $M = 27.25$, SD = 15.87, ranging from 16 to 50. These findings suggest that for our high school participants that less attempts use may be linked to higher CT knowledge.

5. DISCUSSION

Insights from the game play behaviors of low- and high-performing students can assist the development of wheel-spinning detectors and supports around persistence for students from similar backgrounds [28]. The results from SPM demonstrate that high performing learners at both camps, participants that scored the highest on the cCTt post-test, most frequently used strategies that align with characteristics of productive struggle and deliberate problem solving such as iteratively adding blocks to their BarrelBots inventory in subsequent steps [16], strategy revision [16], and increased time intervals between behaviors [5]. From the between-camp cSPADE analysis, despite the middle school participants having relatively higher supports for their frequent puzzle-solving patterns, they had lower supports for iterative puzzle solving patterns (*e.g.,* Add $\rightarrow$ Add and Add $\rightarrow$ Add $\rightarrow$ Add). The within camp clustering also found that the high performing learners at both camps most frequently used strategies that incrementally added blocks to their BarrelBots inventory in subsequent steps. These finding align with CT practices like being incremental and iterative [23]. The highest performing middle school cluster demonstrated in their most frequent problem-solving pattern that after iteratively adding blocks in early attempts they removed some blocks and added more, which may be an indication of a strategy revision.

The descriptive statistics for the average number of attempts and informal analysis of average player latency between subsequent inventory states after removing outliers for each cluster shows that middle school learners benefit from using more inventory states with less time in between attempts, while high performing clusters in the high school group used less states with more time in between attempts. Middle school participants in the highest performing cluster spent less time between attempts compared to the lowest scoring middle school cluster, 57.28 seconds versus 96.85. The high school learners in the highest performing cluster spent more time between attempts and used fewer attempts compared to the high school learners in the other clusters. Together with the SPM results, the relatively large amount of time between attempts and fewer attempts could indicate that high-performing high school participants were more deliberate with their attempts.

The low performing learners at both camps, participants that scored the lowest on the cCTt post-test, most frequently used problem-solving patterns that align with characteristics of wheel-spinning [16]. The presence of removing all blocks may be an indication of wheel-spinning. The participants from the middle school camp in the cSPADE analysis and lowest performing clusters from that camp both had removing all blocks as a state in their most frequent patterns. *Middle school learners also used almost double the amount of attempts compared to participants in the high school camp.* Together these findings indicate that the low-performing middle school learners may have experienced wheel-spinning, they persisted during the task but were more prone to giving up on their ideas causing them to spend extended time on the puzzle without meaningful progress. Although remove all was not a state present in the most frequent patterns for the high school learners, rearranging blocks in subsequent attempts was only used by the lowest-scoring cluster and could be indicative of non-productive game play behaviors. High school participants in the lowest performing cluster used more attempts and seemingly more non-deliberate actions compared to the highest scoring cluster when solving the difficult puzzle.

5.1 Limitations and Future Work

A limitation for our study is our relatively small sample size and choice to conduct our analyses on camps that took place in subsequent years, which could potentially alter our results. For our analyses we also solely rely on log data. Triangulation with other data sources like video and audio of students working on the task could enable us to find more nuanced results and could be a beneficial direction for the field.

Prior work has demonstrated the potential for large language models (LLMs) to provide meaningful, adaptive feedback to students within *Minecraft*-based learning environments [13]. Building on this foundation, our future work will leverage the indicators of persistence identified in this study, along with the strategies students employ during wheel-spinning, to inform the design of system prompts that enable LLMs to deliver targeted, conversational feedback. Overall, we are optimistic that these future supports, grounded by the findings from this study, will enhance student persistence, reduce wheel-spinning, and improve learning outcomes for students.

6. ACKNOWLEDGMENTS

The materials used in this study are based upon work supported by the National Science Foundation, Directorate for Education and Human Resources, and Institute of Education Sciences under Grants 1713609, 1906873, and 2229612.

7. REFERENCES

[1] A. Abbott and A. Tsay. Sequence analysis and optimal matching methods in sociology: Review and prospect. *Sociological methods & research*, 29(1):3–33, 2000.

[2] C. Angeli and M. Giannakos. Computational thinking education: Issues and challenges, 2020.

[3] J. E. Beck and Y. Gong. Wheel-spinning: Students who fail to master a skill. In *International conference on artificial intelligence in education*, pages 431–440. Springer, 2013.

[4] F. C. Billari. The analysis of early life courses: Complex descriptions of the transition to adulthood. *Journal of population research*, 18(2):119–142, 2001.

[5] E. Bumbacher, S. Salehi, C. Wieman, and P. Blikstein. Tools for science inquiry learning: Tool affordances, experimentation strategies, and conceptual understanding. *Journal of Science Education and Technology*, 27(3):215–235, 2018.

[6] B. Eichmann, S. Greiff, J. Naumann, L. Brandhuber, and F. Goldhammer. Exploring behavioural patterns during complex problem-solving. *Journal of computer assisted learning*, 36(6):933–956, 2020.

[7] L. El-Hamamsy, M. Zapata-Cáceres, E. M. Barroso, F. Mondada, J. D. Zufferey, and B. Bruno. The competent computational thinking test: Development and validation of an unplugged computational thinking test for upper primary school. *Journal of Educational Computing Research*, 60(7):1818–1866, 2022.

[8] L. El-Hamamsy, M. Zapata-Cáceres, E. Martín-Barroso, F. Mondada, J. D. Zufferey, B. Bruno, and M. Román-González. The competent computational thinking test (cctt): A valid, reliable and gender-fair test for longitudinal ct studies in grades 3–6: L. el-hamamsy et al. *Technology, Knowledge and Learning*, 30(3):1607–1661, 2025.

[9] E. Fussell. Measuring the early adult life course in mexico: An application of the entropy index. *Advances in Life Course Research*, 9:91–122, 2005.

[10] A. Gabadinho, G. Ritschard, N. S. Müller, and M. Studer. Analyzing and visualizing state sequences in r with traminer. *Journal of statistical software*, 40:1–37, 2011.

[11] B. Hagan. Exploring persistence in computing education: An ordered network analysis of student engagement and persistence in a minecraft learning environment. Unpublished manuscript, 2024.

[12] T.-C. Hsu, S.-C. Chang, and Y.-T. Hung. How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, 126:296–310, 2018.

[13] S. Hum, M. Gadbury, J. Ginger, K. Duda, and H. C. Lane. Barrelbots: Chatgpt feedback for middle school student creative minecraft artifacts. In *International*

Conference on Artificial Intelligence in Education, pages 3–15. Springer, 2025.

[14] S. Kai, M. V. Almeda, R. S. Baker, C. Heffernan, N. Heffernan, et al. Decision tree modeling of wheel-spinning and productive persistence in skill builders. *Journal of Educational Data Mining*, 10(1):36–71, 2018.

[15] J. Kang, M. Liu, and W. Qu. Using gameplay data to examine learning behavior patterns in a serious game. *Computers in Human Behavior*, 72:757–770, 2017.

[16] M. Kapur. Productive failure. *Cognition and instruction*, 26(3):379–424, 2008.

[17] V. Kovanovic, D. Gašević, S. Dawson, S. Joksimovic, and R. Baker. Does time-on-task estimation matter? implications on validity of learning analytics findings. *Journal of Learning Analytics*, 2(3):81–110, 2015.

[18] N. Kumar. How many people play minecraft in 2026 (player count). `https://www.demandsage.com/minecraft-statistics/`, 2026.

[19] H. C. Lane, M. Gadbury, J. Ginger, S. Yi, N. Comins, J. Henhapl, and A. Rivera-Rogers. Triggering stem interest with minecraft in a hybrid summer camp. 2022.

[20] W. J. M. Computational thinking's influence on research and education for all. *Italian Journal of Educational Technology*, 25(2):7–14, 2017.

[21] T. S. McNerney. From turtles to tangible programming bricks: explorations in physical language design. *Personal and Ubiquitous Computing*, 8:326–337, 2004.

[22] F. Murtagh and P. Legendre. Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of classification*, 31(3):274–295, 2014.

[23] V. J. Shute, C. Sun, and J. Asbell-Clarke. Demystifying computational thinking. *Educational research review*, 22:142–158, 2017.

[24] E. J. Slattery, P. Lehane, D. Butler, M. O'Leary, and K. Marshall. Assessing the benefits of digital game-based learning with minecraft in children, adolescents and young adults: A broad systematic review. *Review of Education*, 13(1):e70035, 2025.

[25] J. Voogt, P. Fisser, J. Good, P. Mishra, and A. Yadav. Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and information technologies*, 20(4):715–728, 2015.

[26] K. D. Wang, H. Liu, D. DeLiema, N. Haber, and S. Salehi. Discovering players' problem-solving behavioral characteristics in a puzzle game through sequence mining. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 498–506, 2024.

[27] J. M. Wing. Computational thinking. *Communications of the ACM*, 49(3):33–35, 2006.

[28] G. K. Wong, S. Jian, and H.-Y. Cheung. Engaging children in developing algorithmic thinking and debugging skills in primary schools: A mixed-methods multiple case study. *Education and Information Technologies*, 29(13):16205–16254, 2024.

Positive or Negative? Exploring Graph-based Augmentation for Contrastive Learning in Knowledge Tracing

Soobin Ahn
Seoul National University
ahnsb0418@snu.ac.kr

Shinyoung Lee
Independent Researcher
02.shin.00@gmail.com

Hyojun Lee
Seoul National University
gywnszz@snu.ac.kr

Insub Shin
Seoul National University
iss9802@snu.ac.kr

ABSTRACT

Contrastive learning is a machine learning technique that optimizes the embedding space by pulling positive pairs closer and pushing negative pairs farther. Recently, knowledge tracing (KT) models have adopted contrastive learning to learn the representation of students' interaction sequences. Therefore, augmentation methods should preserve students' knowledge states when generating positive pairs and distort them when generating negative pairs. Which means that augmentation methods should take into account the pedagogical semantics beneath students' interaction sequences. However, random augmentations may ignore the educational context and consequently alter the underlying knowledge representations. To address this issue, we propose a graph-based augmentation method that leverages an interaction graph, where each node denotes a skill-response pair and each edge captures the conditional transition probability between interactions. The interaction graph is constructed using transition probabilities predicted by a trained DKT model for the next interaction. Each augmentation method generates learning paths by following transition probabilities in the interaction graph, enabling the construction of structurally plausible sequences. This design preserves pedagogically meaningful transitions within augmented sequences. Furthermore, we explored whether pairs generated by random augmentation methods and our graph-based augmentation methods function as positive or negative pairs. Our results show that graph-based augmentation can effectively generate sufficient and meaningful pairs. However, we find that knowledge state preservation differs between the human perspective and the model's perspective. Future work is needed to further explore and understand this difference.

Keywords

Data Augmentation, Contrastive Learning, Knowledge Tracing

Soobin Ahn, Shinyoung Lee, Lee Hyojun, and Shin Insub. Positive or Negative? Exploring Graph-based Augmentation for Contrastive Learning in Knowledge Tracing. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 743–747. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040032

1. INTRODUCTION

In recent years, Knowledge Tracing (KT) has been actively studied as a fundamental task in Intelligent Tutoring Systems (ITS). In order to capture the current knowledge state within an ITS, KT estimates the probability of a correct response by taking the student's historical learning path and a target skill as inputs. The estimated knowledge state supports personalized instruction and adaptive learning systems.

To better capture the dynamics of knowledge states, various deep learning approaches have been proposed for KT. Among these, contrastive learning has gained attention as a representation learning framework. It trains models by pulling together semantically similar (positive) pairs while pushing apart semantically dissimilar (negative) pairs. Several studies have adopted contrastive learning in KT by applying data augmentation to construct these pairs from student interaction sequences. For instance, CL4KT[3] was proposed to improve representation learning under sparse interaction data by applying random augmentations to construct positive and negative pairs, encouraging semantically similar learning histories to have closer representations through a contrastive objective. This approach demonstrated the effectiveness of contrastive learning in enhancing knowledge state representations and improving KT performance.

However, in contrastive learning for KT, it remains unclear whether commonly used augmentation methods function as valid positive or negative pairs. Recent work such as QCKT[8] has highlighted that the criteria for selecting positive and negative samples remain insufficiently defined. In KT, semantic similarity reflects the consistency of the students' latent knowledge state. Therefore, in this work, positive pairs are defined as sequence pairs that preserve the students' knowledge state, and negative pairs as those that distort it. Based on this definition, we investigate whether each augmentation method acts as a positive or negative pair, based on its effect on the students' knowledge state.

Despite these successes, we identify several limitations of existing augmentation approaches. First, current models fo-

cus on sequence-level augmentations.[9, 1] However, these methods do not fully capture the structured learning path that emerges from the underlying connections between skills. Second, random augmentations may overlook the educational context of the data and risk distorting the underlying logic of knowledge acquisition.[5] For example, if a prerequisite question is randomly replaced or removed, it creates an invalid sequence that contradicts the actual logic of a students' knowledge acquisition. Such augmentations may lead to sub-optimal contrastive objectives by providing the model with semantically invalid positive pairs.

To address this limitation, we propose graph-based augmentation methods that leverage an interaction graph over student interactions. Rather than applying random perturbations, our approach generates augmented sequences by following transition-weighted paths in the graph, enabling structure-aware augmentations that reflect meaningful pedagogical transitions.

To systematically analyze the role of augmentation methods in contrastive learning for KT, we evaluate both existing random augmentations and the proposed graph-based augmentations in terms of their suitability as positive or negative pairs. Specifically, we assess their suitability based on AUC gain, where higher performance indicates better alignment with the assigned role.

The following research questions are investigated in this study:
RQ1. How do random augmentation methods behave when assigned as positive or negative pairs in contrastive learning for KT?
RQ2. How do graph-based augmentation methods behave when assigned as positive or negative pairs in contrastive learning for KT?
RQ3. How does the performance of the full random augmentation setting compare to that of the full graph-based augmentation setting?

2. METHOD

2.1 Graph-based Augmentation

2.1.1 Definition of interaction graph

Graph-based Knowledge Tracing (GKT) [6] has attempted to capture structured skill relationships and educational context by modeling relationships between skills as graph structures. However, existing studies focus primarily on skill-to-skill correlations.[2, 4, 10] Consequently, there is still a lack of research that utilizes interaction nodes to fully capture the directionality of a students' knowledge state. To address this gap, we employed DKT graph. Given $|S|$ skills, the graph contains $N = 2|S|$ interaction nodes, and each interaction (s, r) with $r \in \{0, 1\}$ is mapped to a node index $v = s+|S|\cdot r$. We define a weighted interaction graph $G = (V, E)$ and its corresponding adjacency matrix $A_{ij} \in \mathbb{R}^{N\times N}$. To capture various pedagogical aspects, we construct an adjacency matrix based on a DKT graph. The DKT graph is a model-based interaction graph constructed from the conditional prediction probabilities of a trained DKT model [7]. Given a student interaction (s_t, r_t) at time step t, we estimate the probability that a student answers a target skill s_{t+1} correctly at the next step. Specifically, the trained DKT model provides

$$p_1 = \Pr(r_{t+1} = 1 \mid s_t, r_t, s_{t+1})$$

which represents the likelihood of correct response on skill s_{t+1} given interaction (s_t, r_t). The probability of an incorrect response is defined as

$$p_0 = \Pr(r_{t+1} = 0 \mid s_t, r_t, s_{t+1}) = 1 - p_1.$$

Based on these predictions, response r_t is determined as 1 if $p_1 \geq p_0$ and 0 otherwise. To ensure reliable transitions, a threshold τ is defined and only transitions satisfying $\Delta = |p_1 - p_0| > \tau$ are retained, with τ set to 0.5 in practice to preserve only transitions with sufficiently meaningful differences. To determine the edge weights, we define $w = \max(p_1, p_0)$, selecting the higher probability between a correct and an incorrect response at the next step. We then retain the top-k transitions with highest w values for each node, defining the weighted adjacency matrix as $A_{ij} = w$.

2.1.2 Graph-based augmentation method

Using the DKT graph defined above, we propose six graph-based augmentation methods. All augmentations leverage the weighted adjacency matrix A to ensure that generated sequences reflect meaningful transitions between interaction nodes.

Graph-based Masking Masking candidates are restricted to interactions that have at least one neighbor within the same sequence with a transition weight above threshold τ. Only these eligible interactions are subject to masking, ensuring that masked positions remain inferable from their structural context, thereby preventing excessive distortion of the students' knowledge state.

Graph-based Crop Transition weights between consecutive interaction pairs are computed using the adjacency matrix A, reflecting how strongly each pair of interactions is connected. Based on these weights, the position with the weakest connection (i.e., the minimum transition weight) is identified and used as the boundary for cropping. The subsequence closer to the target crop length is then retained, ensuring that the remaining segment consists of strongly connected interactions, thereby preserving a consistent and meaningful learning context.

Graph-based Extension For each interaction (s, r), a weighted candidate set is constructed by sampling neighboring interactions (s', r') from the row-normalized adjacency matrix A, excluding same-skill nodes. Using this candidate set, two augmented views $\mathbf{x}^{+1}$ and $\mathbf{x}^{+2}$ are generated via prior and posterior extension, respectively. In prior extension, (s', r') is appended before the head of the sequence; in posterior extension, it is appended after the tail. If the sequence exceeds the maximum length, the opposite end is truncated accordingly.

Graph-based Branching Path Sequence pairs are constructed to share the same initial interactions and diverge from a branching point. From this point, the remaining interactions are generated by traversing the graph, sampling from the top-k neighbors based on transition strength and proximity to a target endpoint. The final interaction is constrained to match the original skill but with a flipped response, ensuring

divergence in the terminal knowledge state while preserving overall structural similarity.

Graph-based Path Substitution The first and last interactions of a sequence are preserved, while an intermediate segment is replaced with an alternative path generated from the interaction graph. Specifically, the intermediate segment is selected and replaced by traversing the adjacency matrix A to construct a new path between the two fixed anchor points. During this process, interactions from the original segment are penalized to encourage the generation of diverse alternative paths, while the initial and final interactions remain consistent.

2.2 Experiment Settings

2.2.1 Benchmark Datasets

To evaluate the performance of the proposed method, we utilized the ASSISTment 2009 dataset. To ensure data integrity, we applied several preprocessing steps before model training. Specifically, users with fewer than five interactions were excluded, and interactions without explicit concept tags were removed to maintain clear concept mapping.

ASSISTments (2009). . These datasets were obtained from an online mathematics tutoring platform. Specifically, we used the ASSISTment 2009-2010 "Skill Builder"set version.

Table 1: Statistics of the Datasets

Dataset	Users	Items	Skills	Interactions	Avg. Seq Length
ASSISTments 09	3,695	17,728	112	282,071	76.34

2.2.2 Experimental design

To investigate whether each augmentation method is more suitable as a positive or negative pair, each augmentation is evaluated under two distinct roles within the InfoNCE loss. When evaluated as a positive, the augmentation is applied to both views $\mathbf{x}^{+1}$ and $\mathbf{x}^{+2}$ with masking. When evaluated as a negative, the augmentation is used to generate the hard negative z_k, while both positive views consist of masking only. We use masking alone as the base positive, as it is considered the most conservative augmentation that is least likely to distort the students' knowledge state. We evaluated two augmentation methods: random augmentations adopted from CL4KT [3], including crop, replace, and permute; and our proposed graph-based augmentations, including prior extension, posterior extension, graph-based crop, graph-based branching path, and path substitution. Each augmentation is tested in both roles, and performance is compared to determine whether it is more appropriate as a positive or negative pair. We also include two additional baselines: No CL, where the model is trained without any contrastive objective, and Full, which represents the best configuration for each augmentation type.

2.2.3 Implementation details

The embedding dimension and hidden dimension were fixed to $d = 64$ for all experiments. Student interaction sequences were sorted chronologically for each user. To standardize sequence lengths, we retained up to the most recent 50 interactions per student; longer sequences were truncated, while shorter ones were zero-padded. All models were evaluated using 5-fold cross-validation, with the Area Under the ROC Curve (AUC) serving as the primary metric. All experiments share masking probability of $r_{\text{mask}} = 0.1$, and the target augmentation method in each condition is applied with a transformation ratio of 0.2. When evaluated as a negative, r_{neg} is set to 1.0; otherwise, it is set to 0. The Full configuration applies both $r_{\text{mask}} = 0.1$ and $r_{\text{crop}} = 0.2$ as positive views, with $r_{\text{neg}} = 1.0$ for hard negative generation. All models were trained using the Adam optimizer with a learning rate of 0.001 and a batch size of 64. We applied a learning rate scheduler that decayed the learning rate by a factor of 0.5 when the validation AUC did not improve for five consecutive epochs, and early stopping was applied if no improvement was observed for 5 consecutive epochs.

3. RESULTS

3.1 RQ 1. Effect of Random Augmentation on Positive and Negative Pairs

Table 2 presents the performance of random augmentation methods when used as either positive or negative pairs. As shown in the table, random augmentations generally achieve higher performance when used as positive pairs than as negative pairs. In particular, replace shows a noticeable performance drop when used as a negative pair, while crop and permute also exhibit a moderate gap. Based on these results, crop, replace, and permute tend to be more suitable when used as positive pairs. From a knowledge state perspective, this provides partial evidence that these random augmentations may better maintain consistency with the original learning sequence when used as positive pairs. These findings highlight that the behavior of random augmentations depends on how they are incorporated within the contrastive learning framework, and that their role assignment can influence their effectiveness.

Table 2: Effect of Random Augmentation on Positive and Negative Pairs

	Positive	Hard negative	Δ
Crop	0.74135	0.73906	+0.00229
Replace	0.73179	0.71676	+0.01503
Permute	0.72977	0.72717	+0.0026
Full random	0.72487		

3.2 RQ 2. Effect of Graph-based Augmentation on Positive and Negative Pairs

Table 3 shows the performance of each graph-based augmentation as a positive or negative pair. Overall, graph crop, branching path, and path substitution tend to perform better as positive pairs, suggesting that these methods are more aligned with the role of positive pairs from a knowledge state perspective. In contrast, prior and posterior extension show slightly better performance as negative pairs, although the differences compared to their positive counterparts are not substantial. For augmentation methods that do not exhibit a clear performance gap between roles, it is less straightforward to determine their appropriate assignment within the

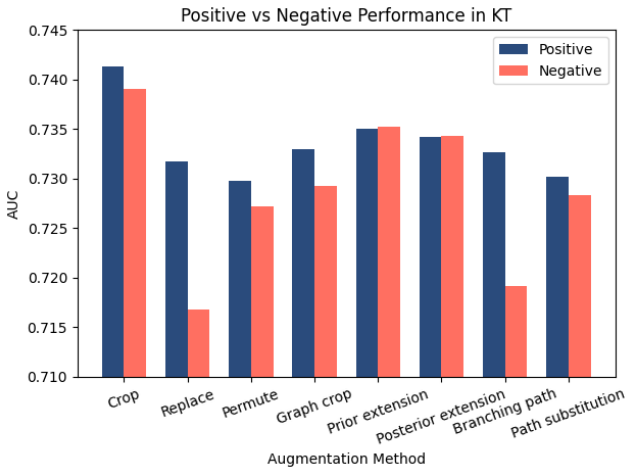


Figure 1: Comparison of AUC performance for each augmentation method under positive and negative pair assignments in knowledge tracing.

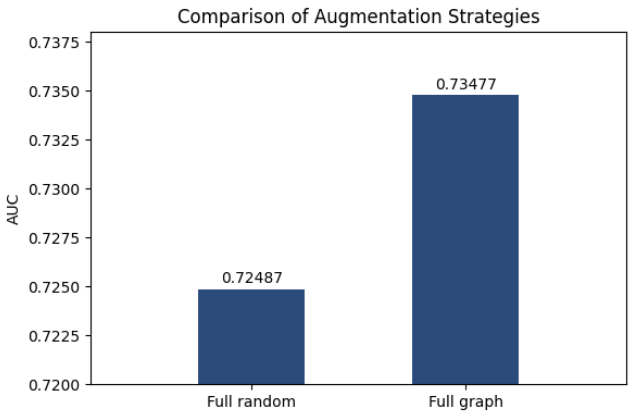


Figure 2: Performance Comparison between Full Random and Graph-based Augmentation Settings

contrastive framework. These methods can be interpreted as producing pairs with only minor semantic differences, rather than clearly positive or strongly negative ones. Additionally, in the case of branching path, although it was originally designed as a hard negative by transitioning from the original sequence to a node with a different response, it shows one of the most pronounced performance differences when used as a positive pair. This observation suggests that its behavior is not fully aligned with its intended role, requires further discussion regarding its underlying cause. The results of RQ1 and RQ2 are further illustrated in Figure 1, where the differences between augmentation methods across roles can be clearly compared.

Table 3: Effect of Graph-based Augmentation on Positive and Negative Pairs

	Positive	Hard negative	Δ
Graph crop	0.73299	0.72922	+0.00377
Prior ext	0.73502	0.73526	−0.00024
Posterior ext	0.73418	0.73428	−0.00010
Branching path	0.73271	0.71912	+0.01359
Path subst	0.73020	0.72834	+0.00186
Full graph		0.73477	

3.3 RQ 3. Comparison of Full Random and Full Graph-based Augmentation

We evaluate the Full configuration for both random and graph-based augmentation settings. In the random setting, all augmentation strategies—masking, crop, permute, and replace—are employed to construct positive views, while negative samples are drawn from the interaction histories of other students within the same mini-batch. For the graph-based setting, the Full configuration is constructed based on the observed performance of each method, where graph mask, graph crop, branching path, and path substitution are assigned as positive pairs, and prior and posterior extension are used as hard negatives. As shown in Figure 2, the graph-based Full configuration achieves the best overall performance (AUC = 0.73477), outperforming the random augmentation-based Full configuration (AUC = 0.72487). This comparison highlights the effectiveness of graph-based augmentation in the full setting. However, it is also observed that the AUC obtained from the Full configurations is lower than the best performance achieved by individual augmentation methods (e.g., random crop and prior extension). This suggests that simply combining multiple augmentation strategies does not necessarily lead to further performance gains, and that further consideration is needed on how to effectively combine them.

4. CONCLUSION

In this work, we investigated graph-based augmentations from the perspective of knowledge state preservation and distortion, and compared them with commonly used random augmentations. To construct structure-aware augmentations, we build an interaction graph using a trained DKT model, where edges are defined by conditional prediction probabilities. This results in a weighted graph that reflects meaningful learning paths, based on which we generate augmented sequences along valid transitions. To evaluate their effectiveness, we analyze how each augmentation method contributes to contrastive learning when assigned as either a positive or negative pair, by examining performance differences across roles. The results show that random augmentations are generally more suitable for positive pairs, while in the graph-based setting, most augmentation methods, except for extension-based approaches, show similar tendencies. In particular, prior and posterior extension exhibit only marginal differences between roles, indicating low sensitivity to role assignment. In contrast, the proposed branching path method, originally designed as a hard negative, demonstrates a clear AUC gain when used as a positive pair. This suggests that its empirical behavior does not fully align with its intended role. Additionally, in this study, the semantic role of each augmentation method was discussed only through downstream AUC performance, which provides an indirect perspective on whether an augmented learning path is truly semantically equivalent to the original sequence. Future work should therefore develop quantitative measures for

semantic equivalence and revisit whether knowledge state preservation and distortion from a human perspective are aligned with those learned by contrastive learning models. Furthermore, it remains unclear how augmentation methods with minor performance differences across roles should be incorporated into the contrastive learning framework. These methods may instead be better suited for data augmentation or regularization, and we suggest this as a direction for future work.

5. REFERENCES

[1] H. Dai, Y. Yun, Y. Zhang, R. An, W. Zhang, and X. Shang. Self-paced contrastive learning for knowledge tracing. *Neurocomputing*, 609:128366, 2024.

[2] W. Han, W. Lin, L. Hu, Z. Dai, Y. Zhou, M. Li, Z. Liu, C. Yao, and J. Chen. Contrastive cross-course knowledge tracing via concept graph guided knowledge transfer. *arXiv preprint arXiv:2505.13489*, 2025.

[3] W. Lee, J. Chun, Y. Lee, K. Park, and S. Park. Contrastive learning for knowledge tracing. In *Proceedings of the ACM web conference 2022*, pages 2330–2338, 2022.

[4] Y. Liu, Q. Liu, Y. Tian, C. Wang, Y. Niu, Y. Song, and C. Li. Concept-aware denoising graph neural network for micro-video recommendation. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 1099–1108, 2021.

[5] S. Mao, G. Li, and Y. Jiang. Boosting knowledge tracing with structure-preserving adaptive contrastive learning. *IEEE Transactions on Consumer Electronics*, 2025.

[6] H. Nakagawa, Y. Iwasawa, and Y. Matsuo. Graph-based knowledge tracing: modeling student proficiency using graph neural network. In *IEEE/WIC/aCM international conference on web intelligence*, pages 156–163, 2019.

[7] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[8] X. Shen, F. Yu, Y. Liu, R. Liang, Q. Wan, T. Yang, M. Shi, and J. Sun. Enhancing knowledge tracing with question-based contrastive learning. *Knowledge-Based Systems*, 325:113899, 2025.

[9] X. Shen, F. Yu, Q. Wan, R. Liang, and J. Sun. Multi-level contrastive learning for knowledge tracing. *ACM Transactions on Knowledge Discovery from Data*, 19(8):1–29, 2025.

[10] X. Song, J. Li, Y. Tang, T. Zhao, Y. Chen, and Z. Guan. Jkt: A joint graph convolutional network based deep knowledge tracing. *Information Sciences*, 580:510–523, 2021.

Predicting Disagreement with Human Raters in LLM-as-a-Judge Difficulty Assessment without Using Generation-Time Probability Signals

Yo Ehara
Tokyo Gakugei University
ehara@u-gakugei.ac.jp

ABSTRACT

Automatic generation of educational materials using large language models (LLMs) is becoming increasingly common, but assigning difficulty levels to such materials still requires substantial human effort. LLM-as-a-Judge has therefore attracted attention, yet disagreement with human raters remains a major challenge. We propose a method for predicting which LLM-generated difficulty ratings are likely to disagree with human raters, so that such cases can be sent for re-rating. Unlike prior approaches, our method does not rely on generation-time probability signals, which must be collected during rating generation and are often difficult to compare across LLMs. Instead, exploiting the fact that difficulty is an ordinal scale, we use a separate embedding space, such as ModernBERT, and identify disagreement candidates based on the geometric consistency of the rating set. Experiments on English CEFR-based sentence difficulty assessment with GPT-OSS-120B and Qwen3-235B-A22B showed that the proposed method achieved higher AUC for predicting disagreement with human raters than probability-based baselines.

Keywords

Disagreement, Difficulty Annotation, Large Language Models

1. INTRODUCTION

In recent years, improvements in the performance of large language models (LLMs) and the development of supporting infrastructure have made it increasingly practical to automatically generate educational content such as practice questions, example sentences, and explanations for learners. As a result, the potential for educational applications has been expanding rapidly. At the same time, whether instructional materials are at an *appropriate difficulty level* for learners is directly related to both learning effectiveness and fairness. Difficulty assessment is therefore a foundational task in the deployment of generated educational materials. However, manual difficulty rating requires expertise and time, and thus easily becomes a bottleneck in large-scale operation.

To address this human-evaluation cost, *LLM-as-a-Judge*, in which an LLM also serves as an evaluator, has attracted increasing attention [16]. LLM-as-a-Judge can reduce the temporal and financial costs of evaluation by lowering the burden on teachers and other human evaluators who would otherwise need to assess large numbers of examples and instructional materials for difficulty and related properties. At the same time, however, reliability, robustness, and alignment with human judgments have repeatedly been identified as major challenges [2, 13, 3]. Therefore, in practical use, it is important not to accept LLM ratings as they are, but instead to efficiently identify cases that are likely to disagree with human judgments, that is, candidate cases of inter-rater disagreement.

One promising cue for identifying such disagreement candidates is to use generation probabilities output by the LLM at rating time, such as log-likelihoods or probability distributions. However, this approach has practical limitations: such signals are available only at generation time. Thus, if they are not collected, they cannot be recovered later. In addition, because generation probabilities differ across models, they are difficult to compare across LLMs.

Building on this motivation, this study extends the framework proposed in [6] and proposes a method for predicting, without using LLM probability signals, which difficulty ratings produced by LLM-as-a-Judge are likely to disagree with human raters. The key idea of this study is that difficulty is an ordinal scale. In language difficulty estimation and readability assessment, deep representations such as BERT embeddings have been shown to be effective [4], and it is also well known that embedding spaces exhibit geometric structure, including directionality and anisotropy [9]. Based on this observation, we prepare, independently of the LLM used as the rater, a separate embedding space such as that of ModernBERT [14], and detect as disagreement candidates those ratings that are geometrically inconsistent with the ordinal difficulty structure, that is, with the alignment and consistency of the rating set as a whole.

This approach has two major practical advantages. First, because it does not depend on probability outputs, it can be applied even when such signals were not collected at the time the LLM-as-a-Judge generated its ratings. Second, unlike

Yo Ehara. Predicting Disagreement with Human Raters in LLM-as-a-Judge Difficulty Assessment without Using Generation-Time Probability Signals. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 748–753. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039851

probability outputs, the quantities used by our method are geometric, which makes them comparable across difficulty labels assigned by different LLMs on the same dataset. Furthermore, the model used for the embedding space is much smaller than the LLM used for rating, with fewer than 1 billion parameters.

In our experiments, we consider an English CEFR-based difficulty assessment task, which provides internationally shared and rigorous difficulty criteria, and predict disagreement with human ratings for sentence difficulty labels produced by the open-weight LLMs gpt-oss-120b [12] and Qwen3-235B-A22B [15]. The results show that the proposed method predicts disagreement with higher AUC than methods based on LLM probability signals, demonstrating its effectiveness as a probability-free and interpretable method for prioritizing cases for re-rating.

2. RELATED WORK

Methods for LLM-as-a-Judge, which assess the extent to which LLM evaluations are correct, have often relied on LLM probability signals. For example, LLM-Rubric presents a framework that calibrates human ratings by using the probability distribution over rubric-based questions as input [10]. However, the use of probability signals still poses practical challenges: (i) they must be obtained at the same time as the rating is generated, and are therefore affected by logging availability and API constraints; and (ii) probability scales and calibration differ across models, making comparison and interpretation difficult. In addition, as a black-box approach to uncertainty estimation that does not rely on probability distributions, previous work has proposed detecting anomalies from the self-consistency of sampled outputs (e.g., SelfCheckGPT) [11], although such methods may require temperature adjustment or other sampling controls.

In language learning research, although assigning difficulty levels based on a common framework is important, it has been noted that constructing sufficiently large, high-quality annotated datasets is itself difficult. For example, a study that constructed a CEFR-based sentence difficulty corpus (CEFR-SP) and demonstrated sentence difficulty estimation using BERT-based models was motivated by this challenge [1].

3. PROPOSED METHOD

The proposed method extends the framework proposed in [6]. We first provide an intuitive explanation of the method using Fig. 1. First, we prepare an embedding vector space that is *completely separate* from the one used for difficulty labeling. For example, we assume a well-known encoder such as BERT [5] or ModernBERT [14], and all embedding vectors are assumed to be normalized to unit length. The main idea of [6] is that, given such an embedding space together with a set of difficulty labels assigned by an LLM, one can uniquely compute a vector that represents the degree of geometric consistency between the embedding space and the difficulty labeling.

Although [6] evaluates only the consistency of the difficulty labeling as a whole, it is also possible to compute the degree of consistency between the embedding space and each individual difficulty label. The proposed method uses this

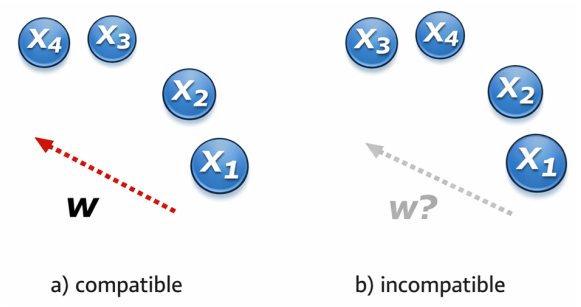


Figure 1: Figure reproduced from [6] with their permission. In (a), let $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$, and $\mathbf{x}_4$ denote two-dimensional word or sentence embeddings. We write $\mathbf{x}_1 \prec \mathbf{x}_2 \prec \mathbf{x}_3 \prec \mathbf{x}_4$ to indicate that $\mathbf{x}_1$ is annotated as easier than $\mathbf{x}_2$, $\mathbf{x}_2$ as easier than $\mathbf{x}_3$, and so on. In this case, if the points are ordered along the direction of $\mathbf{w}$ in the two-dimensional space, the resulting order matches the annotation. In contrast, in example (b), no such direction exists. In practice, most datasets and embedding spaces are closer to the situation in (b), but the degree of alignment is defined by how well one can find a direction that satisfies the ordering in (a) with as few constraint violations as possible. In addition, for a given instance, the degree to which the average of the difference vectors between that instance and all other instances is aligned with $\mathbf{w}$ is defined as the alignment score of that instance.

property: difficulty labels with low consistency with the embedding space are treated as candidates for low inter-rater agreement, that is, cases likely to show disagreement when another rater is introduced. We now describe their method in more detail. Consider N embedding vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, where each $\mathbf{x}_i$ is a D-dimensional vector. We assume that all embedding vectors are normalized, that is, $||\mathbf{x}_i|| = 1$, where $||\mathbf{x}_i||$ denotes the Euclidean norm of the vector. Let $\mathbf{w} \in \mathbb{R}^D$ be a D-dimensional vector representing a direction in the embedding space. They convert difficulty labels into pairwise constraints between two instances. They denote by $\mathbf{x}_i$ the vector judged to be *easier*, and by $\mathbf{x}_j$ the vector judged to be *more difficult*. Then, the condition that instances are aligned in the embedding space in order of difficulty when viewed along the direction $\mathbf{w}$ can be written as $\mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j) < 0$. To transform the inequality constraints into equality constraints, we introduce variables that represent the degree to which each constraint is satisfied (slack variables), and write $\mathbf{w}^\top (\mathbf{x}_{i_k} - \mathbf{x}_{j_k}) + \xi_k = 0$. Here, k is an index over constraints.

At this point, by considering that $\mathbf{w}$ is a direction vector, that is, $||\mathbf{w}||^2 = 1$, and searching for the direction $\mathbf{w}$ that maximizes the sum of all ξ_k values (the degrees to which the constraints are satisfied), we obtain the following mathematical optimization problem. Here, although originally $T = 0$, if we relax it to $T = -\infty$, the optimal $\mathbf{w}$ can be computed in closed form simply as the mean vector of the differences between labeled pairs.

$$\max_{\mathbf{w},\xi} \sum_{k=1}^{K} \xi_k, \tag{1}$$

$$\text{s.t. } \forall k, \quad \mathbf{w}^\top (\mathbf{x}_{i_k} - \mathbf{x}_{j_k}) + \xi_k = 0, \quad \xi_k \geq T, \; ||\mathbf{w}||^2 = 1 \tag{2}$$

The solution to (2) is given by $\mathbf{w} \propto -\sum_{k=1}^{K}(\mathbf{x}_{i_k} - \mathbf{x}_{j_k})$. Let $\bar{\mathbf{x}}_i$ denote the mean vector of level i, and similarly for level j, and let N_i and N_j denote the numbers of instances at levels i and j, respectively. Then, $\mathbf{w}$ can be written as $\mathbf{w} \propto -(N_i\bar{\mathbf{x}}_i - N_j\bar{\mathbf{x}}_j)$. We refer to the method using this direction as "exactw." Although the exact solution can be written in this form, in practice, when the numbers of instances at some levels are small as a result of LLM-based difficulty labeling, computing $\mathbf{w}$ exactly may lead to a biased direction vector. Since $\mathbf{w}$ only needs to be a direction vector, we also propose a simpler method, "most2," in which $\mathbf{w}$ is defined as the difference between the mean vectors of the two labels that were assigned most frequently by the LLM.

Once the direction $\mathbf{w}$ has been obtained, the values of ξ_k can be computed accordingly. Then, by summing the ξ_k values over all constraints involving item i, we can compute the degree of consistency for item i.

4. EXPERIMENTAL SETUP

For the evaluation in this study, a dataset with rigorously obtained human ratings is desirable, and it is also important that the dataset include difficulty labels provided by multiple human raters. For this reason, we used the CEFR-SP dataset [1]. In this dataset, the difficulty of English sentences is rated by two expert annotators on a six-level scale, A1, A2, B1, B2, C1, and C2, based on the Common European Framework of Reference for Languages (CEFR), which is a standard and widely used framework for English proficiency in English language education.

Next, in this study, we needed baseline models from which confidence scores (token generation probabilities) could be obtained. For this purpose, we used GPT-OSS-120B and Qwen3-235B-A22B (hereafter, Qwen3) for labeling, as they were among the largest models that could be run locally on a single GPU as of February 2026.

The input consisted of 10,004 English sentences. The label distribution produced by GPT-OSS-120B was A1: 500, A2: 3,345, B1: 2,899, B2: 3,230, C1: 29, and C2: 1. For Qwen3, the distribution was A1: 635, A2: 1,626, B1: 4,014, B2: 3,293, and C1: 436.

Performance in disagreement detection was evaluated using ROC-AUC, PR-AUC, and Precision@20. ROC-AUC is the area under the ROC curve obtained by thresholding the decision function and plotting the resulting pairs of true positive rate and false positive rate. PR-AUC is the area under the precision–recall curve obtained by similarly sweeping the threshold. Precision@20 denotes the precision when the top 20 instances ranked in descending order of $s(i)$ are regarded as candidates for re-evaluation. For all metrics, sentences for which the label assigned by the LLM differed from that of human Rater A or B were treated as positive instances.

4.1 Difficulty Labeling by LLMs

For the logprob-based baseline, we used the token log-probability of the label sequence output by GPT-OSS-120B or Qwen3 (logprob_label) as the confidence score for disagreement detection.

The prompt included the CEFR criteria descriptions in the user input, and in practice we used the following System/User strings exactly as written:

> System: You are an English proficiency rater who must assign a CEFR level (A1, A2, B1, B2, C1, or C2) to a single sentence using the provided criteria.

After this, we directly inserted the English descriptions of the CEFR criteria for each level from the corresponding Wikipedia page[1].

For the user input, we used the following prompt:

> You must output exactly one label token (A1, A2, B1, B2, C1, or C2) with no explanation.
>
> User: Sentence: "sentence" Respond with exactly one of: A1, A2, B1, B2, C1, C2.
>
> Assistant:

For Qwen3 instruct models, we additionally appended "`\nDo not output anything else. Answer: `" at the end in order to constrain the response to a single label only.

Table 1: Number of parameters in the embedding models

Model	Number of parameters (M)
`modernbert`	394.8
`intfloat/e5-large-v2`	335.1
`BAAI/bge-large-en-v1.5`	335.1
`intfloat/multilingual-e5-large-instruct`	559.9
`sentence-transformers/LaBSE`	470.9
`microsoft/deberta-v3-large`	434.0
`sentence-transformers/sentence-t5-large`	737.7
`sentence-transformers/sentence-t5-base`	222.9
`sentence-transformers/all-roberta-large-v1`	355.4
`Alibaba-NLP/gte-large-en-v1.5`	434.1
`hkunlp/instructor-large`	737.7

4.2 Embedding Models Used

The alignment measure of [6] estimates a single direction w using the differences between class centroids for each CEFR level, and measures the degree of constraint violation by the inner product with w. In this study, we implemented the aforementioned "exactw" as well as "most2," which uses the difference between the centroids of the two most frequent labels. The embedding models used for comparison are shown in Tab. 1. Except for ModernBERT, the names correspond to the model names on Hugging Face (`https://huggingface.co/`).

[1] `https://en.wikipedia.org/wiki/Common_European_Framework_of_Reference_for_Languages`

5. EXPERIMENTAL RESULTS

Let us first restate the goal of this study. We first assign difficulty labels using an LLM. We then use various methods to identify, from those difficulty labels, the cases that are likely to disagree with human raters. Because the CEFR-SP dataset used in this study contains two human raters, we conducted experiments for both disagreement with Rater A and disagreement with Rater B. As noted above, higher values indicate better performance in predicting disagreement for all evaluation metrics, including ROC-AUC, PR-AUC, and Precision@20, which we abbreviate as p@20 in the tables. Tab. 2 and Tab. 3 show the results when GPT-OSS-120B was used for difficulty labeling, whereas Tab. 4 and Tab. 5 show the results when Qwen3 was used for difficulty labeling. Methods with *logprob* in their names are the conventional baselines that use the token generation probabilities of each LLM's output labels. Because no previous method has proposed using embedding vectors separate from the LLM itself, all other methods are proposed methods.

The *exactw* method uses, as the direction of $\mathbf{w}$, the average of all pairwise constraints. In contrast, *most2* uses as $\mathbf{w}$ the direction of the difference between the mean vectors of the two levels that were assigned most frequently and second most frequently by the LLM.

First, across all four combinations of GPT-OSS-120B or Qwen3 with Rater A or B, modernbert (most2) achieved higher PR-AUC than the conventional logprob-based method. Previously, the only option was to use the confidence scores output by the LLM at the time of response generation, and once these were not collected during generation, it was difficult to obtain them again afterward. By contrast, the proposed method makes it possible to obtain, after the response has already been generated, a more effective indicator of confidence for LLM-as-a-Judge. This is a novel and practically useful result.

Second, *for most embedding models, including modernbert, most2 performed better than exactw.* This is an important finding. A likely reason is that the difficulty level labels assigned by each LLM are not evenly distributed. Rather than measuring consistency with the embedding space while also taking into account instances assigned to infrequent levels, better performance was obtained, from the perspective of detecting the magnitude of disagreement with human raters, by closely examining the levels most frequently assigned by the LLM and measuring their consistency with the embedding space.

Furthermore, although various embeddings were tested, *overall, modernbert showed strong performance.* ModernBERT [14] is a method that comprehensively improves upon BERT, and like BERT, it is designed not for a specific dataset but for broad use of embedding vectors across a wide range of tasks. The proposed method in this study uses modernbert in a way that is consistent with its original design intention. It is therefore reasonable to interpret the result as showing that, even with a relatively small number of parameters, an embedding model used in accordance with its intended design is the most useful.

6. DISCUSSION AND LIMITATIONS

Table 2: Evaluation metrics for GPT-OSS-120B (Rater A)

Method (Rater)	ROC-AUC	PR-AUC	p@20
logprob_label (A)	0.561	0.579	0.80
p_i exactw / chain (A)	0.509	0.528	0.55
modernbert [6] most2	0.606	0.652	1.00
modernbert [6] exactw	0.424	0.503	0.80
`e5-large-v2`	0.578	0.624	1.00
`bge-large-en-v1.5`	0.575	0.614	1.00
`multilingual-e5-large-instruct`	0.602	0.652	1.00
`LaBSE`	0.588	0.637	1.00
`deberta-v3-large`	0.589	0.622	0.95
`sentence-t5-large`	0.568	0.604	1.00
`sentence-t5-base`	0.571	0.610	1.00
`all-roberta-large-v1`	0.575	0.608	0.95
`gte-large-en-v1.5`	0.574	0.611	0.95
`instructor-large`	0.577	0.618	1.00

Table 3: Evaluation metrics for GPT-OSS-120B (Rater B)

Method (Rater)	ROC-AUC	PR-AUC	p@20
logprob_label (B)	0.558	0.668	0.95
p_i exactw / chain (B)	0.511	0.653	0.85
modernbert [6] most2 / chain (B)	0.547	0.690	1.00
modernbert [6] exactw / chain (B)	0.427	0.578	0.70
`e5-large-v2`	0.552	0.676	1.00
`bge-large-en-v1.5`	0.535	0.666	1.00
`multilingual-e5-large-instruct`	0.548	0.683	1.00
`LaBSE`	0.551	0.685	1.00
`deberta-v3-large`	0.540	0.675	1.00
`sentence-t5-large`	0.530	0.665	1.00
`sentence-t5-base`	0.531	0.665	1.00
`all-roberta-large-v1`	0.539	0.661	1.00
`gte-large-en-v1.5`	0.535	0.665	0.95
`instructor-large`	0.530	0.668	1.00

The proposed method should be understood as a method for detecting potentially unreliable difficulty labels in settings where the target difficulty construct is ordinal and is at least partly reflected in an independent representation space. In this study, we evaluated the method on a CEFR-based English sentence difficulty assessment. This is a favorable setting for the proposed approach because the target construct is linguistic difficulty, and text embedding models such as ModernBERT are expected to encode lexical, syntactic, and semantic properties that are relevant to language proficiency levels.

This scope is important. The proposed method is not intended to be a domain-independent solution to all forms of educational difficulty assessment. In particular, difficulty in Science, Technology, Engineering, and Mathematics (STEM) domains may depend on conceptual prerequisites, problem-solving procedures, mathematical structure, required operations, common misconceptions, and the interaction between textual and non-textual representations. These factors may not be adequately captured by standard text embeddings.

Table 4: Evaluation metrics for Qwen3 (Rater A)

Method (Rater)	ROC-AUC	PR-AUC	p@20
logprob_label (A)	0.567	0.539	0.55
p_i exactw / chain (A)	0.525	0.508	0.60
[6] most2 / chain (A)	0.593	0.591	0.90
[6] exactw / chain (A)	0.482	0.509	0.75
`e5-large-v2`	0.534	0.537	0.80
`bge-large-en-v1.5`	0.540	0.538	0.75
`multilingual-e5-large-instruct`	0.567	0.568	0.95
`LaBSE`	0.545	0.544	0.95
`deberta-v3-large`	0.565	0.551	0.80
`sentence-t5-large`	0.538	0.532	0.55
`sentence-t5-base`	0.543	0.537	0.60
`all-roberta-large-v1`	0.544	0.536	0.75
`gte-large-en-v1.5`	0.540	0.538	0.80
`instructor-large`	0.542	0.539	0.85

Table 5: Evaluation metrics for Qwen3 (Rater B)

Method (Rater)	ROC-AUC	PR-AUC	p@20
logprob_label (B)	0.539	0.586	0.85
p_i exactw / chain (B)	0.501	0.558	0.75
[6] most2 / chain (B)	0.537	0.610	0.85
[6] exactw / chain (B)	0.424	0.508	0.30
`e5-large-v2`	0.520	0.581	0.70
`bge-large-en-v1.5`	0.506	0.577	0.70
`multilingual-e5-large-instruct`	0.526	0.592	0.75
`LaBSE`	0.533	0.584	0.75
`deberta-v3-large`	0.523	0.585	0.75
`sentence-t5-large`	0.503	0.569	0.75
`sentence-t5-base`	0.496	0.567	0.80
`all-roberta-large-v1`	0.503	0.570	0.70
`gte-large-en-v1.5`	0.505	0.574	0.75
`instructor-large`	0.503	0.573	0.80

In such cases, geometric consistency in a generic embedding space may reflect lexical or semantic similarity rather than the educational difficulty of the item.

Therefore, applying the proposed method to STEM difficulty assessment would require additional domain-specific assumptions and validation. One possible direction is to replace the generic text embedding space with a representation that incorporates prerequisite relations, knowledge-component annotations, worked-solution structure, or other features that are known to be relevant to the target domain. Another requirement is a reliable ordinal difficulty scale with sufficient agreement among human raters, comparable in role to CEFR in the present study.

Accordingly, the main contribution of this paper is not that embedding geometry universally determines educational difficulty. Rather, the contribution is to show that, in language-based difficulty assessment, an independently obtained embedding space can provide a practical and probability-free signal for prioritizing LLM-generated ratings that are likely to disagree with human raters. Relatedly, language-resource work has shown that leveled wordlists can encode different register assumptions that are recoverable from general corpora [8], and inverse-embedding work has suggested that difficulty-related directions in embedding spaces may be diverse rather than unique [7]. Extending the present disagreement detection approach beyond language difficulty assessment is an important direction for future work, but it should be treated as a domain-specific research problem rather than as a direct application of the present method.

7. CONCLUSION

In this study, we proposed a reliability indicator for individual difficulty labels assigned by LLM-as-a-Judge that is easier to obtain and more effective than conventional probability-based approaches. Unlike prior methods, our method requires only the LLM-assigned labels and the original text, without relying on generation-time confidence scores. The method exploits the fact that difficulty is an ordinal scale and measures, for each label, the geometric consistency between an additional embedding space and the difficulty labels assigned by the LLM. Experiments on CEFR-SP showed that the proposed method can detect labels likely to disagree with human raters more accurately than conventional methods. It is also memory-efficient, because it uses only text embedding models with fewer than 1B parameters.

As future work, although this study was evaluated on English text difficulty assessment, extending the method to STEM subjects will require establishing a rigorous difficulty scale comparable to CEFR across human raters. If such a scale can be developed, the proposed method may also be applied to STEM difficulty assessment.

8. ACKNOWLEDGMENTS

This work was supported by JST PRESTO, Japan, Grant Number JPMJPR2363, and by JSPS KAKENHI Grant Numbers JP22K12287 and JP26K15085. We are deeply grateful to the anonymous reviewers for their constructive feedback.

References

[1] Y. Arase, S. Uchida, and T. Kajiwara. CEFR-based sentence difficulty annotation and assessment. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6206–6219, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[2] G. H. Chen, S. Chen, Z. Liu, F. Jiang, and B. Wang. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics.

[3] C.-H. Chiang, W.-C. Chen, C.-Y. Kuan, C. Yang, and H.-y. Lee. Large language model as an assignment evaluator: Insights, feedback, and challenges in a 1000+ student course. In Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2489–2513, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics.

[4] T. Deutsch, M. Jasbi, and S. Shieber. Linguistic features for readability assessment. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–17, Seattle, WA, USA → Online, July 2020. Association for Computational Linguistics.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[6] Y. Ehara. Educational cone model in embedding vector spaces. In *Proc. of ICCE 2025 (short paper)*, 2025.

[7] Y. Ehara. Generating diverse difficulty examples in embedding vector spaces via inverse embedding. In A. I. Cristea, E. Walker, Y. Lu, O. C. Santos, and S. Isotani, editors, *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium, Blue Sky, and WideAIED*, pages 68–76, Cham, 2025. Springer Nature Switzerland.

[8] Y. Ehara. Recovering registers from leveled wordlists. In S. Piperidis, N. Bel, H. van den Heuvel, N. Ide, S. Krek, and A. Toral, editors, *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, pages 1143–1154, Palma, Mallorca, Spain, May 2026. European Language Resources Association (ELRA).

[9] K. Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[10] H. Hashemi, J. Eisner, C. Rosset, B. Van Durme, and C. Kedzie. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13806–13834, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics.

[11] P. Manakul, A. Liusie, and M. Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore, Dec. 2023. Association for Computational Linguistics.

[12] OpenAI. Introducing gpt-oss. OpenAI website, 2025.

[13] V. Raina, A. Liusie, and M. Gales. Is LLM-as-a-judge robust? investigating universal adversarial attacks on zero-shot LLM assessment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7499–7517, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics.

[14] B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, A. Gallagher, R. Biswas, F. Ladhak, T. Aarsen, G. T. Adams, J. Howard, and I. Poli. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In W. Che, J. Nabende, E. Shutova, and M. T. Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547, Vienna, Austria, July 2025. Association for Computational Linguistics.

[15] A. Yang et al. Qwen3 technical report (`https://arxiv.org/abs/2505.09388`), 2025.

[16] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NeurIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Process-Level Analysis of Block-Based Robotics Programming Using Markov Chains

Leon Stjepan Uroic
University of Zagreb Faculty of Electrical Engineering and Computing
leon-stjepan.uroic@fer.hr

Liljana Puskar
University of Zagreb Faculty of Electrical Engineering and Computing
liljana.puskar@fer.hr

Ana Sovic Krzic
University of Zagreb Faculty of Electrical Engineering and Computing
ana.sovic.krzic@fer.hr

ABSTRACT

Understanding student problem-solving strategies in educational robotics is important for developing adaptive instructional support. However, most research in this area focuses on final outcomes rather than the underlying processes. In this exploratory paper, we use first-order Markov chains to model programming strategies used by high school students working with LEGO SPIKE Prime robots. Fine-grained interaction traces are transformed into structured action sequences and compared between low- and high-performing student groups using likelihood-ratio testing and standardized residual analysis. Even in the first task, the Markov chains reveal statistically significant behavioral differences that grow with task complexity, with high-performing students showing more systematic, goal-directed behavior and low-performing students showing more searching, reorganization, and trial-and-error. These exploratory findings suggest that process data from robotics environments can reveal meaningful differences well before final outcomes.

Keywords

Educational robotics, Markov chains, student programming strategies, block-based programming

1. INTRODUCTION

Educational robotics draws heavily on constructivism and Papert's idea that learners develop powerful ideas by building and debugging tangible artifacts [14]. Multiple reviews confirm that robotics activities can improve engagement and learning outcomes [3, 2, 11]. However, the vast majority of studies evaluate these benefits through pre- and post-tests or final solution assessments [13, 20]. This focus on outcomes is at odds with the constructivist premise: in a robotics programming task, iterative debugging and refinement is central, not incidental. Two students can arrive at the same correct solution through very different sequences of exploration, debugging, and self-regulation. An analysis of these processes, rather than only their endpoints, may yield actionable insights for instructors and adaptive systems.

Liljana Puskar, Leon Stjepan Uroic, and Ana Sovic Krzic. Process-Level Analysis of Block-Based Robotics Programming Using Markov Chains. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 754–759. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039754

Process-level analysis has gained traction in block-based programming research more broadly. Dong et al. [6] analyzed trace data from Snap! to detect struggling moments and identify when intervention might be needed. Filvà et al. [7] modified Scratch to log every interaction and used those traces to identify behavior profiles and predict performance. Fu et al. [8] showed that process features extracted from MixToy improve student performance prediction. Beyond block-based environments, trace-based analysis has a longer history in text-based programming contexts, where IDE interaction logs have been used to study debugging strategies [10, 4] and to detect productive versus unproductive programming patterns [15]. However, research that applies interaction trace analysis to educational robot programming specifically remains sparse. Scaradozzi et al. [17] collected and analyzed data from LEGO Mindstorms EV3 workshops to extract problem-solving pathways across student teams. We applied clustering to programming trace data from robotics workshops to study students' programming strategies [21].

Given the sequential nature of trace data, Markov chains are a natural modeling choice. Hensen et al. [9] modeled interactions with a math learning platform as a single Markov chain per student. The resulting chains were subsequently clustered to draw conclusions about different behavioral types. Markov chain transition probabilities have also been used as inputs for a neural network trained to predict student performance [16]. Aside from simple Markov models, Hidden Markov Models are also widely used for modeling learning processes [5, 19], with the hope that hidden states can represent higher-order behaviors than raw interaction logs.

In this work, we present an exploratory process-level analysis of students' block-based programming with LEGO Education SPIKE Prime robots [12]. We propose a four-step pipeline: (1) extraction of semantically meaningful action sequences from raw interaction traces, (2) construction of task-specific first-order Markov chains for high- and low-performing groups, (3) likelihood-ratio testing to identify statistically significant group differences, and (4) standardized residual analysis to pinpoint the transitions that contribute most to these differences. **Our contribution is threefold**: (1) we present, to our knowledge, the first application of Markov-chain process analysis to block-based educational robotics programming, extending a method established in screen-only block environments [7, 8] to a tangible-artifact context; (2) we demonstrate that strategy differences are

detectable in the very first task, before any problem-solving demands, suggesting that early process signals can precede outcome signals by an entire workshop; (3) we identify specific transition patterns (run–reorganize–run, remove-and-rebuild, attention-spreading across irrelevant block types) that are candidate features for adaptive support.

2. DATA COLLECTION AND SEQUENCE EXTRACTION

A total of 97 first-year students (average age = 15.22, SD = 0.35) from a public high school participated[1] in the study during regular class hours. Each 90-minute lesson was delivered as part of the existing programming curriculum. Students programmed a pre-built LEGO SPIKE Prime Break Dancer robot in the accompanying block-based environment based on Scratch. The programming environment features a left panel containing available blocks categorized by type and a right-side canvas where programs are constructed by dragging, connecting, modifying, and removing blocks.

Participants completed six programming tasks arranged in increasing difficulty, structured to build foundational concepts while managing cognitive load. Task 1 introduced the programming interface by asking students to copy a simple motor program from a reference image. Task 2 required students to independently program both arm and leg motors. Task 3 asked students to set specific parameter values (e.g., move legs by 375° at 30% speed and arms for 5 s at 50% speed). Tasks 4–6 progressed into simultaneous motor coordination, conditional logic with sensors, and an open-ended capstone activity. Throughout the workshop, we captured mouse and keyboard interaction data, took screenshots at each mouse event, and recorded complete program snapshots following every modification.

The raw data was transformed into action sequences representing students' interactions with the programming environment. We extracted five action types: block ADDED, block REMOVED, block MOVED, block MODIFIED, and RUN. The RUN action was detected via a computer vision algorithm that identified yellow circular UI elements (the run button) in captured screenshots. The remaining actions were derived by comparing abstract syntax trees (ASTs) of consecutive program snapshots. ADDED indicated a new block on the canvas, MODIFIED indicated a parameter change within an existing block, and REMOVED and MOVED captured deletion or repositioning. Because MOVED and REMOVED actions could affect multiple blocks simultaneously (e.g., dragging a root block also moves all subsequent blocks), we aggregated them into single actions while recording the number and types of affected blocks.

After extracting sequences and evaluating each student's solutions, 76 complete instances remained for analysis. The reduction from 97 to 76 resulted from excluding students with incomplete data: those who missed part of the workshop, experienced technical failures in data recording, or whose solutions could not be reliably evaluated. Summary statistics are shown in Table 1.

[1]The study protocol was reviewed and approved by the Ethics Committee of the University of Zagreb Faculty of Electrical Engineering and Computing. Informed consent in accordance with GDPR was obtained from both participating students and their parents or legal guardians prior to the workshop. All interaction data were pseudonymized at collection, and no personally identifying information is retained in the analysis dataset.

Table 1: Summary Statistics of the Data.

Statistic	Value
Average sequence length	280.09 ± 111.52 events
Number of unique states	38
Total transitions counted	45,396
High performers (top 25%)	20 students
Low performers (bottom 25%)	17 students

3. METHOD

Our main goal in this exploratory study is to ascertain whether students' programming process can be meaningfully analyzed using trace data from the LEGO block-based environment. To this end, we categorized students into high-performing (top 25% by assessment score) and low-performing (bottom 25%) groups and tested for statistical differences between their processes. We adopted this extreme-group design rather than a median split because our exploratory aim is to surface qualitatively distinct strategic patterns: trimming the middle 50% maximizes behavioral contrast between groups and reduces noise from students near the score median, who tend to differ less in programming strategy than in incidental factors such as time on task or momentary attention. This design is well suited to identifying which transitions distinguish strategies; estimating population-level effect sizes is left to future work with larger samples. Because students varied substantially in time on task and total number of actions, raw transition counts are not directly comparable across groups. We therefore constructed, for each group and task, a first-order Markov chain in which states corresponded to the action–block-type combinations described above, and based all subsequent analyses on transition probabilities rather than raw counts.

Transition probabilities were estimated by tallying the frequency of each transition across all students within a given group. Although the first-order assumption is clearly simplified, transition probabilities are influenced by many factors beyond the immediately preceding state, more complex models would be impractical given the relatively small sample size per task. The first-order model proved sufficient to capture meaningful patterns.

The MOVED and REMOVED actions presented a methodological challenge, as they could affect multiple block types simultaneously (e.g., moving a motor block that is connected to a control block moves both). To address this, we counted all pairwise combinations of affected states. For instance, if a student moved motor and sensor blocks and subsequently removed event and control blocks, we incremented the count for each combination: *Moved Motor* → *Removed Event*, *Moved Motor* → *Removed Control*, *Moved Sensor* → *Removed Event*, and *Moved Sensor* → *Removed Control*. Because this pairwise expansion could in principle inflate transition counts for connected subgraphs, we examine its

impact in Section 4.4 by comparing against an alternative encoding that attributes each composite event only to the block type that the student directly grabbed.

A likelihood-ratio test [1] was used to determine whether observed differences between the two groups were statistically significant rather than attributable to sampling variability. This test is particularly important for early tasks, where transition counts are low due to the simplicity of the activities. Following a significant difference, we computed standardized residuals:

$$r_{ij} = \frac{O_{ij} - E_{ij}}{\sqrt{E_{ij}}}, \tag{1}$$

where O_{ij} and E_{ij} denote observed and expected transition counts under the null hypothesis that both groups share the same underlying Markov chain. Expected counts were computed by pooling all sequences, calculating transition frequencies, normalizing them per state, and multiplying by each group's total transitions. Differences between groups were quantified as:

$$r_{ij}^{\textbf{diff}} = r_{ij}^{\text{high}} - r_{ij}^{\text{low}}. \tag{2}$$

Standardized residuals quantify the difference between the observed counts and the counts that we would expect to see if both groups shared the same underlying stochastic process. Furthermore, standardized residual analysis accounts for differing group sizes and normalizes transition counts into comparable distributions, allowing it to identify which transitions drive these differences.

Positive values indicate transitions more frequent among high performers; negative values indicate transitions more common among low performers. Following [18], $|r_{ij}^{\text{diff}}| > 2$ is considered notable and $|r_{ij}^{\text{diff}}| > 3$ highly notable.

4. RESULTS

The high- and low-performer groups originate from statistically different Markov chains on all six tasks ($p < 0.01$). Throughout, we interpret the observed transition patterns in terms of the programming environment's mechanics and the task requirements; these interpretations are intended as plausible explanations to guide future research rather than as established causal claims. We restrict the analysis to Tasks 1–3 because these tasks have the most complete data, all 76 students contributed sequences, whereas later tasks are affected by attrition among low-performing students who did not finish the workshop.

4.1 Task 1: Copying a reference program

The standardized residual differences for Task 1 are shown in Figure 1. Although this task involved only copying code from a reference image and required no debugging or problem-solving, the estimated Markov chains for high- and low-performing students differed significantly ($G = 54.44$, $df = 33$, $p < 0.01$).

The largest residual difference corresponds to the transition Modified Motor → Added Motor ($r = -3.04$), indicating that low-performing students frequently added new motor blocks after modifying existing ones. In the programming environment, this transition means a student changed parameters on an existing motor block and then dragged a new motor block from the side menu onto the canvas. Since the task only requires placing the correct blocks and setting values, repeatedly returning to the menu after modifying a block suggests uncertainty about which block is needed. In contrast, high-performing students more often followed the transition Added Motor → Modified Motor ($r = 2.51$), reflecting a more systematic strategy of first selecting the appropriate block and then refining its parameters.

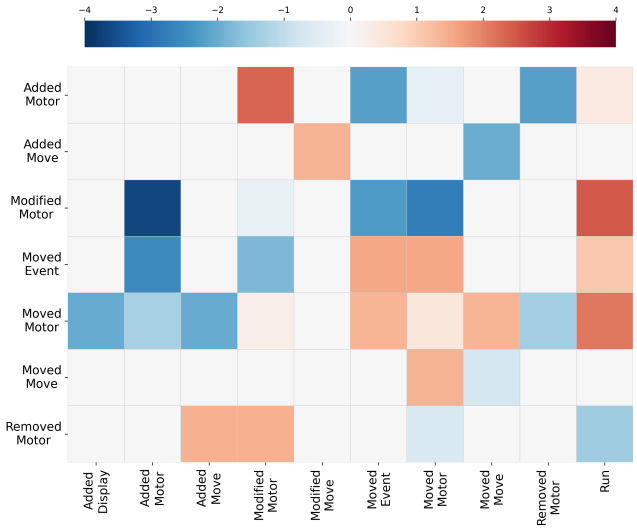


Figure 1: Standardized residual differences (high − low) for Task 1. Rows denote current states and columns denote subsequent states; states with $|r| < 1.0$ are omitted.

Additional low-performer transitions involved event block movements: Moved Event → Added Motor ($r = -2.56$) and Added Motor → Moved Event ($r = -2.09$), suggesting that these students spent time repositioning the program's starting event block alongside their block-searching behavior. Notably, these differences emerge despite the absence of genuine problem-solving demands in the task, suggesting that they reflect differences in familiarity with the interface or approach to the task rather than debugging strategy.

4.2 Task 2: Independent multi-motor programming

Figure 2 shows the standardized residual differences for Task 2 ($G = 62.35$, $df = 40$, $p < 0.01$). In this task, students were asked to independently program both arm and leg motors, and behavioral differences became more pronounced as students began reorganizing their programs.

Low-performing students exhibited substantially more transitions involving movement and modification of move blocks, particularly Added Move → Modified Move ($r = -3.06$) and Modified Move → Moved Move ($r = -3.61$). These patterns indicate frequent restructuring of code: students added move blocks, changed their parameters, and then repositioned them on the canvas. This behavior is consistent with a trial-and-error approach in which students tested partial solutions and subsequently reorganized blocks when the outcome did not match expectations. Importantly, move

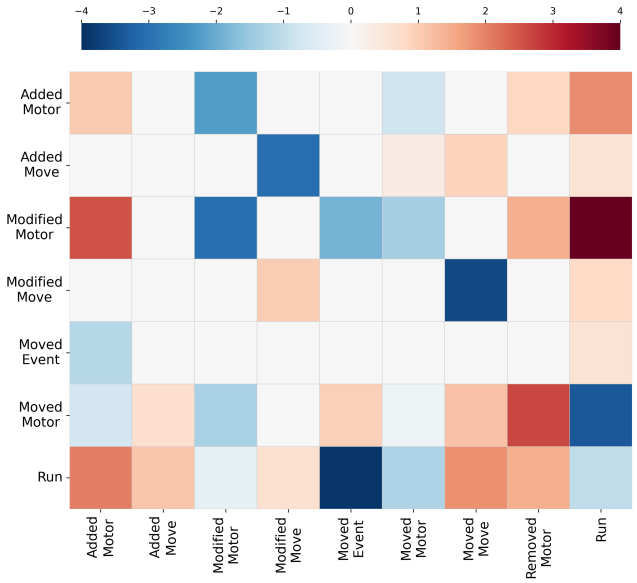


Figure 2: Standardized residual differences (high − low) for Task 2. Rows denote current states and columns denote subsequent states; states with $|r| < 1.0$ are omitted.

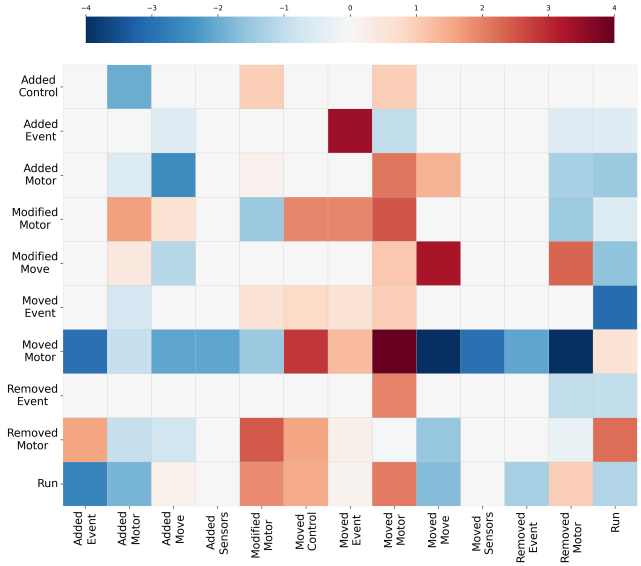


Figure 3: Standardized residual differences (high − low) for Task 3. Rows denote current states and columns denote subsequent states; states with $|r| < 1.9$ are omitted.

blocks were not covered in Task 1, and the subsequent tasks were designed to be solvable using only the previously introduced motor blocks. While Task 2 could technically be solved using move blocks, none of the students successfully did so, meaning these move block manipulations represent unproductive exploration.

A further notable transition was Run → Moved Event ($r = -3.90$), indicating that low-performing students frequently repositioned event blocks immediately after running their programs. Combined with Moved Motor → Run ($r = -3.03$), this pattern suggests a cycle of run–reorganize–run characteristic of unsystematic debugging. High-performing students, by contrast, showed fewer such reorganizations and more direct transitions between adding motor blocks and running (Added Motor → Run, $r = 1.98$; Run → Added Motor, $r = 2.04$), indicating a more stable and anticipatory approach to constructing the program.

4.3 Task 3: Parameter tuning and block ordering

Figure 3 presents the standardized residual differences for Task 3 ($G = 174.54$, $df = 104$, $p < 0.001$). This task required students to set specific motor parameters (e.g., 375° at 30% speed for legs, 5 s at 50% speed for arms), and the common error was placing the speed block after the movement block rather than before it. By this point, differences between groups were both larger in magnitude and more diverse in nature.

A key difference emerged in how students handled block ordering. Low-performing students more often moved motor blocks and then removed them (Moved Motor → Removed Motor, $r = -5.06$), the largest residual for this task, suggesting recognition of incorrect ordering without successfully correcting it through repositioning. Instead, they deleted blocks and re-added them. In contrast, high-performing students more frequently performed consecutive move operations on motor blocks (Moved Motor → Moved Motor, $r = 3.73$), indicating deliberate reordering of the existing program structure rather than removal and reconstruction. High performers also showed a stronger Moved Motor → Run transition ($r = 2.59$), suggesting that after reordering blocks they tested the result, consistent with a hypothesis-driven approach.

Low-performing students were more likely to follow motor movements with structural changes to unrelated block types (Moved Motor → Moved Move, $r = -3.26$; Moved Motor → Moved Sensors, $r = -2.66$; Moved Motor → Added Event, $r = -2.66$), spreading their attention across multiple block categories rather than focusing on the motor blocks central to the task. They also more frequently added motor blocks after running (Run → Added Motor, $r = -2.82$), consistent with a cycle of testing and then starting over with new blocks rather than modifying the existing program.

4.4 Sensitivity to composite-action encoding

To assess whether the pairwise encoding of composite MOVED and REMOVED actions biased the results, we recomputed the Markov chains under a *root-only* alternative in which only the block type that the student directly grabbed contributes to the transition; downstream blocks in the dragged subtree are ignored. This scheme avoids any inflation of transition counts from connected subgraphs at the cost of discarding information about which block types co-occur in composite operations.

The qualitative findings are stable across both encodings. The likelihood-ratio tests remain significant on every task under both schemes (Task 1: $p = 0.011/0.020$; Task 2: $p = 0.013/0.019$; Task 3: both $p < 10^{-4}$), and the ranking, magnitude, and sign of the largest residuals are preserved (Table 2). Across Tasks 1–3, 8–9 of the top-10 transitions per task appear in both schemes; transitions that move in or out are predominantly those near the notability threshold of $|r^{\text{diff}}| = 2$. This indicates that the strategy differences reported in Sections 4.1–4.3 reflect genuine behavioral contrasts rather than artifacts of how composite actions are decomposed.

Table 2: Stability of the largest residuals across encoding schemes (Tasks 1–3). Values are r^{diff} under the pairwise (default) and root-only schemes. The final column indicates whether the transition appears in the top-10 of both schemes for the corresponding task.

T	Transition	Pair.	Root	Top-10
1	Mod. Motor → Add. Motor	−3.04	−3.10	both
1	Add. Motor → Mod. Motor	2.51	2.16	both
1	Mov. Event → Add. Motor	−2.56	−2.00	both
1	Add. Motor → Rem. Motor	−2.09	−2.30	both
1	Add. Motor → Mov. Event	−2.09	−2.30	both
2	Run → Mov. Event	−3.90	−4.02	both
2	Mod. Move → Mov. Move	−3.61	−3.61	both
2	Add. Move → Mod. Move	−3.06	−2.99	both
2	Mov. Motor → Run	−3.03	−2.60	both
2	Mod. Motor → Mov. Motor	−2.70	−2.07	both
3	Mov. Control → Rem. Control	−6.07	−4.82	both
3	Mov. Motor → Rem. Motor	−5.06	−5.37	both
3	Mov. Motor → Mov. Motor	3.73	3.08	both
3	Mod. Move → Mov. Move	3.66	3.61	both
3	Run → Add. Motor	−2.82	−2.85	both
3	Mov. Motor → Run	2.59	2.54	both

5. DISCUSSION

The Markov-chain analysis surfaces three patterns that together sketch a coherent picture of how high- and low-performing students engage with the programming environment. We discuss each in turn, relate it to prior work, and consider its implications for instructional design.

The most striking finding is that Task 1, which required only copying a reference program, already produces statistically significant group differences. Low-performing students repeatedly returned to the block palette after modifying existing blocks (*Modified Motor* → *Added Motor*), while high performers more often followed the reverse sequence. Because the task has no debugging component, the contrast cannot be attributed to differential problem-solving skill; a more plausible reading is that it reflects differences in interface familiarity or in how students plan a sequence of actions before executing it. The practical implication is that process-level signals can precede outcome-level signals by an entire workshop, echoing Jadud's broader argument [10] that trace data serves as a *formative* indicator where assignments and exams only offer *summative* ones, and opening a window for instructional support before students have visibly fallen behind.

A second pattern emerges in Task 2, where low performers exhibit a clear *Run* → *Moved Event* → *Run* cycle together with disproportionate exploration of move-type blocks that none of the students used productively. This is precisely the kind of behavior Dong et al. [6] identify as a "struggling moment" in Snap!: extended periods of activity without meaningful progress, which they argue warrant proactive intervention. Our contribution is to show that these patterns are detectable in a robotics environment with a substantially smaller behavioral vocabulary (38 action–block-type combinations), and that the transitions themselves carry the signal. A natural application is real-time detection: when a student's recent trace shows a high rate of *Run* → *Move* transitions on irrelevant blocks, an adaptive system could surface a reflection prompt asking the student to predict the outcome before the next run.

Task 3 produces the largest residual in the study: low performers recognized incorrect block ordering but resolved it by deletion and reconstruction (*Moved Motor* → *Removed Motor*) rather than repositioning. High performers instead produced consecutive move operations followed by a run, consistent with hypothesis-driven debugging. This mirrors the distinction Piech et al. [15] draw between alpha and gamma trajectory clusters, though their patterns emerge at the level of entire development paths while ours localize to specific transitions. Pedagogically, explicit instruction that blocks can be reordered without being removed could reduce the cost of correction for students who default to deletion. More generally, these transitions are candidate features for the kind of adaptive support that Fu et al. [8] build from aggregate event and sequence features, with the advantage that transition-level signals are directly interpretable as moments where intervention could be inserted.

Taken together, these patterns suggest transition-based indicators that could drive lightweight adaptive support without requiring full sequence modeling, whether through automated prompts or instructor dashboards. The patterns are also entirely behavioral; combining trace data with psychological instruments, eye-tracking, or verbal protocols in future workshops would help disambiguate whether the differences reflect cognitive load, prior experience, or self-regulation.

6. CONCLUSION

This exploratory work-in-progress demonstrates that first-order Markov chains applied to fine-grained programming traces can reveal meaningful behavioral differences between high- and low-performing students in educational robotics, with differences detectable already in the first task, before any problem-solving is required. The consistency of the patterns across tasks suggests Markov chain analysis is a promising tool for surfacing strategic differences well before final outcomes, and the pipeline itself is not specific to the LEGO SPIKE Prime environment but applies to any block-based platform supporting program snapshot extraction. The primary limitation is that data comes from a single workshop context; future work should replicate the analysis across different cohorts and curricula, and evaluate the identified transition patterns as real-time indicators for adaptive instructional support.

7. ACKNOWLEDGMENTS

This work was supported by the Croatian Science Foundation under the project number HRZZ-IP-2022-10-1915 DESCARTES, European Regional Development Fund under grant agreement PK.1.1.10.0007 (DATACROSS). The work of doctoral student Leon Stjepan Uroić has been fully supported by the "Young researchers' career development project – training of doctoral students" of the Croatian Science Foundation DOK-2025-02-8664.

8. REFERENCES

[1] T. W. Anderson and L. A. Goodman. Statistical inference about markov chains. *The Annals of Mathematical Statistics*, 28(1):89–110, 1957.

[2] S. Anwar, N. A. Bascou, M. Menekse, and A. Kardgar. A systematic review of studies on educational robotics. *Journal of Pre-College Engineering Education Research (J-PEER)*, 9(2):Article 2, 2019.

[3] F. B. V. Benitti. Exploring the educational potential of robotics in schools: A systematic review. *Computers & education*, 58(3):978–988, 2012.

[4] P. Blikstein. Using learning analytics to assess students' behavior in open-ended programming tasks. In *Proceedings of the 1st international conference on learning analytics and knowledge*, pages 110–116, 2011.

[5] K. Damevski, H. Chen, D. Shepherd, and L. Pollock. Interactive exploration of developer interaction traces using a hidden markov model. In *Proceedings of the 13th International Conference on Mining Software Repositories*, pages 126–136, 2016.

[6] Y. Dong, S. Marwan, P. Shabrina, T. Price, and T. Barnes. Using student trace logs to determine meaningful progress and struggle during programming problem solving. *International Educational Data Mining Society*, 2021.

[7] D. A. Filvà, M. A. Forment, F. J. García-Peñalvo, D. F. Escudero, and M. J. Casañ. Clickstream for learning analytics to assess students' behavior with scratch. *Future Generation Computer Systems*, 93:673–686, 2019.

[8] Q. Fu, W. Tang, Y. Zheng, H. Ma, and T. Zhong. Predicting programming performance by using process behavior in a block-based programming environment. *Interactive Learning Environments*, 32(6):2371–2385, 2024.

[9] C. Hansen, C. Hansen, N. Hjuler, S. Alstrup, and C. Lioma. Sequence modelling for analysing student interaction with educational systems. *arXiv preprint arXiv:1708.04164*, 2017.

[10] M. C. Jadud. Methods and tools for exploring novice compilation behaviour. In *Proceedings of the second international workshop on Computing education research*, pages 73–84, 2006.

[11] S. E. Jung and E.-s. Won. Systematic review of research trends in robotics education for young children. *Sustainability*, 10(4):905, 2018.

[12] LEGO Education. Lego education spike prime set 45678. Robotics kit, 2020. Available at: `https://education.lego.com/en-us/products/lego-education-spike-prime-set/45678`, accessed February 9, 2026.

[13] S. Lu, Z. Nuryana, X. Ni, W. Xu, and M. N. Alias. A decade of educational robotics: trends and sdg contributions. *Humanities and Social Sciences Communications*, 12(1):1–16, 2025.

[14] S. A. Papert. *Mindstorms: Children, computers, and powerful ideas*. Basic books, 2020.

[15] C. Piech, M. Sahami, D. Koller, S. Cooper, and P. Blikstein. Modeling how students learn to program. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 153–160, 2012.

[16] N. Rohani, K. Gal, M. Gallagher, and A. Manataki. Early prediction of student performance in a health data science mooc. In M. Feng, T. KÃ¤ser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 325–333, Bengaluru, India, July 2023. International Educational Data Mining Society.

[17] D. Scaradozzi, L. Cesaretti, L. Screpanti, and E. Mangina. Identification of the students learning process during education robotics activities. *Frontiers in Robotics and AI*, 7:21, 2020.

[18] D. Sharpe. Your chi-square test is statistically significant: Now what? *Practical Assessment, Research & Evaluation*, 20(8):1–10, 2015.

[19] B. Shih, K. R. Koedinger, and R. Scheines. Unsupervised discovery of student learning tactics. In *Proceedings of the 3rd International Conference on Educational Data Mining*, pages 201–210, 2010.

[20] N. Spolaôr and F. B. V. Benitti. Robotics applications grounded in learning theories on tertiary education: A systematic review. *Computers & Education*, 112:97–107, 2017.

[21] L. S. Uroić, L. Puskar, and A. Sovic Krzic. Preliminary analysis of student programming strategies in an educational robotics workshop. In *16th IEEE Integrated STEM Education Conference*, 2026.

Prompt Sensitivity in LLM-Based Essay Scoring is Model-Specific

Giulio Corsi
University of Cambridge
Cambridge, UK
gc540@cam.ac.uk

Alexandru Marcoci
University of Cambridge
Cambridge, UK
am3159@cam.ac.uk

Maryam Abo-Tabik
Univ. of Central Lancashire
Preston, UK
mabo-tabik@lancashire.ac.uk

Roni Tibon
University of Nottingham
Nottingham, UK
Roni.Tibon@nottingham.ac.uk

Yael Benn
Manchester Metropolitan Univ.
Manchester, UK
Y.Benn@mmu.ac.uk

Deborah Talmi
University of Cambridge
Cambridge, UK
dt492@cam.ac.uk

ABSTRACT

The emergence of Large Language Models (LLMs) has renewed interest in automated essay scoring, yet a barrier to adoption persists: LLM-generated scores are sensitive to prompt formulation, and it is unclear which aspects of a prompt drive this effect or whether findings from one model or institution transfer to another. For educators and institutions considering deployment, this uncertainty is consequential, as without understanding the structure of prompt sensitivity, there is no principled basis for deciding whether to trust an LLM-generated score. This study evaluates three frontier LLMs (GPT-5.4, Claude Opus 4.6, Gemini 3 Flash) across a $3 \times 3 \times 3$ factorial design varying criteria specificity, calibration intervention, and scoring strategy on a calibration sample of 153 undergraduate essays from three UK universities with distinct marking conventions. A main-effects ANOVA on the 27 per-model conditions reveals that prompt sensitivity is substantial but strikingly model-specific: criteria specificity accounts for the largest share of condition-level variance for Gemini (57%, $p < .001$), scoring strategy for GPT (35%, $p < .01$), and scoring strategy and calibration together for Claude (32% and 24% respectively), with no single dimension dominating across models. The optimal configuration also differs across institutions, even within the same model. At each institution, the gap between best and worst configurations spans 4–5 RMSE points on the calibration set, with the worst performing worse than a naïve baseline. For the models tested, these findings indicate that no universal prompt recipe exists, and that responsible deployment requires empirical prompt calibration as a prerequisite rather than an optional refinement – a process more similar to hyperparameter tuning than to simple instruction design.

Giulio Corsi, Alexandru Marcoci, Maryam Abo-Tabik, Georgiana Thorpe Apreutesei, Lyba Razzaq, Roni Tibon, Yael Benn, and Deborah Talmi. Prompt Sensitivity in LLM-Based Essay Scoring is Model-Specific. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 760–764. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039772

Keywords

automated essay scoring, large language models, prompt engineering, hyperparameter tuning

1. INTRODUCTION

The emergence of LLMs has renewed interest in automated essay scoring, with recent work demonstrating that individual models can approximate human scores on standardised tasks [5, 3]. However, a practical barrier to deployment persists: LLM-generated scores are sensitive to prompt formulation, with minor variations in rubric framing or role instructions capable of shifting outputs by a full grade band [1]. For institutions considering the use of LLM-based scoring, this sensitivity is consequential yet poorly understood, and it is unclear whether the effect follows patterns that can inform principled prompt design or whether it is contingent on the specific model and context in use.

Despite this known limitation of LLM-based grading, prompt design for essay scoring is typically treated as a fixed authoring choice, selected on the basis of practitioner intuition. Where studies do evaluate prompt effects, they tend to do so by sequentially adding prompt components within a single model and selecting whichever configuration maximises agreement – an approach that can identify that prompts matter, but not which dimensions of a prompt matter, nor whether those dimensions generalise across models [6, 4]. This leaves two open questions. The first is about decomposition: given that prompt formulation affects LLM-generated scores, which types of prompt intervention are actually responsible for the effect, and how much of the variance does each explain? The second is about generalisability: do the answers to the first question hold across models, or is prompt sensitivity model-contingent in ways that make universal recommendations impossible? The distinction is consequential: if one prompt dimension consistently dominates, practitioners can follow a simple prompting framework; if the dominant dimension varies by model, then empirical calibration is not a convenience but a necessity.

This study addresses that question directly. Three prompt dimensions – targeting what evaluative structure the model

receives, how it is distributionally anchored, and how it is asked to reason – are independently varied in a $3 \times 3 \times 3$ factorial design across three LLMs, yielding 81 model-prompt conditions evaluated on a calibration set of 153 undergraduate essays from three UK institutions. The factorial structure permits decomposition of not only which prompt components matter, but whether they matter in the same way across models.

The contribution of this work, and what attendees will see at the poster, is twofold. First, a per-model variance decomposition that makes the dimensions of prompt sensitivity visible rather than aggregating them into a single "prompts matter" headline. Second, an empirical demonstration that the optimal prompt configuration differs not only across models but also across institutions with distinct marking conventions, with the practical implication that institutions cannot import a prompt validated elsewhere and expect it to behave consistently on their own essays. The poster will display the full grid of 81 conditions alongside the per-institution best/worst gaps, allowing attendees to inspect the model-specific patterns directly and to consider what an empirical calibration workflow would look like in their own context.

2. METHODS

2.1 Dataset

The data is drawn from a corpus of 764 long-form undergraduate psychology essays from three UK universities, spanning 50 modules. All were online submissions of coursework and exam scripts marked by human assessors using the standard UK percentage scale with discrete tick marks (e.g., 62, 65, 68, 72; $M = 63.5$, $SD = 10.1$). A stratified 80/20 split produced a calibration set ($n = 153$, approximately 50 per institution) and a held-out test set ($n = 611$). All analyses in this paper use the calibration set only. The decision to reserve the test set is deliberate rather than incidental: the research question addressed here concerns the structure of prompt sensitivity, which requires evaluating the same essays under all 81 model–prompt conditions and decomposing the resulting variance. The held-out essays are reserved for a separate, downstream question – whether configurations selected through calibration generalise to unseen submissions – which will be addressed in subsequent work and would be premature to evaluate before the present decomposition is established.

2.2 Factorial Design

The three LLMs used – GPT-5.4 (OpenAI), Claude Opus 4.6 (Anthropic), and Gemini 3 Flash (Google) – were selected from different providers to maximise diversity of biases and failure modes. Three prompt dimensions were varied in a fully crossed $3 \times 3 \times 3$ design (Figure 1). The dimensions were chosen to span three theoretically distinct sources of prompt influence: *what* evaluative structure the model receives (criteria specificity), *how* it is distributionally anchored (calibration intervention), and *how* it is asked to arrive at a judgement (scoring strategy). Together, these cover the input-side, distributional, and process-level determinants of LLM scoring behaviour.

(A) Criteria specificity addresses how much evaluative structure the model receives. Prior work has shown that rubric provision can either improve consistency or introduce dimension level artefacts [1], but the relationship between rubric granularity and accuracy has not been isolated. Three levels were tested: no rubric (A1), summary grade boundaries (A2; e.g., "70+: demonstrates critical analysis and original insight"), and the full analytic rubric for a given assignment with per-dimension performance descriptors (A3).

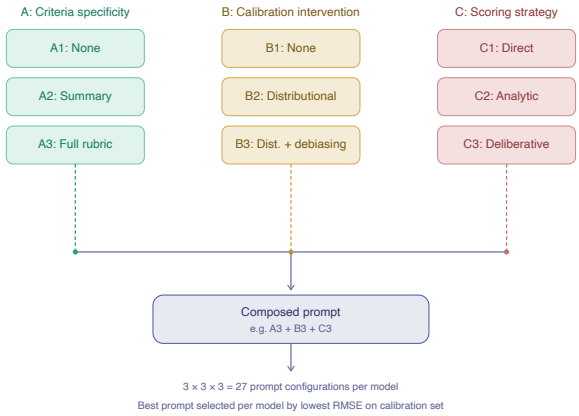


Figure 1: Factorial prompt composition. Three dimensions are crossed in a $3 \times 3 \times 3$ design, yielding 27 prompt configurations per model.

(B) Calibration intervention addresses the well-documented tendency of LLMs to compress scores towards the centre of the scale [2], which can in principle be mitigated by providing distributional anchoring. Three levels were tested: no calibration (B1), distributional base-rate information (B2; e.g., "approximately 25% of essays in this cohort receive a First"), and distributional information with an explicit debiasing instruction (B3; e.g., "...be aware that LLMs tend to compress scores towards the mean").

(C) Scoring strategy addresses how the model is asked to arrive at its judgement, drawing on the distinction between analytic and holistic scoring approaches in educational assessment. Three levels were tested: holistic-direct scoring with no structured reasoning (C1), analytic-decomposed scoring in which the model evaluates the essay separately on substantive quality, use of evidence, and clarity of expression before synthesising a single overall score (C2), and holistic-deliberative scoring in which the model first identifies the appropriate grade band, weighs evidence for and against that classification, and then assigns a specific mark within it (C3).

Each model scored the calibration set ($n = 153$) under all 27 conditions. The best-performing prompt per model was selected by lowest RMSE, independently for each institution. Both prompt selection and the variance decomposition reported below use this same calibration partition. This is appropriate to the research question – which prompt dimensions drive scoring variance – but means that the per-institution best/worst comparisons reflect in-sample performance and are subject to optimistic selection bias, as discussed in Section 3.1.

Table 1: Best and worst model–prompt conditions per institution on the calibration set ($n = 153$). Naïve baseline = predicting the cohort mean. Best/worst are selected as min/max RMSE across 27 conditions; the observed gap is therefore an in-sample upper bound on the true best–worst difference (see text).

Institution	Condition	Model	RMSE	BA (%)	Δ RMSE	Δ BA
Inst. A	Best (A3_B3_C1)	Gemini	5.84	74.1		
	Worst (A1_B1_C1)	Claude	10.94	48.1	+5.09	−25.9
	Naïve baseline	–	6.95			
Inst. B	Best (A3_B1_C3)	GPT	9.74	42.4		
	Worst (A3_B2_C1)	Gemini	13.89	32.2	+4.16	−10.2
	Naïve baseline	–	11.44			
Inst. C	Best (A1_B3_C1)	Gemini	6.40	50.0		
	Worst (A1_B1_C2)	Claude	10.19	32.4	+3.79	−17.6
	Naïve baseline	–	7.02			

3. RESULTS

3.1 The Magnitude of Prompt Effects

Before decomposing which prompt dimensions drive scoring variance, it is important to establish the practical magnitude of the effects at stake. Table 1 presents the best and worst model–prompt conditions for each institution on the calibration set. The in-sample gap between optimal and worst-case conditions spans 3.8–5.1 RMSE points and up to 26 percentage points of band accuracy. At Institution A, the best condition (Gemini, A3_B3_C1) achieves an RMSE of 5.84 with 74.1% band accuracy; the worst (Claude, A1_B1_C1) yields 10.94 and 48.1%.

An important caveat applies: because the "best" and "worst" configurations are selected as the extremes across 27 conditions, the observed gap is subject to selection optimism and will tend to overstate the true best–worst difference on new data. The gap is too large to be explained entirely by sampling variability – the best conditions consistently fall below the naïve baseline while the worst fall above it – but the precise magnitude should be interpreted as an upper bound. Future work will evaluate whether the gap holds on the reserved test set ($n = 611$).

The comparison with the naïve baseline – predicting the cohort mean for every essay – is what makes these differences consequential rather than merely descriptive. The best-calibrated conditions fall below this baseline, indicating that they extract genuine evaluative signal from the essay text. The worst conditions fall clearly above it, meaning that a poorly chosen prompt actively destroys signal that a trivial heuristic would preserve. The difference between a good and bad configuration is therefore not one of marginal refinement but of whether the system functions at all – a point that the poster will make visible by displaying the full distribution of the 27 per-model conditions against the baseline, so that the magnitude of the best–worst spread can be inspected rather than inferred.

Table 2: Condition-level ANOVA: proportion of variance (η^2) in per-condition RMSE explained by each prompt dimension, estimated separately for each model ($n = 27$ conditions per model, 20 residual df). Bootstrap 95% CIs (10,000 resamples) in brackets. $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$.

Dimension	Claude	GPT	Gemini
A: Criteria	ns	ns	57%*** [29, 78]
B: Calibration	24%* [3, 48]	ns	ns
C: Strategy	32%** [6, 56]	35%** [8, 60]	ns

3.2 Prompt Sensitivity is Model-Specific

The central question of this work is whether prompting effects follow general patterns – in which case a universal best-practice recommendation would be possible – or whether they are model-contingent. A condition-level ANOVA on the 27 per-model RMSE values (Table 2) decomposes variance by prompt dimension under a main-effects model. An important caveat applies throughout: with only 27 observations per model and 20 residual degrees of freedom, the η^2 values are point estimates with wide confidence intervals, not precise effect sizes; we report bootstrap 95% CIs (10,000 resamples) to make this uncertainty explicit. With that in mind, the pattern across models is striking. For Gemini, criteria specificity accounts for the largest share of condition-level variance (57%, 95% CI [29, 78], $p < .001$). For GPT, scoring strategy accounts for 35% (95% CI [8, 60], $p < .01$). For Claude, scoring strategy and calibration together account for over half (32%, 95% CI [6, 56], $p < .01$; and 24%, 95% CI [3, 48], $p < .05$). No single dimension is consistently dominant, and dimensions that are significant for one model are non-significant for another.

Two-way interactions (A×B, A×C, B×C) were tested in a full model including all main effects and two-way terms (residual $df = 8$). No individual interaction reached significance at $\alpha = .05$ for any of the three models, and the

combined interaction terms did not improve model fit over the main-effects specification. Given the limited residual degrees of freedom, these null results should not be taken as evidence that interactions are absent, but they support treating the main-effects decomposition as a reasonable approximation.

This pattern is corroborated by a repeated-measures analysis at the essay level ($n = 153$, each scored under all 27 conditions). Gemini shows the largest total prompt sensitivity (partial $\eta^2 = 2.0\%$, driven by criteria at 1.5%, $p < .001$), Claude shows moderate sensitivity (1.2%, spread across calibration and strategy), and GPT shows the least (0.4%). The within-essay effects are naturally small in relative terms – essay quality dominates individual score variance, as it should. But the practical consequence of prompt sensitivity operates at the system level, not the essay level: small per-essay biases that are systematic across a cohort aggregate into the 4–5 point RMSE differences reported in Table 1. A prompt configuration that introduces a consistent directional bias, for instance, may shift each essay by only 2–3 marks, but across a cohort this is enough to meaningfully alter the RMSE. The model-specificity of the pattern is consistent across both levels of analysis, confirming that the condition-level decomposition reflects a real structural difference rather than an artefact of low power.

With that caveat, the practical implication is clear: scoring strategy appears as the largest single dimension for both Claude and GPT, but the two models differ in which other dimensions matter (calibration for Claude, none for GPT), and Gemini is driven by an entirely different dimension (criteria specificity). A practitioner who reads that "scoring strategy is the most important prompt dimension" and applies this as a general rule will miss the dimension that matters most for Gemini and misconfigure calibration for Claude. For the three models tested here, universal prompt guidelines do not merely oversimplify – they actively mislead.

4. CONCLUSION

Prompt sensitivity in LLM-based essay scoring appears to be real, substantial, and model-specific. No single prompt dimension dominates across the three models tested: scoring strategy is the primary driver for both Claude and GPT, but Claude is additionally sensitive to calibration, while Gemini is driven by an entirely different dimension – criteria specificity. The gap between optimal and worst-case configurations spans 4–5 RMSE points per institution on the calibration set, with poorly chosen prompts performing worse than a naïve baseline – in effect, making the difference between a scoring system that extracts evaluative signal and one that removes it completely. For the models tested here, the results suggest that no universal prompt recipe exists, and that empirical calibration may be better understood as a prerequisite for considering deployment than as an optional refinement.

This pattern carries implications that extend beyond predictive accuracy. Because the optimal configuration varies across both models and institutions, a prompt validated in one setting and imported into another may yield systematically different scores for substantively similar work, with consequences for grading consistency at the cohort level and for the comparability of marks across institutions. The institution-level variation in optimal configurations also suggests that prompt sensitivity is not a purely technical artefact but interacts with local marking conventions, which raises fairness considerations when LLM-generated scores are used to compare students who have been assessed under different rubrics or grade distributions. These implications are tentative given the calibration-only scope of the present analysis, but they indicate that the choice of prompt may need to be treated as part of the assessment design itself rather than as a downstream implementation detail. Attendees at the poster will be able to inspect the per-institution differences directly and to consider what an empirical calibration workflow might look like in their own assessment context.

Several limitations constrain these conclusions. All reported comparisons are in-sample: the "best" configuration per model-institution pair is selected by lowest RMSE on the same calibration set used for evaluation, introducing optimistic selection bias. The magnitude of the best–worst gap should therefore be read as an upper bound, and the findings as preliminary evidence rather than deployment-ready conclusions; held-out validation on the reserved test set ($n = 611$) is needed to establish a de-biased estimate. The calibration set comprises approximately 50 essays per institution, yielding η^2 estimates with wide confidence intervals (Table 2); the qualitative pattern of model-specificity is more robust than any individual percentage. Interaction effects between prompt dimensions were tested and found to be non-significant, but with only 8 residual degrees of freedom the test has limited power, and the per-dimension attributions should be treated as approximate. The findings are limited to three models from three providers evaluated at a single point in time, and to undergraduate psychology essays marked under UK conventions; whether the model-specificity pattern extends to other disciplines, marking scales, or model generations warrants further investigation. These limitations notwithstanding, the core practical implication appears clear: prompt configuration for LLM-based scoring may be better treated as an empirical calibration problem than as an authoring decision.

5. ADDITIONAL AUTHORS

Additional authors: Georgiana Thorpe Apreutesei (University of Cambridge, email: `gt485@cam.ac.uk`) and Lyba Razzaq (Manchester Metropolitan Univ., email: `lyba.razzaq@stu.mmu.ac.uk`).

6. REFERENCES

[1] M. Abujadallah, M. Saad, and S. Abudalfa. Evaluating open-source llms for automated essay scoring: The critical role of prompt design. 2025.

[2] S. Lee, Y. Cai, D. Meng, Z. Wang, and Y. Wu. Unleashing large language models' proficiency in zero-shot essay scoring. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 181–198, 2024.

[3] W. Li and H. Liu. Applying large language models for automated essay scoring for non-native japanese.

Humanities and Social Sciences Communications, 11(1):1–15, 2024.

[4] P. Y. Liew and I. K. Tan. On automated essay grading using large language models. In *Proceedings of the 2024 8th international conference on computer science and artificial intelligence*, pages 204–211, 2024.

[5] A. Mizumoto and M. Eguchi. Exploring the potential of using an ai language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2(2):100050, 2023.

[6] W. Xia, S. Mao, and C. Zheng. Empirical study of large language models as automated essay scoring tools in english composition_taking toefl independent writing task for example. *arXiv preprint arXiv:2401.03401*, 2024.

Scalable Argumentative Writing Support: The Efficacy of Small Open-Source Models in K-12 Writing Instruction

Venkata Nagaraju Buddarapu
North Carolina State University
vbuddar@ncsu.edu

Damilola Babalola
North Carolina State University
djbabalo@ncsu.edu

Collin Lynch
North Carolina State University
cflynch@ncsu.edu

ABSTRACT

Writing is an essential skill across grade levels and disciplines. However instructors face critical challenges in providing constructive real-time feedback due to high workloads. While AI services may offer scalable writing support, institutional data privacy and costs remain significant barriers to widespread adoption. This study evaluates the efficacy of small open-source LLMs ($\leq$ 8B parameters) in generating rubric-aligned pedagogical feedback for student essays. Using the PERSUADE corpus (n=25,996), we benchmark small open-source LLMs(SLMs) including Mistral, Gemma, Phi, and LLaMA against proprietary baselines. Our results indicate that these models achieve 68–78% of proprietary performance in feedback quality. We conclude that SLMs represent a viable foundation for scalable, privacy-preserving essay instruction.

Keywords

Automated Writing Evaluation, Large Language Models, Open Source AI, Feedback Quality, Data Privacy.

1. INTRODUCTION

The integration of Artificial Intelligence in education, specifically in **Automated Writing Instruction (AWI) [12]**, offers a potentially transformative solution to the growing crisis in literacy education. Currently, K-12 settings face a dual challenge: declining student writing skills [13] and significant, widespread teacher shortages [36]. This environment has exacerbated a critical "feedback gap," where the demand for personalized, iterative guidance far outpaces the capacity of human educators to provide it.

Researchers have demonstrated that proprietary AI services such as GPT-4 [24] and Claude Sonnet [6] possess the interactive capabilities required to address these complex pedagogical tasks [2]. However, the adoption of these models is often precluded by high operational costs, reliance on external infrastructure, and legitimate data privacy concerns. Consequently, there is an urgent need for SLMs that can serve as private, resource-efficient engines for AWI, capable of providing high-order structural and content feedback at scale.

Within the domain of AWI, generating rubric-aligned feedback for argumentative essays remains a significant challenge. To be effective, models must transcend surface-level linguistic corrections and engage with the "argumentative core" of a text. We evaluate several SLMs such as Mistral, Gemma, Phi, and LLaMA on feedback generation and lead extraction using the PERSUADE corpus. Traditionally, assessing pedagogical feedback requires expert human evaluation, a process that is subjective, time-consuming, and difficult to scale. To mitigate this, we utilize GPT-4o as an automated judge, benchmarking these models against proprietary baselines such as GPT-4 and Claude 3.5 Sonnet (Sonnet 3.5) across four dimensions: specificity, constructiveness, tone, and actionability. Beyond proxy measurements, our study explores how lead extraction provides an additional performance signal for feedback quality when human-gold standards are unavailable. By analyzing the relationship between structural discourse detection accuracy and GPT-4o's feedback quality signal, we investigate whether these SLMs can bridge the feedback gap, successfully replicating the multi-dimensional guidance typically provided by human educators. While SLMs vary in tone and constructiveness, they demonstrate competitive performance in actionability. We further discuss enhancing these capabilities via LoRA [17] and ORPO [15] fine-tuning. This study addresses two primary research questions:

- **RQ1:** How effectively can SLMs replicate the feedback quality of proprietary models across key rubric dimensions?
- **RQ2:** To what extent does a model's accuracy in identifying the "argumentative core" (lead extraction) predict the quality of its generated feedback?

2. RELATED WORK

To address this pedagogical crisis, we must reconcile the reasoning capabilities of LLMs with the constraints of privacy-preserving, local deployment. Below, we review the evolution of language modeling and the current methodologies for Automated Writing Instruction (AWI) that define this research space

Proprietary vs. SLMs: While proprietary models like GPT-4

Venkata Nagaraju Buddarapu, Collin Lynch, and Damilola Babalola. Scalable Argumentative Writing Support: The Efficacy of Small Open-Source Models in K-12 Writing Instruction. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 765–769. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040119

and Sonnet 3.5 set the state-of-the-art for reasoning and dialogue [25, 5], their adoption in K-12 is limited by high costs and data privacy risks [9]. This has catalyzed research into SLMs such as Llama-3, Mistral, Gemma, and Phi-3 [35, 19, 11, 14]. Optimization techniques like Quantization and Low-Rank Adaptation (LoRA) [16] further enable these models to match proprietary performance on domain-specific tasks while maintaining local, private deployment.

Automated Writing Instruction (AWI): AWI has transitioned from shallow, feature-based scoring (e.g., e-rater [7]) to deep generative tutoring [20, 31]. Effective AWI relies on *argument mining* the segmentation of discourse elements like leads, claims, and evidence [30, 21]. While compact models can achieve near-GPT-4 accuracy in discourse classification [27], their capacity to translate this structural detection into high-order pedagogical feedback remains an active area of investigation.

Evaluating Feedback Quality: Pedagogical feedback is multidimensional, requiring assessment across specificity, tone, and actionability [32]. Given the subjectivity and cost of human expert grading and recent work utilizes Large-Model Judges (e.g., G-Eval [22]). We utilize GPT-4 as a reference signal. We treat this as a reasonable baseline while remaining cautious regarding its direct generalization to pedagogical feedback tasks.

Despite progress in discourse classification, the pedagogical reliability of SLMs remains unproven. This study bridges this gap by evaluating a diverse roster of SLMs using the PERSUADE corpus. We investigate lead extraction as a structural proxy for feedback utility measured by specificity, constructiveness, tone, and actionability benchmarked against a GPT-4o reference judge. Isolating lead extraction provides a clean baseline for structural awareness, as effective hooks are strong predictors of essay coherence [10], while avoiding the confounding complexities of multi-span relational tasks [29].

3. METHODS

In this section, we describe the experimental workflow, which integrates model selection and prompt engineering with an evaluation framework.

3.1 Dataset and Model Selection

We utilize the PERSUADE (Persuasive Essays for Rating, Selecting, and Understanding Argumentative Discourse) corpus ($n = 25,996$) [10]. This dataset is particularly suited for our study as it contains student writing across multiple prompts and genres, with each essay featuring a gold-standard *Lead*, the foundational "hook" or thesis providing the ground truth for our structural extraction tasks. The corpus represents a diverse range of student proficiencies (Grades 6–12), as detailed in Table 1. We benchmark a roster of SLMs against proprietary baselines to evaluate the trade-offs between computational efficiency and feedback performance with model specifications outlined in Table 2.

3.2 Prompt Design:

Our pipeline utilized three distinct zero-shot prompt architectures. For *Lead Extraction*, we employed a strict *verbatim constraint*, instructing models to transcribe the exact introductory string without paraphrasing or error correction to ensure metrics reflected architectural alignment. For *Generative Feedback*, we utilized a *Role-Context-Task* framework, anchoring models as expert tutors and providing PERSUADE 2.0 rubric definitions to standardize formative output. Finally, for *Feedback Scoring*, the GPT-4o judge evaluated the student essays and generated feedback using granular Likert-scale rubrics (1–5) detailed in Section 3.3.

Table 1: PERSUADE 2.0 Corpus

Category	Details / Value
Total Essays (N)	25,996
Sample Size (n)	1,000 (Randomly Sampled)
Unique Prompts	15 (e.g., Electoral College)
Grade Levels	6–12 (Middle/High School)
Avg. Length	350–500 words
Elements	Lead, Position, Claim, Counterclaim, Rebuttal, Evidence, Conclusion

3.3 Evaluation Framework

To standardize the automated assessment, the GPT-4o judge applied a 1–5 Likert scale across four pedagogical dimensions developed in consultation with writing instructors. These definitions, detailed below, prioritize revision utility and student motivation:

Specificity: Measures the transition from vague, boilerplate phrases (Score 1) to explicit referencing or quoting of the student's unique text (Score 5).

Constructiveness: Evaluates whether the model moves beyond binary signaling (Score 1) to provide a developmental roadmap explaining the rhetorical "why" behind suggestions (Score 5).

Tone: Assesses the shift from punitive or clinical language (Score 1) to an encouraging, "growth mindset" voice (Score 5).

Actionability: Quantifies the presence of concrete next steps; a Score 5 requires a clear "revision roadmap" with at least two feasible, immediate tasks.

To measure structural identification(lead extraction), we employed two metrics: **SBERT Cosine Similarity** [28] for semantic overlap, and **BLEU** (Bilingual Evaluation Understudy) [26] to measure n-gram precision between the extracted leads and the ground truth. Overall, the reported values for both extraction metrics and feedback quality represent the mean scores calculated across the sampled essays.

3.4 Lead Extraction as a proxy

We prioritize lead extraction as the primary proxy for feedback quality indicator because the lead serves as a critical rhetorical "anchor," requiring the model to establish global context from the outset. While we plan to investigate broader discourse elements such as evidence and counterclaims in future work, the ability to craft an introductory "hook" that previews an argumentative trajectory is a hallmark of rhetorical control [8]; thus, its successful extraction

signals a model's capacity to integrate long-range dependencies and align initial framing with a global thesis.

4. RESULTS

We evaluated model performance across essays using an NVIDIA A100 GPU. The evaluation focused on two primary dimensions: **Lead Extraction** (RQ1), assessing structural identification, and **Feedback Quality** (RQ2), assessing pedagogical utility.

As shown in Table 3, SLMs specifically **LLaMA-3.1-8B** and **Phi-4-mini** surpassed GPT-4o in extraction accuracy. This suggests that structural identification is a mature capability for models $\geq$4B parameters. In the feedback phase, Phi-4-mini proved the most capable SLM, achieving $\sim$68% of the actionability scores recorded by Claude Sonnet.

While SLMs demonstrated high structural accuracy, Table 4 highlights a disparity in feedback substance. SLMs often offer correct but generic advice compared to the targeted revision steps provided by Sonnet 3.5. Furthermore, Phi models occasionally repeated student text verbatim, suggesting that while 8B models are adequate for extraction, tonal nuance and creative synthesis require further tuning.

Notably, we found a correlation (p=0.71) between lead extraction accuracy and feedback generation quality. This suggests that a model's ability to identify lead components can be an indicator of its overall proficiency in providing helpful feedback on argumentative essays such as [10].

5. DISCUSSION

This study demonstrates that compact SLMs represent a viable foundation for scalable, privacy-preserving writing instruction, achieving functional adequacy by matching $\sim$70% of proprietary model performance. While our findings underscore the viability of these models, the observed performance disparities present two critical imperatives for future research.

First, consistent with established correlations between parameter scale and the reasoning capabilities necessary for nuanced feedback [23], our analysis showed a sharp decline when moving from 4-billion to 2-billion parameters. Consequently, future work must investigate whether high-quality training data (synthetic or organic) and specialized instruction tuning (e.g., via LoRA [16] [37]) can offset these parameter deficits to preserve pedagogical efficacy on smaller hardware.

Second, as shown in Table 3, a clear performance hierarchy emerges: LLaMA-3.1-8B and Phi-4-mini dominate the open-source cohort in Lead Extraction (0.87) and Actionability (3.37), respectively, while Gemma-2B represents a performance floor with a collapse in extraction precision (0.22 Cosine). These disparities, contextualized by Table 2, raise critical questions regarding the interplay between architectural "thinness" and pedagogical utility; specifically, it remains to be determined whether the transition from Multi-Query Attention (MQA) [4] and shallow layer depths to Grouped-Query Attention (GQA) [3] and 32-layer profiles is the primary driver of a model's ability to sustain long-range dependencies in argumentative text. While SLMs nearly match proprietary benchmarks in Tone, the persistent "Actionability Gap" suggests that structural awareness alone may not suffice for complex instruction, necessitating further research into whether the observed floor in Gemma-2B is a consequence of its specific attention bottleneck or if parameter density remains the ultimate ceiling for replicating human-level revision roadmaps. Furthermore, does the 8k context window adequate for the PERSUADE corpus constrain holistic feedback for the longer, multi-document assignments prevalent in higher education?

6. LIMITATIONS

The SLMs performance reflects zero-shot capabilities rather than the upper bounds achievable through further prompt engineering and domain-specific fine-tuning. Despite the promising performance of SLMs, several constraints qualify these findings. First, our evaluation relies on the PERSUADE corpus, which consists primarily of argumentative essays from middle and high school students. Consequently, further research is required to evaluate these models across other genres such as narrative or expository writing and within university-level academic discourse to ensure the generalizability of our findings. Second, while GPT-4o serves as a functional proxy for human judgment, its efficacy must be validated in authentic classroom settings for inherit proprietary biases that favor "polished" prose, potentially overlooking subtle pedagogical nuances.

7. CONCLUSION

In conclusion, while SLMs demonstrate significant potential for providing scalable and private writing instruction, their deployment requires targeted research into architecture, datasets, and specialized fine-tuning. By bridging the performance gap between compact models and larger proprietary counterparts, we can develop accessible, high-fidelity tools that provide expert-level pedagogical support. Ultimately, these models offer a viable path toward mitigating instructor workloads while ensuring high-quality, private writing instruction is accessible at scale.

8. REFERENCES

[1] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, M. Harrison, R. J. Hewett, M. Javaheripi, P. Kauffmann, J. R. Lee, Y. T. Lee, Y. Li, W. Liu, C. C. T. Mendes, A. Nguyen, E. Price, G. de Rosa, O. Saarikivi, A. Salim, S. Shah, X. Wang, R. Ward, Y. Wu, D. Yu, C. Zhang, and Y. Zhang. Phi-4 technical report, 2024.

[2] G. Adams, A. Fabbri, F. Ladhak, E. Lehman, and N. Elhadad. From sparse to dense: Gpt-4 summarization with chain of density prompting, 2023.

[3] J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Fedus, S. Sumers, and A. Deshpande. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*, 2023.

[4] J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Lebrón, and S. Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints, 2023.

[5] Anthropic. Claude model card, 2023.

Table 2: Model Roster and Architectural Specifications

Model	Source	Params	Layers	Heads	Attention	Context
Proprietary Baselines						
Sonnet 3.5	Anthropic	–	–	–	–	200k
GPT-4o	OpenAI	∼175B*	–	–	–	128k
SLMs						
LLaMA-3.1-8B [34]	Meta	8B	32	32	GQA	128k
Mistral-7B [18]	Mistral AI	7B	32	32	GQA/SWA	128k
Phi-4-mini [1]	Microsoft	4.2B	32	24	GQA	128k
Phi-3-mini [1]	Microsoft	3.8B	30	32	GQA	128k
Gemma-7B [33]	Google	7B	26	28	MHA	8k
Gemma-2B [33]	Google	2B	18	1	MQA	8k

Table 3: Model Performance Comparison ($n = 1,000$). Best open-source results are bolded and colored.

	Lead Extraction		Feedback Quality (1–5)			
Model	**Cosine**	**BLEU**	**Spec.**	**Constr.**	**Tone**	**Action.**
Proprietary Baselines						
Sonnet 3.5	—	—	4.98	4.94	4.65	4.97
GPT-4o	0.82	0.52	—	—	—	—
SLMs						
LLaMA-3.1-8B	**0.87**	**0.57**	3.69	3.70	4.09	3.22
Phi-4-mini	0.85	0.50	**3.89**	**3.71**	4.09	**3.37**
Phi-3-mini	0.84	0.46	3.47	3.61	4.17	3.15
Mistral-7B	0.77	0.23	3.61	3.58	4.18	3.19
Gemma-7B	0.79	0.49	3.12	3.35	4.20	2.91
Gemma-2B	0.22	0.11	2.78	3.02	3.91	2.60

Table 4: Qualitative Comparison of Generated Feedback (ID: 2241005482)

Model	Feedback Excerpt
Sonnet 3.5	"To strengthen this, cite one specific local park mentioned in your town's budget to ground your argument."
Phi-4-mini	"The essay includes a clear position. However, add more evidence to support claims about the environment."
Gemma-2B	"Good job on this essay. Try to write more sentences and check your spelling next time."

[6] Anthropic. Claude 3.5Sonnet Model Card Addendum. Technical report, Anthropic, 2024.

[7] Y. Attali and J. Burstein. Automated essay scoring with e-rater v.2. *Journal of Technology, Learning, and Assessment*, 2006.

[8] A. Becker. Student writing in the disciplines: Graduate students' navigating the learning of local rhetorical contexts. *Journal of Writing Research*, 7(3):345–382, 2016.

[9] R. e. a. Bommasani. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[10] S. Crossley, Y. Tian, P. Baffour, A. Franklin, M. Benner, and U. Boser. A large-scale corpus for assessing written argumentation: Persuade 2.0. *Assessing Writing*, 61:100865, 2024.

[11] G. DeepMind. Gemma: Lightweight open models by google deepmind, 2024.

[12] L. T. Frase. Knowledge, information, and action: Requirements for automated writing instruction. *Journal of Computer-Based Instruction*, 11(2):55–59, 1984.

[13] A. C. Graesser and D. S. McNamara. Literacy challenges in the 21st century: A perspective on writing and reading in digital environments. *Educational Psychologist*, 56(3):134–146, 2021.

[14] S. e. a. Gunasekar. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

[15] J. Hong, N. Lee, and J. Thorne. Orpo: Monolithic preference optimization without reference model, 2024.

[16] E. e. a. Hu. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[17] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models, 2021.

[18] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril,

T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7b, 2023.

[19] E. e. a. Jiang. Mistral: A sparse mixture of experts for efficient language modeling. *arXiv preprint arXiv:2310.06825*, 2023.

[20] E. Kasneci, K. Sessler, H. Krosse, and G. Kasneci. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274, 2023.

[21] Z. e. a. Ke. Automated assessment of student writing: The persuade corpus. In *Proceedings of LREC*, 2022.

[22] J. e. a. Liu. Gpteval: Nlg evaluation using gpt-4 with better prompting and calibration. *arXiv preprint arXiv:2309.03409*, 2023.

[23] L. C. Magister, J. Mallinson, J. Adamek, E. Malmi, and A. Severyn. Teaching small language models to reason. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, Toronto, Canada, July 2023. Association for Computational Linguistics.

[24] OpenAI. Gpt-4 technical report. *OpenAI*, 2023.

[25] OpenAI, J. Achiam, S. Adler, et al. Gpt-4 technical report, 2024.

[26] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA, 2002. Association for Computational Linguistics.

[27] S. Ramachandran, G. Ilharco, Q. Xie, S. Agrawal, K.-W. Chang, and N. A. Smith. Persuaide: Generating and evaluating personalized feedback on student persuasive writing. In *Proceedings of the 2023 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2023.

[28] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[29] C. Stab and I. Gurevych. Annotating argument components and relations in persuasive essays. In J. Tsujii and J. Hajic, editors, *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland, Aug. 2014. Dublin City University and Association for Computational Linguistics.

[30] C. Stab and I. Gurevych. Argumentation mining in persuasive essays and scientific articles from the discourse structure perspective. In *Argument Mining Workshop*, 2017.

[31] M. Stahl, L. Biermann, A. Nehring, and H. Wachsmuth. Exploring llm prompting strategies for joint essay scoring and feedback generation, 2024.

[32] M. Stevenson. A critical interpretative synthesis: The integration of automated writing evaluation into classroom writing instruction. *Computers and Composition*, 42:1–16, 2016.

[33] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, J. Ferret, P. Liu, P. Tafti, A. Friesen, M. Casbon, S. Ramos, R. Kumar, C. L. Lan, S. Jerome, A. Tsitsulin, N. Vieillard, P. Stanczyk, S. Girgin, N. Momchev, M. Hoffman, S. Thakoor, J.-B. Grill, B. Neyshabur, O. Bachem, A. Walton, A. Severyn, A. Parrish, A. Ahmad, A. Hutchison, A. Abdagic, A. Carl, A. Shen, A. Brock, A. Coenen, A. Laforge, A. Paterson, B. Bastian, B. Piot, B. Wu, B. Royal, C. Chen, C. Kumar, C. Perry, C. Welty, C. A. Choquette-Choo, D. Sinopalnikov, D. Weinberger, D. Vijaykumar, D. Rogozińska, D. Herbison, E. Bandy, E. Wang, E. Noland, E. Moreira, E. Senter, E. Eltyshev, F. Visin, G. Rasskin, G. Wei, G. Cameron, G. Martins, H. Hashemi, H. Klimczak-Plucińska, H. Batra, H. Dhand, I. Nardini, J. Mein, J. Zhou, J. Svensson, J. Stanway, J. Chan, J. P. Zhou, J. Carrasqueira, J. Iljazi, J. Becker, J. Fernandez, J. van Amersfoort, J. Gordon, J. Lipschultz, J. Newlan, J. yeong Ji, K. Mohamed, K. Badola, K. Black, K. Millican, K. McDonell, K. Nguyen, K. Sodhia, K. Greene, L. L. Sjoesund, L. Usui, L. Sifre, L. Heuermann, L. Lago, L. McNealus, L. B. Soares, L. Kilpatrick, L. Dixon, L. Martins, M. Reid, M. Singh, M. Iverson, M. Görner, M. Velloso, M. Wirth, M. Davidow, M. Miller, M. Rahtz, M. Watson, M. Risdal, M. Kazemi, M. Moynihan, M. Zhang, M. Kahng, M. Park, M. Rahman, M. Khatwani, N. Dao, N. Bardoliwalla, N. Devanathan, N. Dumai, N. Chauhan, O. Wahltinez, P. Botarda, P. Barnes, P. Barham, P. Michel, P. Jin, P. Georgiev, P. Culliton, P. Kuppala, R. Comanescu, R. Merhej, R. Jana, R. A. Rokni, R. Agarwal, R. Mullins, S. Saadat, S. M. Carthy, S. Cogan, S. Perrin, S. M. R. Arnold, S. Krause, S. Dai, S. Garg, S. Sheth, S. Ronstrom, S. Chan, T. Jordan, T. Yu, T. Eccles, T. Hennigan, T. Kocisky, T. Doshi, V. Jain, V. Yadav, V. Meshram, V. Dharmadhikari, W. Barkley, W. Wei, W. Ye, W. Han, W. Kwon, X. Xu, Z. Shen, Z. Gong, Z. Wei, V. Cotruta, P. Kirk, A. Rao, M. Giang, L. Peran, T. Warkentin, E. Collins, J. Barral, Z. Ghahramani, R. Hadsell, D. Sculley, J. Banks, A. Dragan, S. Petrov, O. Vinyals, J. Dean, D. Hassabis, K. Kavukcuoglu, C. Farabet, E. Buchatskaya, S. Borgeaud, N. Fiedel, A. Joulin, K. Kenealy, R. Dadashi, and A. Andreev. Gemma 2: Improving open language models at a practical size, 2024.

[34] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models, 2023.

[35] H. e. a. Touvron. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[36] U.S. Department of Education. Teacher Shortage Areas. `https://www.ed.gov/teaching-and-administration/professional-development/teacher-shortage-areas`, 2025. Page last reviewed February 6, 2025.

[37] T. e. a. Wang. Orpo: Offline reinforcement learning with preference optimization, 2023. OpenAI Blog.

Separating Math Errors from Language Barriers: The AI Whisperer Effect in Cross-Lingual Knowledge Tracing

Mario Oscar Pureco-Razo
QuizMyBrainz
mar&mar ideas products & more, LLC
oscarpureco@quizmybrainz.com

Antonieta Ceron-Ponce
QuizMyBrainz
mar&mar ideas products & more, LLC
antonieta@quizmybrainz.com

ABSTRACT

When a bilingual second-grader describes a triangle as "pizza" on a geometry assessment, current AI tutoring systems face a classification dilemma: is this a conceptual error, or evidence of a student using everyday vocabulary (Basic Interpersonal Communicative Skills) because they lack the formal mathematical register (Cognitive Academic Language Proficiency)? No existing educational AI system separates mathematical conceptual understanding from second-language processing at the individual response level. We present CL-KT-X, a neuro-symbolic diagnostic engine that performs cross-lingual knowledge tracing through a three-layer architecture: deterministic linguistic fingerprinting, EXTRACT-NORMALIZE-COMPARE algorithmic prompting, and a Truth-First classification router. In validating CL-KT-X, we document a previously unnamed phenomenon—the *AI Whisperer Effect*—in which architecturally similar local LLMs require model-specific prompt engineering to achieve comparable diagnostic accuracy. Validated on 74 authentic responses from bilingual Title I classrooms across two mathematical domains, the system achieves 98.6% classification accuracy with zero Truth-First violations. This work crosses linguistic, disciplinary, and contextual borders to address equity in AI-assisted mathematics assessment for the 5.3 million emergent bilingual students in U.S. schools.

Keywords

Knowledge Tracing, Cross-Lingual Assessment, Algorithmic Prompting, Bilingual Education, Educational Equity

1. INTRODUCTION

Emergent bilingual students—those developing proficiency in English while maintaining their native language—navigate two linguistic systems simultaneously when responding to mathematical assessments. A student who writes "pizza" for triangle is not demonstrating a geometric misconception; she is applying a Basic Interpersonal Communicative Skills (BICS) analogy because she has not yet acquired the Cognitive Academic Language Proficiency (CALP) term in her second language [4, 5]. Current AI-powered assessment systems cannot make this distinction, systematically misclassifying linguistically mediated responses as conceptual errors.

The consequences are measurable. More than 5.3 million emergent bilingual students attend U.S. public schools, with over one million in Grades 2–3 [8]. English learners are identified for special education at elevated rates in unadjusted models (OR $\approx$ 2.56 at Grade 3), attenuating under full covariate adjustment [9]; among those referred for reading concerns, most did not qualify for the referred disability on evaluation (34 of 44) [10]. Traditional knowledge tracing approaches model mastery as a single latent variable [3, 11, 6], capturing *whether* a student errs but not *why*. They cannot distinguish "doesn't understand circles" from "understands circles but calls them *círculos*."

We present CL-KT-X (Cross-Lingual Knowledge Tracing with Explainable Feedback), a neuro-symbolic diagnostic engine that separates L1 conceptual understanding from L2 language processing by routing student responses through a deterministic pipeline where only the extraction phase is LLM-dependent. Our contributions are threefold:

1. A **neuro-symbolic architecture** for cross-lingual mathematical diagnosis, combining deterministic linguistic fingerprinting with constrained LLM extraction and an invariant Truth-First classification router.
2. **Authentic validation** on 74 classroom responses from bilingual Title I students across two mathematical domains, achieving 98.6% classification accuracy—eliminating reliance on synthetic test batteries.
3. Documentation of the **AI Whisperer Effect**, a previously unnamed phenomenon in which architecturally similar sub-10B parameter LLMs exhibit divergent diagnostic accuracy under identical algorithmic prompts.

This work responds directly to the EDM 2026 theme of *Educational Data Mining Across Borders*. CL-KT-X crosses three borders simultaneously: the *language border* between Spanish and English student responses, the *discipline border* integrating computational linguistics with mathematics education and knowledge tracing, and the *context border*

Mario Pureco-Razo, and Antonieta Ceron-Ponce. Separating Math Errors from Language Barriers: The AI Whisperer Effect in Cross-Lingual Knowledge Tracing. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 770–773. International Educational Data Mining Society (2026).
© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039672

between everyday communicative register (BICS) and academic assessment vocabulary (CALP).

2. RELATED WORK

Knowledge tracing has evolved from Bayesian Knowledge Tracing [3] through deep neural architectures [11, 13] to context-aware attentive models [6]. These advances have substantially improved prediction of student performance trajectories, yet they share a structural limitation: knowledge is modeled as a single latent construct. A student transitioning from "unlearned" to "learned" on a fractions concept may have always understood the mathematics but lacked the second-language vocabulary to express it. To the authors' knowledge, no existing knowledge tracing model separates conceptual from linguistic knowledge states.

Bilingual education research has long recognized the challenge of disentangling language proficiency from content mastery [1, 12]. Cummins' BICS/CALP framework [4, 5] establishes that conversational fluency (BICS) is acquired in 1–3 years, while academic register (CALP) requires 5–7 years. In mathematics, a student may operate at CALP level in Spanish but BICS level in English, producing responses that appear mathematically incorrect under monolingual evaluation. Accommodations such as translated assessments improve validity [12] but do not address real-time diagnostic classification at the individual response level.

The application of large language models to educational assessment has expanded rapidly [7], yet these systems treat language as a transparent medium rather than a variable that interacts with domain knowledge. The emerging field of algorithmic prompting—providing LLMs with step-by-step computational procedures rather than task descriptions—has demonstrated gains in mathematical reasoning [14] but has not been applied to cross-lingual diagnostic contexts. Furthermore, the assumption that prompt engineering transfers uniformly across models within the same parameter class has only recently begun to be examined. Our work addresses both gaps: we apply algorithmic prompting to cross-lingual diagnosis and document empirically that this assumption does not hold.

3. SYSTEM ARCHITECTURE

CL-KT-X employs a three-layer neuro-symbolic architecture. **Layer 1: Linguistic Fingerprint**—a deterministic module that detects Spanish-origin tokens, phonetic approximations (e.g., "tree" for *three*), and BICS vocabulary using curated lookup tables, requiring no LLM invocation. **Layer 2: ENC Algorithmic Prompting**—the LLM performs only EXTRACT (identifying the student's mathematical claim from a free-text response); NORMALIZE, COMPARE, and CLASSIFY execute deterministically in Python using curated BICS-to-CALP equivalence tables. **Layer 3: Truth-First Router**—an architectural invariant ensuring that mathematically incorrect answers *always* resolve to Conceptual_Gap regardless of detected language artifacts.

Algorithm 1 presents the ENC pipeline. The critical design decision is the separation of EXTRACT (LLM-dependent) from NORMALIZE-COMPARE-CLASSIFY (deterministic). This constrains LLM hallucination to the extraction phase, where it is intercepted by the normalizer's curated lookup tables before classification. The Truth-First Router (line 5) cannot be overridden by language detection.

Algorithm 1 EXTRACT-NORMALIZE-COMPARE Pipeline

Require: r (student response), a (expected answer), d (domain)
Ensure: $c \in \{\text{Conceptual_Gap}, \text{Linguistic_Gap}, \text{Correct}\}$

1: $F \leftarrow \text{LinguisticFingerprint}(r)$ ▷ Deterministic: L1 tokens, phonetics, BICS
2: $e \leftarrow \text{LLM_Extract}(r, d)$ ▷ LLM-dependent: "What shape is described?"
3: $n \leftarrow \text{Normalize}(e, \textit{BICS_CALP_Table})$ ▷ Deterministic: e.g., "pizza" → "triangle"
4: $m \leftarrow \text{Compare}(n, a)$ ▷ Deterministic: semantic equivalence
5: **if** r is mathematically incorrect **then**
6: **return** Conceptual_Gap ▷ **Truth-First invariant**
7: **else if** $F.\textit{has_L1_artifacts}$ **and** $m = \text{TRUE}$ **then**
8: **return** Linguistic_Gap
9: **else if** $m = \text{TRUE}$ **then**
10: **return** Correct
11: **end if**

4. METHODOLOGY

4.1 Corpus

The validation corpus comprises 74 authentic student responses collected from bilingual Title I elementary classrooms in the [REDACTED] metropolitan area. Two mathematical domains are represented: Geometry (37 vectors) and Fractions (37 vectors). All responses are unedited, raw classroom artifacts produced by emergent bilingual students (Grades 2–5) during regular instructional assessment. No synthetic augmentation or researcher-generated examples were used. To the authors' knowledge, this represents the first authentic cross-lingual mathematical response dataset used to validate an AI diagnostic system.

Response collection involved de-identified archival artifacts from routine instructional assessment conducted by the Co-PI in their capacity as a certified classroom educator. No personally identifiable information was collected or retained. Student data was handled under FERPA-compliant local processing protocols.

4.2 Dual-Review Protocol

Each response underwent a dual-review classification protocol. First, an independent pedagogical classification by a certified bilingual education specialist (15 years of classroom experience, campus-level teaching recognition) assigned ground-truth labels (Conceptual_Gap, Linguistic_Gap, or Correct) based on professional assessment of the student's mathematical understanding and language use. Second, blind CL-KT-X engine processing produced classifications with no access to the specialist's labels. Discrepancies were reconciled by a certified bilingual educator serving as Principal Investigator. The PI override rate of 17.1% across all domains documents productive human-AI collaboration in the diagnostic loop—evidence that the system augments rather than replaces expert judgment.

Gold-standard labels for the expanded corpus ($N = 97$, math and geometry domains, SO-15-01 stratum filter applied) were established through joint expert consensus scor-

Table 1: Domain certification results on authentic classroom responses. All vectors are unedited student artifacts from bilingual Title I classrooms, classified via dual-review protocol.

Domain	Vectors	Correct	Accuracy	TF Viol.
Geometry	37	36	97.3%	0
Fractions	37	37	100.0%	0
Combined	**74**	**73**	**98.6%**	**0**

ing by the two authors, both certified bilingual educators with extensive TELPAS calibration experience (TEA official records). Labels were assigned collaboratively through expert deliberation rather than independent blind annotation; this joint consensus process constitutes the gold standard against which CL-KT-X is evaluated. Human-system agreement between this gold standard and CL-KT-X output was assessed using Cohen's κ, yielding $\kappa = 0.8785$ (95% CI [0.790, 0.954], BCa bootstrap, $B = 20{,}000$ iterations), exceeding the $\kappa \geq 0.80$ threshold established as the minimum for reliable NLP annotation [2]. This metric reflects agreement between the joint expert consensus gold standard and the system output; it is not a measure of independent inter-annotator agreement. Independent multi-annotator validation is identified as future work (Section 6.2).

4.3 Model Configuration

CL-KT-X was validated using three local LLMs deployed via Ollama: Llama 3 (8B parameters), Mistral 7B-Instruct, and Qwen 2.5 (7B parameters). All models received identical ENC algorithmic prompts. Local deployment ensures FERPA compliance—no student data leaves the school network. The deliberate selection of architecturally similar sub-10B parameter models serves a dual purpose: it tests whether the ENC framework enables diagnostic accuracy with resource-constrained models deployable in Title I school infrastructure, and it creates the controlled conditions under which the AI Whisperer Effect becomes observable.

5. RESULTS

5.1 Domain Certification

Table 1 presents classification accuracy across two mathematical domains. Geometry achieved 36/37 correct classifications (97.3%) and Fractions achieved 37/37 (100.0%), yielding a combined accuracy of 73/74 (98.6%) across all authenticated vectors. Zero Truth-First violations were recorded across both domains—no mathematically incorrect response was classified as a linguistic gap. The single geometry misclassification involved a multi-claim response where the LLM extraction phase truncated a semantically complex student utterance, a known boundary condition documented as the BICS Semantic Ceiling.

5.2 The AI Whisperer Effect

Table 2 reveals a previously undocumented phenomenon we term the *AI Whisperer Effect*. Across three architecturally similar local LLMs and three adversarial prompting strategies applied to the same $N = 97$ corpus, unweighted κ ranges from -0.0047 (Llama 3, Role Injection) to $+0.2449$ (Mistral, Role Injection)—a spread of more than 0.25 κ units among models comparable in parameter count (7–8B) and architecture family. No LLM cell exceeds $\kappa = 0.25$ on either metric; all remain far below the XODA baseline of $\kappa = 0.8785$. These models respond differently to identical algorithmic scaffolding, and no prompting strategy rescues performance.

Table 2: Gate 9 adversarial evaluation ($N = 97$ per cell, parse_rate$= 1.000$). Exact%: exact match. Adj%: within ± 1 tier (adjacent agreement; not accuracy). κ_{uw}: unweighted Cohen's κ. CI: 95% BCa ($B = 10{,}000$, seed 42). GTP-2=Role Inject; GTP-3=CoT Override; GTP-4=Lang. Switch. †CL-KT-X evaluated in-sample; LLMs evaluated zero-shot. Δ should be interpreted under this condition asymmetry.

Model	Strat.	Ex%	κ_{uw}	95% CI
Llama 3	GTP-2	27.8	-0.005	$[-0.106,\ 0.110]$
Llama 3	GTP-3	20.6	$+0.015$	$[-0.063,\ 0.102]$
Llama 3	GTP-4	26.8	$+0.002$	$[-0.093,\ 0.109]$
Mistral 7B	GTP-2	52.6	$+0.245$	$[0.146,\ 0.369]$
Mistral 7B	GTP-3	39.2	$+0.173$	$[0.074,\ 0.287]$
Mistral 7B	GTP-4	45.4	$+0.126$	$[0.049,\ 0.236]$
Qwen 2.5	GTP-2	52.6	$+0.244$	$[0.138,\ 0.370]$
Qwen 2.5	GTP-3	26.8	$+0.036$	$[-0.070,\ 0.151]$
Qwen 2.5	GTP-4	40.2	$+0.077$	$[-0.056,\ 0.219]$
CL-KT-X†	—	—	**0.8785**	**[0.790, 0.954]**

This finding challenges the assumption that prompt engineering transfers across models within the same parameter class. We hypothesize that the AI Whisperer Effect emerges from differences in instruction-tuning procedures and tokenizer behavior that manifest specifically under the constrained extraction tasks required by cross-lingual diagnosis. The practical implication is significant: deployed educational AI systems using local LLMs require empirical validation of prompt–model compatibility, not architectural assumption.

5.3 Prompt Evolution

The 98.6% combined accuracy with Llama 3 represents a substantial improvement over the 64.9% baseline achieved with the initial prompt version on the same fractions corpus—a progression spanning four major revisions. Notably, an intermediate revision that improved geometry performance caused a regression in fractions accuracy (94.6% → 83.8%), demonstrating that prompt modifications for one mathematical domain can cascade into regressions in another. This *Prompt Example Cascade* is a subtype of the AI Whisperer Effect under investigation in ongoing work.

6. DISCUSSION AND CONCLUSION

6.1 Implications

CL-KT-X demonstrates that cross-lingual knowledge tracing is achievable with resource-constrained local LLMs when the system architecture separates LLM-dependent extraction from deterministic classification. The neuro-symbolic design constrains hallucination risk to a single pipeline stage, making the system's reasoning transparent and auditable—properties essential for high-stakes educational assessment.

The Truth-First Router provides a hard guarantee that no mathematically incorrect response can be misclassified as a language barrier, an invariant that purely neural approaches cannot offer.

The AI Whisperer Effect has implications beyond bilingual education. Any deployed system using local LLMs for student response analysis must account for model-specific prompt sensitivity, even among architecturally similar models. This finding reframes what might be dismissed as "brittleness" as a systematic property of small-model behavior under algorithmic prompting constraints.

6.2 Limitations

Five limitations warrant explicit disclosure. First, CL-KT-X was evaluated against the same 97-item corpus used during system development; this in-sample evaluation means the $\kappa = 0.8785$ figure reflects performance under favorable conditions. The LLMs reported in Table 2 were evaluated zero-shot on the same items, creating an asymmetric comparison (in-sample system versus zero-shot baselines). This asymmetry does not invalidate the comparison—both XODA and the LLMs were scored against the same joint expert consensus gold standard using the same metric—but readers should interpret the Δ advantage accordingly. Independent held-out evaluation on unseen data is the primary objective of ongoing work. Second, gold-standard labels were established through joint expert consensus by the two co-authors; as the authors also developed the system being evaluated, an inherent conflict of interest exists. Independent blind annotation by researchers unaffiliated with system development is identified as a necessary condition for broader validity claims and is planned as the next annotation phase. Third, the validation corpus of 74 vectors across two domains, while authentic, is modest in scale; expansion to additional domains (decimals, word problems) and grade levels is in progress. Fourth, the AI Whisperer Effect has been documented across three models and four prompting strategies; broader surveys are needed to characterize the phenomenon fully. Fifth, the current system targets mathematics; whether the ENC framework generalizes to other content areas (e.g., science vocabulary) remains an open question. These represent active research directions rather than architectural constraints.

6.3 Conclusion

This work makes three contributions to educational data mining across borders: a neuro-symbolic architecture for cross-lingual mathematical diagnosis, an authentic validation protocol grounded in bilingual classroom practice, and the identification of the AI Whisperer Effect as a systematic phenomenon in local LLM deployment for educational assessment. Future work includes domain expansion, longitudinal tracking of student knowledge states across assessments, and systematic investigation of the Prompt Example Cascade. CL-KT-X represents a step toward equitable AI-assisted assessment for the 5.3 million emergent bilingual students whose mathematical understanding currently risks being lost in translation.

7. ACKNOWLEDGMENTS

The authors thank the bilingual Title I classroom educators whose instructional practice produced the authentic response artifacts that ground this research. This work is conducted independently by mar&mar ideas products & more, LLC / QuizMyBrainz.

8. REFERENCES

[1] J. Abedi. The No Child Left Behind Act and English language learners: Assessment and accountability issues. *Educational Researcher*, 33(1):4–14, 2004.

[2] R. Artstein and M. Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

[3] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995.

[4] J. Cummins. Cognitive/academic language proficiency, linguistic interdependence, the optimum age question and some other matters. *Working Papers on Bilingualism*, 19:121–129, 1979.

[5] J. Cummins. Age on arrival and immigrant second language learning in Canada: A reassessment. *Applied Linguistics*, 2(2):132–149, 1981.

[6] A. Ghosh, N. Heffernan, and A. S. Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2330–2339, 2020.

[7] E. Kasneci, K. Seßler, S. Küchemann, et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274, 2023.

[8] National Center for Education Statistics. English learners in public schools. U.S. Department of Education, 2024. `https://nces.ed.gov/programs/coe/indicator/cgf`

[9] S. O. Ortiz and B. Y. Chow. Disproportionate identification of English learners with specific learning disabilities: Examining grade-level odds and covariate adjustment. *Research in Special Education*, 3, 2026. `https://doi.org/10.25894/rise.2785`

[10] S. O. Ortiz, M. Fránquiz, and A. Lara. Disentangling language from learning disability: Evaluation outcomes for emergent bilinguals referred for reading concerns. *Bilingual Research Journal*, 43(3):245–252, 2020. `https://doi.org/10.1080/15235882.2020.1823734`

[11] C. Piech, J. Bassen, J. Huang, et al. Deep knowledge tracing. *Advances in Neural Information Processing Systems*, 28, 2015.

[12] G. Solano-Flores. Who is given tests in what language by whom, when, and where? The need for probabilistic views of language in the testing of English language learners. *Educational Researcher*, 37(4):189–199, 2008.

[13] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web*, pages 765–774, 2017.

[14] D. Zhou, N. Schärli, L. Hou, et al. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*, 2022.

Simulating Programming Traces and State Spaces for Hint Generation

Jesper Dannath
Bielefeld University
jdannath@techfak.uni-bielefeld.de

Benjamin Paaßen
Bielefeld University
bpaassen@techfak.uni-bielefeld.de

ABSTRACT

Data-driven approaches for next-step hint generation in programming suffer from a cold-start problem, meaning that high-quality hints depend on the presence of sufficiently similar student data – which is missing whenever systems are deployed for new programming tasks. Pre-trained large language models (LLMs) circumvent this cold-start problem. However, because LLMs are not fine-tuned for specific tasks, their hints may contain irrelevant information, or reveal the solution outright. Further, LLMs require substantial resources and long response times to generate hints. We propose to combine the strengths of both LLMs and data-driven approaches: We harness the flexibility of LLMs to generate diverse artificial student solutions. Then we compare methods for simulating student programming traces, thus addressing the cold-start problem. Importantly, we synthesize traces subtractively, starting from the solution and recursively removing elements, thus achieving a fine-grained simulation of programming progress. From simulated traces, state spaces can be constructed and used to generate hints. Our results show that the simulated state spaces are similar to the ones obtained from real student data. Additionally, student step prediction using a simulated state space leads to comparable results to step prediction via prompting LLMs – while being substantially more efficient at inference time.

Keywords

programming traces, state space simulation, programming hints, intelligent tutoring systems

1. INTRODUCTION

A lack of resources for personalized tutoring remains a major challenge in introductory programming education [12]. Personalized hints provided by intelligent tutoring systems during programming tasks can complement teaching and enhance the quality of programming education [23]. AI tools are already used by many programming students for assistance on homework tasks [28]. However, these tools can lead to over-reliance and an offloading of cognitive effort, potentially hindering learning [26]. According to Vygotsgyan pedagogy and scaffolding theory, teachers should give only enough assistance to students such that they can proceed on the given task by themselves [7]. Next step hints provide a framework for applying scaffolding theory in AI-assisted learning environments [11]. Large Language Models (LLMs) can be used to generate next step hints in programming effectively [21] but consume significant resources and even when prompted not to, they tend to give the students access to multiple steps or even complete solutions [1]. On the other hand, data-driven methods for generating next step hints usually rely on a pool of pre-existing student and hint data [11]. However, these approaches suffer from the cold-start problem, meaning that they rely on the availability of interaction data or teacher-generated data.

In this work, we aim to bridge the gap between LLM-driven and data-driven hint-generation by simulating an initial set of trace data for data-driven algorithms. We utilize LLMs to generate diverse verified solutions across a variety of approaches and solution-complexity levels. We then employ our subtractive trace-simulation approach, meaning that we start from a (student-generated or LLM-synthesized) correct solution to a task and recursively remove elements until the empty program is reached. By analysing the state spaces that arise from such simulated programming traces, we aim to answer the main research question whether and under which circumstances using simulated interaction data is a feasible approach for programming hint-generation? We also address the question whether the simulated student traces resemble real student data and might be useful as synthetic training data or for testing purposes. We approach these questions with the following core contributions:

1. Novel methods for subtractively simulating programming traces from correct solutions including a fine-tuned LLM for reversing programming steps in Python.
2. An experimental comparison of simulated versus real state spaces.
3. An investigation on the feasibility of applying the simulated traces for downstream hint generation methods.

2. RELATED WORK

Data-driven hint generation methods often rely on state spaces to represent student behavior [18]. A state space

Jesper Dannath, and Benjamin Paaßen. Simulating Programming Traces and State Spaces for Hint Generation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 774–779. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039736

consists of a directed graph where the nodes are student states and the edges represent student steps between states. Each state is hereby closely related to the current student program (snapshot). Often, some form of canonicalization or clustering is applied to group slightly different program snapshots and thus achieve a more robust and concise state space [19, 18, 17]. Next step hints can be generated from state spaces using a two-step approach: First, the student's current snapshot is matched to an existing state in the state space [17]. Second, an algorithm selects one of the outgoing states as a prediction of the next state – which is then provided as a hint [11]. One example of such an algorithm would be the hint factory, which employs a Markov Decision Process to select a next step that leads to a correct solution on a short path [18].

For a long time, simulating student behavior has been used to model and understand student learning [24, 22]. The focus used to be on modeling student's cognitive properties and predict the outcome of student task attempts [27]. In programming, it is very difficult to obtain fine-grained simulation data of student interactions, since the solution spaces for programming tasks are vast (combinatorially exploding with the size of the program). Today, LLMs offer a new paradigm for generating synthetic data in domains with vast solution spaces [13]. Previous work has utilized in-context learning and fine-tuning LLMs to generate realistic student attempts based on a reference trace by the same student [14]. Even more recent work is using LLM-Agents that employ reasoning-LLMs and tool-calling in order to derive each student step [6, 27]. In particular, CoderAgent includes memory-retrieval and update steps as well as planning and reflection steps yielding higher Accuracy compared to previous methods [27]. The resulting traces can be used in downstream tasks such as learner modeling and training hint generation models taking student knowledge and learning into account. Our proposed subtractive approach sacrifices some versatility, but circumvents the complexity of predicting a next state by simulating backwards, such that a model merely has to decide which part of a program was likely added last, requiring substantially fewer resources.

3. METHOD

We are approaching programming hint generation through the use of simulated state spaces which are aggregated from simulated programming traces. We can view a programming trace as a sequence of program snapshots s_t for $t \in \{1, ..., T\}$. For simulating traces, we need to predict the mapping from s_t to s_{t+1}, which is a difficult problem, as the number of possible line steps is vast. Additionally, the mapping needs to be applied recursively to obtain a complete trace with no guarantee that a correct solution will be reached (and risk of error accumulation). On the other hand, if s_{t+1} is given, estimating s_t is significantly easier in many cases, as usually we just have to decide which line to delete. Therefore, our strategy is to construct an approximate mapping f s.t. $f(s_{t+1}) \approx s_t$. This mapping can be applied recursively to yield a reverse trace $s_T, s_{T-1}, \ldots, s_0$ with $s_0 = \varepsilon$, the empty program. We call this a *subtractive* approach.

A state space for a task is traditionally build from a range of example attempts, aggregated on a specific step-granularity (programming traces). As outlined above, scaffolding theory provides a pedagogically inspired ideal for step-granularity. A previous user-study indicates that the line-level might be a good approximation of a reasonable scaffolding step-granularity for introductory python tasks [3]. Therefore, we build state spaces from simulated programming traces such that transitioning between nodes is equivalent to applying one line-level change to the current user program (see Figure 3). Below we describe the simulation-method in further detail.

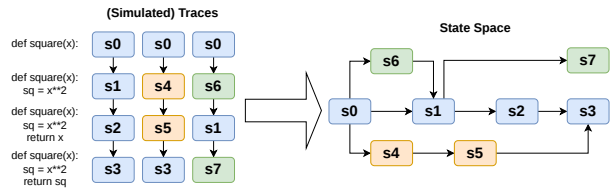


Figure 1: Method: simulated line-based programming traces are aggregated to state spaces

3.1 Generating solutions

For generating diverse correct solutions to a task we are applying a two-step prompting approach. We ask a reasoning-LLM (Qwen3.5-397B) to generate a list of 15 approaches to the task described in one or two sentences. Then, in a second step, a LLM (Qwen3-235B) is prompted without reasoning to implement different versions of each of the approaches. Specifically we first ask the LLM to provide a baseline implementation for a student with medium python skill. Then we repeatedly ask the LLM to make a simpler or a more complex version of it, until the quota (61 solutions total per task) is reached. If a generated solutions wasn't unique after up to four retries, we reduced the quota by one and proceeded to the next attempt. The complete prompt and all other prompts as well as the code for all analysis and supplementary analysis are accessible in our GitLab repository[1].

3.2 Deleting from a random block (RBLL)

We evaluate a programmatic approach for step subtraction which is to simply remove the last line from a random control block. To achieve this, we use the Python abstract syntax tree (AST)-module to list all control structures and obtain the lines which their respective block spans (note that blocks can be nested). Then, we sample one random control block (uniform probability) and delete its last line as well as other lines that might need to be deleted to retain a correct syntax. We call this method RBLL (Random Block Last Line; speak rebel). Note that correct syntax is needed to analyze the block structure of Python code. Hence, some Python-specific mechanisms are imposed to ensure that RBLL never deletes lines in ways that would break the syntax (accounting for multi-line statements, except blocks, inserting pass statements for empty blocks, resolving try blocks if their except is deleted). While this code-processing procedure is specific to Python, the reasoning of first standardizing the syntax, then reversing a step and then fixing a few edge cases of breaking syntax can be applied to other imperative

[1] `https://gitlab.ub.uni-bielefeld.de/publications-ag-kml/step-simulation`

programming languages as well. We do not claim that our approach addresses all possible edge-cases in Python. However, we could not observe any simulation error for RBLL throughout our experiments.

3.3 Prompting LLMs for reversed steps

We compare RBLL with a step-subtraction method based on pretrained LLMs. We utilize an iteratively developed prompt format which is structured into system prompt, context and query. The prompt context contains three elements, starting with a description of how the outputted reversed line steps should be structured. Then, we give the model the correct solution it is deducting steps from. Finally, we embed the first three snapshots and the three most recent snapshots in the model context. In the query we then continue to prompt the model to delete a single line from the current snapshot. Alternatively, we sometimes prompt the model to introduce an error or a misconception into one existing line of the current snapshot. We sample the query such that the model is asked to introduce errors or misconceptions 30% of the time. We did find that this value yields a significant increase in trace diversity while maintaining plausibility. Generally, this method relies on the LLM to branch off into different directions on repeated simulation runs, while RBLL explicitly employs random branching. The complete prompt is available in GitLab. Since LLMs will sometimes fail to produce a reverse step, we are validating the success of the step reversion. When the LLM was unsuccessful (i.e., leaving code unchanged, manipulating more than a single line, or breaking the syntax), we fall back to RBLL.

3.4 Fine-tuning LLMs for reversed steps

Finding that in many cases the LLM was incapable of reversing a line step correctly (see Section 4.2), we concluded fine-tuning might be a viable strategy to improve performance. We used the described prompting format to create a dataset of real reverse traces. In addition to our dataset (Section 4), we used a second keystroke-level dataset from Utah State University (2019) for additional training examples [5]. Then we employed supervised fine-tuning utilising unsloth [2] with the real reverse steps as a target. 50% of the tasks of each of both datasets (20 tasks in total) were selected randomly for training, resulting in 16465 training examples. We trained a 16-bit LoRA [8] with rank 128 for one epoch (lr 1e-4) on a single A40 GPU. The model was inferenced in 8-bit. We used Qwen3-8B [25] as a base model without reasoning traces or thinking mode. We reference the fine-tuned model as Qwen3-8B-SR (SR for Step Reversion).

4. EXPERIMENTS

We evaluate the state space simulations on a real-world dataset. The dataset was collected at a Mid-European university and consists of programming traces from 29 Python tasks recorded via an intelligent tutoring system [4]. In total, 209 students donated 1400 programming traces across Python Introduction sessions between 2023 to 2025. From these traces, which are recorded on a keystroke level, 7370 Line steps could be inferred. After randomly selecting 50% of tasks for fine-tuning, 14 tasks remained for computing the below experiments. For obtaining line steps, We also remove all complete detours from student traces. "Complete" refers to the kind of detours that lead to the exact same state later on. The tasks are about basic python concepts and NumPy as well as Data Mining methods. For evaluation, we used pretrained LLMs of different sizes from the Qwen3 model family, namely Qwen3-8B, Qwen3-coder-30B-A3B (short Qwen3-30b) and Qwen3-235B [25]. Additionally, we utilized RBLL as a baseline and Qwen3-8B-SR as our own fine-tuned model based on Qwen3-8B (Section 3.4). The 8B models were both used in 8-bit quantization on a single NVIDIA RTX 5060 Ti with 16GB of VRAM. Qwen3-30B and Qwen3-235B were used in 16-bit quantization using the scaleway-API. We use three traces per solution for the simulation of state spaces.

Table 1: Fallback rates for step reversion by LLM

Qwen3-8B	Qwen3-30B	Qwen3-235B	Qwen3-8B-sr
76.9%	74.1%	55.0%	35.8%

4.1 Metrics

We compare state spaces of the real student traces (from tasks in the test set) with simulated state spaces. All metrics are based on a matching of states between the real and the simulated state spaces. For matching, we canonicalize all programs using their AST, normalizing variable, function, and class names, and standardizing strings (programmatically formatted parts of strings are kept). To compare states, our metrics are *precision*, meaning the fraction of states in the simulated state space that also occur in the real state space, and *recall*, meaning the fraction of states in the real state space that also occur in the simulated state space. We also report *recall-5*, where we relax the state matching and allow up to five tokens edit-distance (e.g. single function call with one argument; NLTK wordPunctTokenizer). Recall-5 aims to yield a more realistic estimate of a classroom-setting, where one would usually allow relaxed matching to be able to deliver hints to more students. To compare steps in the state space (i.e. edges from one state to another), we search for matches of real states in the simulated state space and compute the number of the available outgoing edges of these matches (0 if the real state does not exist in the simulated state space). We call this metric *average outgoing score*. The outgoing score aims to estimate how many candidate hints we can expect for a randomly selected state (without accounting for state likelihood). All metrics are averaged across tasks.

4.2 Fallback-rates

We found that, in many cases, the pretrained LLMs were not able to follow the prompt of deleting a single line. Instead, the LLMs often returned the exact same snapshot as before, edited multiple lines, or added lines to the current snapshot. On occasions where the LLM fails to reverse a line-step, we simply fallback to RBLL. Throughout all traces simulated for the experiments in this paper, we observe that failure rates decrease with model size, with 76.9% for the Qwen3-8B and 55% for Qwen3-235B (Table 1). Still, fine-tuning is even more effective than model size with merely 35.8% failure rate for the fine-tuned Qwen3-8B-SR.

4.3 Simulation based on existing solutions

In order to evaluate the step-reversion performance in isolation, we examine the performance of the simulation meth-

Table 2: Mean and SD of state space metrics

Method	Precision	Recall	Recall-5	Outgoing
RBLL	**.43 (.19)**	.29 (.09)	.52 (.16)	.46 (.14)
Qwen3-8B	.38 (.15)	.32 (.08)	.55 (.15)	.56 (.16)
Qwen3-30B	.32 (.13)	.31 (.1)	.55 (.17)	.61 (.19)
Qwen3-235B	.24 (.11)	.32 (.11)	.55 (.16)	**.75 (.26)**
Qwen3-8B-SR	.31 (.14)	**.33 (.09)**	.57 (.16)	.63 (.15)

Table 3: Mean and SD of state space metrics for AI-generated solutions

Method	Precision	Recall	Recall-5	Outgoing
RBLL	**.06 (.05)**	.1 (.06)	.31 (.17)	.36 (.21)
Qwen3-8B	.05 (.03)	.11 (.07)	.3 (.14)	.47 (.31)
Qwen3-30B	.05 (.03)	.11 (.07)	.31 (.16)	.5 (.31)
Qwen3-235B	.03 (.02)	**.12 (.07)**	.32 (.17)	**.67 (.43)**
Qwen3-8B-SR	.05 (.05)	.11 (.07)	.29 (.13)	.53 (.36)

ods for recovering state spaces from corresponding real student solutions (Table 2). The results show that, while the control-block based simulation method has the highest precision (.43), it is slightly outperformed by the LLM-based simulation methods on recall and recall-5. In terms of outgoing score, the LLM-based methods outperform the programmatic method by a larger margin, with the Qwen3-235B performing best (.75). For the use-case of synthetic trace-dataset generation for testing or model training, one would have to decide between using RBLL for maximum precision or using the LLM-based approach for more variability and slightly improved recall. Also, while being sufficient on our introductory dataset, the performance of all methods might decrease for longer tasks (See Section 5 and Appendix A).

4.4 Simulation with generated solutions

In many scenarios such as introducing a completely new task, it is reasonable to assume that little to no correct solutions to the task are available. In this section, we investigate how well we can recover real state spaces using only LLM-generated solutions to tasks. LLM-generated solutions are verified using unit tests (therefore, two tasks without unit tests had to be excluded). Table 3 shows the metrics based on up to 61 generated solutions per task (A task might have fewer as described in Section 3.1). In fact, the number of unique solutions that could be generated per task varies between 10 and 61 with a mean of 50.0. The relative performance between the different methods remains similar compared to using the real student solutions: RBLL achieves highest precision, LLMs better recall and more variability. For all methods, overall performance is lower compared to having real student solutions available. Interestingly, the outgoing metric degrades only slightly, whereas precision and recall decrease by a factor of 3. Recall-5 remains more stable between the two scenarios decreasing only by approximately 1/3. We expect that an increased number of generated solutions will generally lead to a higher number of recovered states. In further experiments we can observe that the recall indeed increases for all methods up to 61 generated solutions but the increase diminishes significantly (Figure 2).

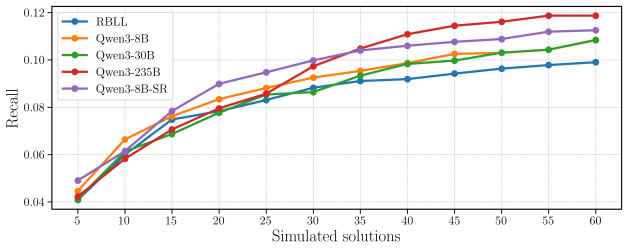


Figure 2: Avg. Outgoing states by number of generated solutions

Table 4: Comparing direct-LLM and state space based step prediction

Method	Tree-edit dist.	chrF	Acc	Acc-5
direct-LLM (8B)	**6.88**	0.76	**0.2**	**0.43**
direct-LLM (235B)	7.27	0.76	0.2	0.41
Sim RBLL	7.03	0.76	0.10	0.37
Sim 235B	7.37	0.76	0.06	0.32
Sim 8B-SR	7.55	**0.77**	0.07	0.33

4.5 Predicting steps

Deriving next step hints using simulated state spaces is not a trivial task. First, a student's new state needs to be matched against an existing state in the simulated state space. We do so by using the closest state according to tree edit distance (TED) on AST [15]. To select a next-step hint we consider the successor of the matched state in the simulated state space closest to the student's state (in terms of TED). If the successor differs in exactly one line from the student's state, this is defined as the next-step hint. Otherwise, we compute the line edit between the matched state and the successor and apply them to the student state, inspired by [10]. If this does not yield correct syntax, we attempt syntax repair by fixing indentation-levels; and if this does not succeed, we predict the successor directly (even if it differs by more than one line). As a baseline, we compare with an LLM prompted to predict the next line step. The prompt contains information on the expected format and step-granularity (line-steps) as well as the task description and current student step and is also available in the linked repository.

Since step prediction is a different task compared to the above evaluations, we employ different metrics. In particular, we report the TED to the actual next step of a student (lower is better), the character-level F-score (chrF; higher is better), which has been shown to align well with human assessment on code generation tasks [16], and the accuracy, meaning the rate of exact matches after canonicalization (higher is better). We also report Accuracy-5, where we mark a prediction as correct based on a relaxed matching up to an token edit-distance of 5 (similar to recall-5). Table 4 shows the results for predicting the next student step on the test-dataset. Among the simulation-based step predictors, RBLL performs best in terms of accuracy and TED. The LLM next step hints perform best in terms of accuracy but are slightly outperformed in terms of chrF. TED is comparable between RBLL simulation and LLMs.

5. CONCLUSIONS

In this work, we investigated the potential of simulating programming traces and state spaces using a subtractive approach based on LLMs and/or programmatic line removal. Such simulation has two major purposes: creating synthetic datasets and providing data for generating next step hints. Our experiments confirm that we can recover real intermediate student states if realistic solutions are available. The recall and precision vary by the employed simulation method: random block last line (RBLL) provides generally higher precision compared to LLMs; and larger LLMs generally achieve lower precision but higher average number of matched outgoing states. This is likely explained by larger LLMs generating more diverse programming traces. Throughout all experiments, the fine-tuned LLM Qwen3-8B-SR performs mostly on par with the much larger 235B model while also retaining a bit more precision. Additionally, the fine-tuned model has a much lower fallback rate, indicating that the task of removing lines from existing code is beyond the scope of the model's pre-training procedure. Using the simulated state spaces for step prediction shows that the LLM-based approaches do not have a benefit compared to RBLL. Given that RBLL is also much more efficient, it is currently the most advisable method for next step hint generation with simulated state spaces.

We also believe that the potential of the selection in state spaces is not fully realized in this work yet, as we didn't investigate more advanced methods of predicting steps within the simulated state spaces. Future work could investigate this and might also find ways to take advantage of the more diverse traces that LLM-based step simulation produces. Also, we did not investigate the effect of combining the simulated state spaces with student or teacher-provided trace data, as might be the case in real-world deployment. Some caution is advised for applying the presented methods for more complex tasks since our dataset only consists of easy introductory python tasks. Our supplementary experiments show decreased performance for state space reconstruction with real solutions on the Utah State keystroke-level programming dataset from 2019 [5], which consists of print statement heavy longer tasks like turtle tasks or CLI based games (see Appendix A). These tasks are considerably longer compared to the ones in our own dataset, but unfortunately lack unit tests and therefore could not be tested for state space reconstruction with LLM-generated solutions and step prediction. Generally, there is a lack of openly available line level or sub-line level python-trace datasets that could be used for evaluating the performance of our method for a larger variety of tasks [9].

Future work should also investigate whether the generated hints from simulated state spaces are effective in a pedagogical sense. This could also be supported by additional effort for personalizing the retrieved simulated states to the students current snapshot, like proposed in [20]. Finally, the large difference in performance between Table 2 and Table 3 suggest that future approaches might be able to improve the quality of simulated state spaces by focusing on the generation of more realistic student solutions. Still, RBLL provides a cheap and effective baseline for future work on simulated state spaces.

6. ACKNOWLEDGMENTS

This work is funded by the Ministry of Culture and Science of the State of North Rhine-Westphalia within the scope of the SAIL project. Funding ref: NW21-059A.

APPENDIX

A. UTAH STATE 2019 DATASET RESULTS

Table 5: Utah 2019 - Mean and SD of state space metrics

Method	Precision	Recall	outgong
RBLL	**.1 (.05)**	.1 (.03)	.15 (.04)
Qwen3-8B	.09 (.04)	.1 (.03)	.16 (.04)
Qwen3-30B	.09 (.04)	.11 (.03)	.16 (.04)
Qwen3-235B	.07 (.03)	.11 (.03)	**.21 (.05)**
Qwen3-8B-SR	.07 (.03)	.11 (.03)	.18 (.04)

2. REFERENCES

[1] A. Birillo, E. Artser, A. Potriasaeva, I. Vlasov, K. Dzialets, Y. Golubev, I. Gerasimov, H. Keuning, and T. Bryksin. One step at a time: Combining llms and static analysis to generate next-step hints for programming tasks. In *Proceedings of the 24th Koli Calling International Conference on Computing Education Research*, New York, NY, USA, 2024.

[2] M. H. Daniel Han and U. team. Unsloth, 2023.

[3] J. Dannath, A. Deriyeva, and B. Paassen. What is a Step? A User Study on How to Sub-divide the Solution Process of Introductory Python Tasks. In *Proceedings of the 18th International Conference on Educational Data Mining*, Palermo, Italy, July 2025.

[4] A. Deriyeva, J. Dannath, and B. Paaßen. SCRIPT: Implementing an intelligent tutoring system for programming in a german university context. In C. Stracke, S. Sankaranarayanan, P. Chen, and E. Blanchard, editors, *Proceedings of the 26th International Conference on Artificial Intelligence in Education (AIED 2025) Practitioner's Track*, Palermo, Italy, 2025.

[5] J. Edwards, K. Hart, and R. Shrestha. Review of CSEDM Data and Introduction of Two Public CS1 Keystroke Datasets. *Journal of Educational Data Mining*, 15(1):1–33, 2023.

[6] W. Gao, Q. Liu, L. Yue, F. Yao, R. Lv, Z. Zhang, H. Wang, and Z. Huang. Agent4Edu: Generating Learner Response Data by Generative Agents for Intelligent Education Systems. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, Philadelphia, PA, USA, Feb. 2025. AAAI Press.

[7] J. Hammond and P. Gibbons. Putting scaffolding to work: The contribution of scaffolding in articulating ESL education. *Prospect*, 20(1):6–30, 2005.

[8] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *Proceedings of the 10th International Conference on Learning Representations*, pages 1–15, Virtual, Apr. 2022.

[9] N. Kiesler, J. Impagliazzo, K. Biernacka, A. Kapoor, Z. Kazmi, S. G. Ramagoni, A. Sane, K. Tran, S. Taneja, and Z. Wu. Where's the Data? Finding and Reusing Datasets in Computing Education. In *Working Group Reports on 2023 ACM Conference on*

Global Computing Education, pages 31–60, Hyderabad India, Sept. 2024. ACM.

[10] T. Lazar and I. Bratko. Data-driven program synthesis for hint generation in programming tutors. In S. Trausan-Matu, K. E. Boyer, M. Crosby, and K. Panourgia, editors, *Intelligent Tutoring Systems*, pages 306–311, Cham, 2014. Springer International Publishing.

[11] J. McBroom, I. Koprinska, and K. Yacef. A Survey of Automated Programming Hint Generation – The HINTS Framework. *ACM Computing Surveys*, 54(8):1–27, Nov. 2022.

[12] R. P. Medeiros, G. L. Ramalho, and T. P. Falcão. A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education*, 62(2):77–90, 2019.

[13] M. Nadas, L. Diosan, and A. Tomescu. Synthetic Data Generation Using Large Language Models: Advances in Text and Code. *IEEE Access*, 13:134615–134633, 2025.

[14] M. H. Nguyen, S. Tschiatschek, and A. Singla. Large Language Models for In-Context Student Modeling: Synthesizing Student's Behavior in Visual Programming. In P. Benjamin and D. E. Carrie, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, Atlanta, GA, USA, July 2024. International Educational Data Mining Society.

[15] B. Paaßen. Revisiting the tree edit distance and its backtracing: A tutorial. *arXiv*, 1805.06869, 2022.

[16] D. G. Paul, H. Zhu, and I. Bayley. Benchmarks and Metrics for Evaluations of Code Generation: A Critical Review. In *Proceedings of the 1st IEEE International Workshop on Testing and Evaluation of Large Language Models*, San Jose, CA, USA, July 2024. IEEE.

[17] T. Price, R. Zhi, and T. Barnes. Evaluation of a Data-driven Feedback Algorithm for Open-ended Programming. In *Proceedings of the 10th International Conference on Educational Data Mining*, pages 318–323, Wuhan, China, 2017. International Educational Data Mining Society.

[18] T. W. Price, Y. Dong, and T. Barnes. Generating Data-driven Hints for Open-ended Programming. In *Proceedings of the 9th International Conference on Educational Data Mining*, Raleigh, North Carolina, 2016.

[19] K. Rivers and K. R. Koedinger. A Canonicalizing Model for Building Programming Tutors. In S. A. Cerri, W. J. Clancey, G. Papadourakis, and K. Panourgia, editors, *Intelligent Tutoring Systems*, volume 7315 of *Lecture Notes in Computer Science*, pages 591–593. Springer, Berlin, Heidelberg, 2012. Series Title: Lecture Notes in Computer Science.

[20] K. Rivers and K. R. Koedinger. Automating Hint Generation with Solution Space Path Construction. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, A. Kobsa, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, D. Terzopoulos, D. Tygar, G. Weikum, S. Trausan-Matu, K. E. Boyer, M. Crosby, and K. Panourgia, editors, *Intelligent Tutoring Systems*, volume 8474, pages 329–339. Springer International Publishing, Cham, 2014. Series Title: Lecture Notes in Computer Science.

[21] L. Roest, H. Keuning, and J. Jeuring. Next-step hint generation for introductory programming using large language models. In *Proceedings of the 26th Australasian Computing Education Conference*, Sydney, Australia, 2024.

[22] X. Tian, G. Pan, and H. Wang. Large Language Models in Student Simulation: A Survey, Nov. 2025. https://doi.org/10.21203/rs.3.rs-8148343/v1.

[23] K. VanLehn. The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems. *Educational Psychologist*, 46(4):197–221, Oct. 2011.

[24] K. VanLehn, S. Ohlsson, and R. Nason. Applications of Simulated Students: An Exploration. *Journal of Artificial Intelligence in Education*, 5(2):135–135, 1994.

[25] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, C. Zheng, D. Liu, F. Zhou, F. Huang, F. Hu, H. Ge, H. Wei, H. Lin, J. Tang, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Zhou, J. Lin, K. Dang, K. Bao, K. Yang, L. Yu, L. Deng, M. Li, M. Xue, M. Li, P. Zhang, P. Wang, Q. Zhu, R. Men, R. Gao, S. Liu, S. Luo, T. Li, T. Tang, W. Yin, X. Ren, X. Wang, X. Zhang, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Zhang, Y. Wan, Y. Liu, Z. Wang, Z. Cui, Z. Zhang, Z. Zhou, and Z. Qiu. Qwen3 Technical Report, May 2025.

[26] C. Zhai, S. Wibowo, and L. D. Li. The effects of over-reliance on AI dialogue systems on students' cognitive abilities: a systematic review. *Smart Learning Environments*, 11(1):28, June 2024.

[27] Y. Zhan, Q. Liu, W. Gao, Z. Zhang, T. Wang, S. Shen, J. Lu, and Z. Huang. CoderAgent: Simulating Student Behavior for Personalized Programming Learning with Large Language Models. In *Proceedings of the 34th International Joint Conference on Artificial Intelligence*, Montreal, Canada, Aug. 2025. IJCAI.

[28] R. Zviel-Girshin. The Good and Bad of AI Tools in Novice Programming Education. *Education Sciences*, 14(10):1089, Oct. 2024.

Temporal Clustering of Flight Simulator Data for Learner Profiling: A Methodological Comparison

Cheker Neffati
ETS Montréal
cheker.neffati@etsmtl.ca

Hakim Trachet
ETS Montréal
hakim.trachet@etsmtl.ca

Georges Ghazi
ETS Montréal
georges.ghazi@etsmtl.ca

Ange Adrienne Nyamen Tato
Université Laval
ange-adrienne.nyamen-tato@fse.ulaval.ca

ABSTRACT

Characterizing pilot behavior during takeoff is a prerequisite for enabling adaptive interventions in aviation Intelligent Tutoring Systems (ITS). Traditional High-Gain/Low-Gain dichotomies are insufficient to capture the behavioral diversity required for fine-grained learner modeling. This paper presents an unsupervised learning pipeline applied to multivariate time-series data collected from 12 pilots performing A320 takeoffs in X-Plane, resulting in a dataset of 112 rotation sequences. Four temporal clustering methods are evaluated during the rotation phase (S6): G-WVDTW, Temporal-DEC, BiLSTM Autoencoder, and Transformer. The two best-performing methods are subsequently validated on the ground roll (S5) and initial climb (S7) phases. On S6, the BiLSTM suggests five algorithmically-derived behavioral groupings extending beyond the binary paradigm, while the Transformer yields a duration-based separation. Cross-phase validation further indicates that rotation phase is the only segment that provides sufficiently rich and exploitable behavioral diversity for learner profiling, supporting a phase-specific rather than phase-agnostic approach to takeoff profiling.

Keywords

pilot behavioral profiling, temporal clustering, intelligent tutoring systems, BiLSTM autoencoder, transformer, flight simulator

1. INTRODUCTION

Intelligent Tutoring Systems (ITS) for aviation training require fine-grained learner models to provide personalized feedback. However, existing characterizations of pilot control behavior are still largely based on the traditional High-Gain/Low-Gain dichotomy [8]. This binary framework, originally derived from linear control theory, is insufficient to represent behavioral diversity observed in real flight data. Although the takeoff phase represents only about 2% of total flight time, it accounts for roughly 13% of fatal accidents worldwide [2], making it a priority phase for adaptive training interventions.

Prior work on data-driven pilot profiling has explored clustering of flight recorder data for anomaly detection [7] and non-parametric classification of pilot styles [5]. More directly relevant to ITS, Tato et al. [11, 12] demonstrated that unsupervised methods can extract behavioral profiles from simulator data to support adaptive coaching. However, these studies were limited to one or two clustering methods and did not systematically compare deep temporal architectures, nor evaluate the sensitivity of profile extraction to takeoff phase selection.

This paper addresses these gaps with three methodological contributions: (1) a systematic comparison of four temporal clustering paradigms on 112 A320 rotation sequences from 12 pilots with varying expertise; (2) a cross-phase validation on the ground roll (S5) and initial climb (S7) confirming that behavioral diversity is phase-dependent; (3) an analysis of the inductive biases of BiLSTM and Transformer autoencoders, with a discussion of their respective relevance for future learner modeling in ITS. By bridging temporal machine learning, human factors in aviation, and adaptive educational systems, this work contributes to the interdisciplinary foundations of data-driven learner modeling.

2. DATA AND PREPROCESSING

2.1 Experimental Context

Data were collected using the X-Plane desktop flight simulator configured for an Airbus A320 under standard conditions. Twelve pilots across three expertise levels (5 experts, 3 licensed, 4 novices) performed multiple takeoffs, yielding 115 sequences for the ground roll (S5), 112 for the rotation (S6), and 89 for the initial climb (S7) after quality filtering. Input variables were selected per phase to reflect the degrees of freedom available to the pilot, ranging from 4 variables for the highly constrained rotation phase (S6) to 12 for the ground roll (S5) and 13 for the initial climb (S7).

2.2 Preprocessing Pipeline

For the rotation phase (S6), sequence lengths range from 18 to 81 timesteps (mean = 35.4, std = 10.3). Min-Max normalization to $[0, 1]$ was applied to the training set. Variable-length sequences were zero-padded to the per-phase maxi-

Cheker Neffati, Ange Adrienne Nyamen Tato, Georges Ghazi, and Hakim Trachet. Temporal Clustering of Flight Simulator Data for Learner Profiling: A Methodological Comparison. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 780–785. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040146

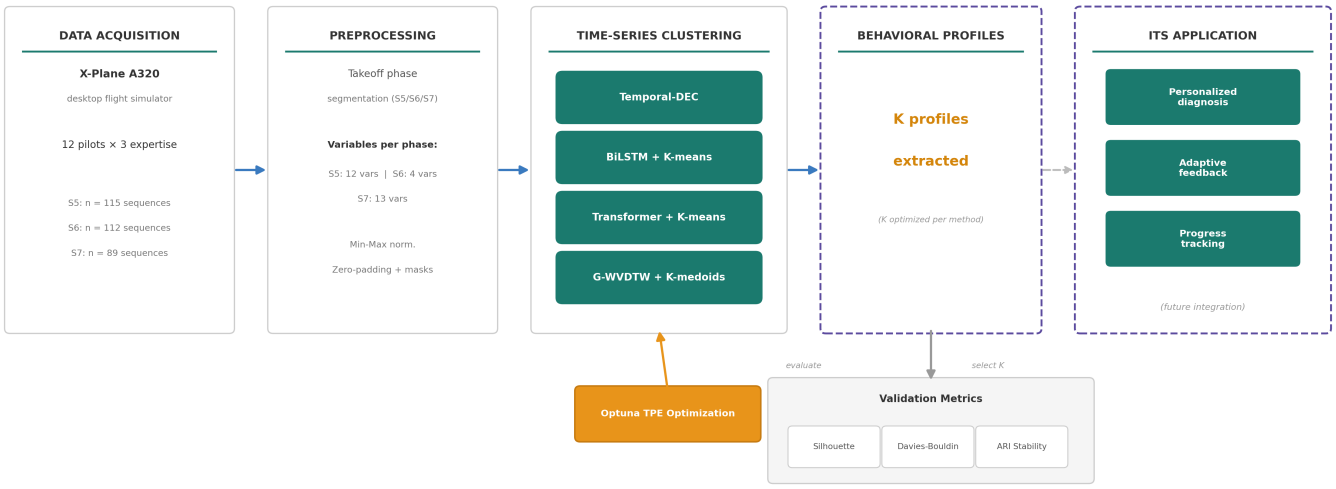


Figure 1: Methodology pipeline. Flight simulator data undergoes phase-specific preprocessing and temporal clustering via four methods. The three deep methods are optimized with Optuna [1]. The optimal K is selected using internal clustering metrics. Resulting profiles constitute a foundation for future ITS integration.

mum length ($T_{\max} = 81$ for S6) with binary masks to prevent gradient contributions from padded timesteps. The dataset was split 80/20 stratified by pilot identity (seed = 42) to avoid data leakage across sequences from the same pilot.

2.3 Evaluation Metrics

In the absence of ground-truth behavioral labels, clustering quality is assessed through three internal metrics and one stability measure. The **Silhouette score** [6] $S \in [-1, 1]$ measures the ratio of intra-cluster cohesion to inter-cluster separation; values above 0.5 indicate reasonable structure. The **Davies-Bouldin index** (DB) [4] penalizes clusters that are dispersed or poorly separated; lower is better, with DB < 1.0 considered satisfactory. The **Adjusted Rand Index** (ARI) measures assignment consistency across 50 independent runs with random initializations; ARI ≥ 0.8 indicates high reproducibility. The optimal number of clusters K is selected by maximizing the Silhouette score over $K \in \{2, \ldots, 7\}$.

These internal metrics assess the geometric quality of the partitions in latent space. They do not, by themselves, establish that the discovered groupings are educationally meaningful or instructionally actionable; such validation requires future expert review and integration into an ITS prototype.

3. METHODS

Four temporal clustering methods are evaluated, spanning three paradigms: distance-based, end-to-end deep, and modular deep. We use the term *cluster* for algorithmically-derived groupings and *profile* for their behavioral interpretation.

3.1 G-WVDTW + K-medoids

A distance-based approach using variance-weighted Dynamic Time Warping [3] to compute pairwise sequence dissimilarities across variables, with weights inversely proportional to per-variable variance. K-medoids clustering is then applied on the resulting $n \times n$ distance matrix. K-medoids was preferred over hierarchical or spectral alternatives for its native compatibility with arbitrary precomputed distance matrices, its robustness to outliers, and the interpretability of medoids as actual representative sequences — a property useful for downstream domain expert review.

3.2 Temporal-DEC

An end-to-end deep clustering framework adapting the Deep Embedded Clustering objective [16] to time series via a BiLSTM encoder. Representation learning and cluster assignment are jointly optimized by minimizing a combined reconstruction and KL-divergence loss.

3.3 BiLSTM Autoencoder + K-means

A modular pipeline separating representation learning from clustering [10]. A bidirectional LSTM autoencoder with temporal attention compresses each sequence into a fixed-size latent vector; K-means is then applied in the latent space. The bidirectional architecture captures both anticipatory and reactive control patterns.

3.4 Transformer + K-means

A modular pipeline replacing the recurrent encoder with a Transformer architecture [15] using sinusoidal positional encoding. The encoder projects input sequences through multi-head self-attention blocks into a latent vector; K-means is applied in latent space. Hyperparameters for the three deep methods were optimized with Optuna [1] (30 trials) using a composite objective balancing ARI stability, Silhouette, and Davies-Bouldin.

4. RESULTS

4.1 Rotation Phase (S6)

Table 1 reports clustering performance across the four methods on the 112 rotation sequences. Modular deep approaches outperform both G-WVDTW and Temporal-DEC on all metrics. G-WVDTW yields a weak Silhouette (0.383) and insufficient stability (ARI = 0.866). Temporal-DEC

achieves intermediate results but remains below modular approaches in cluster separation. Following the interpretation scale of [6], BiLSTM ($S = 0.608$) and Transformer ($S = 0.647$) both fall in the "reasonable structure" range (0.51–0.70).

Table 1: Clustering performance on the rotation phase (S6, $n = 112$).

Method	K	Sil.↑	DB↓	ARI
G-WVDTW + K-medoids	2	0.383	0.990	0.866
Temporal-DEC	3	0.530	0.562	1.000
BiLSTM + K-means	5	0.608	0.452	0.998
Transformer + K-means	2	0.647	0.469	1.000

The BiLSTM Silhouette is maximized at $K = 2$; however, $K = 5$ is retained as the optimal compromise: it maintains a reasonable Silhouette ($S = 0.608$) while achieving the lowest Davies-Bouldin index across all tested values of K (DB = 0.452), indicating compact and well-separated clusters. This $K = 5$ solution suggests five algorithmically-derived behavioral groupings distinguished by dynamic profile shape: a progressive profile (C0, $n = 26$), high-gain inputs (C1, $n = 18$), delayed pitch-up onset (C2, $n = 28$), oscillatory corrections (C3, $n = 23$), and low-gain inputs (C4, $n = 17$). C1 and C4 appear consistent with the High-Gain and Low-Gain categories described in the literature [8]; C0, C2, and C3 suggest intermediate profiles extending beyond the binary paradigm. These labels are assigned post-hoc by visual inspection. A preliminary review by a domain expert confirmed the operational coherence of the five groupings given the available data, though this does not constitute a systematic validation of their educational or instructional significance.

To better characterize the structure of the discovered profiles, we further examined their distribution across pilots and expertise levels. The resulting patterns indicate that the five clusters are shared across the cohort rather than being confined to isolated individuals. The pilot-by-cluster matrix nonetheless reveals mild pilot-specific tendencies in a few clusters, such as a relative concentration of P13 and P7 in C2 and of P11 in C4. These patterns should be interpreted as partial skews within a broader multi-pilot distribution rather than as exclusive cluster ownership, while the expertise distribution confirms that experts and novices contribute to each cluster to varying degrees.

Figure 2 illustrates the dynamic shape differences underlying these post-hoc labels. C1 exhibits the earliest and steepest pitch stick input, producing the fastest pitch rate and the shortest rotation, consistent with high-gain control. C4 displays the opposite pattern: low and delayed stick deflection, slow pitch rate, and the longest rotation, consistent with low-gain control. C2 shows a delayed onset of pitch-up despite eventually reaching comparable angles, while C3 combines an early stick input with intermediate dynamics. C0 occupies an intermediate position with a smooth, progressive trajectory. Across all clusters, vind_kias evolves nearly linearly, confirming that cluster separation is driven by control input dynamics rather than by airspeed differences.

Figures 3 and 4 provide a finer-grained view of how the discovered clusters are distributed across pilots and expertise levels.

The Transformer ($K = 2$, $S = 0.647$, ARI = 1.000) yields a different separation: the two clusters differ primarily in rotation duration rather than dynamic shape, consistent with the expected sensitivity of sinusoidal positional encoding to absolute sequence position. At $K = 5$, the Transformer produces a Silhouette of 0.603 but the five groups remain organized along a continuous duration gradient rather than exhibiting distinct behavioral modes.

Latent space analysis reveals that both architectures produce quasi-2D representations (PCA explains 99.7% and 99.8% of variance for BiLSTM and Transformer respectively), but with fundamentally different geometries: a V-shaped structure with discrete cluster positions for the BiLSTM (separation by dynamic shape), versus a continuous linear gradient for the Transformer (separation by duration). Expertise level does not drive cluster assignment in either model; experts, licensed pilots, and novices are distributed across clusters, though this observation should be interpreted cautiously given the small sample size ($n = 12$ pilots).

4.2 Cross-Phase Validation (S5, S7)

The two best-performing methods (BiLSTM and Transformer) were applied to S5 (115 sequences, 12 variables) and S7 (89 sequences, 13 variables) to assess whether behavioral diversity is phase-dependent. Results are reported in Table 2.

Table 2: Cross-phase validation results (BiLSTM and Transformer).

Segment	Method	K	Sil.↑	DB↓	ARI
S5 (Ground Roll)	BiLSTM	2	0.568	0.629	0.992
	Transformer	2	0.381	1.017	1.000
S7 (Initial Climb)	BiLSTM	2	0.587	0.473	0.989
	Transformer	2	0.760	0.394	1.000

Both methods converge to $K = 2$ on S5 and S7. On S5, the BiLSTM suggests a 76.5%/23.5% split between standard and non-standard pitch input patterns, consistent with the constrained Airbus nose-down procedure between 80 and 100 KIAS. The Transformer yields a weaker separation ($S = 0.381$, DB > 1.0), suggesting the duration-based bias is less discriminative on longer, higher-dimensional sequences. Neither method extracts exploitable stylistic diversity on S5.

On S7, the BiLSTM isolates a single atypical sequence ($n = 1/89$, 98.9% in the nominal cluster), indicating near-uniform behavior consistent with the highly standardized initial climb procedure (flight director, V2 + 10 KIAS, gear and flap retraction). The Transformer achieves the highest Silhouette across all segments ($S = 0.760$), though driven by duration rather than behavioral content: the divergence between the two clusters appears only at the end of longer sequences, with no difference in pilot inputs in the early seconds. This confirms that on standardized phases, the Transformer transitions from behavioral profiling toward sequence-length anomaly detection.

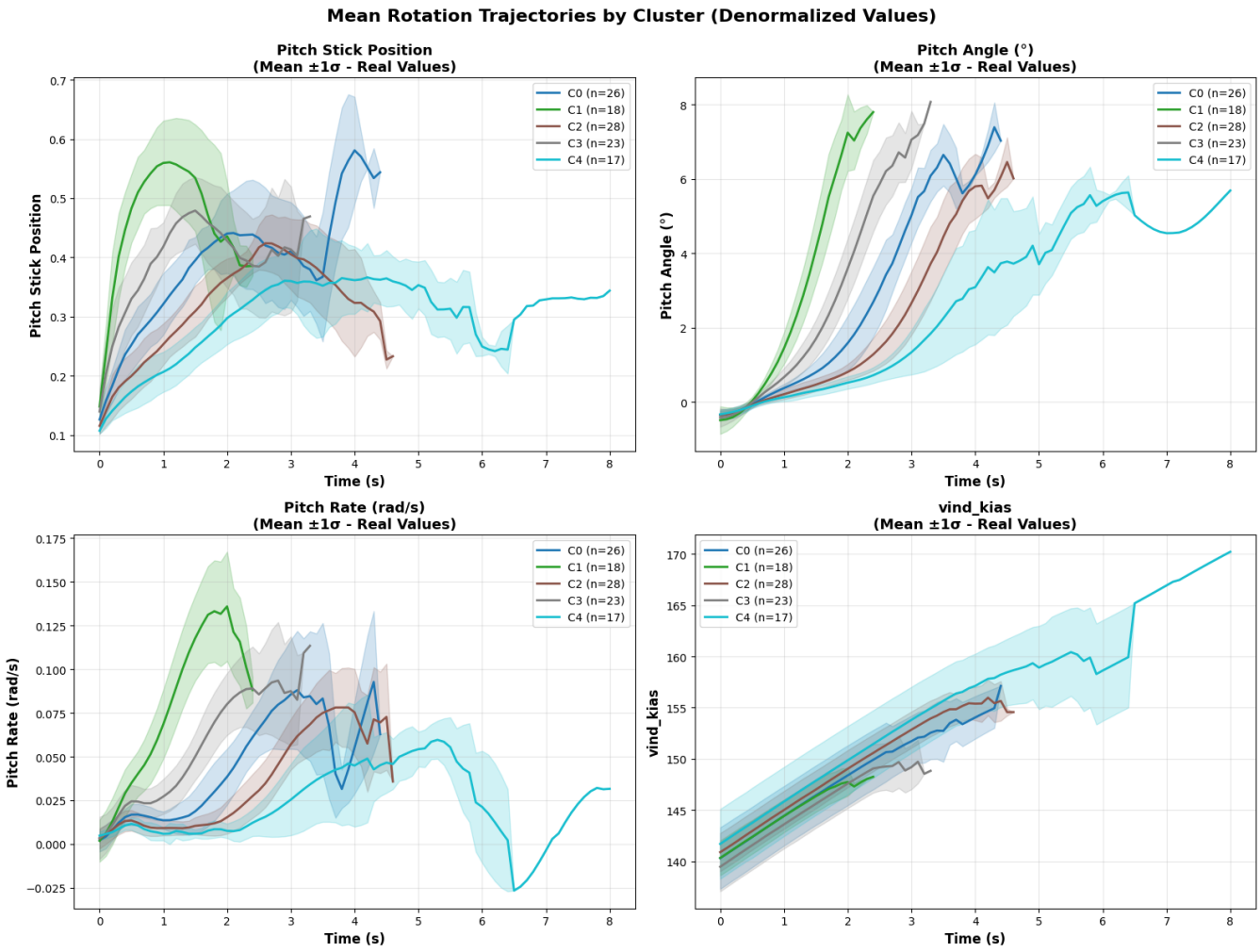


Figure 2: Mean denormalized trajectories ($\pm 1\sigma$) of the four S6 variables across the five BiLSTM clusters.

These results confirm that behavioral diversity is phase-dependent: S5 and S7, being procedurally constrained, yield binary structures without stylistic diversity. Only S6 produces a rich multi-cluster structure ($K = 5$) suitable for further investigation as candidate learner profiles.

5. DISCUSSION

5.1 Inductive Biases and Method Selection

The performance gap between modular deep methods and distance-based or end-to-end approaches suggests that separating representation learning from clustering appears advantageous on this small, high-dimensional flight dataset. G-WVDTW, despite its interpretable distance metric, suffers from the quadratic complexity of pairwise DTW computation and produces coarse separations. Temporal-DEC's joint optimization may cause the clustering objective to interfere with representation quality on limited data.

The contrasting behaviors of BiLSTM and Transformer reveal complementary inductive biases. The BiLSTM accumulates sequential state across timesteps, naturally capturing local dynamic transitions such as pitch rate oscillations and anticipatory stick movements. The Transformer's sinusoidal positional encoding embeds absolute temporal position into every representation, making sequence duration a primary discriminating factor. These biases are not defects but reflect different aspects of pilot behavior: the BiLSTM captures *how* the rotation is performed, while the Transformer captures *how long* it takes.

5.2 Relevance for Learner Modeling in ITS

The five BiLSTM groupings constitute candidate profiles whose operational coherence has been confirmed by a preliminary domain expert review [11, 12], suggesting they capture recognizable piloting patterns rather than statistical artifacts. Operational coherence does not, however, establish educational validity or instructional actionability. If validated against learning outcomes, such profiles could inform differentiated feedback strategies — for instance, exercises targeting smooth control for high-gain pilots or timing drills for delayed-onset patterns. The high clustering stability (ARI = 0.998) is a necessary but not sufficient prerequisite for consistent profile assignment.

The Transformer's duration-based separation, while less

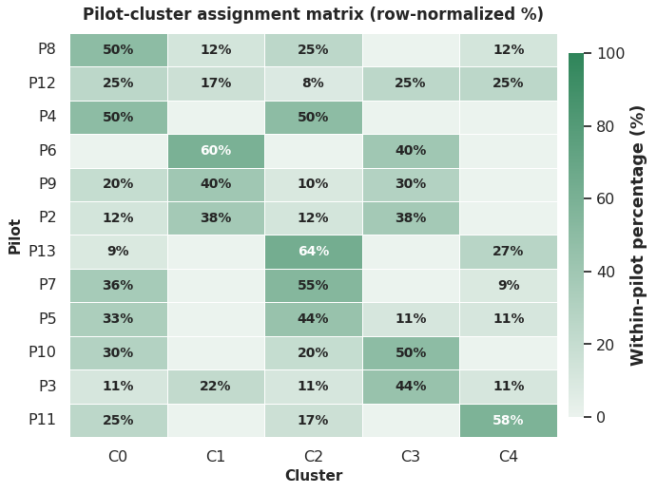


Figure 3: Pilot-by-cluster assignment matrix (row-normalized %).

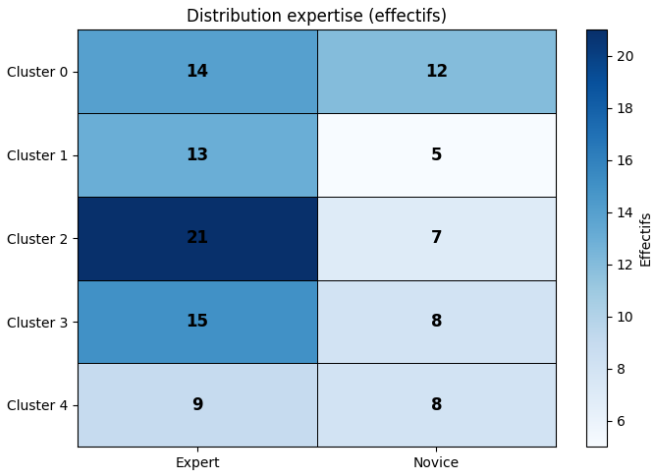


Figure 4: Distribution of cluster assignments by expertise level.

suited to style profiling, could complement the BiLSTM perspective by monitoring rotation efficiency — a metric relevant to procedural compliance assessment.

Overall, this work provides a methodological foundation rather than a validated ITS component. The transition from coherent clusters to educationally actionable profiles requires systematic multi-assessor validation and integration into a tutoring loop with measurable learning outcomes.

5.3 Limitations

Several limitations warrant acknowledgment. First, the dataset comprises 12 pilots and at most 115 sequences, which is sufficient for a proof-of-concept but not for establishing a generalizable taxonomy [9]. Small-sample settings can amplify apparent structure and make estimates optimistic, so the observed patterns should be interpreted as methodological candidates rather than definitive learner categories [13, 14].

Overfitting risk on 112 rotation sequences is mitigated by three design choices: (i) purely reconstructive training without label supervision, (ii) constrained latent dimensionality (8–16 units) tuned via Optuna with dropout and weight decay, and (iii) clustering stability verified across 50 random initializations (ARI $\geq$ 0.99). Latent representations may nonetheless encode pilot-specific idiosyncrasies, making the per-pilot distribution analysis (Figure 3) essential to interpret groupings as shared patterns rather than pilot signatures.

Other limitations include the use of a desktop simulator (simulator fidelity), post-hoc visual labeling validated by a single expert [11, 12], the inability of internal metrics to establish educational meaningfulness, and the restriction to the takeoff phase.

6. CONCLUSION

We presented an unsupervised pipeline for extracting candidate pilot behavioral profiles from multivariate flight simulator data, comparing four temporal clustering methods across three takeoff phases. On the rotation phase (S6), modular deep approaches outperform distance-based and end-to-end methods: the BiLSTM autoencoder suggests five behaviorally distinct groupings extending beyond the binary High-Gain/Low-Gain paradigm [8], while the Transformer yields a complementary duration-based separation. Cross-phase validation confirms that behavioral diversity is phase-dependent: only S6 produces a multi-cluster structure, while S5 and S7 converge to binary splits reflecting procedural constraints.

These results provide a methodological foundation for data-driven learner diagnosis in aviation ITS. The complementarity between BiLSTM (control style) and Transformer (temporal efficiency) perspectives suggests richer learner models combining both. Future work will focus on broader expert validation, larger pilot populations, and integration into an ITS prototype to assess profile-specific feedback effects on learning outcomes.

7. ACKNOWLEDGMENTS

This work is funded by CRIAQ (Consortium de Recherche et d'Innovation en Aérospatiale au Québec) and NSERC (Natural Sciences and Engineering Research Council of Canada). We gratefully acknowledge the financial and logistical support of our industry partners, Bombardier Inc, Beam me up Augmented Intelligence, and Cognitive Group. We also thank the members of the C-Pilot project for their valuable insights and contributions to this work.

8. REFERENCES

[1] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD*, pages 2623–2631, 2019.

[2] Boeing. Statistical summary of commercial jet airplane accidents, worldwide operations 1959–2022. Technical report, Boeing Commercial Airplanes, 2023.

[3] D. Cao, Z. Lin, D. Liu, and X. Chai. G-WVDTW: A generalised weighted variance dynamic time warping algorithm for subsequence matching in multivariate time series. *Expert Systems*, 42(5):e70036, 2025.

[4] D. Davies and D. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[5] F. Garcia Lorca, S. Gururajan, and S. Belt. Characterization of pilot profiles through non-parametric classification of flight data. In *AIAA Information Systems-AIAA Infotech @ Aerospace*, page 0914, Grapevine, TX, 2017. AIAA.

[6] L. Kaufman and P. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.

[7] L. Li, S. Das, R. John Hansman, R. Palacios, and A. Srivastava. Analysis of flight data using clustering techniques for detecting abnormal operations. *Journal of Aerospace Information Systems*, 12(9):587–598, 2015.

[8] D. McRuer and E. Krendel. Mathematical models of human pilot behavior. Technical Report AGARD-AG-188, AGARD, 1974.

[9] S. Raudys and A. Jain. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):252–264, 1991.

[10] A. Sagheer and M. Kotb. Unsupervised pre-training of a deep LSTM-based stacked autoencoder for multivariate time series forecasting problems. *Scientific Reports*, 9(1):1–16, 2019.

[11] A. Tato, R. Nkambou, and G. Nana Tato. Towards adaptive coaching in piloting tasks: Learning pilots behavioral profiles from flight data. In *International Conference on Intelligent Tutoring Systems*, pages 105–114, Cham, 2022. Springer.

[12] A. Tato, R. Nkambou, and G. Nana Tato. Automatic learning of piloting behavior from flight data. In *International Conference on Intelligent Tutoring Systems*, volume 13891 of *LNCS*, pages 541–552, Cham, 2023. Springer.

[13] A. Vabalas, E. Gowen, E. Poliakoff, and A. Casson. Machine learning algorithm validation with a limited sample size. *PLoS ONE*, 14(11):e0224365, 2019.

[14] S. Varma and R. Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7(1):91, 2006.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5998–6008, Long Beach, CA, 2017. Curran Associates.

[16] J. Xie, R. Girshick, and A. Farhadi. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning (ICML)*, volume 48, pages 478–487. PMLR, 2016.

The "Easy Trap": Why LLMs Underestimate Misconception-Driven Difficulty

Amanda La Hadi
Monash University, Indonesia
amanda.lahadi@monash.edu

Muhammad Johan Alibasa
Monash University, Indonesia
johan.alibasa@monash.edu

Guanliang Chen
Monash University, Australia
guanliang.chen@monash.edu

A. Taufiq Asyhari
Monash University, Indonesia
taufiq.asyhari@monash.edu

ABSTRACT

Large language models (LLMs) are increasingly used for estimating item difficulty in educational assessment. However, it remains unclear whether such estimates reflect how learners actually experience difficulty. This study investigates the alignment between LLM-generated difficulty ratings and empirical student performance on basic mathematics tasks. Four widely used LLM-based systems generated difficulty ratings (1-100 scale) for 32 arithmetic items across multiple runs (N = 640 ratings). These were compared with empirical difficulty derived from responses of 770 Indonesian undergraduates using Classical Test Theory (CTT) and Item Response Theory (2PL). Results show moderate rank correlations (Spearman's $\rho = 0.52 - 0.7$), indicating that LLMs capture coarse ordering of item difficulty. However, substantial and systematic misalignment emerges in fraction items. Several items consistently rated as "easy" by LLMs were among the most difficult for students (e.g., only 34.16% correct for $100 \div 12$). We argue that LLMs approximate curricular difficulty—what should be easy based on instructional sequencing—rather than cognitive difficulty driven by learner misconceptions. This leads to systematic underestimation of misconception-driven items, a phenomenon we term the "Easy Trap." These findings highlight a critical limitation of LLM-based difficulty estimation and suggest that relying on such estimates without empirical grounding may introduce bias in assessment design and adaptive systems.

Keywords

Educational Data Mining, Item Response Theory, Adaptive Assessment, Cognitive Modeling, Psychometric Calibration

1. INTRODUCTION

Large language models (LLMs) are rapidly being integrated into educational practice, extending far beyond basic search or item drafting to functions that directly affect assessment—providing feedback, assisting with grading for multiple-choice and open-ended responses, recommending remedial content, generating curriculum-aligned items, and even estimating item difficulty from text alone [5, 8, 19, 27, 29, 35, 38]. While these capabilities promise substantial gains in efficiency, they also introduce crucial measurement risks: LLM-generated judgments often reflect curricular expectations rather than empirical patterns of student performance, potentially leading to mis-calibrated difficulty labels when real learners diverge from what the curriculum prescribes [1, 23].

This risk is becoming salient for so-called "basic" mathematics. Foundational number concepts—arithmetic with whole numbers, integers, fractions, decimals, and exponents — appear early in schooling yet continue to challenge many undergraduates, with well-documented downstream effects on later success in STEM disciplines [4, 16, 21, 33, 36]. Among these domains, fractions are repeatedly identified as a bottleneck skill and a strong predictor of attainment in algebra, statistics, and calculus [14, 15, 21, 30]. Formal instruction in fractions begins in Grade 3, but persistent misconceptions—e.g., whole-number bias, treating numerators and denominators independently, or assuming "division makes smaller"—are widely reported across contexts and often persist into higher education [20, 22]. When LLMs classify such fraction items as "easy" on the grounds that they are introduced early in curricula, their predictions may systematically underestimate actual difficulty for undergraduate learners.

Empirical difficulty can be estimated from student performance using Classical Test Theory (CTT), which defines difficulty as the proportion of students answering correctly (p-value), and Item Response Theory (IRT), which models the interaction between latent ability and item characteristics [9, 12, 15]. Recent studies have explored the use of LLMs to approximate expert judgments of difficulty or to infer IRT parameters from item text [1, 28]; however, findings suggest that surface textual features are weak predictors of latent difficulty compared to empirical response data.

This study evaluates whether LLM-generated difficulty estimates align with empirical undergraduate performance on basic arithmetic items. Beyond measuring alignment, we investigate a systematic source of misprediction: the distinction between curricular difficulty (what is expected to be easy based on instructional progression) and cognitive difficulty (what is actually difficult due to persistent mis-

Amanda La Hadi, Johan Alibasa, Guanliang Chen, and A. Taufiq Asyhari. The "Easy Trap": Why LLMs Underestimate Misconception-Driven Difficulty. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 786–792. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039732

conceptions). We hypothesize that LLMs primarily encode curricular expectations, leading to systematic underestimation of items that are procedurally simple but conceptually challenging. Using responses from 770 Indonesian undergraduates and difficulty estimates from four LLM systems, we examine where and why these mismatches occur, and what they imply for the use of LLMs in educational assessment.

Research question (**RQ**): *"To what extent do LLM-based difficulty estimates align with empirical student performance on undergraduate basic arithmetic, and how does the disconnect between these measures reveal specific misconception-driven demands that models fail to capture?"*

2. RELATED WORKS

Item difficulty is traditionally measured using Classical Test Theory (CTT) and Item Response Theory (IRT). In the 2-parameter logistic (2PL) IRT model, item difficulty and discrimination parameters enable sample-invariant estimation under standard conditions and support fine-grained diagnostics [12, 15]. Emerging work explores using large language models (LLMs) to approximate expert judgments of difficulty or infer psychometric properties directly from item text [23, 28]. However, evidence suggests that surface textual features are weak predictors of latent difficulty compared to empirical response data, and many approaches rely on simulated responses or proxy indicators rather than real student performance [1].

LLMs are increasingly used to support assessment workflows, including item generation, grading, and feedback [5, 8, 37]. Recent studies have investigated whether LLMs can estimate item difficulty, with promising correlations reported in controlled settings [1, 28]. However, these approaches largely depend on simulated learners or benchmark datasets, leaving open questions about how well LLM-generated difficulty reflects actual student performance, particularly in conceptually demanding domains.

Fractions are widely recognized as a challenging domain in mathematics education, with difficulties persisting into higher education [4, 21]. Prior work shows that fraction understanding predicts later success in algebra and overall mathematical achievement [17, 31]. These difficulties are often driven by persistent misconceptions, such as whole-number bias and incorrect reasoning about operations, which continue to affect even university students [10, 11].

Despite this extensive literature, there is limited empirical data on fraction performance among undergraduates. This gap makes it difficult to assess whether LLM-generated difficulty estimates reflect young adult learner performance or primarily encode curriculum-based expectations. In this study, we combine CTT and 2PL IRT with real response data from 770 Indonesian undergraduates to evaluate the alignment between LLM-generated difficulty estimates and empirical difficulty. By grounding the analysis in observed learner performance, we examine how discrepancies between model predictions and student outcomes reveal the limitations of LLM-based difficulty estimation.

3. METHOD

3.1 Participant

Participants were 770 second-year undergraduate students enrolled during 2020 to 2025 in mandatory mathematics courses at a large public university in Indonesia (79.74% female, 20.26% male; mean age = 19.2 years). All students had completed Indonesia-K13 mathematics curricula through Grade 9 – 12 (before the national shift to the Merdeka Curriculum in 2021). Participation was voluntary, and informed consent was obtained. The study used previously collected data approved for secondary analysis by the university ethics board (Authorization No. 511/In.23/L.1/TL.01/12/2025) and received additional approval from the Human Research Ethics Committee (Project ID: 50488).

3.2 Instrument

3.2.1 BMA Test

The Basic Mathematics Ability (BMA) assessment consisted of 32 multiple-choice items covering arithmetic operations aligned with K13 Grades 3-7 standards. The distribution was as follows: whole number operations (n = 5), integer operations (n = 10), fraction operations (n = 11), decimal operations (n = 4), and exponents/roots (n = 2). Items were drawn from validated Indonesian textbooks (published by the Ministry of Education) and prior validated research [20]. Each item had four response options.

The present analysis focuses on eleven fraction items (Items 3, 11, 20, 23, 25, 27, 28, 29, 30, 31, 32) spanning addition, subtraction, multiplication, division, and magnitude comparison. These eleven items represent all fraction-related items from the 32-item assessment. The test was administered via Google Forms and paper-and-pencil formats in proctored classroom settings (90-minute time limit). Calculators were not permitted to ensure assessment of procedural fluency.

3.2.2 LLM Difficulty Estimation Prompt

Difficulty predictions were obtained from four frontier LLM based AI systems:

1. Claude Sonnet 4.5 (Anthropic, September 2025)
2. Gemini 3 (Google DeepMind, November 2025)
3. ChatGPT 5.2 (OpenAI, GPT-5 series, December 2025)
4. Microsoft Copilot (Powered by GPT-5, December 2025)

For each item, each model was queried independently across four repetitions (N = 4 models × 5 repetitions × 32 items = 640 total difficulty scores). A standardized prompting template was used as illustrated in Figure 1 (detailed outputs are available in the GitHub repository). We evaluated consistency across multiple runs using flip-rate, which measures the proportion of questions where the model produced different answers between runs [13].

3.3 Empirical Difficulty Measures

We calculated the empirical item difficulties using Classical Test Theory (CTT) and 2-parameter logistic (2PL) IRT. These approaches provide complementary perspectives on item difficulty. CTT offers intuitive, sample-dependent difficulty indices based on the proportion of correct responses, whereas IRT estimates latent item difficulty parameters (β) while accounting for differences in student ability levels. The

Prompt for Generate Item Difficulty

You are an expert in mathematics assessment design.
Evaluate the following item:
ITEM: [Full item picture with options]
CONTEXT:

- Target population: Indonesian undergraduate students
- All content taught in elementary school under Indonesian Kurikulum 2013
- Purpose: Evaluate item characteristics and estimate functional difficulty for undergraduate students on a Basic Mathematics Ability (BMA) instrument.
- Item Review: Identification of mathematical topic, domain, targeted skill, and curriculum grade alignment.
- Cognitive Demand: Assessment of skill difficulty, cognitive load, multi-step reasoning, and prerequisite concepts.
- Difficulty Scale: Constrained numerical scale reflecting automatic fluency, basic procedural skill, or basic multi-step reasoning.
- Provide an exact difficulty score (1 - 100 scale):
  1 - 10 = Automatic basic fluency
  11 - 20 = Basic procedural skill
  21 - 30 = Basic multi-step
  31+ = Higher cognitive load
- Justify your rating briefly.

OUTPUT: Structured item-level difficulty table with mathematical topic, domain, prerequisite concepts, multi-step reasoning, and curriculum alignment.

Figure 1: Standardized prompting template that used in all LLMs.

2PL model was selected instead of the 1PL (Rasch) model because preliminary inspection indicated substantial variation in item discrimination across the 32 problems (a range from 0.28 - 1.83, M = 1.20, SD = 0.37). Item spanned all five discrimination categories defined by Baker [2], from very low (n = 2) to very high (n = 2), with the majority falling in the moderate (n = 16) ranges. Although a detailed analysis of discrimination is beyond the scope of this study, this heterogeneity is inconsistent with the equal-discrimination assumption of the 1PL model and motives the use of 2PL. In the 2PL model, difficulty parameter can then be compared directly with LLMs-estimated difficulty [12].

We estimated 2PL IRT model parameters using marginal maximum likelihood (MML) via the mirt package in R [7]. IRT's item difficulty theoretically range from $-\infty$ to $+\infty$, but typically falls between -3 and +3 in practical testing. Values near 0 indicate average difficulty, negative values reflect easier items, and positive values correspond to harder items [2].

To evaluate alignment between LLM-generated difficulty estimates and empirical difficulty measures, we computed Spearman's rank correlation (), Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). We rated consistency across multiple runs using two different metrics: (a) flip-rate, which measures the proportion of questions where the model produced different answers between runs [13], and (b) intra-class correlation (ICC) calculated using python pingouin package.

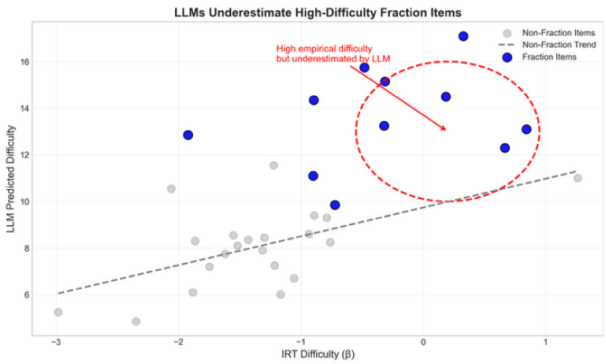


Figure 2: As empirical difficulty increases, LLMs systematically assign lower-than-expected difficulty scores to fraction items.

4. RESULTS

We distinguish between two forms of difficulty throughout the analysis: (1) curricular difficulty, reflected in LLM generated estimates based on procedural and textual features, and (2) cognitive difficulty, reflected in empirical performance shaped by learner misconceptions. We evaluated the alignment between LLM-generated difficulty estimates and empirical difficulty using multiple metrics. Across models (32 arithmetic items), LLM predictions show moderate rank alignment with empirical difficulty (Spearman's $\rho = 0.52 - 0.77$; Table 1), indicating that models capture coarse ordering of item difficulty. Similar patterns were observed for both IRT and CTT measures. However, rank alignment masks substantial discrepancies in absolute difficulty. RMSE and MAE values vary notably across models (Table 1), showing that models with similar correlations differ in their ability to estimate difficulty magnitude. This indicates that correlation alone overstates practical predictive accuracy.

Figure 2 compares LLM-predicted difficulty with empirical IRT difficulty. For CTT, we are using (1 – p) to align the magnitude with other measurements. Non-fraction items follow the expected positive trend, indicating general alignment between model predictions and empirical measures. However, fraction items systematically deviate from this relationship. As empirical difficulty increases ($\beta > 0$), fraction items become much harder for students, but the increase in LLM-predicted difficulty is smaller than expected.

This judgement suggests that LLMs fail to capture sources of cognitive difficulty specific to fraction concepts. Although isolated cases of underestimation appear among non-fraction items, these do not form a consistent pattern. In contrast, the repeated and directional deviation observed for fraction items indicates a domain-specific bias rather than random error (also showed in flip rate at Figure 3). While within-model variance was observed, it did not consistently correspond to prediction error, suggesting that variability alone is not a reliable indicator of misalignment.

Fraction items account for the most severe misalignment between LLM-predicted and empirical difficulty. Table 2 lists the five most underestimated items, all of which were consistently rated as "easy" by LLMs despite empirical failure rates exceeding 66.8%. In the most extreme case, only 34.2%

Table 1: Correlation between individual LLM predictions and empirical difficulty measures

LLM Model	IRT**	CTT**	RMSE (IRT)	MAE (IRT)	Rank
Claude Sonnet 4.5	0.700*	0.776	0.809	0.638	Best rank alignment
Gemini 3 Pro	0.624*	0.737	0.857	0.665	Balanced
ChatGPT 5.2	0.567*	0.716	0.974	0.721	Worst magnitude
Microsoft Copilot	0.515*	0.595*	0.943	0.727	Best magnitude
CTT	0.912*		0.366	0.257	Empirical baseline

*Significant at $\alpha = 0.05$.
**Correlation using Spearman ρ.
For CTT, we use $(1 - p)$ to align the magnitude with other measurements.

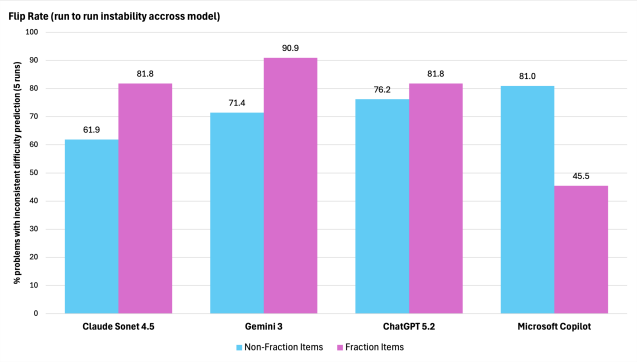


Figure 3: Flip Rate (Run-to-run Instability Across Models).

of students correctly solved division between whole number and fraction (100 12), although it was consistently categorized as a basic procedural task. We refer to this systematic misclassification as the "Easy Trap": items that are procedurally simple but conceptually conflicting are consistently underestimated by LLMs.

Analysis of student response distributions (Table 3) reveals that errors are not random but reflect systematic misconception patterns. Common incorrect responses include "division makes smaller" beliefs, whole-number bias, inappropriate operation selection, and incorrect reasoning about magnitude. These patterns indicate that empirical difficulty is driven primarily by conceptual misunderstandings rather than procedural complexity.

This pattern is further supported by the flip rate analysis shown in Figure 3). Across most models, fraction items produced higher rates of inconsistent difficulty predictions than the overall item set (for 5 runs). Especially for Gemini, which shows almost 91% inconsistency during repetition, which means 10 out of 11 fractions get different difficulty estimation more than 3 times. This suggests that LLM judgments on misconception-driven fraction problems are not only systematically misaligned with empirical student difficulty, but also less stable across repeated evaluations. The consistently higher flip rates for fraction items indicate that conceptually conflicting problems introduce greater uncertainty in LLM difficulty estimation compared to arithmetic items overall.

Taken together, these results suggest that LLM-generated difficulty estimates align more closely with procedural complexity and curricular expectations, but fail to fully capture cognitive difficulty driven by persistent misconceptions.

5. DISCUSSION

Our findings reveal a systematic distinction between curricular and cognitive difficulty in LLM-based estimation. While LLMs achieve moderate rank alignment with empirical measures, they consistently underestimate items whose difficulty is driven by persistent misconceptions. This suggests that LLMs rely primarily on surface-level procedural cues and curricular sequencing, rather than modeling the cognitive processes that shape student performance. As a result, LLM-generated difficulty estimates reflect what should be easy according to instruction, rather than what is actually difficult for learners.

Error analysis (Table 3) provides insight into the cognitive sources of this misalignment. Undergraduates continue to retain foundational misunderstandings typically associated with earlier grades. Common errors—"division makes smaller" reasoning (Item P32), inappropriate operation selection (Item P20), and whole-number overgeneralization (Items P11, P28)—indicate many students lack robust mental models of fractions. These patterns align with established research on fraction difficulty, particularly whole number bias and magnitude understanding [4, 17, 18, 24, 32]. Our study extends this line of work with a higher-education perspective: despite completing 12 years of schooling, many students still lack stable conceptual understanding and frameworks for elementary fraction operations.

Stability analyses reveal that fraction items are vulnerable to run-to-run variability. Figure 3) show high flip-rates meaning that users may receive different difficulty labels for identical items across attempts. This align with argument by Gonzales [13] that GPT-4 family models change their answer on ∼35% of problems in 3 repetitions. This volatility reflects both stochastic generation and inconsistent sensitivity to cognitive features, underscoring the need for response aggregation and uncertainty reporting when difficulty scores inform instructional decisions or assessment design.

These results have direct implications for educators and institutions increasingly relying on LLM-generated items as part of classroom assessment. In Indonesian higher education, classroom practice already uses LLM-generated items to evaluate student understanding [25, 26, 34]. Our findings caution against depending on single difficulty estimates for high-stakes decisions, as our repeated query analyses show that LLMs often produce varying interpretations of identi-

Table 2: Fraction Items with the Largest LLM Underestimation of Empirical Difficulty

Item	LLM's Difficulty Estimation Range				LLM Mean	IRT β	CTT
	Claude Sonnet	Gemini Pro	Chat GPT	Copilot			
P32	13–18	10–15	8–18	6–12	12.30	0.667	0.342
P3	10–18	12–16	6–9	10–16	13.10	0.843	0.384
P11	17–20	18–25	10–18	12–18	17.10	0.328	0.426
P20	10–16	15–18	9–16	9–18	14.50	0.185	0.449
P28	14–16	12–14	9–22	8–13	13.25	-0.319	0.553

Note: LLM difficulty estimates represent the range across five runs for each model. LLM Mean is the average across all models and runs. IRT β is the 2PL difficulty parameter. CTT is the proportion of students answering correctly (p-value). Lower CTT values indicate higher empirical difficulty.

Table 3: Distribution of student responses and dominant misconception patterns for the five most underestimated fraction items

Item	Correct (%)	Common Incorrect Answer Pattern(s)	Misconception Type
P32: $100 \div \frac{1}{2}$	34.2%	50, 100	Operation misinterpretation ("division makes smaller")
P3: Estimate the sum of $\frac{12}{13} + \frac{7}{8}$	38.4%	19, 21	Fraction magnitude misconception
P11: $2\frac{3}{5} + 1\frac{2}{3}$	42.6%	$3\frac{5}{8}$, $3\frac{8}{15}$	Component-wise addition error
P20: $\frac{2}{3} - \frac{3}{5}$	44.9%	$\frac{1}{2}$, $\frac{1}{5}$	Whole-number bias
P28: $\frac{12}{15} \div \frac{4}{15}$	55.3%	$\frac{3}{15}$, $\frac{1}{5}$	Fraction division misconception

Note: Correct (%) shows the proportion of students who answered correctly. Common Incorrect Pattern(s) lists the most frequent wrong answers chosen by students. Misconception Type describes the underlying conceptual error driving the incorrect responses.

cal prompts across users and attempts—instability also documented by Gonzales [13] and Castleman et al. [6] across leading LLMs on mathematics problems. Instead, LLM estimates should be treated as provisional heuristics requiring empirical validation, not ground validated difficulty values. The misalignment between algorithmic and cognitive difficulty means LLMs may systematically misclassify which items will challenge students.

This difficulty misalignment is particularly important for educational data mining applications. In adaptive testing and automated item generation, difficulty estimates play a central role in sequencing content and personalizing learning experiences. When misconception-driven items are systematically rated as easier than they actually are, assessment systems may select inappropriate items, miscalibrate learner ability, and reduce the effectiveness of instructional interventions. These findings suggest that LLM-based assessment systems should be grounded in empirical learner data rather than relying solely on model-generated difficulty estimates. Overall, this study should be interpreted as an exploratory baseline for understanding how LLMs estimate mathematical difficulty in misconception-driven contexts. These results provide an initial foundation for future work on cognitively informed LLM-based assessment systems.

Several limitations should be considered when interpreting these findings. First, this study focuses on fraction arithmetic among Indonesian undergraduate students. Although fractions are well known for misconception-driven errors [3, 17], it remains unclear whether the "Easy Trap" phenomenon also appears in other mathematical domains of undergraduate learning such as algebra, geometry, or calculus. Future studies should examine whether similar patterns emerge across different topics. Second, while the dataset included 770 students, the analysis was based on only 32 arithmetic items, with the clearest evidence of systematic underestimation found in 11 fraction items. A broader evaluation involving more items and mathematical domains would provide stronger evidence regarding the consistency of LLM difficulty misalignment. In addition, the data were collected from a single institution, so further validation across institutions and cultural contexts is needed. Third, our prompting approach was intentionally designed to reflect realistic educational use, where educators ask LLMs to estimate item difficulty without detailed information about student misconceptions or error patterns. Although this mirrors current practice in AI-assisted assessment design, richer prompts that include misconception patterns or cognitive analysis may improve alignment between LLM predictions and empirical difficulty. Future research should investigate whether misconception-informed prompting can reduce this systematic bias.

6. CONCLUSIONS

This study shows that LLM-based difficulty estimation captures the relative ordering of item difficulty but systematically fails to represent cognitive difficulty driven by misconceptions. As a result, fraction items—where performance depends on conceptual understanding rather than procedural complexity—are consistently underestimated. We interpret this as evidence that LLMs encode curricular expectations rather than learner-centered difficulty, leading to a measurable and predictable bias.

Taken together, these findings suggest that while LLMs are promising tools for rapid item prototyping and scalable content generation, their effective use in assessment requires approaches that integrate model efficiency with empirical validation, psychometric calibration, and human expertise. Although this study focuses on fraction arithmetic, this domain provides a well-established testbed for examining persistent misconceptions and cognitive bottlenecks in mathematics learning. The observed misalignment therefore points to broader limitations of LLM-based assessment tools when

applied to conceptually demanding domains.

7. ACKNOWLEDGMENTS

This work was supported by the Indonesia Endowment Fund for Education (LPDP), Ministry of Finance of the Republic of Indonesia.

8. REFERENCES

[1] C. Acquaye, Y. T. Huang, M. Carpuat, and R. Rudinger. Take out your calculators: Estimating the real difficulty of question items with llm student simulations. *arXiv preprint arXiv:2601.09953*, 2026.

[2] F. B. Baker and S.-H. Kim. *Item response theory: Parameter estimation techniques*. CRC press, 2004.

[3] R. S. Baker and A. Hawn. Algorithmic bias in education. *International journal of artificial intelligence in education*, 32(4):1052–1092, 2022.

[4] B. Bentley and M. J. Bossé. College students' understanding of fraction operations. *International electronic journal of mathematics education*, 13(3):233–247, 2018.

[5] G. Biancini, A. Ferrato, and C. Limongelli. Multiple-choice question generation using large language models: Methodology and educator insights. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*, pages 584–590, 2024.

[6] J. Castleman, N. Nadeem, T. Namjoshi, and L. T. Liu. Rethinking math benchmarks for llms using irt. *Proceedings of Machine Learning Research*, 273:66–82, 2025.

[7] R. P. Chalmers. mirt: A multidimensional item response theory package for the r environment. *Journal of statistical Software*, 48:1–29, 2012.

[8] Y. Chu, P. He, H. Li, H. Han, K. Yang, Y. Xue, T. Li, J. Krajcik, and J. Tang. Enhancing llm-based short answer grading with retrieval-augmented generation. *arXiv preprint arXiv:2504.05276*, 2025.

[9] L. Crocker and J. Algina. *Introduction to classical and modern test theory*. ERIC, 1986.

[10] J. Cui, L. Wang, D. Li, and X. Zhou. Verbalized arithmetic principles correlate with mathematics achievement. *British Journal of Educational Psychology*, 94(1):41–57, 2024.

[11] C. Deeken, I. Neumann, and A. Heinze. Mathematical prerequisites for stem programs: What do university instructors expect from new stem undergraduates? *International Journal of Research in Undergraduate Mathematics Education*, 6(1):23–41, 2020.

[12] S. E. Embretson and S. P. Reise. *Item response theory: Foundations for psychologists and social scientists*. Routledge, 2025.

[13] M. A. A. Gonzalez, M. B. Hernandez, M. A. P. Perez, B. L. Orozco, J. T. C. Soto, and S. Malagon. Do repetitions matter? strengthening reliability in llm evaluations. *arXiv preprint arXiv:2509.24086*, 2025.

[14] S. S. Gray, B. J. Loud, and C. P. Sokolowski. Calculus students' use and interpretation of variables: Algebraic vs. arithmetic thinking. *Canadian Journal of Science, Mathematics and Technology Education*, 9(2):59–72, 2009.

[15] R. K. Hambleton and H. Swaminathan. *Item response theory: Principles and applications*. Springer Science & Business Media, 2013.

[16] B. M. Hijji. An indispensable requirement for medical dosage calculation: Basic mathematical skills of baccalaureate nursing students. *Nursing Reports*, 15(5):150, 2025.

[17] J. Hwang and P. J. Riccomini. A descriptive analysis of the error patterns observed in the fraction-computation solution pathways of students with and without learning disabilities. *Assessment for Effective Intervention*, 46(2):132–142, 2021.

[18] K. Kadir, Kodirun, E. Cahyono, A. Hadi, A. Sani, and Jafar. The ability of prospective teachers to pose contextual word problem about fractions addition. In *Journal of Physics: Conference Series*, volume 1581, page 012025. IOP Publishing, 2020.

[19] A. P. Kumar, A. Nayak, M. S. K, Chaitanya, and K. Ghosh. A novel framework for the generation of multiple choice question stems using semantic and machine-learning techniques. *International Journal of Artificial Intelligence in Education*, 34(2):332–375, 2024.

[20] A. La Hadi and D. Dedyerianto. Analisis data miskonsepsi siswa sekolah menengah pertama dalam menyelesaikan operasi aritmatika dasar. *Al-Ta'dib: Jurnal Kajian Ilmu Kependidikan*, pages 18–33, 2020.

[21] H.-J. Lee and I. Boyadzhiev. Underprepared college students' understanding of and misconceptions with fractions. *International Electronic Journal of Mathematics Education*, 15(3), 2020.

[22] M. Lestari, R. Johar, M. Mailizar, and A. Ridho. Measuring learning loss due to disruptions from covid-19: Perspectives from the concept of fractions. *Jurnal Didaktik Matematika*, 10(1):131–151, 2023.

[23] M. Li, H. Jiao, T. Zhou, N. Zhang, S. Peters, and R. W. Lissitz. Item difficulty modeling using fine-tuned small and large language models. *Educational and Psychological Measurement*, 85(6):1065–1090, 2025.

[24] Y. Li and G. Kulm. Knowledge and confidence of pre-service mathematics teachers: The case of fraction division. *ZDM*, 40(5):833–843, 2008.

[25] N. Nasution. Using artificial intelligence to create biology multiple choice questions for higher education. *Agricultural and Environmental Education*, 2(1):4–8, 2023.

[26] S. Nita, K. Sussolaikah, and J. D. Aldida. The role of artificial intelligence-based technology with chatgpt as an educational learning media innovation in indonesia. *International Journal of Multidisciplinary Sciences and Arts*, 2(4):235–241, 2023.

[27] C. Ormerod, S. Lottridge, A. E. Harris, M. Patel, P. van Wamelen, B. Kodeswaran, S. Woolf, and M. Young. Automated short answer scoring using an ensemble of neural networks and latent semantic analysis classifiers. *International Journal of Artificial Intelligence in Education*, 33(3):467–496, 2023.

[28] P. Razavi and S. Powers. Estimating item difficulty using large language models and tree-based machine learning algorithms. *arXiv preprint arXiv:2504.08804*, 2025.

[29] E. Rudolph, H. Seer, C. Mothes, and J. Albrecht. Automated feedback generation in an intelligent tutoring system for counselor education. In *2024 19th Conference on Computer Science and Intelligence Systems (FedCSIS)*, pages 501–512. IEEE, 2024.

[30] M. Ryals, S. Hill-Lindsay, M. E. Pilgrim, and L. C. Burks. 'simple mistakes' in college algebra: an analysis of students' perceptions of their errors using attribution theory. *International Journal of Research in Undergraduate Mathematics Education*, pages 1–28, 2025.

[31] R. S. Siegler, G. J. Duncan, P. E. Davis-Kean, K. Duckworth, A. Claessens, M. Engel, M. I. Susperreguy, and M. Chen. Early predictors of high school mathematics achievement. *Psychological science*, 23(7):691–697, 2012.

[32] M. W. H. Spitzer and K. Moeller. Predicting fraction and algebra achievements online: A large-scale longitudinal study using data from an online learning environment. *Journal of Computer Assisted Learning*, 38(6):1797–1806, 2022.

[33] S. Stewart and S. Reeder. Algebra underperformances at college level: What are the consequences? In *And the rest is just algebra*, pages 3–18. Springer, 2016.

[34] D. A. Susanto, A. Priyolistiyanto, F. Pinandhita, A. P. KA, and D. S. Bimo. Utilizing chatgpt on designing english language teaching (elt) materials in indonesia: Opportunities and challenges. *Celt: A Journal of Culture, English Language Teaching & Literature*, 24(1):157–171, 2024.

[35] T. Susnjak. Beyond predictive learning analytics modelling and onto explainable artificial intelligence with prescriptive analytics and chatgpt. *International Journal of Artificial Intelligence in Education*, 34(2):452–482, 2024.

[36] V. Tariq. Diagnosis of mathematical skills among bioscience entrants. *Diagnostic testing for mathematics*, pages 14–15, 2003.

[37] K. VanLehn, F. Milner, C. Banerjee, and J. Wetzel. A step-based tutoring system to teach underachieving students how to construct algebraic models. *International journal of artificial intelligence in education*, 34(2):224–246, 2024.

[38] R. Weegar and P. Idestam-Almquist. Reducing workload in short answer grading using machine learning. *International Journal of Artificial Intelligence in Education*, 34(2):247–273, 2024.

Toward a Cyclic Evidence–Task Design Framework: A Preliminary Study of Generating Formative Scaffolding Tasks from Solution Logs with a Large Language Model

Yoonji Song
Seoul National University
songyj@snu.ac.kr

Insub Shin
Seoul National University
iss9802@snu.ac.kr

Yun Joo Yoo
Seoul National University
yyoo@snu.ac.kr

ABSTRACT

Formative assessment in mathematics should attend not only to answer correctness but also to the procedural evidence embedded in students' problem-solving processes. Evidence-centered design (ECD) offers a principled basis for connecting learner evidence to task design. However, in classroom contexts, teachers face substantial time and cognitive burden when designing formative assessment tasks based on students' evidence. This study presents a preliminary implementation of a task–evidence–task cycle for formative assessment using a large language model (LLM). Using 1,406 solution logs collected from 29 students, the LLM infers required knowledge components (KCs) from students' step-by-step solution logs and constructs diagnostic evidence (e.g., procedural differences, targetable causes, and required KCs). It then uses this evidence to generate two scaffolding tasks for each original task. Expert evaluation showed moderate agreement on the generated KCs and substantial agreement on the appropriateness of the scaffolding tasks. A small follow-up activity with four students further suggested that the generated tasks may help make missing intermediate reasoning steps more visible during re-solving. These findings provide preliminary feasibility evidence for one implemented task–evidence–task cycle, while repeated cyclic use is left for future work.

Keywords

Evidence-centered Design, Scaffolding Task Generation, Step-by-Step Solution Logs, Large Language Model

1. INTRODUCTION

In mathematics education, formative assessment plays a key role in supporting students' learning throughout the problem-solving process. It should therefore attend not only to the correctness of final answers but also to the processes through which those answers are produced.

Mathematical problem solving involves interpreting problem situations and coordinating multiple cognitive components. Koedinger et al. [7] define these as knowledge components (KCs), or learnable units of knowledge or cognitive function. Because procedural and conceptual knowledge interact and develop together in mathematics learning [13], students' procedural work can be interpreted as evidence of KC use during problem solving. Assessment should therefore move beyond answer correctness to examine process-level evidence in students' work.

Evidence-centered design (ECD) provides a principled framework for making such inferences by specifying relationships among claims, evidence, and tasks [10, 12]. However, many ECD implementations remain largely task-driven: tasks are specified first, and evidence models are used mainly to interpret task outcomes [1]. As a result, the process by which diagnostic evidence informs subsequent task design often remains underspecified or is treated as a black box [5].

To address this gap, this study adopts KCs as an interpretive unit for constructing diagnostic evidence and implements a task–evidence–task cycle in which evidence from students' solution logs informs subsequent task design. Operationalizing such redesign in classrooms is challenging because extracting KCs from student work and redesigning formative tasks require substantial time and cognitive effort from teachers [2]. Although recent studies have explored large language models (LLMs) for analyzing student work or generating assessment-related artifacts [8, 9, 11, 16], most have focused on isolated functions such as task generation, automated scoring, or feedback rather than integrated, evidence-based task redesign [6].

This paper presents a preliminary implementation of an ECD-grounded task–evidence–task cycle for formative assessment. The LLM is used for two related functions: (1) constructing diagnostic evidence from students' solution logs, including inferred KCs, and (2) generating scaffolding tasks based on that evidence. The generated tasks are reviewed by two experts, and their appropriateness and potential educational usefulness are examined.

Accordingly, this study addresses the following research questions (RQs):

RQ1. To what extent can an LLM generate meaningful KC candidates required for problem solving from students' solution logs?

RQ2. To what extent can an LLM generate formative scaffolding tasks based on KCs inferred from students' solution

Yoonji Song, Insub Shin, and Yunjoo Yoo. Toward a Cyclic Evidence–Task Design Framework: A Preliminary Study of Generating Formative Scaffolding Tasks from Solution Logs with a Large Language Model. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 793–797. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039827

logs?

2. BACKGROUND

2.1 Evidence-Centered Design for Formative Assessment

Evidence-centered design (ECD) frames assessment as a principled approach to making inferences about learning from observable performance [10, 12]. By organizing assessment around claims, evidence, and tasks, ECD emphasizes that validity depends not only on task format or difficulty but also on what evidence is collected from learners' work and how it is interpreted [8].

This perspective is especially relevant to formative assessment, which aims to identify learners' current difficulties and provide targeted support [2]. In ECD-based formative assessment, tasks are not only measurement tools but also means of making learners' thinking visible. When scaffolding is evidence-based, it should respond to procedural bottlenecks, omissions, and error patterns observed in students' actual work rather than rely on fixed hints [15].

Although ECD presupposes an ongoing interplay between evidence and task design [10], its applications have typically proceeded in a forward direction: tasks are defined first, and evidence models are then developed to interpret the resulting performances. In such task-driven implementations, the link between diagnosis and the design of subsequent tasks is often left implicit [1, 5]. Recent LLM-based studies have addressed parts of this broader process, including the analysis of student work and the generation of assessment-related outputs [8, 9, 11, 16]. Most, however, have focused on isolated functions, such as scoring, feedback, or task authoring, rather than on evidence-driven formative task redesign as an integrated process. Moreover, many existing approaches still generate tasks primarily on the basis of predefined learning objectives or answer structures [3], with relatively little work systematically using evidence from students' problem-solving processes to guide subsequent task design.

In this study, we adopt an ECD-grounded cyclic Evidence–Task Design Framework in which evidence derived from student solution logs is translated into subsequent formative scaffolding task design.

2.2 Knowledge Components as Units of Evidence

Knowledge components (KCs) are acquired units of cognitive function or knowledge structure that can be inferred from performance on related tasks [7]. In mathematics, a task typically requires multiple KCs rather than a single one [13]. Accordingly, assessments based only on answer correctness may fail to capture learners' actual difficulties.

Prior work has shown that errors, omissions, and incomplete solution steps can serve as evidence for inferring learners' knowledge states [4]. In this study, KCs are operationally defined as units of knowledge that enable mathematical performance, such as definitions, conditions, meanings, and interpretation rules [7]. KC-related errors or omissions in students' solution processes are interpreted as indicators of learning gaps and used as fine-grained evidence for subsequent scaffolding task design. Recent studies have also explored LLMs as tools for supporting this kind of evidence interpretation and generation process [4, 11, 16].

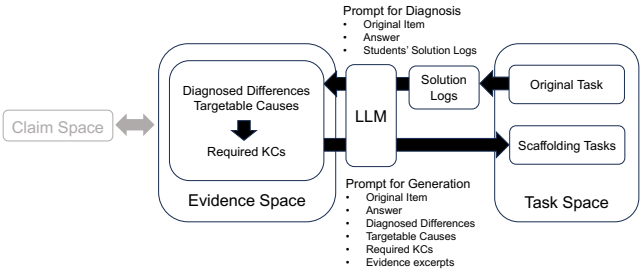


Figure 1: Overview of one implemented task–evidence–task cycle using an LLM.

3. TASK–EVIDENCE–TASK CYCLE

This study presents one implemented task–evidence–task cycle (Figure 1) in which an LLM converts student solution logs into diagnostic evidence and then uses that evidence to design scaffolding tasks. The cycle links a task space to an evidence space through two prompt-based transformations: diagnosis (task space → evidence space) and generation (evidence space → task space). Knowledge components (KCs) function as the shared representational basis across these stages, linking what is inferred from students' work to what is subsequently targeted in scaffolding-task design.

3.1 Prompt Design

We implemented the prompting pipeline in two stages: a diagnosis prompt for constructing evidence from student logs, and a generation prompt for producing scaffolding tasks from the diagnosed evidence.

3.1.1 Prompt from Task Space to Evidence Space

The diagnosis prompt takes as input the original task, the final answer, and students' line-level solution logs. For each task, the logs are grouped into correct and non-correct responses (including incorrect and unfinished responses) based on final-answer correctness. The prompt then asks the LLM to compare the two groups and identify recurring procedural differences, such as omitted conditions, incorrect relational expressions, or points at which solutions stop before completion. Based on these differences, the LLM is prompted to identify targetable causes, that is, aspects of performance that appear to be missing, incomplete, or insufficiently activated and that could plausibly be addressed through instructional support. The prompt also asks the model to infer 10 required KCs associated with those causes and needed to solve the original task. During prompt development, candidate set sizes of 5, 10, and 15 were examined across tasks of different difficulty levels, and 10 was adopted for the present implementation.

To make the diagnosis explicit and traceable, the prompt requires all outputs to be grounded in concrete excerpts from the solution logs, including equations, symbols, and written statements. These excerpts serve as evidence for the identified procedural differences, targetable causes, and related KCs.

3.1.2 Prompt from Evidence Space to Task Space

The generation prompt takes as input the original task, the final answer, the diagnosed procedural differences, the targetable causes, the generated KC candidates, and supporting evidence excerpts from the solution logs. Using these inputs, the LLM generates scaffolding tasks intended to address the diagnosed difficulties revealed in the earlier stage.

The prompt constrains the LLM not to generate tasks that simply rehearse isolated KCs or lead directly to the final answer. Instead, the generated tasks are designed to elicit intermediate products that make students' reasoning visible, such as stating relevant conditions, constructing relational expressions, or explaining the meaning of a result. Tasks that primarily emphasize answer production or direct computation are excluded.

For each original task, the prompt generates two scaffolding tasks. For each generated task, it also specifies the targetable cause addressed, the supporting evidence excerpt, the related KC, and the intended intermediate product. In this way, the evidence constructed from student work is carried forward into subsequent task design within one implemented task–evidence–task cycle.

4. EVALUATION

This section evaluates the present implementation by reporting (1) the dataset used in the evaluation, (2) the quality of LLM-based KC extraction (RQ1), and (3) expert and student evaluations of the generated scaffolding tasks (RQ2).

Grade 10 students ($N = 29$) in South Korea provided handwritten solutions, and all personally identifiable information was removed. These were reviewed by the researchers, transcribed into LaTeXformat, and segmented into a line-level solution log dataset for KC-based diagnosis, with each line representing one written step in a student's solution. The generated scaffolding tasks were reviewed by two experts (one mathematics teacher and one mathematics education researcher) and tested in a small-scale follow-up activity examining changes in answer accuracy and solution processes. The diagnosis and task generation processes were implemented using GPT-5.2.

4.1 Data Description

Problem-solving data were collected from 15 tasks on quadratic equations, quadratic functions, and quadratic inequalities at the Grade 10 level. These tasks were categorized into three difficulty levels (easy, intermediate, and difficult) based on their cognitive and procedural demands.

Twenty-nine students freely chose which tasks to solve. One solution per student per task was collected and decomposed into line-by-line units, yielding 1,406 solution logs.

Table 1 summarizes answer accuracy and the average number of solution steps by difficulty level. Easy-level tasks showed 85.1% accuracy with 3.08 steps on average, intermediate-level tasks 75.6% with 4.15 steps, and difficult-level tasks 62.7% with 5.99 steps. Overall, lower accuracy tended to be associated with longer solution processes.

At the individual-task level, the two lowest-accuracy tasks showed 17.6% and 33.3% accuracy, respectively; the latter also had the second-highest average number of solution steps.

Table 1: Summary of student performance by difficulty level

Level	Items	Attempts	Accuracy (%)	Steps
Easy	5	87	85.1	3.08
Intermediate	4	90	75.6	4.15
Difficult	6	118	62.7	5.99

4.2 KC Extraction from Solution Logs (RQ1)

To examine whether the LLM could meaningfully extract KCs from students' solution logs, the model generated 10 KC candidates for each of the 15 tasks. Two experts judged the appropriateness of each of the 150 KC candidates using a binary scale (appropriate = 1, inappropriate = 0).

Inter-rater agreement indicated moderate consistency (Cohen's $\kappa = 0.52$; overall agreement = 0.78), suggesting that judgments about individual KC candidates were not fully stable.

Table 2: Task-level validity and stability of LLM-generated KC candidates

Measure	Value
Strict validity rate (both experts = valid)	0.533
Lenient validity rate ($\geq$1 expert = valid)	0.753
Jaccard similarity (median)	0.750

Task-level analysis further showed a strict validity rate of 0.533, a lenient validity rate of 0.753, and a median Jaccard similarity of 0.750 (Table 2). These results suggest substantial task-level overlap between the LLM-generated KC sets and those identified by expert raters, while also indicating limited agreement on individual KC candidates. The generated KCs should therefore be interpreted as provisional rather than definitive diagnostic units.

We also examined the relationship between the average number of solution steps per task and the average number of valid KCs identified by expert raters. A significant positive Spearman rank correlation was observed ($\rho = 0.52$, $p = .048$), indicating that tasks with longer solution logs tended to yield more expert-validated KC candidates.

4.3 Appropriateness of Generated Scaffolding Tasks (RQ2)

To examine the appropriateness of the generated scaffolding tasks for formative assessment, two experts independently evaluated them using a five-point Likert scale (Table 3). For each original task, two scaffolding tasks were generated, resulting in 30 tasks in total.

Inter-rater agreement, assessed with quadratic weighted Cohen's kappa, was substantial (0.76 for Scaffolding Task 1 and 0.73 for Scaffolding Task 2). Both experts also assigned mean ratings above the midpoint of the scale (Expert 1: M = 3.53, SD = 1.50; Expert 2: M = 3.83, SD = 1.46), indicating generally positive evaluations and suggesting that the generated scaffolding tasks were reasonably appropriate for formative use.

Table 3: Evaluation criteria for the appropriateness of scaffolding tasks

Score	Description
1	The task is unrelated to the original task or is mathematically invalid
2	The task is almost identical to the original task and is not appropriate as a scaffolding task (mere repetition)
3	The scaffolding task focuses on a peripheral KC and should be revised to target a more central step
4	The KC level targeted by the scaffolding task is appropriate, but minor revisions to numbers or wording are needed
5	Very appropriate as a scaffolding task for the original task

Table 4: Example of an original task and two LLM-generated scaffolding tasks

Task	Description
Original task	You plan to create a rectangular garden with an area of at least $400\,m^2$. The perimeter of the garden is $100\,m$, and the length of the longer side is $x\,m$. Find the range of possible values of x. (Assume that the length is greater than the width.)
Scaffolding task 1	For a rectangle with perimeter $100\,m$, express the width in terms of x when the length is $x\,m$. Then write the condition "the length is greater than the width" as a single inequality in x.
Scaffolding task 2	For a rectangle with perimeter $100\,m$, when the length is $x\,m$, rewrite the condition that the area is at least $400\,m^2$ as a single quadratic inequality involving only x.

To obtain preliminary qualitative observations, a follow-up re-solving activity was conducted with four students who had answered incorrectly on the two original tasks with the lowest initial answer accuracy. The students were provided with the original task together with scaffolding tasks that experts had judged to be appropriate.

For the lowest-accuracy task, two of the four students answered correctly in the re-solving attempt, whereas two remained incorrect. For the second-lowest-accuracy task, which involved a longer solution process (Table 4), three students answered correctly and one remained incorrect. Qualitative review of the re-solving processes showed that some students stated previously omitted conditions or added intermediate expressions that were absent from the original solutions. These observations indicate that student responses differed across the two tasks in terms of both final correctness and the procedural changes observed during re-solving.

5. DISCUSSION

Regarding RQ1, the results suggest that an LLM can generate plausible KC candidates from students' step-by-step solution logs, with moderate consistency in expert judgments. The agreement and task-level overlap measures indicate that the generated KC candidates generally fell within a range experts considered appropriate, although the exact KC sets did not always coincide. Thus, the generated KCs are better interpreted as provisional diagnostic units than as definitive representations of student knowledge. The positive relationship between the number of solution steps and the number of valid KCs further suggests that the LLM responded systematically to differences in procedural evidence rather than producing arbitrary KC sets.

Regarding RQ2, expert evaluations showed substantial agreement on the appropriateness of the generated scaffolding tasks, suggesting that the LLM-produced tasks were generally suitable for formative use. Rather than directly drilling individual KCs or simplifying the original tasks, these tasks were designed to elicit intermediate products that make learners' thinking visible, such as stating conditions, constructing relational expressions, or articulating the meaning of a solution. This design characteristic aligns with the view that formative assessment tasks should reveal learners' reasoning processes and support the reconstruction of missing or weak procedural steps [2, 14].

The follow-up study with four students provided additional qualitative observations about the potential usefulness of the generated scaffolding tasks. In these cases, some students stated previously omitted conditions or added intermediate expressions after working on the generated tasks. These observations indicate that the tasks may have prompted procedural actions that were not visible in the original solutions, although this interpretation remains tentative given the small-scale design. Differences in responses across the two tasks may reflect differences in their cognitive demands, suggesting that the usefulness of scaffolding may depend partly on task characteristics. Future research could examine how the cognitive demand levels of KCs and interactions among multiple KCs relate to scaffolding design and effectiveness.

Theoretically, this study illustrates how a task–evidence–task cycle can be implemented between the task space and the evidence space within an ECD framework. The present design shows how evidence derived from learners' performances can be organized into diagnostic representations and used to inform subsequent task design in a formative assessment context. In this sense, the study provides a preliminary illustration of how an LLM may support diagnostic interpretation and subsequent task generation within an ECD-based assessment process.

Practically, we envision the proposed pipeline as a teacher-in-the-loop support tool that generates candidate scaffolding tasks from student solution logs. This workflow may help reduce teacher burden by supporting the design of follow-up formative tasks while preserving teacher control over task selection, revision, and classroom use. Future work could examine how this process functions in authentic classroom settings and how the framework may be extended to include the Claim Space within a broader ECD process.

6. CONCLUSIONS AND LIMITATIONS

This study implemented one task–evidence–task cycle by connecting students' solution logs (task space) with KC-based diagnostic evidence (evidence space) to generate formative

scaffolding tasks. Preliminary results suggest that an LLM can generate plausible KC candidates from students' solution logs and that tasks with richer procedural evidence tended to yield more expert-validated KC candidates. Expert review and a small-scale follow-up activity further suggested that the generated scaffolding tasks may help students recognize and produce intermediate products required for problem solving. Together, these findings provide preliminary feasibility evidence that an LLM may support one evidence-informed task–evidence–task cycle for formative assessment grounded in learners' observed problem-solving processes.

Several limitations should be noted. First, the present study examined only a single implemented task–evidence–task cycle rather than repeated cycles across multiple rounds. Second, the study was limited to algebraic topics, specifically quadratic equations, quadratic functions, and quadratic inequalities, and did not examine other areas of mathematics such as geometry or statistics. Third, our evaluation of KC extraction focused on precision, but could not assess recall without predefined reference KC sets for each task. Future work should establish such reference sets for comprehensive evaluation. Fourth, the follow-up study involved only a small number of students and did not include a control group. Therefore, the follow-up findings should be interpreted as preliminary qualitative observations.

7. REFERENCES

[1] Meirav Arieli-Attali, Sue Ward, Jay Thomas, Benjamin Deonovic, and Alina A. von Davier. The expanded evidence-centered design (e-ECD) for learning and assessment systems: A framework for incorporating learning goals and processes within assessment design. *Frontiers in Psychology*, 10:853, 2019.

[2] Paul Black and Dylan Wiliam. Developing the theory of formative assessment. *Educational Assessment, Evaluation and Accountability*, 21(1):5–31, 2009.

[3] Kuang Wen Chan et al. Automatic item generation in various stem subjects using large language model prompting. *Computers and Education: Artificial Intelligence*, 8:100344, 2025.

[4] Mehmet Arif Demirtas. Identifying and evaluating novel knowledge component models for programming skills. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 969–973, 2024.

[5] Mingyu Feng, Eric G. Hansen, and Diego Zapata-Rivera. Using evidence-centered design for learning (ECDL) to examine the assistments system. In *Proceedings of the Annual Meeting of the American Educational Research Association*, San Diego, CA, USA, 2009.

[6] Enkelejda Kasneci et al. ChatGPT for good? on opportunities and challenges of large language models for education. *Computers and Education: Artificial Intelligence*, 4:100193, 2023.

[7] Kenneth R. Koedinger, Albert T. Corbett, and Charles Perfetti. The Knowledge-Learning-Instruction Framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, 36(5):757–798, 2012.

[8] Holger Maus, Paul Tschisgale, Fabian Kieser, Stefan Petersen, and Peter Wulff. Developing and evaluating a large language model-based automated feedback system grounded in evidence-centered design for supporting physics problem solving. *arXiv preprint arXiv:2512.10785*, 2025.

[9] Jennifer Meyer et al. Using LLMs to bring evidence-based feedback into the classroom: AI-generated feedback increases secondary students' text revision, motivation, and positive emotions. *Computers and Education: Artificial Intelligence*, 6:100199, 2024.

[10] Robert J. Mislevy, Russell G. Almond, and Janice F. Lukas. A brief introduction to evidence-centered design. *ETS Research Report Series*, 2003(1):i–29, 2003.

[11] Hyeongdon Moon, Richard Lee Davis, Seyed Parsa Neshaei, and Pierre Dillenbourg. Using large multimodal models to extract knowledge components for knowledge tracing from multimedia question information. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 342–353, 2025.

[12] James W. Pellegrino. Assessment as a positive influence on 21st century teaching and learning: A systems approach to progress. In Patrick Griffin, Barry McGaw, and Esther Care, editors, *Assessment and Teaching of 21st Century Skills*, pages 167–183. Springer, Dordrecht, 2014.

[13] Bethany Rittle-Johnson and Martin Schneider. Developing conceptual and procedural knowledge in mathematics. In Roi Cohen Kadosh and Ann Dowker, editors, *The Oxford Handbook of Mathematical Cognition*, pages 275–290. Oxford University Press, 2014.

[14] Lorrie A. Shepard. The role of assessment in a learning culture. *Educational Researcher*, 29(7):4–14, 2000.

[15] Janneke van de Pol, Monique Volman, and Jos Beishuizen. Scaffolding in teacher–student interaction: A decade of research. *Educational Psychology Review*, 22(3):271–296, 2010.

[16] Yumou Wei, Paulo Carvalho, and John Stamper. Kcluster: An LLM-based clustering approach to knowledge component discovery. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 228–240, 2025.

Unsupervised Anomaly Detection of Learning Behaviors Considering Contextual Information

Yoshihiro Yasuda
Chubu University
yoshi@mprg.cs.chubu.ac.jp

Takayoshi Yamashita
Chubu University
takayoshi@fsc.chubu.ac.jp

Tsubasa Hirakawa
Chubu University
hirakawa@fsc.chubu.ac.jp

Hironobu Fujiyoshi
Chubu University
fujiyoshi@fsc.chubu.ac.jp

ABSTRACT

With the widespread adoption of digital learning environments, learning support utilizing log data collected from Learning Management Systems (LMS) has garnered significant attention. However, many conventional learning analytics rely on aggregated metrics such as total study time or the number of accesses, thereby failing to capture complex temporal dynamics, such as the sequential context of learning behaviors. In this study, we propose an unsupervised anomaly detection method that extracts atypical behaviors that deviate from standard learning patterns using students' behavior stream data. Specifically, we use a Transformer Encoder to generate contextualized embedding vectors from the time-series sequences of learning behaviors, and apply an Isolation Forest to detect anomalous behaviors without incurring labeling costs. Experimental results using actual university learning log data confirmed that the majority of the detected anomalous behaviors were immediate exits (i.e., merely opening and closing materials) lacking the context of page transitions. While these anomalous behaviors were predominantly observed among low-performing students (F and D grades), approximately 30% of the logs classified as anomalous belonged to high-performing students (A grades). Atypical behaviors extracted through unsupervised anomaly detection suggest the inclusion of not only indications of learning deficiencies, such as immediate exits, but also advanced learning strategies adopted by high-performing students, such as the concurrent study of multiple materials and the marking of crucial sections.

Keywords

Educational Data Mining, Anomaly Detection, Behavior Stream Data, Transformer Encoder, Isolation Forest

1. INTRODUCTION

The widespread adoption of digital learning environments has led to increased interest in learning analytics utilizing log data collected from Learning Management Systems (LMS). Many conventional analyses have relied on aggregated metrics, such as access frequency and total learning time, which fail to capture complex temporal dynamics, including the sequential context of learning behaviors. For example, even with the same number of actions, the learning process of a student who rapidly skips through pages differs fundamentally from that of a student who occasionally returns to review previous content. To address this issue, we propose an unsupervised anomaly detection method that considers the context of learning behaviors. In educational data, clearly defining what constitutes effective or ineffective learning and manually labeling such data entails substantial labor and computational costs. Furthermore, supervised learning carries the risk of overlooking unknown learning strategies that do not fit into predefined categories. Therefore, we adopted an approach that establishes the standard learning behaviors exhibited by the majority of students as a baseline and detects anomalous behaviors deviating from this norm in an unsupervised manner. Specifically, we use a Transformer Encoder [9] to convert temporal sequences into feature vectors that retain contextual information, and then detect anomalous behavior patterns using an Isolation Forest [6]. The main contributions of this study are as follows:

- Context-aware feature extraction: We proposed an embedding method that captures the structure of temporal behavior sequences by combining Inverse Document Frequency (IDF) weighting based on action frequency with a Masked Language Modeling (MLM) task.
- Construction of an unsupervised anomaly detection pipeline: We built a framework that eliminates the costs and biases associated with labeling, directly extracting distinctive behaviors from behavioral stream data.
- Validation of effectiveness using real-world data: Through experiments using actual university learning logs, we demonstrated that the proposed method can accurately extract not only a lack of learning, such as immediate exits, but also advanced learning strategies employed by high-performing students.

2. RELATED WORK

In Educational Data Mining, analyses utilizing learning behavior logs are actively being conducted. Kohama et al.

Yoshihiro Yasuda, Takayoshi Yamashita, Tsubasa Hirakawa, and Hironobu Fujiyoshi. Unsupervised Anomaly Detection of Learning Behaviors Considering Contextual Information. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 798–802. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040087

proposed a method for predicting academic performance by representing the occurrence counts of actions within digital materials as histogram features [5]. While this method is effective in capturing overall grade trends, it relies on the mere aggregation of actions, leading to the loss of sequential and contextual specificities of learning. Recently, sequence modeling using Transformer architectures has gained significant traction in EDM. For instance, models such as Self-Attentive Knowledge Tracing (SAKT) [8] and SAINT [1] have successfully applied attention mechanisms to model complex dependencies in student interaction sequences. While these models excel in processing sequential data, they are primarily designed for supervised tasks, such as predicting correctness or final grades. Consequently, they require substantial labeled data and risk overlooking novel or atypical learning strategies that fall outside predefined categories. On the other hand, as an attempt at anomaly detection considering sequential context, Jiang et al. proposed an approach that models sequential data using self-supervised learning and detects anomalous learning behaviors based on the prediction error of the next action [4]. However, because this method relies on unidirectional prediction from the past to the future, it is limited by its inability to fully capture bidirectional dependencies of preceding and succeeding actions. To resolve these issues, this study proposes an anomaly detection approach that extracts unknown, atypical learning patterns in an unsupervised manner using an Isolation Forest, while simultaneously preserving the contextual information of the action sequences through the use of a Transformer Encoder.

3. PROPOSED METHOD

The proposed method consists of three components: training an embedding model via an MLM task, generating vector representations of behavior patterns by averaging the vector representations of individual actions, and detecting action sequences containing anomalous behaviors using an Isolation Forest. An overview of the proposed method is shown in Figure 1.

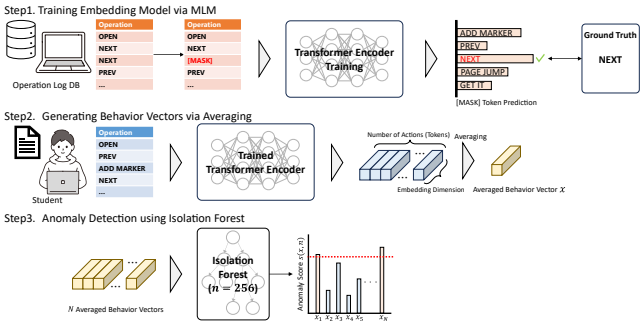


Figure 1: Overview of the proposed method.

3.1 Training the Embedding Model via MLM Task

A Transformer Encoder is utilized to acquire vector representations that capture contextual information from the time-series data of learning behaviors. During training, a Masked Language Modeling (MLM) task was employed to account for the relationships with preceding and succeeding actions. Within the learning behavior logs serving as training data, a significant imbalance exists between actions that are frequently performed by many students and those that occur rarely. Because it is crucial to consider not only the simple frequency of occurrence but also how common an action generally is, this method adopts Inverse Document Frequency (IDF)a metric focusing on the proportion of students who performed the action as a weight for the loss function. Building upon the architecture of BERT-Base [2], the model maps each discrete action into a dense continuous space. The embedding dimension was set to 768, following the standard configuration of BERT-Base, to preserve sufficient representational capacity for modeling complex behavioral patterns. This allows the model to learn structural dependencies, such as in what sequence or context a certain action is likely to occur, factoring in the specificity of the learning behavior rather than relying solely on raw action frequencies.

3.2 Generation of Behavior Pattern Vector Representations via Averaging

The Transformer Encoder, optimized through the MLM task, is used to extract features from the learning behavior sequences targeted for anomaly detection. Each learner's sequence of actions is fed into the model to obtain contextualized vector representations for the individual actions constituting the sequence. Subsequently, the obtained vector representations of each action are averaged over the length of the action sequence. Through this process, a fixed-length vector representing the entire sequence of learning behavior patterns is generated while preserving the context of individual actions. This enables the comparison and evaluation of diverse action sequences of varying lengths within a unified feature space.

3.3 Construction of the Isolation Forest

An Isolation Forest, an unsupervised anomaly detection algorithm based on decision trees, is applied to the generated averaged behavior vectors. During the construction phase, the data is recursively partitioned into two by randomly selecting a feature and a split point randomly determined between its maximum and minimum values. By repeating this process, numerous decision trees are constructed to form the forest.

3.4 Anomaly Detection Based on Path Length

Using the constructed Isolation Forest, the anomaly score for each behavior sequence is calculated based on the average path length required for the data point to become isolated in a leaf node. The anomaly score is defined by Equation (1).

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{1}$$

Here, x is the data point, $h(x)$ is the path length required for data x to reach a leaf node, $E(h(x))$ is the average path length across all decision trees, and $c(n)$ is the average path length for a dataset of size n. Normal data require many splits to be isolated, whereas anomalous data are isolated with fewer splits, generally resulting in shorter path lengths. This property is leveraged to detect anomalous behaviors.

Table 1: Results of anomaly detection.

Result	Count	Ratio [%]	Avg. Actions	Grade Distribution [%]				
				A	B	C	D	F
Normal	1,389	94.9	152.2	38.7	27.1	19.8	9.8	4.6
Anomalous	74	5.1	30.6	28.4	16.2	23.0	21.6	10.8

4. EVALUATION EXPERIMENTS

In this experiment, an Isolation Forest was applied to the feature vectors generated by the proposed method to calculate the anomaly score of each action sequence. Subsequently, the top 5% of the calculated anomaly scores were extracted as anomalous action sequences.

4.1 Datasets

In this study, we used learning behavior log data from 1,446 students, collected via the learning management system BookRoll [7, 3] during lectures conducted at Kyushu University between the 2019 and 2022 academic years. The data consists of seven courses, labeled A through G, which vary in lecture content and instructors. For the experiment, the log data was restricted to activities occurring within lecture hours, and was divided into training data (data up to the 2021 academic year) and evaluation data (data from the 2022 academic year). The logs contain detailed records of learning operations, such as user IDs, operation types, and operation timestamps. Among all operations, NEXT accounts for approximately 61% and PREV for about 27%, meaning these two page transition operations alone constitute nearly 90% of the total.

4.2 Experimental Results

Table 1 presents the results of the anomaly detection. In the evaluation data, while the average number of actions for normal data was approximately 152.2, the anomalous data averaged only 30.6 actions, confirming a significant tendency for fewer actions in the latter group. Examining the grade distribution, the anomalous data group exhibited a higher proportion of F and D grades. Conversely, students with an A grade also accounted for nearly 30% (28.4%) of the anomalous data. This is likely because the detection captured weeks where students exhibited atypical behavior patterns differing from standard learning behaviors, or weeks where the number of actions significantly decreased due to extremely short study times caused by arriving late or leaving early.

4.3 Ablation Study on IDF Weighting

To verify the effectiveness of IDF weighting, we compared the anomaly detection results with a model that does not use IDF (the baseline). Table 2 shows the anomaly detection results without IDF. Compared to Table 1, while no major differences were observed between the presence and absence of IDF in terms of the total number of detections or the average number of actions, the proposed method incorporating IDF showed a slight tendency to improve the detection rate of students in the lower academic tiers (F and D grades). Furthermore, approximately 75% (55 instances) of the extracted anomalous data were common to both models, and these were primarily data with an extremely low number of actions, such as immediate exits.

Table 2: Anomaly detection results without IDF weighting (baseline).

Result	Count	Ratio [%]	Avg. Actions	Grade Distribution [%]				
				A	B	C	D	F
Normal	1393	95.2	151.9	38.6	27.1	19.7	9.9	4.7
Anomalous	70	4.8	30.7	28.6	17.1	25.7	20.0	8.6

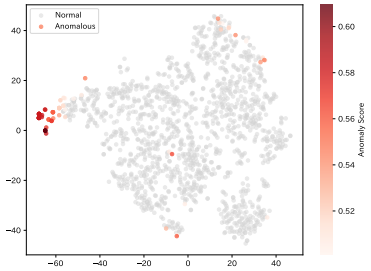


Figure 2: Anomaly score distribution.

5. ANALYSIS AND DISCUSSION

Based on the experimental results from the previous section, we present a detailed analysis of the behavior histories classified as anomalous.

5.1 Qualitative Evaluation by Clustering

To qualitatively evaluate the characteristics of the features generated by the proposed method, we visualized the distribution of the averaged behavior vectors. We employed t-SNE as a visualization method to compress the high dimensional feature space into a two-dimensional plane. For clustering, we utilized the k-means algorithm and set the number of clusters to 14. Figure 2 shows the anomaly score distribution of the generated averaged behavior vectors, and Figure 3 shows the results of clustering and color-coding each action sequence. From Figure 2, it is evident that data with high anomaly scores are densely concentrated at the edge of the feature space, forming an independent region. Correlating this region with the clustering results in Figure 3 revealed that it corresponds to a specific cluster primarily consisting of actions like OPEN (opening materials) and CLOSE (closing materials), accompanied by a low total number of actions. This indicates behaviors where the student immediately exited without reading the content after accessing the system. Furthermore, because these sequences lack the context of page transitions commonly seen among most students, they are isolated in the feature space and are thus considered to have been classified as anomalous. In contrast, the region corresponding to normal data was confirmed to represent typical learning behavior sequences characterized by a high number of actions and frequent page transitions, predominantly NEXT. Here, to confirm the effectiveness of the IDF weighting described in Section 4.3, we compare our findings with the clustering results of the baseline (Figure 4). As shown in Figure 4, in the baseline, sequences primarily consisting of OPEN and CLOSE are mixed within the same cluster, grouped together simply as the opening and closing of materials. In contrast, in the proposed method (Figure 3), these are placed in distinct clusters, indicating that the differences between each operation are captured more clearly.

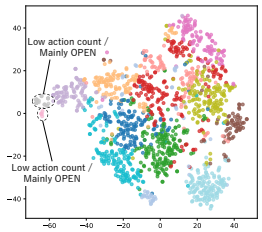


Figure 3: Clustering results of the proposed method.

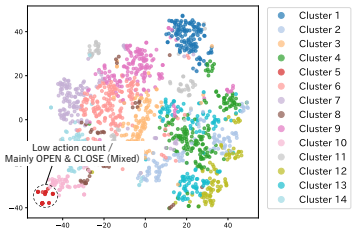


Figure 4: Clustering results of the baseline.

Table 3: Comparison of operations between the top 5 and lowest 5 anomaly sequences.

	OPEN	CLOSE	NEXT	PREV
Top 5 anomalies	3	4	1	0
Lowest 5 anomaly scores	14	10	601	206

5.2 Comparison of Operations between Anomalous and Normal Data

To clarify the specific breakdown of anomalous behaviors, we compared the constituent operations and total action counts of the top 5 anomaly sequences and the sequences with the lowest 5 anomaly scores (Table 3). The top 5 sequences exhibited an extremely low number of operations; while OPEN occurred 3 times and CLOSE 4 times, the primary page transition operations NEXT and PREV occurred only 1 and 0 times, respectively. This indicates instances of closing materials without reading or sessions ending immediately, demonstrating that the sheer lack of learning behavior is successfully detected as anomalous. On the other hand, the sequences with the lowest 5 anomaly scores involved over 800 total operations, with NEXT (601 times) and PREV (206 times) accounting for the vast majority. The ratio of NEXT to PREV is approximately 3:1, suggesting a steady learning pattern of advancing through the material and occasionally returning to review. Stable learning sessions are also evident, with OPEN and CLOSE occurring more than 10 times each. These results demonstrate that the proposed method distinguishes sequences based not merely on quantitative differences, but structurally on the presence or absence of continuous learning behaviors involving NEXT and PREV.

5.3 Analysis of High-Performing Students in the Abnormal Group

As shown in the previous sections, the majority of anomalous behaviors consist of immediate exits. However, among the A-grade data classified as anomalous, there are exceptional cases featuring a high number of operations. To elucidate this specificity, we extracted the operational breakdowns of A-grade students who recorded 100 or more operations (Table 4). Table 4 reveals that these are distinct learning behaviors entirely different from simple exits. A specific student (ID: D-2022_U89) stood out with 23 material openings and closings and 303 page transitions, strongly suggesting comparative learning by frequently navigating back and forth between multiple materials. The other three students demonstrated a high frequency of operations related to markers and memos relative to their page transitions, indicating a meticulous study style of remaining on specific pages to organize important sections. While normal data is predominantly characterized by continuous page transitions, these anomalous instances represent behaviors distinct from standard browsing, such as the concurrent study of multiple materials and the extensive marking or note-taking of critical sections. Here, when compared with the baseline detection results, it is evident that the proposed method captures more of these distinctive behaviors of high-performing students. Specifically, while the baseline detected only two of these students (D-2022_U89 and D-2022_U68), the proposed method successfully detected all four, including those utilizing markers and memos (D-2022_U14 and D-2022_U47). This outcome proves that the proposed method can detect not only a simple lack of learning but also the advanced learning behaviors adopted by high-performing students.

Table 4: Operational breakdown of A-grade students classified as anomalous with 100 or more operations.

Student ID	Open/Close	Page Transition	Marker	Bookmark	Memo	Handwritten Memo	Total	Baseline	Proposed
D-2022_U89	23	303	0	0	0	0	326	✓	✓
D-2022_U68	7	46	84	5	23	1	166	✓	✓
D-2022_U14	2	44	62	0	5	0	113	–	✓
D-2022_U47	2	35	70	0	0	0	107	–	✓

6. CONCLUSION

In this study, we proposed and evaluated an unsupervised framework for anomalous behavior detection using context-aware behavioral features and an Isolation Forest. Experimental results demonstrated that anomalous data frequently lacked context due to immediate exits and contained a high percentage of F and D grades, whereas normal data involved continuous page transitions and contained a high percentage of A grades. Detailed analysis suggested that continuous learning actions involving page transitions, the concurrent study of multiple materials, and the marking or note-taking of important sections are advanced behaviors that contribute to academic improvement. Future research will focus on developing more sophisticated anomaly detection methods by incorporating temporal information, such as page stay time, to capture qualitative differences in browsing behavior.

7. ACKNOWLEDGEMENTS

This work was supported by JST CREST Grant Number JPMJCR22D1, Japan.

8. REFERENCES

[1] Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, and J. Heo. Towards an appropriate query, key, and value computation for knowledge tracing. *CoRR*, abs/2002.07033, 2020.

[2] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[3] B. Flanagan and H. Ogata. Learning analytics platform in higher education in japan. *Knowledge Management & E-Learning: An International Journal*, 10(4):469–484, 2018.

[4] L. Jiang and N. Bosch. Mining and assessing anomalies in students' online learning activities with self-supervised machine learning. In *Proceedings of the 15th International Conference on Educational Data Mining (EDM)*, pages 549–554, 2022.

[5] H. Kohama, Y. Ban, T. Hirakawa, T. Yamashita, H. Fujiyoshi, A. Itai, and H. Usami. Recommending learning actions using neural network. In *Proceedings of the 31st International Conference on Computers in Education (ICCE)*, 2023.

[6] F. T. Liu, K. M. Ting, and Z.-H. Zhou. Isolation forest. In *IEEE International Conference on Data Mining (ICDM)*, pages 413–422, 2008.

[7] H. Ogata, C. Yin, M. Oi, F. Okubo, A. Shimada, K. Kojima, and M. Yamada. E-book-based learning analytics in university education. In *Proceedings of the 23rd International Conference on Computers in Education, ICCE 2015*, pages 401–406. Asia-Pacific Society for Computers in Education, 2015.

[8] S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. *CoRR*, abs/1907.06837, 2019.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Vizart: A Multimodal LLM System to Support Visual Learning

Susanna Atanessian
University of California, Berkeley
susanna_a@berkeley.edu

Narges Norouzi
University of California, Berkeley
norouzi@berkeley.edu

ABSTRACT

Creating high-quality educational visualizations remains time-consuming and difficult to personalize at scale. We introduce *Vizart*, a multi-agent framework for end-to-end text-to-visual generation in Scalable Vector Graphics (SVG) that supports visual learning. By treating visualization as a code-generation task in SVG, *Vizart* utilizes the reasoning and coding capabilities of Multimodal Large Language Models (MLLMs) while addressing their known limitations in spatial reasoning and pedagogical alignment. The framework combines retrieval-augmented in-context guidance, staged scenario and code planning, incremental SVG generation, rubric-based iterative refinement across specialized agents, and user-driven refinement of generated visuals. In an initial study of 100 prompts spanning five subject domains — biology, physics, chemistry, mathematics, and computer science — *Vizart* demonstrates promising early results in generating clear, structured, and pedagogically aligned explanatory visualizations. All code and products are open-source and accessible at `https://github.com/susanna-at/vizart`.

Keywords

Educational Visualizations, MLLMs, Agentic AI, SVG

1. INTRODUCTION AND RELATED WORK

Visual learning is an effective pedagogical modality across subject areas and is widely used to support explanations of complex systems and deepen understanding of verbal and written material. Prior work in education and learning sciences has shown that visualizations can be a powerful support for learning, understanding, and long-term recall [1, 11]. Research on intelligent tutoring systems has demonstrated the value of graphic visualizations and representational competency supports for enhancing learning outcomes [7, 10]. However, while students often require visuals to address specific learning needs, designing high-quality educational visualizations can be a significant burden on educators, requiring substantial time and often specialized design skills. These findings motivate the development of systems that target visual learning and can make high-quality educational visualizations more accessible, adaptable, and personalized. The emerging capabilities of Multimodal Large Language Models (MLLMs) create opportunities to generate visualizations at a scale and level of accessibility that were previously difficult to achieve. *Vizart* is inspired by this opportunity. Recent efforts in computing education have explored generative AI for tutoring-style assistance, instructional support, immediate feedback, and coding/debugging support [6]. Systems such as 61A Bot, an LLM-based homework assistant for a large introductory programming course, illustrate this potential at scale [12]. However, these efforts primarily focus on dialogue-based tutoring and code-centered support, leaving generative AI for visual learning comparatively underexplored.

MLLMs have demonstrated strong capabilities in code generation, contextual understanding, dynamic debugging, and iterative optimization [3]. Many visualization formats allow visual generation to be formulated as a code generation task. One particularly useful representation is Scalable Vector Graphics (SVG), an XML-based markup language for creating high-resolution vector graphics. Treating visual generation as code generation enables us to leverage the inherent strengths of MLLMs and is especially promising for scientific and educational applications, which often require high specificity, precision, and clarity. Representing visuals as code, therefore, offers important advantages: it enables finer-grained control over the generation process and allows direct editing and refinement after the image is produced.

Prior work has explored SVG and text-to-SVG generation frameworks. StarVector [9] demonstrates that MLLMs can translate semantic understanding of visuals into SVG primitives. SVGenius, a recent benchmark for SVG understanding, editing, and generation, shows that although current models have promising SVG capabilities, they still face substantial challenges in robust SVG processing, particularly as task complexity increases [2]. Some prior work has studied SVGs as a medium for visualizing hints in mathematics problems [5]; however, frameworks for more involved pedagogical visualization tasks that generalize across subject domains remain underexplored.

Most notably, general-purpose MLLMs are primarily optimized for visual aesthetics rather than instructional clarity. Consequently, existing approaches struggle to handle complex instructional diagrams and often fail to maintain the

Susanna Atanessian, and Narges Norouzi. Vizart: A Multimodal LLM System to Support Visual Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 803–807. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039847

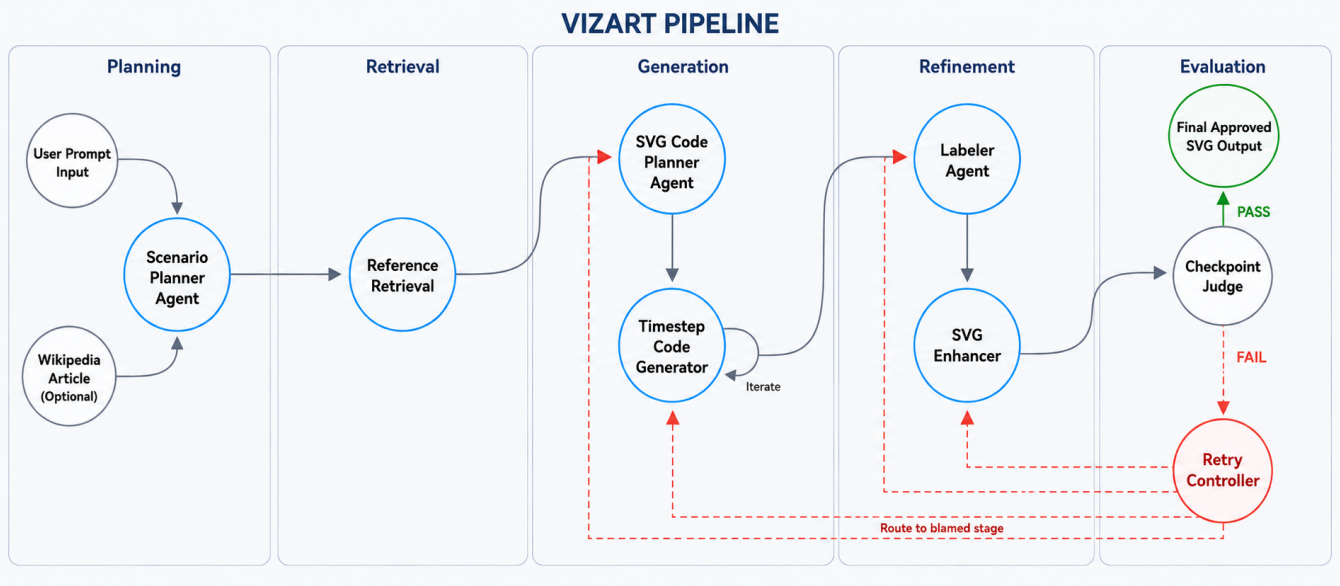


Figure 1: End-to-end overview of the *Vizart* pipeline across its five stages: planning, retrieval, generation, refinement, and evaluation.

precise spatial reasoning, semantic relationships, and geometric constraints required for accurate pedagogical representation [2]. EduVisBench and EduVisAgent further show that, although foundation models can produce step-by-step textual reasoning, they still struggle to generate pedagogically effective visual explanations that are clear, logically structured, and aligned with human cognitive processes [4]. Thus, general-purpose, single-pass generation pipelines fundamentally lack the scaffolding required for education.

To address these limitations, we introduce ***Vizart***, a multi-level human-in-the-loop framework for end-to-end text-to-SVG generation tailored for visual learning. *Vizart* is centered on structured element-level generation and iterative refinement. It aims to leverage the coding abilities of foundation MLLMs while addressing their limitations in spatial reasoning, clarity, and pedagogical alignment.

2. METHODOLOGY AND PIPELINE

Vizart is a multi-agent framework for end-to-end text-to-SVG generation designed to support visual learning. We adopt SVG as the intermediate representation between the language model and the visualization. This choice is motivated by both the broad availability of SVG/XML data on the web and the demonstrated ability of MLLMs to generate and manipulate structured code. As part of *Vizart*, we curate a dataset of more than 5,000 SVG visualizations, equally balanced across five subject areas: *biology, physics, chemistry, mathematics, and computer science*. The dataset is constructed by scraping English-language SVG visualizations from Wikimedia Commons and recording their associated articles as topical context. We applied automatic quality filters such as rendering validation, followed by manual review for subject relevance. We preserve original source URLs and Wikimedia Commons license metadata for attribution. The resulting files contain a median of 86 SVG elements per visualization and are 72 KiB on average. The dataset serves as both an example source and a retrieval base for in-context generation guidance. Given a user request and optional background context, *Vizart* performs a text-based retrieval step over the topic-matched SVG dataset using BM25 [8] and keeps the top-k most relevant examples as references for in-context learning in the downstream generation task of the *Code Planner* stage.

As illustrated in Figure 1, the pipeline begins with a user's natural-language query and optional supporting context, such as a linked article, book section, or other educational resource. In our experiments, we use Wikipedia articles from the dataset metadata as an example of external context. All agents reason over a shared unit of construction called an *element*. An element is defined as a logical structural unit of the figure; it may consist of one or more SVG tags, but it is treated as a single visual unit during planning and generation.

The **Scenario Planner** interprets the request and produces a step-by-step pedagogical visualization plan. It determines how the concept should be visualized, what should be shown, in what order, and at what level of detail for the intended learner.

The **Code Planner** translates this high-level visual plan into an SVG implementation blueprint. It creates a guided plan with actionable coding steps for translating the scenario specification into SVG code. It specifies the approximate layout, visual structure, and the SVG/XML elements needed to realize each part of the figure.

The **Timestep SVG Code Generator** executes the code plan incrementally, stage-by-stage, rather than in a single pass. At each stage, it receives the current SVG state, the instructions for the next step, and the original user intent, and progressively updates the figure. This incremental ap-

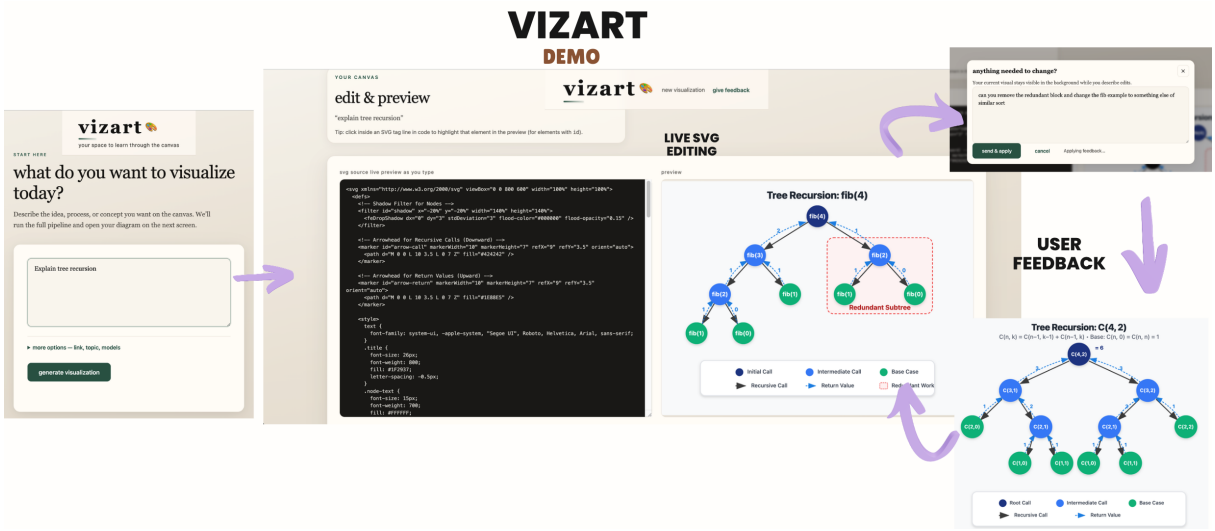


Figure 2: *Vizart* interface showing prompt input, generated SVG rendering, editable code, and controls for iterative refinement.

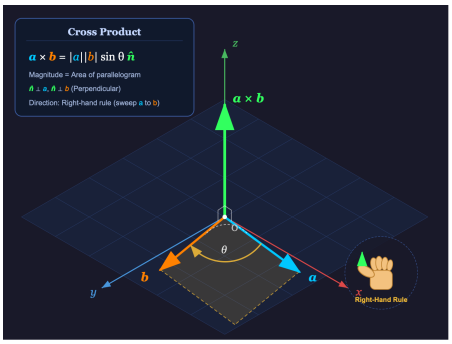


Figure 3: *Vizart* output for mathematics: cross product.

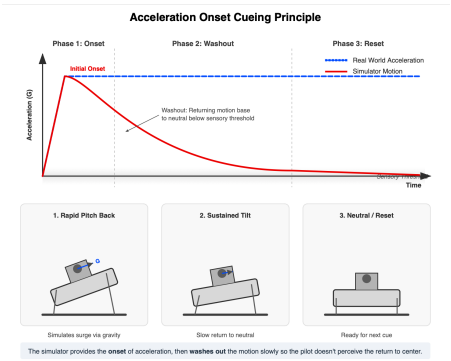


Figure 4: *Vizart* output for physics: onset cueing.

proach mirrors how a human developer would construct a complex diagram — building up the visual element by element, verifying intermediate results, and adjusting as the figure takes shape.

The **Labeler** controls annotations. The *Labeler* determines which labels are necessary for the pedagogical goal, audits existing labels, and issues add, remove, or adjust operations on text tags to improve readability and instructional value.

The **Enhancer** performs final visual refinement. It improves spacing, alignment, and element placement, focusing on presentation quality without altering the intended concept.

The **Judge** evaluates the final visualization using a rubric across four dimensions: visual clarity, layout and visual organization, pedagogical utility, and fidelity to the target concept, averaging to obtain an overall score. Each dimension is scored 1–100. The Judge outputs a binary pass/no-pass decision, concise feedback, and the stage most responsible for the most prominent issue in the case of a no-pass. If figure generation fails, the pipeline can restart from the identified stage with judge feedback injected into downstream agents.

Vizart is designed as a human-in-the-loop system. Because the visualization remains in SVG form throughout generation, intermediate and final outputs are editable. This allows the system not only to regenerate from intermediate stages using judge feedback, but also to support direct refinement and control by instructors or learners. Figure 2 shows the *Vizart* interface. The *Vizart* interactive demo supports all stages of the pipeline with customizable parameters. Users can refine generated visualizations by directly editing the SVG code, manipulating visual elements through the interface, or providing natural-language feedback for regeneration. When feedback is provided, the system can route the revision back through the relevant pipeline stage while preserving the editable SVG representation.

3. RESULTS

In a preliminary study, we evaluated *Vizart* on 100 prompts. Half of the prompts were randomly selected high-school to college-level questions distributed across the five subject categories, while the other half were prompts for which a corresponding ground-truth visualization was available in our Wikipedia-derived SVG dataset. All 100 evaluations were run in fully automated mode (no human intervention) to

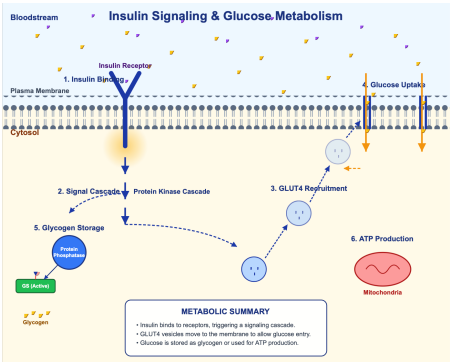


Figure 5: *Vizart* biology output: insulin signaling.

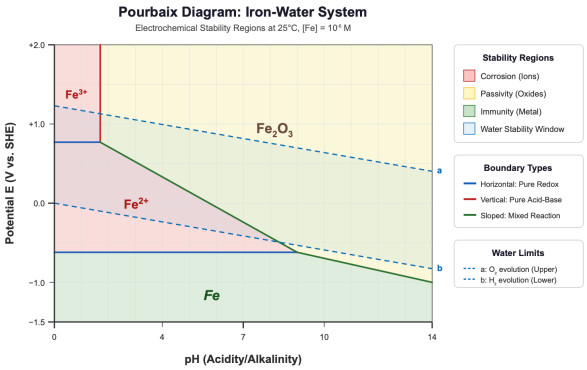


Figure 6: *Vizart* output for chemistry: Pourbaix diagram.

establish a baseline. Judge-based regeneration loops were capped at two iterations. The agent stack used a combination of Gemini-3 Flash Preview and Claude Sonnet 4-6. Gemini was used for planning and judging, while Claude was used for code-generation stages, assigned based on preliminary qualitative comparisons; the architecture is model-agnostic by design. Figures 3–7 show representative *Vizart* outputs across five subject domains.

The average documented judge scores across the four rubric items, alongside human annotator scores on the same rubric, are reported in Table 1. Based on the Judge's binary pass/no-pass decision, the passing rate was 40%. The results in Table 1 show overall alignment between human critique and the LLM as a judge. Across individual criteria, it is notable that the human annotator rated the fidelity to the subject matter and pedagogical value lower than the LLM judge, motivating human-in-the-loop refinement.

4. FUTURE WORK

Vizart creates an exciting foundation for future work in educational visualization. As users interact with the system, provide rubric-based feedback, iterate with the tool, or manually edit generated SVGs, these actions can serve as signals of user preferences and desirable visualization criteria. Such interaction data could support more personalized generation pipelines, including fine-tuning the Judge agent and related components to better match preferences in style, level of detail, and pedagogical presentation. Because different foundation models exhibit different stylistic tendencies, future versions of *Vizart* could also adapt model selection to user preferences and task type. More broadly, as larger amounts of user feedback, interaction traces, and edited outputs are collected, *Vizart* could support stronger human-in-the-loop optimization, including fine-tuning and reinforcement learning from human feedback.

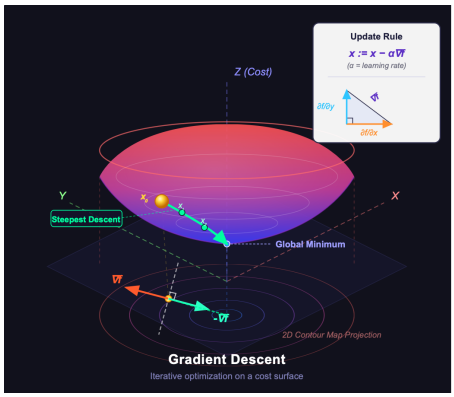


Figure 7: *Vizart* output for computer science: gradient descent.

Table 1: Average LLM judge and human scores across rubric items.

Source	Organization	Clarity	Pedagogy	Fidelity	Overall
LLM Judge	73.3	71.8	81.2	88.1	78.6
Human	68.2	74.9	70.5	75.6	72.3

A second direction is a larger benchmarking and evaluation framework for educational text-to-SVG generation. Such a framework could include broader baseline comparisons, additional metrics for pedagogical quality and visual correctness, and agent-level analysis of where different foundation models succeed or fail. It could also support comparison across domains, levels of contextual input, and degrees of user interaction, helping identify which components of the framework contribute most to final visualization quality.

5. CONCLUSIONS

Generating educational visualizations has long required either design expertise or significant time — often both. *Vizart* offers a different path: by representing visualizations as structured SVG code and orchestrating specialized agents across planning, generation, labeling, refinement, and evaluation, the system enables the production, inspection, and iteration on explanatory figures through natural language alone. Our initial study across five subject domains suggests that this structured, multi-stage approach can meaningfully address the spatial reasoning, label clutter, and pedagogical alignment failures that limit single-pass MLLM generation. The more significant contribution, however, may be architectural. By keeping the visualization in an editable, inspectable format throughout the pipeline, *Vizart* positions instructors and learners as active participants rather than passive recipients of generated content. The human-in-the-loop design is not a fallback for when the model fails — it is central to the system's pedagogical philosophy.

6. REFERENCES

[1] E. Bobek and B. Tversky. Creating visual explanations improves learning. *Cognitive Research: Principles and Implications*, 1(1):27, 2016.

[2] S. Chen, X. Dong, H. Xu, X. Wu, F. Tang, H. Zhang, Y. Yan, L. Wu, W. Zhang, G. Hou, Y. Shen, W. Lu, and Y. Zhuang. Svgenius: Benchmarking llms in svg understanding, editing and generation, 2025.

[3] Y. Dong, X. Jiang, J. Qian, T. Wang, K. Zhang, Z. Jin, and G. Li. A survey on code generation with llm-based agents, 2025.

[4] H. Ji, S. Qiu, S. Xin, S. Han, Z. Chen, D. Zhang, H. Wang, and H. Yao. From eduvisbench to eduvisagent: A benchmark and multi-agent framework for reasoning-driven pedagogical visualization, 2025.

[5] J. Lee, J. Lee, W. Feng, and A. Lan. From text to visuals: Using llms to generate math diagrams with vector graphics, 2025.

[6] J. Prather, J. Leinonen, N. Kiesler, J. Gorson Benario, S. Lau, S. MacNeil, N. Norouzi, S. Opel, V. Pettit, L. Porter, B. N. Reeves, J. Savelka, D. H. Smith, S. Strickroth, and D. Zingaro. Beyond the hype: A comprehensive review of current trends in generative ai research, teaching practices, and tools. In *2024 Working Group Reports on Innovation and Technology in Computer Science Education*, ITiCSE 2024, pages 300–338. Association for Computing Machinery, 2025.

[7] J. Rho, M. A. Rau, and B. D. Van Veen. Preparing future learning with novel visuals by supporting representational competencies. In *Artificial Intelligence in Education: 23rd International Conference, AIED 2022, Proceedings, Part I*, pages 66–77. Springer, 2022.

[8] S. Robertson and H. Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

[9] J. A. Rodriguez, A. Puri, S. Agarwal, I. H. Laradji, P. Rodriguez, S. Rajeswar, D. Vazquez, C. Pal, and M. Pedersoli. Starvector: Generating scalable vector graphics code from images and text, 2025.

[10] S. Schez-Sobrino, C. Gómez-Portes, D. Vallejo, C. Glez-Morcillo, and M. A. Redondo. An intelligent tutoring system to facilitate the learning of programming through the usage of dynamic graphic visualizations. *Applied Sciences*, 10(4):1518, 2020.

[11] J. Schoenherr, A. R. Strohmaier, and S. Schukajlow. Learning with visualizations helps: A meta-analysis of visualization interventions in mathematics education. *Educational Research Review*, 45:100639, 2024.

[12] J. D. Zamfirescu-Pereira, L. Qi, B. Hartmann, J. DeNero, and N. Norouzi. 61a bot report: Ai assistants in cs1 save students homework time and reduce demands on staff. (now what?). In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*, SIGCSE TS 2025, pages 1309–1315. Association for Computing Machinery, 2025.

When the Finish Line Moves: Immediate Corrective Practice and Implications for Design in Digital Math Learning *

Wen-Chiang Lim
Worcester Polytechnic Institute
wlim@wpi.edu

Yanping Pei
Worcester Polytechnic Institute
ypei@wpi.edu

Adam Sales
Worcester Polytechnic Institute
asales@wpi.edu

Neil Heffernan
Worcester Polytechnic Institute
nth@wpi.edu

ABSTRACT

Digital learning platforms often give students a chance to solve a similar problem immediately after an incorrect response, a design we refer to here as "*immediate corrective practice*" (ICP). While prior work suggests that immediate feedback and targeted retries can support learning, less is known about how various implementations of ICP affects performance and engagement in real-world settings.

We study ICP using extant large-scale log data from a K–12 mathematics platform where teachers can choose whether to enable a feature that presents supplementary similar problems after errors on eligible items within an assignment. Focusing on students who answered the first eligible problem in an assignment incorrectly, we compare those shown ICP with those assigned the same problem sets but not shown ICP because their teachers did not use the feature. Students who encountered ICP performed better on the next problem, but were also less likely to complete the assignment.

One possible explanation is that additional practice after an error shifts students' perceived endpoint of the assignment, creating a moving-goalpost effect that discourages continued engagement. As large language models make it easier to generate corrective problems and feedback at scale, our findings offer design insights for supporting learning while minimizing disengagement.

Keywords

immediate feedback, immediate corrective practice, observational data

*(Does NOT produce the permission block, copyright information nor page numbering). For use with edm_article.cls.

Wen Chiang Lim, Yanping Pei, Adam Sales, and Neil Heffernan. When the Finish Line Moves: Immediate Corrective Practice and Implications for Design in Digital Math Learning. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 808–812. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039687

1. INTRODUCTION

Digital learning platforms in K–12 mathematics often provide students with immediate feedback on auto-gradable, non-open-response problems. This can take the form of correctness feedback, hints, or a worked explanation. To reinforce understanding, students may also be given opportunities for "*immediate corrective practice*" (ICP), such as solving similar problems they had difficulty with, either immediately after the current problem or after completing the assignment. Examples include "Start Over" feature in Khan Academy [9], "Redo" in ASSISTments [15], and "Quick Retake" in ALEKS [1]. While prior work suggests that immediate feedback and targeted retries can support learning [14, 4], less is known about how different implementations affect student learning and engagement in real-world settings.

This paper studies one particular implementation of ICP in a real-world DLP context. Specifically, we examine the effects of providing students with a similar supplementary problem immediately after an incorrect response within an ongoing assignment. We focus on a naturally occurring setting in a K–12 mathematics platform in which teachers can enable or disable an ICP feature that presents students with a similar problem immediately after they answer a problem incorrectly on the first attempt, provided a similar problem is available. We investigate the following research questions:

- **RQ1**: Does immediate corrective practice improve student performance on the immediate next problem?
- **RQ2**: Does immediate corrective practice affect student completion of the assignment?
- **RQ3**: If the evidence suggests "Yes" to RQ2, is the decrease in persistence purely due to longer assignment length?

2. RELATED WORK

2.1 Benefits of Immediate Feedback and ICP

Immediate corrective practice is closely related to research on feedback and mastery learning. Mastery learning frameworks emphasize iterative cycles of assessment, feedback, and corrective opportunities, where students are given additional chances to practice until proficiency is reached [3,

5]. Similarly, digital learning platforms often provide immediate feedback and targeted retry opportunities after errors, and such approaches have been associated with improved learning, especially for weaker students [2, 11, 14].

The instructional logic is straightforward: when a student makes an error, the platform uses that moment to provide a closely related opportunity to apply what was learned from the immediate feedback. This is especially plausible for procedural mathematics, which benefits from repeated practice.

2.2 Potential Tradeoffs for ICP

Implementations of immediate corrective practice vary across platforms. Students may practice a fixed skill until they reach mastery, practice on similar problems after an assignment, or receive supplementary similar problems woven directly into the assignment they are working on.

Although adding similar problems after an error is consistent with mastery learning, it may not always be beneficial from the learner's perspective. Supplementary ICP within an assignment can provide a timely opportunity to revisit a skill, but it also increases the amount of work. Prior work has shown that assignment size matters for student performance, with evidence that shorter assignments can be associated with better performance than longer ones [12]. This raises the possibility of assignment fatigue, which may reduce students' willingness to complete the assignment.

In addition, research on goal-distance theory [10, 7] suggests that motivation depends on perceived distance to completion. When ICP problems are added after an error, students may perceive the assignment's endpoint as shifting after they have already started the assignment. Thus, while ICP is aligned with mastery learning, its implementation within an assignment may also lower persistence by making the finish line feel farther away than expected.

2.3 Implications for scaling of ICP

Prior to the advent of large language models (LLMs), creating similar problems often required substantial human labor, alongside work on crowdsourcing and learner-sourcing to produce high-quality educational content. Recent research suggests that LLMs can now generate and scale such problems with much less effort [6, 13]. As a result, it is increasingly important to move beyond content quality alone and examine existing implementations to identify better designs for future randomized controlled trials and field testing.

3. DATA AND SETTING

This study used extant de-identified log data collected by a K–12 mathematics digital learning platform, ASSISTments, between Jan 2023 and Dec 2025 [8], with the data collection process approved by IRB. The platform has an ICP feature that, when enabled for an assignment, provides students with a similar problem after an incorrect response on eligible items. We focus on data with these criteria:

- a student attempts the first item in an assignment for which a corresponding similar corrective practice problem is available, even though this is not necessarily the first item in the assignment,
- the student answers this item incorrectly on the first attempt, and
- there is a later non-open-response autogradable item.

These instances represent cases in which immediate corrective practice could potentially be triggered while still allowing measurable downstream outcomes without contamination from earlier ICP exposure. Our two analytic samples consist of student-problem observations meeting these criteria: a *first-instance* dataset with students' first exposure to the feature, and a *full* dataset with all such exposures. We compare outcomes between students exposed to immediate corrective practice and students assigned the same problem sets but not exposed because their teachers did not enable the feature. Assignments with no later auto-gradable item after the ICP-eligible item are excluded.

Finally, to examine whether lower assignment completion is simply due to longer assignments under ICP, we analyzed an *expanded* dataset containing all student assignment attempts between Jan 2023 and December 2025. Using this dataset, we estimated the association between teacher-enabled ICP and assignment completion while controlling for the maximum possible assignment length if ICP were triggered. We use the maximum possible length because ICP is triggered only after incorrect responses, and not all students will answer all ICP-eligible problems incorrectly.

4. MEASURES

Outcome variables. We consider the following outcomes on the immediate next autogradable problem, using the average when the problem has more than one part, as well as completion of the rest of the assignment.

Average next-problem score (RQ 1,2 only): We analyzed these scores in two ways: first, among only students who completed the next problem, and second, with students who skipped the next problem coded as 0. Scoring reflected both accuracy and support use: first-attempt correct responses earned 100%, correct responses after hints or within the second or third attempts earned partial credit (33–66%), and responses requiring more than three attempts or revealed-answer responses earned 0%.

Assignment completion (RQ 1,2,3): Whether the student goes on to complete the assignment, i.e., a binary variable.

Covariates. To control for students' general ability and current performance in the relevant skill, we include the following variables centered on the median:

Prior 5-problem average performance (RQ 1,2,3): student's average performance on the previous five problems before the assignment, used as a proxy for general ability

Initial-item score (RQ 1,2 only): score on the initial item that was answered incorrectly on first attempt, or average score if the problem had more than one part.

Assignment length (RQ3 only): Natural log of the base number of problems if ICP is not enabled, and theoretical maximum number of problems, including ICP, if it is enabled.

4.1 Analytic Approach

We estimated the association between immediate corrective practice and subsequent student outcomes using multilevel regression models fit separately in RQ1 and RQ2 on two datasets: a *first-instance* dataset and a *full* dataset. This allowed us to examine whether results were consistent when focusing on a student's first eligible exposure versus all eligible exposures. Our main predictor for RQ1 and RQ2 was `ICP_seen`, an indicator of whether a student received an ICP problem after an incorrect response.

In RQ3, the main predictor was `use_ICP`, an indicator of whether the teacher enabled the ICP feature for the assignment, and analyzed using the *expanded* dataset. The models include random intercepts for students and teachers, as well as for problems in RQ1 and RQ2 and problem sets in RQ3, to acccount for differences across these units that are not captured by the observed covariates.

4.2 Models for Continuous Outcomes

For next-problem outcomes, we fit linear mixed-effects models using `lmer` from `R`'s `lme4` package with maximum likelihood (`REML = FALSE`).

$$\begin{aligned} Y_{ijkm} = \beta_0 &+ \beta_1 \texttt{ICP_seen}_{ijkm} \\ &+ \beta_2 \texttt{prior5avg}_{ijkm} \\ &+ \beta_3 \texttt{prior5avg}_{ijkm} * \texttt{ICP_seen}_{ijkm} \\ &+ \beta_4 \texttt{initial_score}_{ijkm} \\ &+ u_i + v_j + w_k + \varepsilon_{ijkm} \end{aligned}$$

where Y_{ijkm} denotes the continuous outcome for observation m. Here, u_i, v_j, and w_k denote random intercepts for student, teacher, and next problem respectively, and assumed to be normally distributed with mean 0. Residual error ε_{ijkm} is also assumed to be normally distributed with mean 0.

We applied this specification to the two continuous outcomes: next-problem score average, and its corresponding variant in which students who did not complete the next problem were scored as zero. This allowed us to test whether the observed associations were robust to a more conservative coding of non-completion.

4.3 Models for Binary Assignment Completion

For the binary outcome indicating whether the student completed the remainder of the assignment in RQ1 and RQ2, we fit a generalized linear mixed-effects model with a logit link using `glmer` from `lme4` in `R`. As in the linear models, we included random intercepts for student, teacher, and next problem, with the same distributional assumptions.

$$\begin{aligned} \text{logit}\left(Pr(Y_{ijkm} = 1)\right) = \beta_0 &+ \beta_1 \texttt{ICP_seen}_{ijkm} \\ &+ \beta_2 \texttt{prior5avg}_{ijkm} \\ &+ \beta_3 \texttt{prior5avg}_{ijkm} * \texttt{ICP_seen}_{ijkm} \\ &+ \beta_4 \texttt{initial_score}_{ijkm} \\ &+ u_i + v_j + w_k \end{aligned}$$

where $Y_{ijkm} = 1$ indicates assignment completion. For interpretation, fixed effects from the logistic models were exponentiated and reported as odds ratios. In RQ3, the model is similar, except that we replace `ICP_seen` and `initial_score` with `use_ICP` and `assignment_length` respectively. Due to the size of the *expanded* dataset ($>$ 4 million records), we used `glmmTMB` in `R` which is more efficient for large datasets.

4.4 Bootstrap Inference

Given the complexity of the mixed-effects structure, we used parametric bootstrap resampling with `bootMer` to obtain empirical standard errors for the fixed effects in the models for RQ1 and RQ2. For each fitted model, we generated 500 bootstrap replications by simulating new response values from the fitted model and refitting the model to each simulated dataset, using `type = "parametric"` and `use.u = FALSE`. Bootstrap standard errors were then calculated as the standard deviation of the bootstrapped fixed-effect estimates across replications. However, we did not use bootstrapping for RQ3 because of the dataset size and computational cost of repeatedly refitting the model. Given the much larger sample size in RQ3, we do not expect the substantive conclusions to be sensitive to this choice.

5. RESULTS

Our results suggest a mixed pattern. On one hand, immediate corrective practice appears to provide short-term benefits for students who engage with it. For RQ1, students exposed to ICP show better subsequent performance, consistent with the idea that immediate targeted reinforcement can support learning after an error. This pattern holds even after accounting for students who skipped the next problem.

On the other hand, these benefits come with important engagement trade-offs. For RQ2, exposure to ICP is associated with lower assignment completion. In other words, while ICP may support learning after an error, the additional problems may also discourage students from finishing the assignment.

One way to interpret these results is to distinguish between the *instructional value* of the feature and its *psychological experience*. The same intervention that helps one student consolidate learning may be experienced by another as an unexpected increase in workload. In RQ3, we find that even after controlling for assignment length, ICP remains associated with lower odds of assignment completion. This suggests that lower completion is not due to assignment length alone, and that a perceived shift in the assignment endpoint, consistent with goal-distance theory, may also play a role. *We elaborate on the results in greater detail below.*

Across both the first-instance and full datasets, results were highly consistent. Students who saw the similar next problem (ICP) performed better on the immediate next problem, and this pattern held even when non-responses were recoded as 0 (Table 1). The ICP $\times$ Prior5avg interaction was negative across specifications, suggesting that the performance benefit of ICP was smaller for students with higher prior performance.

At the same time, students who saw ICP had lower odds of completing the assignment in both the first-instance and full

Table 1: Next Problem Score (Inclusive of non-responses coded as 0) Mixed Effects Linear Regression Results

	First-Instance	Full
(Intercept)	0.467***	0.470***
	[0.452, 0.481]	[0.458, 0.481]
ICP seen	0.036***	0.029***
	[0.023, 0.050]	[0.021, 0.036]
Prior5avg	0.248***	0.241***
	[0.237, 0.258]	[0.233, 0.248]
Initial score	0.295***	0.284***
	[0.285, 0.304]	[0.277, 0.291]
ICP seen × Prior5avg	-0.032*	-0.040***
	[-0.062, -0.001]	[-0.056, -0.023]
Total Observations	92,875	161,625
ICP observations	8,974	29,856
Unique Students	39,972	50,482
Unique Teachers	1,481	1,605
Unique Next Problems	858	1,279

Note. Entries are coefficient estimates with statistical significance indicated: $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$., with 95% confidence intervals in brackets below. The interaction term shows how the association between prior 5-problem average and next-problem score differs when ICP was seen.

Table 2: Student Completed Assignment Mixed Effects Logistic Regression Results

	First-Instance	Full
(Intercept)	6.94***	6.89***
	[6.05, 7.97]	[6.11, 7.77]
ICP seen	0.82**	0.84***
	[0.72, 0.94]	[0.79, 0.90]
Prior5avg	0.93	0.89**
	[0.86, 1.01]	[0.83, 0.95]
Initial score	1.21***	1.22***
	[1.11, 1.31]	[1.15, 1.30]
ICP seen × Prior5avg	1.42**	1.16*
	[1.11, 1.81]	[1.01, 1.34]

Note. Entries are odds ratios with statistical significance indicated: $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$, with 95% confidence intervals in brackets below. Odds ratios above 1 indicate higher odds of assignment completion, while odds ratios below 1 indicate lower odds.

datasets (Table 2). This negative association was somewhat weaker for students with higher prior performance.

In the analysis using the expanded dataset, ICP remained associated with lower completion even after controlling for median assignment length (Table 3) and student performance. The interaction results also point in the similar directions across specifications: the negative association between ICP and assignment completion appears weaker for students with stronger prior performance and stronger for lower-performing students. However, this moderation was estimated more precisely when prior performance was modeled continuously than when it was coded as a lowest-quartile indicator. Given that lower completion is not explained by assignment length alone, it lends credibility that moving the goalpost has led to decreased persistence.

Table 3: [Expanded Dataset] Assignment Completion Mixed Effects Logistic Regression Results

	Prior5Avg	LowPrior5Avg
(Intercept)	6.49***	6.50***
	[6.13, 6.87]	[6.13, 6.88]
Use ICP	0.88***	0.88***
	[0.86, 0.90]	[0.86, 0.90]
Asg length	0.41***	0.41***
	[0.40, 0.41]	[0.40, 0.41]
Prior5avg	1.10***	
	[1.08, 1.12]	
Low_Prior5avg		0.96***
		[0.95, 0.97]
Use ICP × Asg length	1.03***	1.03***
	[1.02, 1.05]	[1.02, 1.05]
Use ICP × Prior5avg	1.04*	
	[1.01, 1.08]	
Use ICP × Low_Prior5avg		0.98
		[0.96, 1.00]
Total Observations		4,154,258
Unique Students		204,175
Unique Teachers		3,586
Unique Problems Sets		85,329

Note. Entries are odds ratios with statistical significance indicated: $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$, with 95% confidence intervals in brackets below. Odds ratios above 1 indicate higher odds of assignment completion, while odds ratios below 1 indicate lower odds.

6. DISCUSSION, LIMITATIONS, AND DESIGN IMPLICATIONS

This study is limited by its observational design, so the results should be interpreted as associations rather than causal effects. Because the sample was restricted to students who reached the first eligible problem, we cannot rule out selection or survival bias. Our outcomes also focus on short-term performance and assignment completion, rather than longer-term learning, and the findings are specific to the platform studied.

Within these limits, the results suggest a consistent tradeoff: immediate corrective practice within an assignment is associated with slightly better next-problem performance, but also with lower assignment completion. This suggests that the value of ICP depends not only on the quality of the added problems, but also on how they are implemented. For platform developers, the key question is not simply whether to provide similar problems after errors, but how to design their use in ways that support learning without reducing persistence. This also points to promising directions for future randomized experiments, such as testing alternative implementations that vary how workload changes during an assignment, including reverse moving-goalpost designs that remove problems when students answer correctly. As large language models make it easier to generate similar problems at scale, the main challenge may shift from producing such content to designing better ways to use it.

7. REFERENCES

[1] ALEKS Corporation. Assignment suite update, 2013. https://www.aleks.com/highered/math/Assignment_

Suite_Update.pdf. Accessed: April 6, 2026.
[2] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive Tutors: Lessons Learned. *Journal of the Learning Sciences*, 4(2):167–207, Apr. 1995.
[3] B. S. Bloom. Learning for mastery. *Evaluation Comment*, 1(2):1–12, 1968.
[4] G. Campitelli and F. Gobet. Deliberate Practice: Necessary But Not Sufficient. *Current Directions in Psychological Science*, 20(5):280–285, Oct. 2011.
[5] T. R. Guskey and S. L. Gates. Synthesis of research on the effects of mastery learning in elementary and secondary classrooms. *Educational Leadership*, 43(8):73–80, 1986.
[6] J. He-Yueya, N. D. Goodman, and E. Brunskill. Evaluating and optimizing educational content with large language model judgments. In B. PaaÃŸen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 68–82, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society.
[7] C. Heath, R. P. Larrick, and G. Wu. Goals as reference points. *Cognitive Psychology*, 38(1):79–109, 1999.
[8] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, Dec. 2014.
[9] Khan Academy. Can a student restart an exercise, quiz, or unit test? https://support.khanacademy.org/hc/en-us/articles/35941807811213-Can-a-student-restart-an-exercise-quiz-or-unit-test. Accessed: March 31, 2026.
[10] R. Kivetz, O. Urminsky, and Y. Zheng. The Goal-Gradient Hypothesis Resurrected: Purchase Acceleration, Illusionary Goal Progress, and Customer Retention. *Journal of Marketing Research*, 43(1):39–58, Feb. 2006.
[11] K. R. Koedinger, J. R. Anderson, W. H. Hadley, and M. A. Mark. Intelligent Tutoring Goes To School in the Big City. *International Journal of Artificial Intelligence in Education*, 8:30–43, 1997. Artwork Size: 165322 Bytes.
[12] S. Mojarad. Studying assignment size and student performance using propensity score matching. In *Proceedings of the 9th International Conference on Educational Data Mining (EDM)*, pages 701–702, 2016.
[13] K. A. Norberg, H. Almoubayyed, L. De Ley, A. Murphy, K. Weldon, and S. Ritter. Rewriting Content with GPT-4 to Support Emerging Readers in Adaptive Mathematics Software. *International Journal of Artificial Intelligence in Education*, 35(2):587–626, June 2025.
[14] R. Razzaq, K. S. Ostrow, and N. T. Heffernan. Effect of Immediate Feedback on Math Achievement at the High School Level. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, and E. Millán, editors, *Artificial Intelligence in Education*, volume 12164, pages 263–267. Springer International Publishing, Cham, 2020. Series Title: Lecture Notes in Computer Science.
[15] The ASSISTments Foundation. Redo feature, 2025. https://www.assistments.org/academy-videos/redo-feature. Accessed: March 31, 2026.

8. ACKNOWLEDGMENTS

Neil and Cristina Heffernan thank their many past and current funders including NSF (2118725, 2118904, 1950683, 1917808, 1931523, 1940236, 1917713, 1903304, 1822830, 175 9229, 1724889, 1636782, 1535428, 2215842, 2225091, 2341948, & 2153481); IES (R305N210049, R305D210031, R305A17013 7, R305A170243, R305A180401, R305A12 0125, R305R22001 2, R305T240029, R305J250058); DOE (U411B190024, S411A 240012, S411A240012); GAANN (P200A120238, P200A1800 88), Schmidt Futures, BMGF, CZI, Arnold, Hewlett, the Jaffe Foundation, and anonymous donors. None of the opinions expressed here are those of the funders.

Who Benefits from Looking-for-Team Forums? Examining Expert-Novice Differences in Online Team Formation

Ran Bi
SAS Institute
ran.bi@sas.com

Shiyao Wei
Florida State University
sw22b@fsu.edu

ABSTRACT

Crowdsourcing platforms like Kaggle provide community-driven mechanisms for team formation and informal learning, yet little is known about how these mechanisms operate across expertise levels. We analyze 1,577 comments from 671 participants across 18 Kaggle "Looking for a Team" megathreads (2023) and identify a systematic attention imbalance by expertise level. Low-tier users (Novice and Contributor) constitute 89.3% of participants, yet higher-tier users attract disproportionate attention in the reply network ($H = 16.46$, $p = .003$). Dyadic regression further confirms that receiver expertise predicts tie formation (OR = 1.28), consistent with preferential attachment. Using linked final team records for 16 competitions, however, we do not observe a statistically significant expertise gap in ending on a multi-member team (41.0% for low-tier vs. 34.7% for high-tier user-competition observations, $\chi^2 = 1.26$, $p = .261$). These results show that megathreads reproduce expertise-based differences in visibility more clearly than in observed final team membership. We identify preferential attachment as a key mechanism underlying this visibility imbalance and translate our findings into design recommendations for more equitable team formation systems.

Keywords

Team Formation, Expert-Novice Differences, Crowdsourcing, Preferential Attachment, Informal Learning

1. INTRODUCTION

Crowdsourcing platforms such as Kaggle have evolved beyond task completion into spaces for professional development and informal learning [2, 12]. As crowdsourcing tasks grow in complexity, collaboration has become essential, encouraging knowledge workers to form teams to tackle data science challenges [6]. The first step of collaboration is team formation, a process that unfolds differently in virtual communities than in traditional organizations because individuals have the autonomy to choose their own collaborators [5].

Ran Bi, and Shiyao Wei. Who Benefits from Looking-for-Team Forums? Examining Expert-Novice Differences in Online Team Formation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 813–817. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040055

To facilitate this process, platforms provide spaces for team-seeking behavior. Kaggle's "Looking for a Team" (L4T) megathreads, introduced in 2022, are competition-specific forums where participants post team-seeking messages and reply to potential collaborators [18]. While prior work has examined the general dynamics of megathread conversations and self-representation patterns [18], a critical question remains: who actually benefits from these spaces?

This question has equity implications. If these spaces disproportionately serve users who already have established networks and credentials, they may widen the gap between experts and newcomers rather than close it [7]. We address three research questions:

- **RQ1.** How does expertise level relate to participation patterns and network position?
- **RQ2.** What network mechanisms (preferential attachment and homophily) drive tie formation?
- **RQ3.** Do expert and novice users benefit equally from megathread participation?

Drawing on preferential attachment [1] and homophily [11], we hypothesize that higher-tier users attract a disproportionate share of replies in team-seeking forums and that this visibility advantage may carry over into observed team membership outcomes.

2. RELATED WORK

Group Formation in Online Communities. Research on group formation distinguishes automatic from agency-driven approaches across different contexts [5]. In MOOCs, automatic grouping by activity or demographics can improve satisfaction [14, 19], but such automation potentially removes individual agency. In open communities like Kaggle, human agency dominates. Participants actively seek information about potential collaborators before forming teams [10]. Language proficiency and past collaborations predict team-up decisions positively on Kaggle, while geographical distance and skill disparity show negative effects [3]. However, most studies assume pre-formed teams rather than examining how teams naturally emerge in these communities [6].

Expert-Novice Differences in Crowdsourcing Communities. Crowdsourcing platforms such as Kaggle make expertise visible through status hierarchies: the performance tier system

publicly labels users from Novice, Contributor, Expert, Master, to Grandmaster, and these labels travel with users into team-seeking interactions. Prior work has shown that newcomers in open-source communities face barriers to gaining trust and attracting collaboration [10], and that sharing online profiles and expertise signals predicts collaboration success [13]. What remains underexplored is whether team formation spaces specifically designed to help all users actually achieve equitable outcomes across expertise levels.

Preferential Attachment, Homophily, and Marginality. Preferential attachment, the tendency for well-connected nodes to attract yet more connections [1], and homophily, the tendency for similar individuals to associate [11], are two fundamental mechanisms in social network formation. Wang [17] has argued that marginality, defined as being located at the periphery of a network, might form a team easier than more experienced users in crowdsourcing contexts. Twyman et al. [15] found that network positioning shapes Kaggle competition performance, while Li et al. [9] showed social influence effects in this community. We extend this work by examining how these mechanisms manifest in team-seeking online forums and whether they produce inequitable outcomes for novices.

3. METHODS

3.1 Data Collection

We collected data from Kaggle's official L4T megathreads for all competitions held in 2023 using the Kaggle API. Of 31 competitions identified, 18 featured L4T megathreads created by Kaggle staff. Our dataset comprises two components: (1) megathread comments, totaling 1,577 from 676 unique users across the 18 competitions, and (2) user profiles for 671 users matched to Kaggle's performance tier system (99.3% match rate, five participants do not have a profile).

Kaggle's performance tier system assigns users a tier from 0 (Novice, no activity) to 4 (Grandmaster), with tier 5 reserved for Kaggle Staff. Tiers are awarded based on cumulative competition medals: Novice (no medals), Contributor (profile complete), Expert ($\geq$2 bronze medals), Master ($\geq$1 gold + 3 silver), and Grandmaster ($\geq$5 gold including a solo gold). We operationalize expertise using this tier and create a binary split: Low-Tier (Novice and Contributor, tiers 0–1, indicating no or minimal verified competition achievement) and High-Tier (Expert through Grandmaster, tiers 2–4, indicating at least two verified competition medals).

3.2 Data Analysis

Our analysis proceeds in five stages. First, we constructed a directed reply network from the megathread data using NetworkX [4]. Each node represents a user, and a directed edge from A to B indicates that user A replied to a comment posted by user B. Edge weights represent the number of replies between a pair. The resulting network contains 665 nodes and 1,300 directed edges (density = 0.003). Figure 1 visualizes the reply network, with node color indicating Kaggle performance tier and node size reflecting weighted in-degree (replies received).

Then, we characterized the expertise composition of megathread participants and their comment activity patterns through a tier distribution analysis. Third, we computed in-degree (replies received), out-degree (replies sent), and betweenness centrality for each node, then compare distributions across tiers using Kruskal-Wallis tests. Then, we tested for preferential attachment by comparing the proportion of replies received by each tier against their population share with a chi-squared goodness-of-fit test, and for homophily by computing the E-I index [8], defined as the ratio of cross-tier to within-tier edges, with significance assessed via a permutation test (1,000 permutations). Finally, we fit a dyadic logistic regression that models tie formation as a function of sender tier, receiver tier, tier difference, same-tier indicator, and account tenure, sampling negative dyads at a 3:1 ratio and evaluating with five-fold cross-validation.

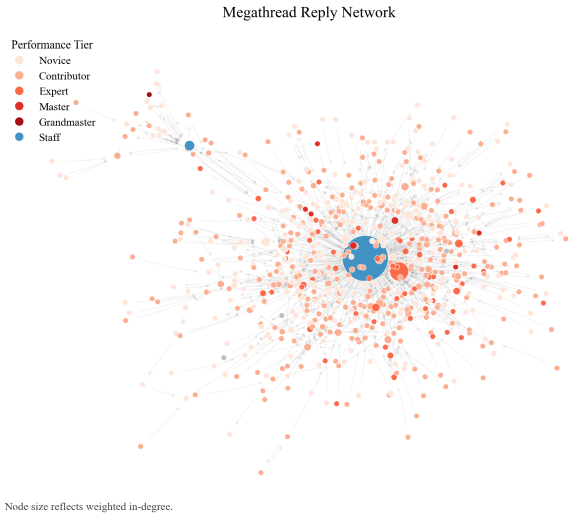


Figure 1: Reply network constructed from Kaggle L4T megathreads. Node color denotes performance tier, and node size reflects weighted in-degree.

3.3 Team-up Success Operationalization

We use a linked competition-team file that connects megathread participants to final team records for 16 competitions. Success is defined as ending on a multi-member team in a given competition (i.e., `NumbersPlayer` > 1). This linked subset contains 382 user-competition observations from 219 unique users, of which 377 observations have matched tier labels. This measure captures observed final team membership directly, in contrast to indirect signals such as medal counts or reciprocal replies.

4. RESULTS

4.1 RQ1: Expertise and Participation

The megathread participant pool is heavily skewed toward lower tiers. As shown in Figure 2, Novice users (tier 0) comprise 42.2% of participants and Contributors (tier 1) comprise 47.1%, yielding a combined low-tier share of 89.3%. Expert users (tier 2) account for only 9.2%, with Master (1.0%) and Grandmaster (0.1%) users nearly absent.

Despite their minority presence, higher-tier users occupy more central network positions. Expert users receive significantly more replies (mean in-degree = 3.32) compared

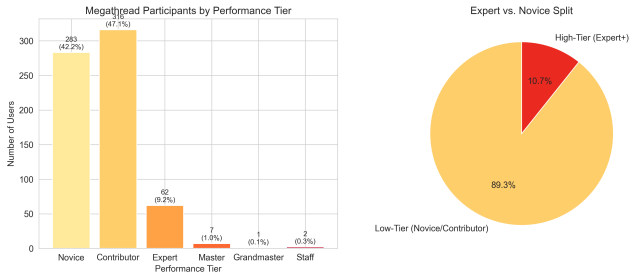


Figure 2: Megathread participants by performance tier (left) and binary expertise split (right). Low-tier users dominate at 89.3%.

Table 1: Population share vs. reply-target share by tier, demonstrating preferential attachment toward Expert users.

Tier	Pop. %	Reply %	Ratio
Novice	42.2	26.2	0.62
Contributor	47.1	29.9	0.63
Expert	9.2	13.9	1.51
Master	1.0	1.1	1.10

to Novice (1.39) and Contributor (1.43) users, with a statistically significant difference across tiers (Kruskal-Wallis $H = 16.46$, $p = .003$). Betweenness centrality also differs significantly ($H = 16.09$, $p = .003$), with Expert users serving as more central bridges in the network. However, outdegree does not differ significantly ($H = 3.82$, $p = .431$), indicating that all tiers send a similar number of replies but receive attention unevenly.

4.2 RQ2: Attachment and Homophily

Preferential attachment. Table 1 compares population share with reply-target share. Excluding Staff (who post the initial megathread announcements), Expert users receive 1.51× their population share in replies, while both Novice (0.62×) and Contributor (0.63×) users receive less than their expected share. A chi-squared test confirms this disproportionality is significant ($p < .001$), supporting preferential attachment toward higher-tier users.

Homophily. The E-I index of 0.125 shows slightly more interaction across tiers than within them, consistent with novices reaching out to experienced users. The permutation test yields $p = .062$, which does not reach conventional significance ($\alpha = .05$), which should be treated as suggestive rather than conclusive. The trend points toward upward-directed attention, but a larger sample is needed before claiming a confirmed homophily effect. Figure 3 shows the reply mixing matrix; low-tier users direct a substantial share of replies toward Staff-initiated threads and cross-tier targets.

Dyadic regression. The logistic regression model achieves a five-fold cross-validated AUC of 0.558. The receiver's tier has a positive coefficient (OR = 1.28), confirming that higher-tier users attract more connections. Tier difference has a negative coefficient (OR = 0.83), indicating that larger expertise gaps slightly reduce tie formation probability. The

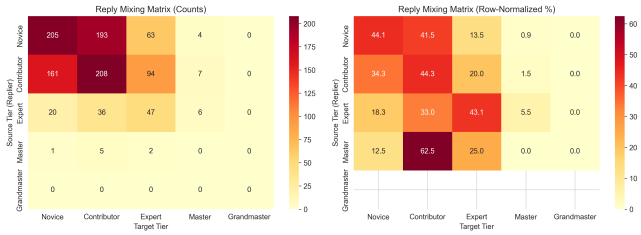


Figure 3: Reply mixing matrix showing interaction patterns between tiers. Left: raw counts. Right: row-normalized percentages.

overall model performance is modest, which is expected for sparse social networks, but the direction and significance of the coefficients support both preferential attachment and weak homophily effects.

4.3 RQ3: Differential Success Rates

Using the linked team-membership data, we do not find a statistically significant expertise gap in final teaming outcomes. As shown in Figure 4, low-tier user-competition observations end on multi-member teams in 41.0% of cases, compared with 34.7% for high-tier observations; this difference is not statistically significant ($\chi^2 = 1.26$, $p = .261$).

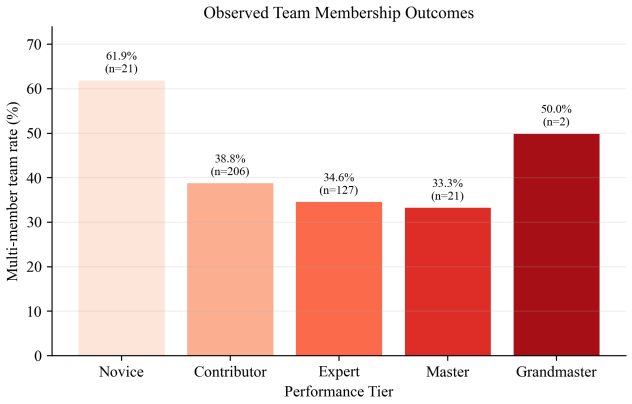


Figure 4: Observed multi-member team rates by performance tier in the linked 16-competition subset. Aggregated to binary groups, low-tier observations reach 41.0% and high-tier observations 34.7%, with no significant difference ($\chi^2 = 1.26$, $p = .261$).

5. DISCUSSION

5.1 Attention Inequality and Team Outcomes

Our findings reveal a mixed pattern. Megathreads are populated mainly by low-tier users, but higher-tier users receive more attention once interaction begins. While higher-tier users receive more attention, they do not necessarily convert that attention into more frequent teaming outcomes, as the multi-member team rates do not differ significantly by tier. Preferential attachment seems only to reproduce inequality in visibility more clearly than in observed final teaming outcomes. Therefore, the megathread mechanism appears to amplify visibility disparities without necessarily

translating them into a significant teaming gap, suggesting that the relationship between visibility and teaming is complex and mediated by other factors [10].

5.2 Why Visibility Does Not Map Onto Teaming

Several mechanisms can explain this misalignment. First, low-tier users overwhelmingly outnumber high-tier users (89.3% vs. 10.7%), so even when high-tier users attract proportionally more replies, the absolute volume of low-tier participants leaves opportunity for novices to find one another and form teams among themselves. Second, attention is not the same as acceptance. High-tier users who received many replies may decline most invitations, screen for quality, or already have established collaborators outside the megathread, so the replies they receive do not necessarily convert into joint team registrations [15]. Third, megathreads are only one of several channels through which Kaggle teams form. Teammates can be recruited via private messages, off-platform networks, or prior competitions, and these channels likely distribute more evenly across tiers than the public reply network does. These dynamics suggest that the visibility imbalance is real and consequential for whose contributions get seen, but it is buffered by the over-representation of novices and by team formation pathways that operate outside the channel.

5.3 Implications

While Wang [17] has argued that marginality is an important indicator of whether a user can form a group in crowdsourcing platform, our finding reveal a challege for novice participants in the crwodsourcing space. Reframing the finding through an equity lens, the absence of a teaming gap is not evidence that megathreads are equitable. Visibility shapes whose questions are answered, whose skills are recognized, and whose presence is registered by the rest of the community, all of which carry weight beyond the binary outcome of ending on a multi-member team. A novice who posts and receives no replies may eventually team with another novice, but the experience of being overlooked is itself an equity-relevant outcome that the multi-member rate alone cannot capture. This motivates design responses targeted at the visibility rather than only the final teaming layer.

Our results suggest several design implications for more equitable team formation. Platforms could provide templates that prompt users to describe their skills, learning goals, and availability, making it easier for novices to present relevant information in comparable ways [13], and temporarily anonymize tier information in team-seeking forums to reduce preferential attachment effects so that evaluation rests on stated skills rather than status labels. Because experts rarely use megathreads, platforms could also design mentorship pathways that incentivize cross-tier collaboration through recognition systems or reduced entry barriers, and offer practice collaborations before competitive events [16] so newcomers can build trust incrementally. These implications extend beyond Kaggle to other online learning communities, such as open-source development, online courses, and professional networking platforms, where similar expert-novice dynamics may exist.

5.4 Limitations

Two limitations bound the scope of our claims. First, Kaggle's performance tier conflates platform experience with domain expertise and is partly circular: tiers are awarded for cumulative competition medals, so higher-tier users are by definition those who have already performed well, and seasoned data scientists new to Kaggle will be misclassified as novices. We treat this as a substantive feature rather than a flaw, since tier labels are the community's shared expertise signal and our study examines how those socially recognized markers shape interaction, but readers should interpret tier as visible status rather than as ground-truth skill. Second, ending on a multi-member team is a coarse outcome that does not establish that the team was actually formed through the megathread, and it does not capture collaboration quality. Future work should link megathread exchanges to teammate identities directly and incorporate more behavioral signals such as notebook sharing and code co-authorship.

6. CONCLUSION

We examined expert-novice differences in Kaggle's "Looking for a Team" megathreads and found that these platform-provided team formation spaces disproportionately attract novice users while directing disproportionate attention toward higher-tier participants. Higher-tier users receive more network attention, but the linked final team records do not show a statistically significant expertise gap in observed multi-member team membership. These findings contribute to ongoing work on equity in online learning communities, showing that platform mechanisms designed for universal benefit can reproduce hierarchies in visibility even when downstream team outcomes are more mixed. We offer concrete design recommendations for creating more equitable team formation systems in competitive learning communities.

7. REFERENCES

[1] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[2] W. Chow. A pedagogy that uses a Kaggle competition for teaching machine learning: An experience sharing. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, pages 1–5. IEEE, 2019.

[3] I. Dissanayake, N. Mehta, P. Palvia, V. Taras, and K. Amoako-Gyampah. Competition matters! self-efficacy, effort, and performance in crowdsourcing teams. *Information & Management*, 56(8):103158, 2019.

[4] A. Hagberg, P. Swart, and D. S. Chult. Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Lab., 2008.

[5] A. M. Harris, D. Gómez-Zará, L. A. DeChurch, and N. S. Contractor. Joining together online: The trajectory of CSCW scholarship on group formation. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–35, 2019.

[6] K. Huang, J. Zhou, and S. Chen. Being a solo endeavor or team worker in crowdsourcing contests? it is a long-term decision you need to make. *Proceedings*

of the ACM on Human-Computer Interaction, 6(CSCW2):1–27, 2022.

[7] R. F. Kizilcec, J. Reich, M. Yeomans, C. Dann, E. Brunskill, G. Lopez, S. Turkay, J. J. Williams, and D. Tingley. Scaling up behavioral science interventions in online education. *Proceedings of the National Academy of Sciences*, 117(26):14900–14905, 2020.

[8] D. Krackhardt and R. N. Stern. Assessing the political landscape: Structure, cognition, and power in organizations. *Administrative Science Quarterly*, 33(2):342–369, 1988.

[9] X. Li, Y. Bai, and Y. Kang. Exploring the social influence of the Kaggle virtual community on the M5 competition. *International Journal of Forecasting*, 38(4):1507–1518, 2022.

[10] J. Marlow, L. Dabbish, and J. Herbsleb. Impression formation in online peer production: Activity traces and personal profiles in GitHub. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, pages 117–128. ACM, 2013.

[11] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[12] J. Polak and D. Cook. A study on student performance, engagement, and experience with Kaggle inclass data challenges. *Journal of Statistics and Data Science Education*, 29(1):63–70, 2021.

[13] R. Ren, B. Yan, and L. Jian. Show me your expertise before teaming up: Sharing online profiles predicts success in open innovation. *Internet Research*, 30(3):845–868, 2020.

[14] L. Sanz-Martínez, E. Er, A. Martínez-Monés, Y. Dimitriadis, and M. L. Bote-Lorenzo. Creating collaborative groups in a MOOC: A homogeneous engagement grouping approach. *Behaviour & Information Technology*, 38(11):1107–1121, 2019.

[15] M. Twyman, G. Murić, and W. Zheng. Positioning in a collaboration network and performance in competitions: a case study of kaggle. *Journal of Computer-Mediated Communication*, 28(4):zmad024, 2023.

[16] F. L. Vinella, R. Mosch, I. Lykourentzou, and J. Masthoff. The impact of digital nudging techniques on the formation of self-assembled crowd project teams. In *Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, pages 265–275. ACM, 2022.

[17] R. Wang. Marginality and team building in collaborative crowdsourcing. *Online Information Review*, 44(4):827–846, May 2020.

[18] S. Wei and R. Bi. Social network and self-representation in megathread: Group formation in a data science crowdsourcing community. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 434–437. International Educational Data Mining Society, 2024.

[19] Q. Zhang, K. L. Peck, A. Hristova, K. W. Jablokow, V. Hoffman, E. Park, and R. Y. Bayeck. Exploring the communication preferences of MOOC learners and the value of preference-based groups: Is grouping enough? *Educational Technology Research and Development*, 64:809–837, 2016.

Would a Short Explanation Immediately Help? Examining Causal Effects of Students' Help-Seeking in Math

Shan Zhang
University of Florida
zhangshan@ufl.edu

Eamon Worden
Worcester Polytechnic Institute
elworden@wpi.edu

Walter Leite
University of Florida
walter.leite@coe.ufl.edu

Neil T. Heffernan
Worcester Polytechnic Institute
nth@wpi.edu

Anthony F. Botelho
University of Florida
abotelho@coe.ufl.edu

ABSTRACT

Educational research often seeks causal evidence about instructional and content design choices, yet randomized experiments can be difficult to run at scale. Although separate content-level experimental studies are technically possible, they are often impractical without substantial infrastructure and coordination. Quasi-experimental studies, whether through design or through post-hoc analysis of non-experimental data, offer opportunities to infer causality from student interactions with learning content. In the context of digital learning platforms, however, the rich process data collected through student interactions has remained underutilized and represents a strong opportunity to better account for confounding that may otherwise go unaccounted for by covariates available in student course, enrollment, and survey data. To address this gap, in this study, we conduct a causal analysis to estimate the effectiveness of an on-demand explanation using a propensity score weighting analysis with data from the ASSISTments digital learning platform. We draw from prior work in educational data mining and AI in education to leverage Deep Knowledge Tracing as a method to estimate propensity scores. We find that requesting an explanation is associated with lower next-problem performance, but we further identify heterogeneous causal effects, suggesting improved short-term learning for higher-effort students who request explanations compared with those who request only the answer. Overall, this study demonstrates how learner-model representations and process measures can strengthen quasi-experimental analyses of platform logs, while emphasizing that the effectiveness of on-demand explanations depends on students' engagement and help-seeking strategies.

Keywords

Causal Inference, Propensity Scoring Weighing, Deep Knowledge Tracing, Explanation, Help-seeking

Shan Zhang, Eamon Worden, Walter Leite, Neil T. Heffernan, and Anthony F. Botelho. Would a Short Explanation Immediately Help? Examining Causal Effects of Students' Help-Seeking in Math. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 818–822. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039683

1. INTRODUCTION

Educational research increasingly seeks causal evidence to inform instructional and content design decisions. Randomized experiments provide a strong foundation for such evidence and have been successfully deployed in digital learning platforms (DLPs), including studies that systematically vary instructional supports such as hints, feedback, and explanations [14, 15, 17]. However, running experiments comprehensively across the wide range of problems, supports, and learning contexts in large-scale platforms requires substantial infrastructure and coordination, and is often impractical for systems that are already widely deployed. As a result, researchers frequently rely on observational interaction data, raising the challenge of how to draw credible causal inferences from non-experimental platform logs [3, 7].

Causal inference in observational DLP data is particularly challenging because students' interactions with instructional supports are highly non-random. Students who request help often differ systematically from those who do not, not only in prior knowledge but also in engagement, persistence, and other learner characteristics that jointly influence both help-seeking behavior and learning outcomes [11]. These selection effects can substantially bias naïve estimates of instructional effectiveness if not adequately addressed. Recent work in educational data mining (EDM) has therefore emphasized the need for clearly defining causal estimands, evaluating common support, diagnosing covariate balance, and assessing robustness to unobserved confounding [4, 8, 10, 23].

Despite this progress, a persistent limitation of many observational studies is that adjustment for confounding often relies on relatively coarse or static proxies for prior knowledge, such as aggregate accuracy. This is problematic because prior outcomes are often among the most important confounders of treatment [20]. This mismatch is particularly striking given that digital learning platforms record rich, high-dimensional, and temporally structured process data that summarize students' learning histories in far greater detail than is typically used in causal analyses [19].

To address this gap, in this study, we conduct a quasi-experimental causal analysis that leverages learner-model representations to strengthen control for confounding in observational platform data. Using data from ASSISTments, we estimate the short-term causal effect of requesting an

on-demand explanation for a single target problem using propensity score weighting [9]. We first train a Deep Knowledge Tracing (DKT) model on students' historical interactions to produce fine-grained, skill-level mastery estimates that summarize each learner's prior learning history [16]. These mastery estimates are then incorporated as covariates in propensity score estimation [13], following contemporary guidance for machine learning–based causal inference in educational data mining [8, 10]. We compare explanation use against two counterfactual baselines—no on-demand help-seeking and answer-only requests—and examine heterogeneity by engagement, motivated by the hypothesis that explanations may support sense-making use but not rapid task completion. Specifically, we address the following research questions:

RQ 1. What is the effect of requesting on-demand explanations in comparison to students who do not seek help on learning, as measured by correctness on the next problem?

RQ 2. What is the effect of requesting on-demand explanations in comparison to students who ask for the answer on learning, as measured by correctness on the next problem?

RQ 3. Are the effects of explanation use heterogeneous across students with different levels of engagement?

2. METHODS

2.1 Dataset

The data[1] were drawn from ASSISTments, a digital learning platform widely used in U.S. middle school mathematics instruction. We focus on students who attempted a target problem and a subsequent problem assessing the same skill (7.RP.A; proportional reasoning). The target problem was completed by 12,697 students, of whom 7,179 answered incorrectly and were included in the analytic sample.

Students could optionally request on-demand supports while solving the target problem. As shown in Figure 1, the explanation provides step-by-step guidance, whereas Figure 2 shows the answer-only view, which reveals the final solution without intermediate steps. The outcome is a binary indicator of correctness on the subsequent problem (968 correct; 6,211 incorrect). Student interaction logs included problem attempts, correctness, skill identifiers, and help-seeking actions. These data were used to construct prior knowledge measures and behavioral indicators such as time-on-task.

2.2 Data Processing

Problem-level logs were transformed into student–skill sequences. Each problem attempt was mapped to a skill identifier, and each unique skill, including multi-skill combinations, was treated as a distinct category (as suggested in [22]), resulting in $K = 189$ skills. (All 189 skill labels are available on the Open Science Framework (OSF))[2]. The final processed dataset consisted of 1,700,234 problem-attempt rows from 7,179 unique students, with each row aligned to a single student, a single encoded skill, and a unique row identifier to support downstream model training and post hoc analyses.

[1]This study involves post hoc analysis of platform log data conducted with approval from an Institutional Review Board (IRB). Data access may be granted through a data-sharing agreement with ASSISTments.

[2]`https://osf.io/vgd76/overview?view_only=4c03f1f48f4d47c9a170a78a487c128a`

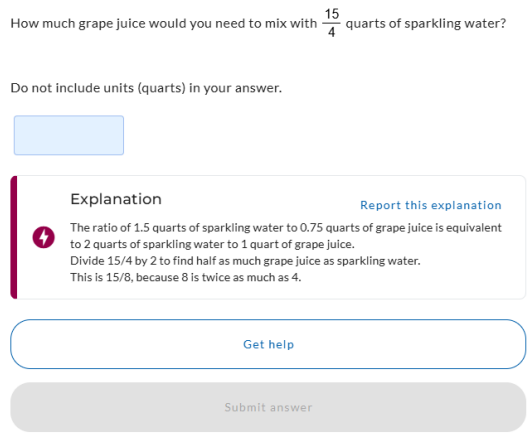


Figure 1: An explanation as it appears in ASSISTments.

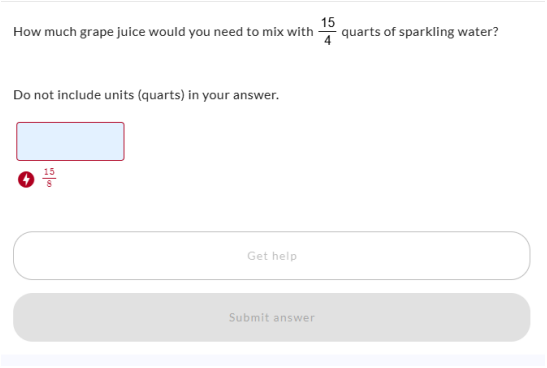


Figure 2: The answer as it appears in ASSISTments.

2.3 Analytic Approach

To estimate the causal effect of explanation use, we combined learner modeling with propensity score weighting (PSW). First, we trained a DKT model on students' interaction sequences to obtain skill-level mastery estimates. Students with fewer than two interactions ($n = 30$) were excluded, yielding 7,149 students. The model was trained using an LSTM architecture with 10-fold cross-validation to generate out-of-fold predictions (AUC = 0.742), which were used as high-dimensional covariates representing prior knowledge.

Next, we constructed mutually exclusive help-seeking groups based on students' behavior on the target problem. Students were categorized into three groups: (1) *Explanation*, students who requested an explanation only; (2) *Answer-only*, students who requested an answer only; and (3) *No-help*, students who did not request any help. The final analytic sample included 821 students in the Explanation group, 3,659 in the Answer-only group, and 1,535 in the No-help group.

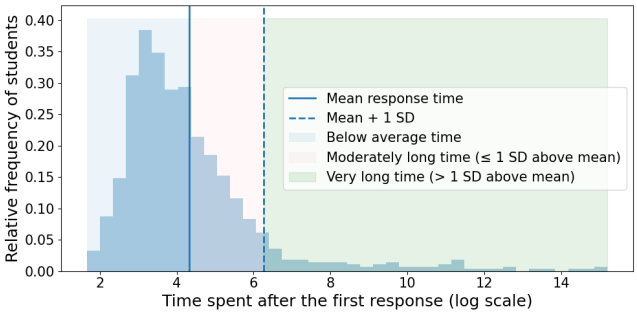


Figure 3: Distribution of time spent after the first response (log scale) for students in the Explanation group ($n = 816$)

To examine heterogeneity, we defined an effort-based subset within the Explanation group using adjusted time-on-task derived from log data (Figure 3). Time was computed as problem duration minus time to first response and log-transformed due to skew. The *Moderate-Effort* subset included students whose adjusted time fell between the mean and mean plus 0.5 standard deviations ($n = 153$), while the *High-Effort* subset included students whose adjusted time fell between the mean and mean plus 1 standard deviation ($n = 228$).

For causal estimation, we applied PSW to adjust for observed differences between groups [13, 18]. Propensity scores were estimated using multiple models, including logistic regression (LR), gradient boosting machines (GBM), random forests (RF), Bayesian additive regression trees (BART), and neural networks (NN), following recent recommendations for machine learning–based causal inference [10]. Separate propensity score models were estimated for each treatment–control contrast (explanation vs. no-help, explanation vs. answer-only, and effort-based subsets).

For each estimator, the common-support assumption was evaluated by examining the overlap in propensity score distributions between treated and control groups. Models exhibiting limited overlap or extreme propensity scores were excluded. Covariate balance was assessed using absolute standardized mean differences (SMDs), with acceptable balance defined as $|\text{SMD}| \leq 0.25$ [21]. Weights targeting the average treatment effect on the treated (ATT) were constructed such that treated units received weight 1 and control units received $PS/(1-PS)$. Residual imbalance ($0.05 \leq |\text{SMD}| \leq 0.25$) was addressed by including covariates in outcome models.

Treatment effects were estimated using a parametric standardization approach with weighted logistic regression models predicting next-problem correctness [5]. Cluster-robust standard errors were computed to account for classroom-level dependence [12], and statistical uncertainty was quantified using cluster bootstrap resampling (100 replications) to obtain 95% confidence intervals. Sensitivity analyses were conducted using Frank's robustness framework to assess the impact of potential unobserved confounding [6].

3. RESULTS

3.1 RQ1: Effects of On-Demand Explanations Compared to No Help-Seeking

We examined whether requesting explanations affected subsequent performance compared to not seeking help ($n_{control} = 1535$, $n_{treat} = 821$) using PSW based on 189 DKT-derived covariates. Common-support diagnostics indicated adequate overlap for LR, GBM, and BART, while RF and NN exhibited limited or unstable support and were excluded from further analysis. All common-support diagnostics are available on OSF. After weighting, covariate balance improved substantially for LR, GBM, and BART (max |SMD| reduced from 0.973 to 0.167 under LR, 0.177 under BART, and 0.224 under GBM, respectively), indicating that observed differences between groups were effectively reduced.

Among retained estimators, results were consistent in both direction and magnitude. The LR-based ATT was -0.185 (SE = 0.028, 95% CI [-0.250, -0.147]), and BART yielded a similar estimate of -0.191 (SE = 0.029, 95% CI [-0.256, -0.149]). Although GBM produced a negative estimate, it was excluded due to instability under bootstrap resampling. Sensitivity analyses indicated that these negative effects were robust to potential unobserved confounding. These results suggest that, after adjusting for prior knowledge and observed covariates, students who requested explanations were less likely to correctly answer the subsequent problem than comparable students who did not seek help.

3.2 RQ2: Effects of On-Demand Explanations Compared to Answer-Only

We next compared explanation use to answer-only help ($n_{\text{control}} = 3659$, $n_{\text{treat}} = 821$). Common-support diagnostics indicated adequate overlap for LR, GBM, and BART, while RF and NN showed weaker or unstable support. Propensity score weighting improved covariate balance, with all retained estimators achieving acceptable post-weighting balance ($|SMD| \leq 0.25$) and most covariates showing low imbalance ($|SMD| < 0.10$).

Across all retained estimators, treatment effects were consistently near zero and not statistically significant. The LR-based ATT was 0.000 (SE = 0.012, 95% CI [-0.021, 0.026]), with similar near-zero estimates observed for GBM and BART. These findings indicate that, on average, requesting an explanation does not lead to improved short-term performance compared to simply viewing the answer.

3.3 RQ3: Heterogeneity of Explanation Effects by Student Engagement

We examined heterogeneity by focusing on an effort-based subset of students within the Explanation group ($n = 228$), representing moderate-to-high engagement.

Explanation (effort-based subset) vs. no help-seeking. Adequate common support was observed for LR and BART, and weighting improved covariate balance (max |SMD| reduced to 0.162 for LR and 0.146 for BART). Both estimators yielded statistically significant negative effects. The LR-based ATT was -0.139 (SE = 0.024, 95% CI [-0.179, -0.093]), and BART produced -0.184 (SE = 0.040, 95% CI [-0.249, -0.131]). Although these effects remain negative, they are smaller in magnitude compared to the full-sample

results, suggesting that greater engagement may partially mitigate the negative association observed in RQ1.

Explanation (effort-based subset) vs. answer-only. Common support was broader across estimators, and weighting achieved acceptable covariate balance. In contrast to the previous comparison, estimated effects were small but positive. The LR-based ATT was 0.048 (SE = 0.024, 95% CI [0.002, 0.095]), and BART yielded 0.057 (SE = 0.025, 95% CI [0.016, 0.100]). These findings indicate that, among more engaged students, explanations may provide modest benefits relative to answer-only support.

4. DISCUSSION AND FUTURE WORK

This study uses observational log data from ASSISTments to estimate the short-term causal effects of on-demand explanation use on subsequent problem performance. Methodologically, we integrate learner modeling with causal inference by incorporating DKT-derived skill mastery estimates as high-dimensional covariates in PSW models. This approach provides a richer adjustment for prior knowledge than commonly used proxies and illustrates how learner-model representations can strengthen causal analyses of platform log data [8, 10]. We also follow contemporary guidance for machine learning–based propensity score estimation, including overlap checks, balance diagnostics, estimator comparison, sensitivity analyses, and documentation of retained models [4, 8, 10]. Overall, these design choices illustrate how learner-model outputs and process data can strengthen post-hoc causal analyses of platform logs [23].

Across comparisons, explanation use was not associated with improved short-term performance on average and was instead linked to lower subsequent correctness when compared to students who did not seek help (RQ1). One plausible explanation for this pattern is the presence of unproductive help-seeking behaviors, such as gaming the system, that are not fully captured by the DKT-derived prior knowledge estimates. While DKT controls for students' historical performance at the skill level, it does not explicitly model behavioral tendencies related to strategic help use, such as rapidly requesting explanations to complete assignments rather than to support understanding. As a result, explanation use in this comparison may partially proxy for underlying gaming behavior, which has been shown to undermine learning outcomes in tutoring systems [1, 2]. In this sense, the estimated effects may reflect the impact of *clicking to access help and receiving an explanation as enacted in the platform*, rather than the effect of engaging with an explanation under conditions of productive use.

At the same time, alternative mechanisms may contribute to the observed negative average effects. Students in the no-help-seeking control group may engage more often in self-regulated problem solving, including persistence through errors, metacognitive monitoring, and strategy refinement. They may also seek support from external sources, such as peers, teachers, or other materials, which are not captured in platform data. These unobserved differences highlight an important limitation of log-based analyses and suggest that comparisons between help-seeking and non-help-seeking students may conflate the effects of instructional supports with broader differences in learning strategies and resource use.

In contrast, the explanation-versus-answer-only comparison (RQ2) yielded near-zero average effects across retained estimators, with confidence intervals spanning zero. This finding suggests that, for the target problem examined, viewing an explanation rather than only the final answer is associated with little to no immediate difference in performance on the subsequent problem. These near-zero estimates should not be interpreted as evidence that explanations lack educational value more broadly. Instead, they indicate that the effectiveness of explanations likely depends on how students engage with the support, rather than on the mere availability of help [8].

Results from the effort-based subset analysis (RQ3) further clarify this heterogeneity. When the explanation subgroup with moderate-to-higher adjusted time-on-task was compared to no-help-seeking controls, estimated effects remained negative, but with reduced magnitude. When the same effort-based subset was compared to answer-only students, estimated effects were small and positive. These contrasts suggest that explanation use does not operate as a uniformly beneficial treatment. Instead, outcomes vary with students' engagement patterns and with the counterfactual baseline. From a design perspective, these findings motivate platform features that better distinguish sense-making help use from rapid answer-seeking and that scaffold students toward more constructive engagement with explanations.

Several limitations remain. First, although we control for prior knowledge using DKT and include log-based engagement measures, we do not directly observe other learner characteristics such as motivation, metacognition, or affect. Second, the analysis focuses on a single problem and performance on the next problem, which may limit generalizability to other content areas or longer learning sequences. Third, although we examine heterogeneity using an effort-based proxy, engagement is likely multi-dimensional and may not be fully captured by time-on-task alone. With that, future work should extend this approach by incorporating richer measures of student behavior, including finer-grained indicators of self-regulated learning. In addition, applying this framework across multiple problems would help clarify the longer-term effects of explanation use.

5. CONCLUSION

This study demonstrates how learner-model representations and process data can be integrated into a rigorous quasi-experimental framework to estimate causal effects from observational digital learning platform logs. By incorporating Deep Knowledge Tracing–based skill mastery estimates as high-dimensional confounding controls within propensity score weighting, the approach extends common EDM practice and supports more transparent and principled causal inference. Applied to on-demand explanation use in ASSISTments, the findings show that explanations do not yield uniform short-term performance benefits and that their effects depend on students' engagement and help-seeking behaviors. Overall, these results demonstrate the importance of evaluating instructional supports as they are enacted in real learning contexts and highlight the methodological value of combining learner modeling with causal inference to study heterogeneous effects in educational data.

6. ACKNOWLEDGMENTS

We would like to thank the National Science Foundation (#2331379), Institute of Education Sciences (#R305B230007), Gates Foundation (#078981), Support from the Learning Engineering Tools Competition, and other anonymous philanthropy.

7. REFERENCES

[1] R. Baker, J. Walonoski, N. Heffernan, I. Roll, A. Corbett, and K. Koedinger. Why students engage in "gaming the system" behavior in interactive learning environments. *Journal of Interactive Learning Research*, 19(2):185–224, 2008.

[2] R. S. Baker, J. E. Richey, J. Zhang, S. Karumbaiah, J. M. Andres-Bray, H. A. Nguyen, J. M. A. L. Andres, and B. M. McLaren. Gaming the system mediates the relationship between gender and learning outcomes in a digital learning game. *Instructional Science*, 53(2):203–238, 2025.

[3] J. E. Beck, K.-m. Chang, J. Mostow, and A. Corbett. Does help help? introducing the bayesian evaluation and assessment methodology. In *International conference on intelligent tutoring systems*, pages 383–394. Springer, 2008.

[4] A. F. Botelho, A. H. Closser, A. C. Sales, N. T. Heffernan, and K. P. Vanacore. 2024 workshop on causal inference in educational data mining. In *Proceedings of the Seventeenth International Conference on Educational Data Mining (EDM 2024)*, 2024. Workshop proposal.

[5] B. A. Brumback. *Fundamentals of causal inference: with R*. Chapman and Hall/CRC, 2021.

[6] K. A. Frank, S. J. Maroulis, M. Q. Duong, and B. M. Kelcey. What would it take to change an inference? using rubin's causal model to interpret the robustness of causal inferences. *Educational Evaluation and Policy Analysis*, 35(4):437–460, 2013.

[7] J.-E. Gustafsson. Causal inference in educational effectiveness research: A comparison of three methods to investigate effects of homework on student achievement. *School Effectiveness and School Improvement*, 24(3):275–295, 2013.

[8] K. Kitto, B. Hicks, and S. Buckingham Shum. Using causal models to bridge the divide between big data and educational theory. *British Journal of Educational Technology*, 54(5):1095–1124, 2023.

[9] W. Leite. *Practical propensity score methods using R*. Sage Publications, 2016.

[10] W. Leite, H. Zhang, Z. Collier, K. Chawla, L. Kong, Y. Lee, J. Quan, and O. Soyoye. Machine learning for propensity score estimation: A systematic review and reporting guidelines. *Psychological Methods*, 2025.

[11] R. Li, N. Che Hassan, and N. Saharuddin. College student's academic help-seeking behavior: A systematic literature review. *Behavioral Sciences*, 13(8):637, 2023.

[12] T. Lumley. Analysis of complex survey samples. *Journal of Statistical Software*, 9(8):1–19, 2004.

[13] J. K. Lunceford and M. Davidian. Stratification and weighting via the propensity score in estimation of causal treatment effects: a comparative study. *Statistics in medicine*, 23(19):2937–2960, 2004.

[14] K. S. Ostrow and N. T. Heffernan. Studying learning at scale with the assistments testbed. In *Proceedings of the third (2016) ACM conference on learning@ scale*, pages 333–334, 2016.

[15] T. Patikorn and N. T. Heffernan. Effectiveness of crowd-sourcing on-demand assistance from teachers in online learning platforms. In *Proceedings of the Seventh ACM Conference on Learning @ Scale*, L@S '20, page 115–124, New York, NY, USA, 2020. Association for Computing Machinery.

[16] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[17] S. Ritter, A. Murphy, and S. Fancsali. Curriculum-embedded experimentation. In *Third Workshop on A/B Testing and Platform-Enabled Research (Learning@ Scale 2022), New York, NY, USA.*, 2022.

[18] P. R. Rosenbaum and D. B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

[19] A. Sales, A. Botelho, T. Patikorn, and N. T. Heffernan. Using big data to sharpen design-based inference in a/b tests. In *Proceedings of the eleventh international conference on educational data mining*, 2018.

[20] P. M. Steiner, T. D. Cook, W. R. Shadish, and M. H. Clark. The importance of covariate selection in controlling for selection bias in observational studies. *Psychological methods*, 15(3):250, 2010.

[21] E. A. Stuart. Matching methods for causal inference: A review and a look forward. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 25(1):1, 2010.

[22] X. Xiong, S. Zhao, E. G. Van Inwegen, and J. E. Beck. Going deeper with deep knowledge tracing. In *Proceedings of the Ninth International Conference on Educational Data Mining*. International Educational Data Mining Society, 2016.

[23] L. Yao, Z. Chu, S. Li, Y. Li, J. Gao, and A. Zhang. A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(5):1–46, 2021.

AI-Assisted Ultrasound Instruction for Novices: Can We Avoid Overreliance?

Jack Fitzgerald
Colorado State University
jack.fitzgerald@colostate.edu

Nathaniel Blanchard
Colorado State University
nathaniel.blanchard@colostate.edu

ABSTRACT

Being trained to competently use an ultrasound device typically takes a few years of training, and access to training programs is limited. With the rapid advancements being made in the field of machine learning, researchers are exploring ways to bridge this gap by developing AI-powered systems that can quickly upskill ultrasound novices. These systems have been shown to improve novice performance in the short term, but it is unclear if they are truly upskilling their users. It's possible that users of such systems are overrelying on the guidance that they provide, placing a hurdle in their path towards mastering the necessary skills to become competent sonographers. I believe that it's important to keep overreliance in mind when developing a system for AI-guided ultrasound, since it could potentially have the opposite effect and ultimately decrease the pool of competent sonographers in the long run. I plan to explore the cutting edge of the field to see what tools we have at our disposal, in particular, I have found world modeling and learning latent action spaces to be an exciting direction that could be applied to this problem. The generalizability of world modeling and the ability to learn a latent action space can be utilized to create an agent that imitates and aligns with the actions that professional ultrasound instructors take during traditional training programs. Ideally, such an agent could reduce the overreliance problem by performing actions that are conducive to learning and is not constantly giving away "correct answers" to every problem a sonogorapher trainee encounters during hands-on exercises.

Keywords

Ultrasound, AI-guidance, Overreliance, Latent Actions, World Modeling

1. PROBLEM INTRODUCTION

Point-of-care ultrasound (POCUS) has become a powerful tool in the arsenal of medical practitioners thanks to decades of innovation and miniaturization, donning the title of the "new stethoscope" [6, 25]. Despite it's demonstrated value and importance in modern medicine, becoming trained in the use of an ultrasound device is costly and difficult. Introducing ultrasound training early in a medical professionals career, such as in their undergraduate studies, has shown to pay off [24]. However, early career ultrasound training is limited by equipment costs, lack of faculty funding, and decisions over how to fit ultrasound training into an already packed undergraduate curriculum. As the demand for competent sonographers increases, utilizing AI to guide and teach early-career medical practitioners and ultrasound novices appears to be a promising direction towards addressing these shortcomings

1.1 Alternative Approaches to Teaching Ultrasound

Less conventional approaches to teaching the usage of an ultrasound device have been proposed to increase access to hands-on practice. [4] developed a system that tracked the position of a probe on a manikin, and a monitor would display ultrasound images collected from real patients based on the probe's position. [28] developed the Perk Tutor system, which provided visualizations and assistance for learning ultrasound-guided needle insertions. A follow-up study with Perk Tutor showed that undergraduate students that practiced with it had improved performance compared to a control group afterwards in a test without the Perk Tutor assistance [18]. [15] developed a similar system to Perk Tutor for ultrasound-guided needle insertions and found improved novice performance in certain cases, however they did not investigate if performance persisted when not using the guidance system. While these studies show promising results, one study has shown that fully self-directed ultrasound training with simulation software is insufficient [19]. One work developed a serious game, where the player scans boxes in a toy factory, to help with spatially understanding ultrasound [8]. Mixed reality has also been used to develop a sophisticated ultrasound teaching tool [29].

1.2 AI Ultrasound Guidance

Much work has been done in the last decade to leverage AI to aid in ultrasound scans, including assistance in scan plane acquisition and detection of anatomy within the ultrasound image. [5] developed a CNN model for obstetric ultrasounds to detect standard planes and localize anatomy. [9] Learns from experts performing echocardiograms by training a model to predict the next rotational movement of the ultrasound probe that aligns with a desirable scan plane of the heart. Further work expanded on this and trained a model

Jack Fitzgerald, and Nathaniel Blanchard. AI-Assisted Ultrasound Instruction for Novices: Can We Avoid Overreliance?. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 823–825. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040028

on expert eye-gaze, indicating to a novice sonographer where they should be looking at in an ultrasound image during a scan [20]. Reinforcement learning has been used to predict movements toward the desired scan plane by collecting ultrasound scans in such a way that allows for offline training [21].

World modeling techniques [3, 13] have been used to perform probe movement guidance [31, 14] by making good use of unlabeled data. However, this task necessitates often expensive equipment such as robot arms or optical tracking systems to obtain accurate 6D poses of the ultrasound probe during scanning. [9] makes use of an inertial measurement unit (IMU) device, which tends to be cheaper, but noise in the accelerometer makes translational data unusable and thus is only able to predict rotational adjustments to the probe.

1.3 Novice Performance with AI Guidance

AI-guided ultrasound systems that provide probe position guidance [17, 22] and segmented anatomy overlays [7] have been shown to improve novice performance. However, these works do not investigate if these methods of AI guidance facilitate long-term learning or induce overreliance. Although these systems have demonstrated performance improvement in novices, is it enough to fully address the shortage of trained sonographers? If overreliance proves to be an issue, then that would be concerning towards the long term development of skills [16]. I propose that to avoid overreliance on AI guidance, the system should try to closely align to traditional ultrasound training that happens during hands-on practice with volunteers or phantoms.

2. PROPOSED SOLUTION

Developing an AI guidance system that teaches sonography to novices on par with traditional instruction requires the system to understand what types of actions and behaviors (verbal, gestures, etc.) that traditional ultrasound instructors display during hands-on exercises. Existing work has introduced and analyzed datasets containing teacher actions and behavior in classroom settings [1, 33, 26, 11, 23]. However, similar datasets that label teacher behavior and actions for ultrasound instructors do not exist. Work has been done to wholistically represent the actions that sonographers take with the aim of improving sonography education [10], which can be useful in informing dataset curation and possible actions, but these could be different than the actions instructors take. I propose that curating a dataset of instructors guiding students through an ultrasound scanning procedure and annotating relevant actions is the penultimate step towards a guidance system that aligns with traditional ultrasound training.

A promising line of research that has recently picked up steam is in latent action modeling [12, 32, 30, 27], which seeks to learn possible actions from a dataset and represent them in latent space. Difficulties with privacy and costs immediately arise when collecting data of ultrasound instruction (e.g. one-on-one scanning sessions with a patient, sonography student, and an instructor), and labeling such a dataset with appropriate actions adds another layer of costs. These problems in data collection motivate a self-supervised approach towards learning a useful action space for the task of modeling actions that sonogrpahy make. Moreover, pretraining on web-scale data via world modeling to learn a rich action space from unrelated tasks [12] could be frutiful, since certain generic actions may transfer (such as pointing, or verbally indicating mistakes). Models trained in such a way, such as VJEPA2 [2], have demonstrated excellent sample efficiency on downstream tasks. Therefore, perhaps only a relatively small amount of ultrasound instructor action-labeled data would be enough to fine tune a powerful model. This would be the ultimate step of the proposed solution.

3. IMPACT

I anticipate that this research will provide insights to the benefits and shortcomings in applying world modeling and latent action space modeling to education, but more specifically to the domain of hands-on ultrasound training. Developing an action-labeled dataset for ultrasound instruction is also critical, as it can directly impact the ability of the model to derive appropriate representations of the instructor actions that are actually taking place during such training.

4. REFERENCES

[1] B. Albreiki, N. Zaki, and H. Alashwal. A systematic literature review of student'performance prediction using machine learning techniques. *Education Sciences*, 11(9):552, 2021.

[2] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[3] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15619–15629, 2023.

[4] C. J. Banker, P. C. Pedersen, and T. L. Szabo. Interactive training system for medical ultrasound. In *2008 IEEE Ultrasonics Symposium*, pages 1350–1354. IEEE, 2008.

[5] C. F. Baumgartner, K. Kamnitsas, J. Matthew, T. P. Fletcher, S. Smith, L. M. Koch, B. Kainz, and D. Rueckert. Sononet: real-time detection and localisation of fetal standard scan planes in freehand ultrasound. *IEEE transactions on medical imaging*, 36(11):2204–2215, 2017.

[6] A. Bledsoe and J. Zimmerman. Ultrasound: The new stethoscope (point-of-care ultrasound). *Anesthesiology clinics*, 39(3):537–553, 2021.

[7] J. S. Bowness, A. J. Macfarlane, D. Burckett-St Laurent, C. Harris, S. Margetts, M. Morecroft, D. Phillips, T. Rees, N. Sleep, A. Vasalauskaite, et al. Evaluation of the impact of assistive artificial intelligence on ultrasound scanning for regional anaesthesia. *British journal of anaesthesia*, 130(2):226–233, 2023.

[8] B. Byl, M. Süncksen, and M. Teistler. A serious virtual reality game to train spatial cognition for medical ultrasound imaging. In *2018 IEEE 6th International Conference on Serious Games and Applications for Health (SeGAH)*, pages 1–4. IEEE, 2018.

[9] R. Droste, L. Drukker, A. T. Papageorghiou, and J. A. Noble. Automatic probe movement guidance for freehand obstetric ultrasound. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 583–592. Springer, 2020.

[10] L. Drukker, H. Sharma, R. Droste, M. Alsharid, P. Chatelain, J. A. Noble, and A. T. Papageorghiou. Transforming obstetric ultrasound into data science using eye tracking, voice recording, transducer motion and ultrasound video. *Scientific Reports*, 11(1):14109, 2021.

[11] Z. Gang, Z. Wenjuan, H. Biling, C. Jie, H. Hui, and X. Qing. A simple teacher behavior recognition method for massive teaching videos based on teacher set. *Applied Intelligence*, 51(12):8828–8849, 2021.

[12] Q. Garrido, T. Nagarajan, B. Terver, N. Ballas, Y. LeCun, and M. Rabbat. Learning latent action world models in the wild. *arXiv preprint arXiv:2601.05230*, 2026.

[13] D. Ha and J. Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3):440, 2018.

[14] H. Jiang, Z. Sun, N. Jia, M. Li, Y. Sun, S. Luo, S. Song, and G. Huang. Cardiac copilot: Automatic probe guidance for echocardiography with world model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 190–199. Springer, 2024.

[15] E. J. Kim, J. Min, J. Song, K. Song, J.-H. Song, and H. J. Byon. The effect of electromagnetic guidance system on early learning curve of ultrasound for novices. *Korean journal of anesthesiology*, 69(1):15–20, 2016.

[16] N. Kosmyna, E. Hauptmann, Y. T. Yuan, J. Situ, X.-H. Liao, A. V. Beresnitzky, I. Braunstein, and P. Maes. Your brain on chatgpt: Accumulation of cognitive debt when using an ai assistant for essay writing task. *arXiv preprint arXiv:2506.08872*, 4, 2025.

[17] A. Kumar, E. Baum, C. Parmer-Chow, and J. Kugler. Limited echocardiogram acquisition by novice clinicians aided with deep learning: A randomized controlled trial. *Biology Methods and Protocols*, 10(1):bpaf083, 2025.

[18] H. Lia, Z. Keri, M. S. Holden, V. Harish, C. H. Mitchell, T. Ungi, and G. Fichtinger. Training with perk tutor improves ultrasound-guided in-plane needle insertion skill. In *Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10135, pages 232–239. SPIE, 2017.

[19] F. D. Mackay, F. Zhou, D. Lewis, J. Fraser, and P. R. Atkinson. Can you teach yourself point-of-care ultrasound to a level of clinical competency? evaluation of a self-directed simulation-based training program. *Cureus*, 10(9):e3320, 2018.

[20] Q. Men, C. Teng, L. Drukker, A. T. Papageorghiou, and J. A. Noble. Multimodal-guidenet: Gaze-probe bidirectional guidance in obstetric ultrasound scanning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 94–103. Springer, 2022.

[21] F. Milletari, V. Birodkar, and M. Sofka. Straight to the point: Reinforcement learning for user guidance in ultrasound. In *International Workshop on Preterm, Perinatal and Paediatric Image Analysis*, pages 3–10. Springer, 2019.

[22] V. Mor-Avi, B. Khandheria, R. Klempfner, J. I. Cotella, M. Moreno, D. Ignatowski, B. Guile, H. J. Hayes, K. Hipke, A. Kaminski, et al. Real-time artificial intelligence–based guidance of echocardiographic imaging by novices: image quality and suitability for diagnostic interpretation and quantitative analysis. *Circulation: Cardiovascular Imaging*, 16(11):e015569, 2023.

[23] J. Naim, J. Cao, F. Tasneem, J. Jacobs, B. Milne, J. Martin, and T. Sumner. Towards actionable pedagogical feedback: A multi-perspective analysis of mathematics teaching and tutoring dialogue. In *Proceedings of the 18th International Conference on Educational Data Mining*, pages 328–341, 2025.

[24] S. M. Provenzano, I. Mageto, A. González Frontera, C. Uter, S. Casey, A. Badillo, and A. O. Fakoya. Ultrasound as a foundational tool in medical education: A literature review. *Advances in Medical Education and Practice*, pages 2167–2179, 2025.

[25] E. Sabath. Point of care ultrasonography as the new "laennec sthetoscope". *World Journal of Nephrology*, 13(1):90542, 2024.

[26] V. Sharma, M. Gupta, A. Kumar, and D. Mishra. Star-3d: A holistic approach for human activity recognition in the classroom environment. *Information*, 15(4):179, 2024.

[27] B. Tharwat, Y. Nasser, A. Abouzeid, and I. Reid. Latent action pretraining through world modeling. *arXiv preprint arXiv:2509.18428*, 2025.

[28] T. Ungi, D. Sargent, E. Moult, A. Lasso, C. Pinter, R. C. McGraw, and G. Fichtinger. Perk tutor: an open-source training platform for ultrasound-guided needle insertions. *IEEE Transactions on Biomedical Engineering*, 59(12):3475–3481, 2012.

[29] J. Wang, J. Zhang, J. N. Capizzano, M. Sigakis, X. Wang, and V. Popov. explainmr: Generating real-time textual and visual explanations to facilitate ultrasonography learning in mr. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2025.

[30] S. Ye, J. Jang, B. Jeon, S. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin, et al. Latent action pretraining from videos. *arXiv preprint arXiv:2410.11758*, 2024.

[31] Y. Yue, Y. Wang, H. Jiang, P. Liu, S. Song, and G. Huang. Echoworld: Learning motion-aware world models for echocardiography probe guidance. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 25993–26003, 2025.

[32] C. Zhang, T. Pearce, P. Zhang, K. Wang, X. Chen, W. Shen, L. Zhao, and J. Bian. What do latent action models actually learn? *arXiv preprint arXiv:2506.15691*, 2025.

[33] J. Zhao, H. Zhu, and L. Niu. Bitnet: A lightweight object detection network for real-time classroom behavior recognition with transformer and bi-directional pyramid network. *Journal of King Saud University-Computer and Information Sciences*, 35(8):101670, 2023.

Astronauts' Cognitive Mapping: Evolving Identity through the Chronospatial Frontiers

Ginger[1] and Joo Young Park[1]
[1]Florida Institute of Technology
chenj2021@my.fit.edu

ABSTRACT
This research aims to construct a theoretical model of the cognitive maps contributing to astronauts' academic and professional success. By integrating Interpretative Phenomenological Analysis (IPA) and grounded theory (GT) to identify in-depth sensemaking as well as philosophical patterns within key factors, the researcher conducted nine semi-structured interviews to deepen the inquiry into astronauts' life-long narratives. The study introduces an original cognitive-mapping model comprising three interlocking systems: the chronosystem, ecosystem, and identity system. It explores how astronauts define pivotal moments, perceive environments, and establish self-identity to succeed in academic study and career development.

To address the analytical challenges of large-scale qualitative data, this study applies a human–AI collaborative approach, through which 92 axial categories and 20 selective themes were identified. Three original frameworks—Heavenly Hour, Terrain Triumph, and Human Harmony—were established, alongside two substantive theories and one emerging middle-range theoretical proposition. The findings further suggest a bicameral theorizing model for human–AI qualitative analysis, in which human interpretive depth and AI-enabled analytical scale converge to support complementary forms of meaning-making and theory construction. To operationalize this process, the researcher developed the GOLDEN Framework, a six-step methodology for ethical and reflexive human–AI collaboration designed to support rigorous analysis of large-scale qualitative data and strengthen synergistic human–AI learning.

Keywords
Cognitive mapping, Interpretative Phenomenology Analysis, Grounded Theory, Human-AI Collaboration, Qualitative analysis

1. INTRODUCTION
The contemporary challenges including the biological weapon threats [1, 2], environmental crises, geopolitical tensions [3], and technological shifts driven by AI have contributed to an education crisis [4]: how to guide the current and future generations towards success? Astronauts, as modern-day way finders, with their life experiences and informed decision-making could delineate a cognitive map [5] for navigating educational and social landscapes in an era marked by unprecedented uncertainty. This study examines how astronauts perceive pivotal moments, formative environments, and self-identity, focusing on revealing the underlying factors that enable them to excel in academic study and career development. Using a qualitative approach that combines IPA [6] and GT [7], while IPA grounds the inquiry in the richness of individual lived experience, GT provides the systematic scaffolding to build transferable conceptual patterns across cases, the researcher seeks the shared meanings and emergent conceptual patterns within their lived experience to address the three research questions: to have excelled in academic study and career development, 1) What are astronauts' pivotal moments? 2) What are the main environmental factors? 3) How do astronauts define their self-identity?

Ginger Chen, and Joo Young Park. Astronauts' Cognitive Mapping: Evolving Identity through the Chronospatial Frontiers. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 826–831. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040006

2. METHODS

2.1 Sampling, Participants, and Data
Purposive sampling method is chosen to conduct semi-structured interviews [8], which allows the deliberate selection of individuals with the specific criteria necessary to address the research questions. The sample consists of nine retired astronauts who have travelled to space. Four astronauts' lifelong partners also consented to join the project, helping the participant better recall details and strengthen the depth of the conversation by providing crucial relational context. The primary astronaut sample includes female (n=2) and male astronauts (n=7). The average age of total 13 participants (nine astronauts and four partners) is 76. All of the participants were given a pseudonym unless they explicitly gave the consent to use their real names.

A research facilitator supported the recording and transcribing with Voice Recorder & Notes Pro and Adobe Premium Pro for the six in-person interviews. The researcher applied Google Meets, Team Meeting to record and transcribe the three virtual interviews. The average time of each audio-recorded semi-structured interview is 2.56 hours, and the average word count of each transcript is 20,460. Along with the nine formal interviews, the researchers applied reflexive journals to analyze data [9].

2.2 Data Analysis
The researcher adopted an iterative human-AI approach to construct meaning not only accurately but also artistically [10], and developed the GOLDEN Framework, acronymed for Ground, Operate, Lead, Deepen, Ethics, and Nature, a six-step ethical and reflexive protocol for qualitative analysis. Figure 1 shows three types of AI tools structurally applied, and Table 1 demonstrates how to operate AI4People Principles across the GOLDEN stages [11]. Across all three works (Heaven Hour, Terrain Triumph, and Human Harmony), data saturation was assessed using theoretical saturation criteria [7] and then using design effect/effective sample

size framework [12], paired with Silverman's [13] data density principles. This dual quantitative–theoretical approach ensures saturation is justified both by demonstrated conceptual closure across axial and selective levels and by statistical sufficiency.

To enhance the credibility of the research, the synthesized member checking in addition to traditional member checking approaches was adopted [14]. As Spall [15] suggests, the researcher invited four experts with knowledge of the research but not participants provide a review of the data collection and analysis to ensure the absence of bias, adding to the study's validity. To further increase the study's trustworthiness, three external auditors from diverse professional backgrounds without knowledge of the study are invited to provide an unbiased examination of the findings, interpretations, and conclusions [16].

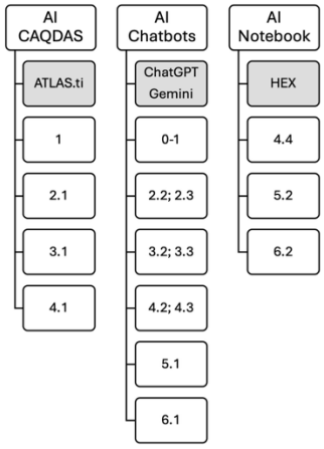


Figure 1. Multi-Modal AI Tools Deployment

Table 1. AI4People Principles across Iterative Analysis Stages

AI4People	Stage	Stage Description
Beneficence	0-1; 1	Data Preparation
Non-maleficence	2.1	AI Open Coding
Autonomy	2.2	Human-AI Coding
Justice	2.3	Preliminary Theoretical Structuring
Non-maleficence	3.1	AI Open Coding
Autonomy	3.2	Human-AI Coding
Explicability	3.3	Data Saturation Assessment & Audit Trail Management
Non-maleficence	4.1	AI Open Coding
Autonomy	4.2	Human-AI Coding
Explicability	4.3; 4.4	Data Saturation Confirmation and Consolidation
Justice	5.1; 6.1	Theoretical Framework Alignment and Theoretical Validation
Beneficence	5.2; 6.2	Quantitative Validation

3. THEORETICAL BACKGROUND

Having identified the four stages of astronauts' mental representations, Suedfeld [17] proclaims that space psychology must undoubtedly maintain its concern with predicting, preventing, and countering adverse reactions. To truly understand what space exploration means to those who engage in it, it is necessary to look at how the explorers respond to challenges. This means that the early myths- the formative years of astronauts' experiences with deliberate decision making are not negligible. Thus, the research applies cognitive maps to seek key decision-making points through a whole-life approach to reflect on how personal needs are satisfied as well as personal values are realized over the life course [18].

3.1 Cognitive Mapping

Edward C. Tolman's pioneering work introduced the concept of cognitive maps - mental representations that individuals develop to understand and navigate their environments [5]. The researcher then integrates the western cognitive mapping theory and the Chinese ancient strategic thoughts to achieve success: Heaven, Earth, and Men.

3.2 Heavenly Hour, Terrain Triumph, Human Harmony (HTH Model)

Mencius (c. 372 – 289 B.C.) highlighted the superlative significance of humans in the trio relationship: tiān shí is inferior to terrain triumph, and dì lì is inferior to rén hé. This ideology reflecting the Chinese essential worldview of the interconnectedness of the divine trinity of the universe has been translated and applied in various fields by Western scholars. Lau [19] interpreted, "opportune times are not as good as advantageous terrain; advantageous terrain is not as good as harmonious people." Van Norden [20] simplified, "The timing is not as important as the terrain, and the terrain is not as important as having unity among the people." To keep the accuracy as well as fluency, the researcher translated with the simplest version, "Heavenly Hour matters. Terrain Triumph more. Human Harmony most."

Then the interview protocol was formulated based on the HTH model to explore astronauts' defining moments, formative environments, and self-identity. Compared to Piaget's learning stages theory [21] and Vygotsky's development of learning zones [22], the researcher gives voices to the participants to define their own terms to inform teaching methodologies and education policies for customized plans for each unique individual learner.

4. RESULTS

Through a rigorously iterative human-AI qualitative analysis from more than 8,751 open codes, the researcher selected 20 emergent themes and refined 92 axial categories across three original theoretical frameworks. Directed by Finlay's [23] "maps' of five variants of reflexive activities, the researcher found that comparative analysis of human-refined and AI-generated outcomes reveals a complementary relationship: human sensemaking provides interpretative depth, while algorithmic processing enables indispensable scalability. The findings indicate that human-AI synergistic workflows support effective model for high-volume textual analysis.

4.1 Heavenly Hour Framework and Theory

The Heavenly Hour framework encapsulates the five pivotal moments shaping astronauts' trajectories: spark-ling interest, honing transition, awakening passion, reaching fulfillment, and pledging commitment, acryonymed SHARP. Often honed through

military training or rigorous academic disciplines, astronauts demonstrate an exceptional ability to recognize and capitalize on opportune moments, reflecting their cognitive precision, time consciousness, and decisiveness. Figure 2 shows how the human- and AI-based coding result theoretical frameworks exhibiting greater interpretive complementarity in three dimensions.

• MONOLOOP astronauts: Top Tier forces dominate on a single path. Inner Compass is unwavering. Agency appears high because the path is clear.

• BILOOP astronauts: Middle Tier becomes critical because Inner Compass must reconcile two domains. Relational Scaffolding supports the dual commitment. Serendipity often determines which loop becomes primary.

• POLYLOOP astronauts: All seven Nexus forces operate at maximum complexity. Each loop generates its own Capital, Trajectory, and Crucibles, while the Inner Compass integrates across loops. This is where the Enabling Tier becomes most visible: navigating multiple loops requires both luck and deliberate choice.

```
DIMENSION 1: TRAJECTORY PATTERN (Lollipop Fractal)
How many study/career loops does the astronaut traverse?

  Monoloop    Biloop       Polyloop
     ○          ∞            ✿
  single     dual path    multiple
  trajectory  convergence  overlapping loops

DIMENSION 2: PIVOTAL MOMENTS (SHARP)
What phenomenological sequence marks each loop?

  S → H → A → R → P
  Spark  Hone  Awaken  Reach  Pledge
  (each loop contains its own SHARP sequence)

DIMENSION 3: FORCES (Nexus)
What 7 forces power each moment within each loop?

  Top:     Capital + Trajectory + Crucible     (56.7%)
  Middle:  Inner Compass + Rel. Scaffolding    (29.2%)
  Enabling: Serendipity + Agency               (13.1%)
```

Figure 2. Heavenly Hour Theoretical Alignment

4.2 Terrain Triumph and A Middle-Range Theoretical Proposition

The researcher upgraded Bronfenbrenner's ecological systems [24] into the ISEE Terrain Triumph framework. ISEE is acryonymed for four ecological systems: the ideological and cultural macrosystem (IDEAL), social and interpersonal mesosystem (SOAP), environment and physical exosystem (EPIC), and emotional and psychological microsystem (EMO). Each system comprises both catalysts and challenges, which yielded eight themes in Terrain Triumph, e.g. EPIC Challenge and EPIC Catalyst. This framework maps out the dynamic interplay between catalysts and their counterparts-challenges, revealing the nuanced progression of astronauts' diversified life stages.

Despite statistical evidenced pass for workability (Figure 3) and hypotheses tests [7] (Figure 4), the researcher suggests a middle-range theory with emerging patterns [25], tentatively termed Death Is Not the End of the Story (DINTEOTS), which characterizes how individuals transform adversity, loss, and challenges into sustained trajectories of growth and meaning-making.

4.3 Human Harmony Framework and Theory

Although the researcher's seven human-coded themes — acronymed as the iNSPIRE framework — diverge in nomenclature from the seven AI-generated themes, human-AI collaboration revealed a notable convergence of independently derived conceptual structures, lending empirical support to AI's socio-emotional attributes [26]. As illustrated in Figure 5, both codlings organize around a shared four-stratum architecture — 元 Core, 基 Foundation, 变 Dynamic, and 合 Integration — functioning as a common interpretive scaffold across 41 human axial categories (268 refined open codes) and 36 AI axial categories (1,145 initial open codes). AI-generated thematic functions of harmony aligned with the researcher's human-centered identity framework, yielding two complementary theoretical formulations: a researcher's reflective All-Mirror-in-One perspective emphasizing unity through mutual interpretation, and the AI-layered 4Strata model describing the operational dynamics of identity across multiple dimensions. This convergence demonstrates how distinct analytic pathways can arrive at shared higher-order abstractions. While human contributions provide philosophical depth and contextual nuance, AI enables empirical rigor through expansive coding, statistical saturation testing, theme-shift quantification, and convergence–divergence metrics.

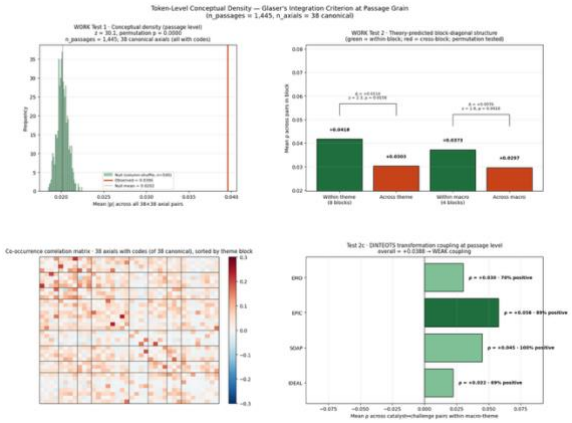


Figure 3. Terrain Triumph Work Test

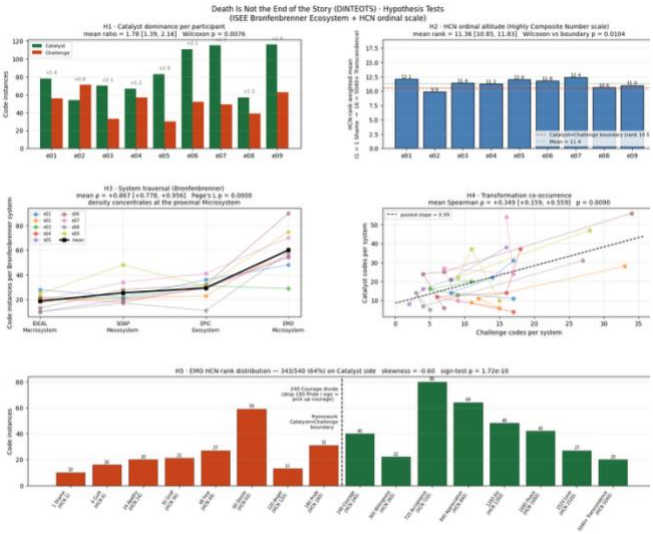


Figure 4. Terrain Triumph Hypotheses Testing

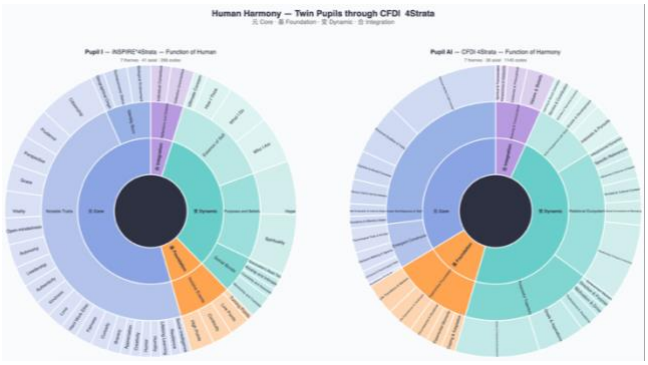


Figure 5. Human Harmony Frameworks: Comparative Sunburst Visualization of Human-Coded iNSPIRE*4Strata (Pupil I) and AI-Coded CFDI*4Strata (Pupil AI)

4.4 HTH Model and Bicameral Theorizing

Drawing on Jaynes's [27] foundational conception of the bicameral mind that cognition operates through two distinct yet interacting chambers, the researcher extends this metaphor into bicameral theorizing, a Human-AI qualitative analytic structure in which researcher-led (I + AI) and AI-surfaced (AI + I) reasoning function as complementary generative modes rather than interchangeable contributions. This bicameral architecture organizes the HTH Model across three frameworks — Heavenly Hour, Terrain Triumph, and Human Harmony — each partitioned into two epistemically distinct halves. The Structural/Locational half anchors identity in its situated, relational, and contextual dimensions, yielding the collaborative constructs Lollipop Fractal, Heart-Land, and All-Mirror-in-One grounded theories. The Transformational/Teleological half orients identity toward purpose, trajectory, and emergent meaning, producing AstroNexus, DINTEOTS, and 4Strata Navigation. This deliberate alternation of analytic authorship across halves reflects a principled division of generative labor.

The overall bicameral structure — structural grounding paired with teleological projection — mirrors the dual functions of identity itself: where one is and where one is becoming. Importantly, the distribution of theorizing labor is not fixed but contingent upon the researcher's personality, cognitive dispositions, and analytic capabilities. In this study, the researcher's inclination toward structural and locational reasoning anchored the first half of each framework, while AI's complementary teleological capacity completed the theorizing arc in Heavenly Hour and Human Harmony. Terrain Triumph, however, illustrates a different configuration: one in which the researcher independently drove the theoretical construction from the ground up, yet AI's contribution to statistical validation through procedures such as the Work test and hypothesis testing remained integral to substantiating its middle-range theory claim. This suggests that bicameral theorizing functions less as a rigid division of labor and more as a dynamic, researcher-sensitive scaffold whose outcomes reflect the unique epistemic posture each inquirer brings to the analytic encounter.

Table 2. Bicameral Theorizing across HTH Model

Framework	Structural/ Locational Half	Transformational/ Teleological Half
Heavenly Hour	Lollipop Fractal (I + AI)	AstroNexus (AI + I)
Terrain Triumph	Heart-Land (I + AI)	DINTEOTS (I + AI)
Human Harmony	All-Mirror-in-One (I + AI)	4Strata Navigation (AI + I)

5. DISCUSSION

Resonating with Silver and Lewins's emphasis on engaging with emerging technology for qualitative analysis [28], this study highlights the irreversibility and uncertainty inherent in intensive analytical processes. It remains an open question whether comparable AI-generated outcomes would have emerged without the researcher's prior extensive validation of over 8,751 open codes and the subsequent development of theoretical frameworks and grounded theories.

Extending ongoing debates on data saturation [29], this study argues that saturation can be reexamined in light of AI-supported analysis through indicators such as information power, data density comparison, effective sample size, empirical saturation evidence, and methodological precedent. Meanwhile, critical questions remain: to what extent can human-AI collaboration reproduce - or transform - the trajectories of qualitative sensemaking?

In summary, this study underscores the importance of human-first iterative engagement prior to AI integration to preserve methodological rigor and to mitigate risks associated with irreversible analytical decisions [30]. Such considerations point toward a broader rethinking of how human agency, technological augmentation, and epistemic responsibility co-evolve in the future of data-intensive research [31].

6. REFERENCES

[1] Lyon, R. F. 2021. The COVID-19 Response Has Uncovered and Increased Our Vulnerability to Biological Warfare, *Military Medicine, 186* (7-8) 193-196. DOI= https://doi.org/10.1093/milmed/usab061

[2] Haddad, H. 2025. Redefining biological weapons in the evolving AI, CRISPR, and Biothreat Landscape, Ethics, *Medicine and Public Health, 33*, 101176. DOI= https://doi.org/10.1016/j.jemep.2025.101176

[3] Meshko, H. M., Meshko, O. I., and Habrusieva, N. V. 2023. The impact of the war in Ukraine on the emotional well-being of students in the learning process, *Journal of Intellectual Disability Diagnosis and Treatment, 11*(1), 55-65. DOI= https://doi.org/10.6000/2292-2598.2023.11.01.7

[4] Chan, C. K. and Hu, W. 2023. Students' voices on Generative AI: Perceptions, benefits, and challenges in Higher Education, *International Journal of Educational Technology in Higher Education, 20*(43). DOI= https://doi.org/10.1186/s41239-023-00411-8

[5] Tolman, E. C. 1948. Cognitive maps in rats and men, *Psychological Review, 55*(4), 189-208. DOI= https://psycnet.apa.org/doi/10.1037/h0061626

[6] Smith, J. A., Jarman, M., and Osborn, M. 1999. Doing interpretative phenomenological analysis, *Qualitative health*

psychology: Theories and Methods, 1(1), 218-240. DOI= https://doi.org/10.4135/9781446217870.n14

[7] Glaser, B. G. and Strauss, A. L.1967. *The discovery of grounded theory: Strategies for qualitative research*. Aldine de Gruyter (Trans), Aldine Transaction, United States. DOI= https://doi.org/10.4324/9780203793206

[8] Guest, G., Bunce, A. and Johnson, L.2006. How Many Interviews Are Enough? An Experiment with Data Saturation and Variability, *Field Methods, 18*(1), 59-82. DOI= https://doi.org/10.1177/1525822X05279903

[9] Guba, E.G. and Lincoln, Y.S. 1989. *Fourth Generation Evaluation.* SAGE Publications.

[10] Williams, M. and Moser, T. 2019. The art of coding and thematic exploration in qualitative research, *International Management Review, 15*(1), 45-55.

[11] Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., Luetge, C., Madelin, R., Pagallo, U., Rossi, F., Schafer, B., Valcke, P., and Vayena, E. 2018. AI4People-An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations. *Minds and machines, 28*(4), 689–707. DOI= https://doi.org/10.1007/s11023-018-9482-5

[12] Kish, L. 1965. Sampling Organizations and Groups of Unequal Sizes. *American Sociological Review, 30*(4), 564–572. DOI= https://doi.org/10.2307/2091346

[13] Silverman, B. W. (2018). Density Estimation for Statistics and Data Analysis. (1st ed.). Routledge. DOI= https://doi.org/10.1201/9781315140919

[14] Birt, L., Scott, S., Cavers, D., Campbell, C., and Walter, F. 2016. Member Checking: A Tool to Enhance Trustworthiness or Merely a Nod to Validation?. *Qualitative health research, 26*(13), 1802–1811. DOI= https://doi.org/10.1177/1049732316654870

[15] Spall, S. 1998. Peer debriefing in qualitative research: Emerging operational models. *Qualitative inquiry, 4*(2), 280-292. DOI= https://doi.org/10.1177/107780049800400208

[16] Miller, D. L. 1997. One Strategy for Assessing the Trustworthiness of Qualitative Research: Operationalizing the External Audit. DOI= https://eric.ed.gov/?id=ED411268

[17] Suedfeld, P., Brcic, J., Johnson, P. J., and Gushin, V. 2015. Coping strategies during and after spaceflight: data from retired cosmonauts. *Acta Astronautica, 110*, 43-49. DOI= https://doi.org/10.1016/j.actaastro.2012.04.039

[18] Litano, M. L. and Major, D. A. 2015. Facilitating a whole-life approach to career development: the role of organizational leadership, *Journal of Career Development, 43*(1), 52–65. DOI= https://doi.org/10.1177/0894845315569303

[19] Mencius. 2003. *Mencius* (Lau, D. C. Trans), Penguin Classics, London.

[20] Mencius. 2008. *Mengzi with Selections from Traditional Commentaries.* (Van Norden, B. W. Trans), Hackett Publishing, Cambridge.

[21] Piaget, J.1977. *The Development of Thought: Equilibration of Cognitive Structures* (Rosin, A. Trans), Viking Press, New York.

[22] Vygotsky, L. S. 1978. *Mind in Society: The Development of Higher Psychological Processes*, Harvard University Press, Cambridge.

[23] Finlay, L. 2002. Negotiating the swamp: The opportunity and challenge of reflexivity in research practice, Qualitative Research, 2(2), 209-230. DOI= https://doi.org/10.1177/146879410200200205

[24] Bronfenbrenner, U. 1979. *The Ecology of Human Development: Experiments by Nature and Design*, Harvard University Press, United Kingdom.

[25] Merton, R.K. 1968. *Social Theory and Social Structure* (Enlarged ed.), The Free Press, New York.

[26] Kolomaznik, M., Petrik, V., Slama, M., and Jurik, V. (2024). The role of socio-emotional attributes in enhancing human-AI collaboration. *Frontiers in Psychology, 15*. DOI= https://doi.org/10.3389/fpsyg.2024.1369957

[27] Jaynes, J. 1976. *The Origin of Consciousness in the Breakdown of the Bicameral Mind.* Houghton Mifflin. (Reprinted in 1982).

[28] Silver, C. and Lewins, A. 2014. *Using software in qualitative research.* SAGE Publications Ltd. DOI= https://doi.org/10.4135/9781473906907

[29] Wutich, A., Beresford, M., and Bernard, H. R. 2024. Sample Sizes for 10 Types of Qualitative Data Analysis: An Integrative Review, Empirical Guidance, and Next Steps. *International Journal of Qualitative Methods, 23*. DOI= https://doi.org/10.1177/16094069241296206

[30] Sunstein, C. R. 2018. Irreparability as Irreversibility, *Supreme Court Review,* Harvard Public Law Working Paper. DOI= https://ssrn.com/abstract=3113571 or http://dx.doi.org/10.2139/ssrn.3113571

[31] Teilhard de Chardin, P. 1965. *The Phenomenon of Man* (with an introd. by Julian Huxley. translation by Bernard Wall), Harper & Row.

Columns on Last Page Should Be Made as Close As Possible to Equal Length

Beyond Seeing: Alternative Representations for Image-Dependent Mathematics

Ethan Croteau
Worcester Polytechnic Institute
Worcester, MA, USA
ecroteau@wpi.edu

Neil Heffernan
Worcester Polytechnic Institute
Worcester, MA, USA
nth@wpi.edu

ABSTRACT

Multimodal large language models (MLLMs) are increasingly proposed for tutoring, feedback, and other educational uses, yet they remain brittle on mathematics problems that require interpreting diagrams, graphs, number lines, and other figures. This dissertation investigates whether many of these failures are fundamentally representational: models may have the mathematical knowledge needed to solve a problem, but fail because the visual information is not encoded in a sufficiently faithful form for downstream reasoning. To address this challenge, I study alternative representations of image-dependent mathematical content, including high-quality alt-text, structured textual descriptions, and executable Python reconstructions of the original figure. I first analyze difficult image-dependent middle-school mathematics problems to identify recurring visual bottlenecks in MLLM performance. I then evaluate whether alternative representations improve not only problem solving, but also educationally meaningful tasks such as hint generation, explanation, scaffolding, accessibility support, and related-problem creation. The dissertation contributes both an empirical account of representational failure in multimodal educational AI and a design framework for making visual mathematical content more usable in learning environments.

Keywords

multimodal large language models, mathematics education, visual representations, accessibility, educational AI

1. INTRODUCTION

Multimodal large language models (MLLMs) are increasingly being explored for tutoring, feedback, problem solving, and other educational applications. In mathematics education, however, many tasks depend on diagrams, graphs, geometric figures, tables, and other visuals that provide essential problem context. These are not simply text questions with accompanying illustrations; rather, the visual representation often contains information required to determine the correct answer or construct a valid solution. Such visually grounded tasks also arise in classroom-authentic tutoring and assessment settings, including platforms such as ASSISTments [3].

This challenge is especially important for educational data mining and AI in education because model success on visually grounded tasks depends on more than symbolic reasoning alone. A model may appear mathematically capable while still failing to count objects accurately, read a scale, bind labels to the correct visual elements, or preserve spatial relations needed for reasoning. As a result, strong performance on broad multimodal benchmarks can obscure weaknesses that matter in authentic learning environments [4]. Recent work on authentic middle-school image-dependent mathematics suggests that such failures are often driven by visual misreading rather than symbolic mathematics alone [2].

My dissertation is motivated by the hypothesis that many of these failures are fundamentally representational. In other words, the model may possess the mathematical reasoning required once the relevant information is available, but fail because the original image is not transformed into a sufficiently faithful internal or external representation for downstream reasoning. This dissertation therefore asks whether alternative representations of visual mathematical content can make image-dependent mathematics more usable for MLLMs in educational settings. Rather than treating the original image as the only viable input, I investigate forms such as high-quality alt-text, structured textual descriptions, and executable Python scripts that reconstruct the figure while preserving its mathematical structure.

2. PROBLEM AND RESEARCH GAP

MLLMs have become a central topic in educational technology research, with growing interest in their use for tutoring, automated feedback, accessibility, and content generation. At the same time, prior work in mathematics education and the learning sciences has shown that representations are not incidental to learning; they shape interpretation, inference, and strategy selection [1, 5]. This makes image-dependent mathematics an important testbed for educational AI.

Existing multimodal work often focuses on benchmark accuracy or broad demonstrations of capability [4]. While this work is valuable, it does not fully explain why visually grounded mathematics problems remain difficult in authentic educational settings, especially when success depends on

Ethan Croteau, and Neil Heffernan. Beyond Seeing: Alternative Representations for Image-Dependent Mathematics. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 832–834. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040024

precise visual interpretation. In such cases, failure may reflect not only reasoning limitations but also bottlenecks in how task-relevant visual information is represented.

A second gap is that most work treats the original image as the primary or only form of visual input. In educational systems, however, the same mathematical content can often be expressed in multiple forms. A diagram may be represented as alt-text, as a structured description of entities and relations, or as an executable program that reconstructs the figure. These alternatives may preserve different aspects of the original content and may vary in how well they support reasoning, explanation, accessibility, and content generation. Accessibility guidance emphasizes that such alternatives must preserve task-relevant structure, not merely summarize appearance [6, 7]. Despite this promise, representation choice has not been studied systematically for image-dependent mathematics in educational contexts.

3. RESEARCH QUESTIONS AND THEORETICAL FRAMING

The dissertation is organized around three questions:

RQ1: What kinds of image-dependent mathematics problems remain especially difficult for MLLMs, and what recurring visual bottlenecks contribute to these failures?

RQ2: To what extent can alternative representations of visual mathematical content improve MLLM understanding and problem solving on image-dependent tasks?

RQ3: Which forms of alternative representation are most useful for downstream educational tasks such as hint generation, explanation, scaffolding, accessibility support, and the creation of related practice problems?

These questions reflect a dissertation arc that moves from *diagnosis* to *intervention* to *educational application*. The central theoretical claim is that representation is not merely a delivery format for visual information; it is part of the mechanism by which a model can or cannot reason successfully about a task. Accordingly, this dissertation treats alternative representation as both an analytic lens for explaining multimodal failure and a design space for building more useful educational AI systems.

4. CURRENT PROGRESS

A substantial portion of the dissertation's diagnostic phase is already complete. Earlier work on authentic middle-school mathematics problems in which the image is required for solving showed that even strong contemporary MLLMs remain brittle when the figure contains task-critical information not recoverable from text alone. Across repeated evaluations with and without the figure, these studies suggested that visual misreading is a dominant failure mode, with recurring errors involving counting, measurement, label binding, and geometric relation extraction.

This prior result motivates the dissertation's shift from diagnosing image-dependent failure to studying representation as an intervention. Rather than viewing each failure only as evidence of insufficient model capability, I ask whether the same visual information can be expressed in forms that are more accessible to the model while remaining faithful to the underlying mathematics. To explore this possibility, I conducted early experiments using prompting to generate Python scripts that reconstruct selected images from difficult mathematics problems. These scripts, typically implemented with plotting or drawing libraries, serve as executable representations of the original figure. Unlike a static image, they can be inspected, edited, parameterized, and potentially reused for multiple educational purposes. Building on these experiments, I have also begun implementing an early-stage web prototype that converts structured math-task inputs into schema-constrained visual specifications, deterministic scene representations, executable Matplotlib code, and SVG outputs. The prototype logs intermediate artifacts and run-level metadata, enabling inspection and comparison of alternative representations, but it should currently be understood as research infrastructure rather than as a fully evaluated educational intervention.

These early experiments suggest that executable representations may have value beyond answer production. Because they expose the structure of a figure in a machine-readable and human-interpretable form, they may support hint generation, explanation, scaffolding, and the creation of related-but-not-identical problems. Likewise, alt-text and structured descriptions may improve accessibility for learners who cannot fully access the original image while also providing models with more explicit task-relevant information.

5. PROPOSED METHODOLOGY

The next phase of the dissertation will systematically compare multiple representations of the same image-dependent mathematical content. I currently anticipate studying at least four representation conditions: (1) the original image, (2) high-quality alt-text, (3) structured textual descriptions designed to preserve task-relevant visual relations, and (4) executable Python reconstructions of the original figure. Hybrid conditions that combine these forms may also be explored.

These representations will be evaluated across several educationally meaningful tasks. The first is direct problem solving on image-dependent mathematics items. Here I will measure whether alternative representations improve model performance and reduce specific types of visual failure. The second is hint and explanation generation, where the goal is not simply to produce a correct answer but to generate support that is instructionally useful and grounded in the relevant visual structure. The third is scaffolding and accessibility support, including whether alternative representations help make visual mathematical information more understandable to both models and learners. The fourth is related-problem generation, where executable or structured representations may enable the creation of new problems that are similar in concept but not identical in surface form.

Methodologically, I plan to combine quantitative evaluation with qualitative analysis. Quantitative measures will include problem-solving accuracy, consistency across repeated trials, and performance differences across representation conditions. Qualitative analyses will examine which visual bottlenecks persist, which are reduced, and what kinds of down-

stream outputs become more educationally useful under alternative representations. For generative tasks such as hints or related-problem creation, expert review may also be used to assess mathematical fidelity, pedagogical usefulness, and alignment with the intended concept.

6. EXPECTED CONTRIBUTIONS AND RELEVANCE TO EDM

This dissertation aims to contribute at three levels. First, it will provide an empirical characterization of the visual bottlenecks that make some image-dependent mathematics problems especially difficult for MLLMs. Second, it will contribute a methodological framework for studying alternative representations of visual mathematical content and their effects on downstream reasoning. Third, it will offer practical guidance for educational applications, including tutoring, accessibility support, grounded explanation, and content generation.

The work is relevant to EDM in several ways. It studies how representational choices shape model behavior on authentic educational tasks; it addresses human factors and explainability by examining why models fail on visually grounded items; and it connects to the broader EDM interest in comparing human and artificial intelligence on learning-relevant tasks. More broadly, the dissertation is intended to speak to both learning sciences and computer science perspectives by combining attention to representational meaning, learner accessibility, and instructional usefulness with rigorous evaluation of model performance.

7. FEEDBACK SOUGHT

I would especially value feedback in three areas. First, I seek guidance on how best to position the dissertation's central contribution: as a study of multimodal failure, a study of alternative representation, or a broader effort to support educational uses of visual mathematical content. Second, I seek feedback on the design of the representation comparisons, including which representation conditions and evaluation tasks should be prioritized for the dissertation. Third, I would value advice on how to strengthen the link between EDM, AIED, and learning sciences perspectives when evaluating educational usefulness beyond answer accuracy.

Overall, this dissertation aims to understand why "seeing" is often not enough for current MLLMs on image-dependent mathematics, and how alternative representations may help bridge that gap in ways that are meaningful for learners, educators, and educational AI systems.

8. ACKNOWLEDGMENTS

We would like to thank NSF (e.g., 2118725, 2118904, 1950683, 1917808, 1931523, 1940236, 1917713, 1903304, 1822830, 1759229, 1724889, 1636782, & 1535428), IES (e.g., R305N210049, R305D210031, R305A170137, R305A170243, R305A180401, & R305A120125), GAANN (e.g., P200A180088 & P200A150306), EIR (U411B190024 & S411B210024), ONR (N00014-18-1-2768), NIH (R44GM146483), and Schmidt Futures. None of the opinions expressed here are those of the funders.

9. REFERENCES

[1] S. Ainsworth. Deft: A conceptual framework for considering learning with multiple representations. *Learning and Instruction*, 16(3):183–198, 2006.

[2] E. Croteau and N. Heffernan. Seeing is solving: MLLMs, reasoning, and refusal in visual math. *Journal of Educational Data Mining*, 18(1):244–285, 2026.

[3] N. T. Heffernan and C. L. Heffernan. The ASSISTments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[4] P. Lu, H. Bansal, T. Xia, J. Liu, C. Li, H. Hajishirzi, H. Cheng, K.-W. Chang, M. Galley, and J. Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *The Twelfth International Conference on Learning Representations*, Vienna, Austria, 2024. OpenReview.net.

[5] R. E. Mayer. *Multimedia Learning*. Cambridge University Press, Cambridge, UK, 3rd edition, 2020.

[6] W3C Accessibility Guidelines Working Group. Web content accessibility guidelines (wcag) 2.2. W3C Recommendation, Dec. 2024. Latest version: `https://www.w3.org/TR/WCAG22/`.

[7] W3C Web Accessibility Initiative. Images tutorial: Complex images. `https://www.w3.org/WAI/tutorials/images/complex/`, 2022. Updated 17 January 2022. Accessed 10 April 2026.

Cognitive Agent Compilation for Explicit Problem Solver Modeling

Hyeongdon (Doni) Moon
Carnegie Mellon University
donim@andrew.cmu.edu

John Stamper
Carnegie Mellon University
jstamper@andrew.cmu.edu

Carolyn Rose
Carnegie Mellon University
cp3a@andrew.cmu.edu

ABSTRACT

Large language models (LLMs) are widely used for tutoring, feedback generation, and content creation, but their broad pretraining makes them hard to constrain and poor substitutes for controllable learners. Educational systems often require inspectable and editable knowledge states: educators want to know what a system assumes the learner knows, and learners benefit when the system can justify actions in terms of explicit skills, misconceptions, and strategies. Inspired by cognitive architectures, we propose Cognitive Agent Compilation (CAC), a framework that uses a strong teacher LLM to compile problem-solving knowledge into an explicit target agent. CAC separates (i) knowledge representation, (ii) problem-solving policy, and (iii) verification and update rules, with the goal of making bounded problem solving more inspectable and editable in educational settings. We present an early proof of concept implemented with Small Language Models that surfaces key design trade-offs, particularly between explicit control and scalable generalization, and positions CAC as an initial step toward bounded-knowledge AI for educational applications.

Keywords

Cognitive Architecture, Knowledge Representation, Learner models, Explainable AI, Intelligent Tutoring systems

Hyeongdon Moon, John Stamper, and Carolyn Rosé. Cognitive Agent Compilation for Explicit Problem Solver Modeling. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 835–835. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040042

Doctoral Consortium Application: Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression

Megan Caruso
University of Colorado Boulder
Boulder, CO, USA
Megan.Caruso@colorado.edu

ABSTRACT

Understanding how learning unfolds moment-to-moment remains a central challenge in capturing real-time learning processes. Much of the field has focused on aggregated behavioral signals or prediction of outcomes, often overlooking the fine-grained temporal dynamics through which learning processes emerge. This work introduces a multimodal, temporally explicit framework for modeling reading comprehension by integrating eye tracking, EEG, and textual features at the level of individual fixations during naturalistic reading.

We analyzed data from 116 participants reading five long expository texts while eye movements and EEG were recorded concurrently. We then constructed fixation-level, multivariate time series that included gaze (fixation duration), neural responses (fixation-related potentials), and lexical properties (surprisal and word frequency). We applied multivariate vector autoregression (mlVAR) to model how these signals unfold over time, both within fixations (contemporaneous) and across fixations. We also examined how comprehension, reading time, prior knowledge, and mind wandering covary with network connectivity.

Across models, robust lexical-gaze relationships were observed, with surprisal positively associated with fixation duration and word frequency negatively associated with fixation duration, both within and across fixations. These patterns extend classic findings from reading research into a temporal, multivariate setting. Neural associations were more sensitive to modeling choices, suggesting that brain-behavior coupling may depend on how temporal structure is represented. At the participant level, comprehension and reading time were associated with differences in contemporaneous network connectivity, indicating that variation in learning outcomes corresponds to differences in multimodal coordination.

Overall, this work positions temporal network modeling as a promising direction. By integrating multimodal signals at a fine temporal scale in naturalistic settings, it provides a framework for studying learning as a dynamic, evolving process rather than a static outcome. Beyond reading comprehension. This approach offers a generalizable methodology for modeling temporal dependencies across modalities in a wide range of educational contexts.

Keywords

Multimodal modeling, eye tracking, EEG, reading comprehension, temporal modeling

Megan Caruso. Doctoral Consortium Application: Exploring Temporal Dynamics Underlying Eye Gaze, EEG, and Text Characteristics during Naturalistic Reading with Multi-Level Vector Autoregression. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 836–836. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039986

Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations

E. Margaret Perkoff[1]
eperkoff@upenn.edu

Megan Lyons[2]
lyonsm3@chop.edu

Alexandria Mulqueen[2]
mulqueena@chop.edu

Ryan S. Baker[3]
ryan.baker@adelaide.edu.au

Alan Gee[4]
alan.gee@amiralearning.com

Jaclyn Ocumpaugh[5]
jocumpau@central.uh.edu

Adam Porsch[4]
adam.porsch@amiralearning.com

Neville Ryant[6]
nryant@ldc.upenn.edu

Julia Parish-Morris[2]
parishmorrisj@chop.edu

Lyle Ungar[1]
ungar@upenn.edu

[1]University of Pennsylvania [2]Children's Hospital of Philadelphia [3]Adelaide University
[4]Amira Learning [5]University of Houston [6]Linguistic Data Consortium

ABSTRACT

The prevalence of online reading platforms has surged along with improvements to automatic speech recognition (ASR) systems. These platforms facilitate personalized reading instruction, but like all learning systems, work best when students find them engaging. The Educational Data Mining (EDM) community has made great efforts to detect student engagement, but these efforts have largely been confined to school and lab environments without intentionally including neurodiverse populations. In the present study, we evaluate engagement levels in children with attention deficit hyperactivity disorder (ADHD, n=55), autism (n=38), both autism and ADHD (n=38) and with no known clinical diagnoses (n=22) as they interact with a digital literacy learning system from their homes. Engagement was measured using parent report, and acoustic features were extracted from audio recordings of the students while they interacted with the Amira literacy learning system. Our results indicate that acoustic measures related to temporal features, pitch, and vocal quality are related to engagement levels across all clinical groups. We also identify key temporal features that vary across different clinical groups. These findings demonstrate the validity of using non-invasive acoustic features to measure engagement levels during reading comprehension tasks at home. However, interaction effects also highlight the importance of taking into account children's clinical diagnosis when measuring engagement.

Keywords

Reading Engagement, Online Reading Comprehension, Neurodivergent

Margaret Perkoff. Engagement-Driven Differences in Acoustic Features During Online Reading Comprehension Conversations. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 837–850. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039929

1. INTRODUCTION

Digital reading platforms have emerged as a promising way to provide personalized literacy instruction to students, both at home and at school. These platforms guide the student through level-appropriate passages where they can read text out loud, receive real-time interventions, and respond to reading comprehension questions[69]. Compared to traditional instruction, Intelligent tutoring systems (ITSs) for reading can even produce larger learning gains [65, 70], even among students with learning differences that sometimes interfere with reading comprehension, such as Autism Spectrum Disorder (ASD) [39]. However, meta-reviews of literacy applications have demonstrated that they are most effective when (1) the systems are highly interactive and (2) children use them with higher intensity (at least 75 minutes per week) and duration (at least 3 months) [39, 69, 70]. This leads to a direct need to understand how engaged students are with a platform in order to provide effective instruction and keep them coming back for regular sessions.

Reading engagement is often defined as the student's active involvement with reading [32], and it is sometimes distinguished from motivation by considering motivation as a mindset that is necessary in order to deeply engage with reading [2]. Although exact definitions of reading engagement vary, it is often described as including cognitive, affective, and behavioral dimensions [41]. Reading engagement is not only indicative of children's enjoyment, but is also linked to improved reading abilities and reading comprehension [6, 31, 41, 63].

Despite an abundance of research on engagement measures, there is no standardized method for measuring student's engagement with online reading platforms. Some conversational agent based literacy tools operationalize verbal engagement in terms of child response length [33] or semantic features including topic relevance and intelligibility [68]. However, these measures are proxies for the actual construct, which is likely to involve a range of cognitive, behavioral, and affective experiences [26]. Moreover, they do not account for key acoustic elements, including those that may be more in-

dicative of a students' affective experience with the reading activity [44].

In this study, we explore the use of acoustic features to detect reading engagement among neurodivergent students with different clinical diagnoses, including Autism and Attention Deficit Hyperactivity Disorder (ADHD). We examine the use of acoustic features as they can be easily extracted by voice-based reading software without the use of specialized equipment that may be difficult to acquire for use in either the home or classroom. We evaluate the interaction between clinical diagnosis and engagement levels by analyzing acoustic features that indicate voice quality, speaking rates, and prosody, and which have been associated with engagement in prior research [34, 44].

To our knowledge, this is one of the first studies to focus on understanding which acoustic features most effectively model engagement during digital reading instruction, and one of the first in the EDM community to consider how clinical diagnoses might affect those modeling efforts. Our results indicate that temporal acoustic features are robust indicators of engagement across groups in our study, but other acoustic features (e.g., variation of voiced segment length) are diagnosis-sensitive. With this work, we aim to contribute to a better understanding of how engagement manifests differently in diverse populations of learners while also putting forward tools to help detect these differences, provide personalized support, and ultimately improve outcomes.

2. BACKGROUND AND RELATED WORK

Research on student engagement defines it across many dimensions, including behavioral, emotional, and cognitive domains [26]. Real-time models of student engagement have become a common way for researchers in the Educational Data Mining community to study engagement at scale, including both affective engagement and cognitive engagement. Such models have been trained on a variety of label types, including those produced from text-based video [76], video data [10, 36], speech [71, 44] physiological signals including skin conductance data [12, 77], posture data [56], and self-report [74, 18]; and from classroom observations [5, 50].

Much of this work has focused on affective engagement [13], although behavioral and cognitive engagement [30, 42] are also modeled. Such models have been particularly important for researchers studying affect dynamics [20, 51]. They have also been useful for making short-term and long-term predictions about student learning outcomes [27]. However, given the rich history of definitions involved with studying student engagement in the broader research community [26], the EDM community continues to explore novel ways to capture this construct.

2.1 Detecting Reading Engagement in Learning Systems at Home

One particularly difficult challenge in this space is appropriately creating high quality engagement labels for students who use learning software at home. Many sensors used to detect engagement can be highly invasive (e.g., video), while others are difficult to use in the home environment (e.g., posture sensors). Meanwhile, interaction-based detectors [50] rely on classroom observation or self-report data to provide the labels. Although these detectors can certainly be applied to students in the home environment, it is difficult to verify whether detectors trained on labels from a classroom environment would generalize to students' home environments. Additionally, self-reports are susceptible to self-presentation effects and to differences in students' ability to gauge their own level of participation [21, 28].

A few studies have sought to leverage measures that could be more easily collected in a home environment. For example, models based on student response times leverage item response theory to distinguish between engaged and disengaged students in online settings [7, 35]. These could be used in home environments, even if they might need to be renormed for different clinical populations. Another approach is to measure student talk time while engaging with an online tutor[17]. Some existing reading software measures verbal engagement based on the number of turns with a conversational agent or word counts [33]. Xu et al. [68] evaluated the nuances of children's responses to comprehension conversations asked by conversational agents versus humans via five aspects of verbal engagement: lexical diversity, intelligibility, productivity, topic relevance, and accuracy [66, 64]. Similarly, Gorgun et al. leveraged linguistic features in online discussion posts to estimate engagement based on third-party manual coding [30]. Their scheme defined four attributes of cognitive engagement: interactive, constructive, active, and social.

In addition to semantic-based engagement features, some researchers have explored the use of acoustic features in the online setting. Litman et al. [44] evaluated prosodic cues of disengagement and uncertainty while students engaged with a physics tutor. Their research demonstrated that when students are disengaged, they respond significantly more slowly than engaged students. Disengaged students were also found to have lower maximum and mean pitch values than their engaged peers. Furthermore, disengaged students demonstrated more steady pitch values as well as lower levels of energy. Our study builds upon these findings, with the intention of exploring whether they apply to non-neurotypical learners.

2.2 Reading Engagement in Individuals with ASD and ADHD

As our ability to detect engagement in online settings has advanced, there have been increasing calls for better understanding for a more diverse range of learners [52], and subsequent research has shown that additional demographic information can improve models. For example, Karumbaiah et al. [38], shows that some behavioral engagement measures (i.e., help-seeking) are better understood when school-level demographics are taken into account. Likewise, other research has shown that we can improve our models of mind-wandering by accounting for neurodivergence [67].

Accounting for clinical diagnoses in engagement research is particularly important since certain conditions are characterized by developmental challenges that begin to emerge during early childhood and impact multiple life domains (e.g. school, home) across the lifespan [49]. These increasingly prevalent conditions are characterized by factors that

Table 1: Demographic and clinical characteristics of participant groups. Number of activities signifies the number of comprehension conversations that were included from each participant subset.

Clinical Group	Mean Age (SD)	Male	Female	Ethnicity	Activities
ADHD	8.527 (1.386)	35	20	Not Hispanic or Latino (48)	305
				Hispanic or Latino (7)	
ASD	8.345 (1.632)	14	15	Not Hispanic or Latino (22)	129
				Hispanic or Latino (7)	
ASD+ADHD	8.316 (0.904)	22	16	Not Hispanic or Latino (34)	169
				Hispanic or Latino (4)	
Not ASD or ADHD	8.409 (1.054)	11	11	Not Hispanic or Latino (18)	103
				Hispanic or Latino (4)	
Overall	8.417 (1.276)	82	62	Not Hispanic or Latino (122)	709
				Hispanic or Latino (22)	

could interfere with our measurement of student engagement [14]. For example, individuals with Autism Spectrum Disorder (ASD) face persistent challenges in social communication, as well as restricted or repetitive patterns of behaviors, interests, and activities [4]. Attention-deficit/hyperactivity disorder (ADHD) is a neurodevelopmental condition diagnosed in about 1 in 11 (10.5%) of U.S. school-aged children [15] and is distinguished by persistent patterns of inattention and/or hyperactivity-impulsivity. While autism and ADHD are distinct neurodevelopmental conditions, they frequently co-occur. 50-70% of people with ASD meet diagnostic criteria for ADHD [58] and 12.5% of individuals with ADHD meet diagnostic criteria for ASD [73]. This co-occurrence (colloquially referred to as AuDHD) could result in additional nuances in learning behavior that would further complicate accurate measurement of engagement in this population.

Although ASD and ADHD often co-occur with disabilities that directly impair reading skills (e.g. dyslexia; [11]), various clinical features specific to ASD and ADHD can also contribute to challenges with reading engagement and comprehension for children with either condition. Specifically, impairments in social communication, word recognition, and oral language may compromise many autistic children's reading comprehension skills [57, 46, 40]. Furthermore, some early autistic readers exhibit a striking discrepancy between their reading comprehension and word recognition skills such that their reading comprehension skills lag behind their growing proficiency in word recognition [16]. Reading comprehension may also pose difficulty for children with ADHD. Working memory impairments that are characteristic of ADHD can constrain reading comprehension and limit children's ability to recall central ideas from text directly after reading it [47]. Additionally, on average, children with ASD and/or ADHD may struggle to maintain sustained attention on tasks they are not interested in or motivated to complete. This suggests that children with ASD and/or ADHD may exhibit reduced engagement when reading texts they did not choose [37] or that is not related to their interests [22]. Thus, several clinical characteristics associated with ASD and ADHD may exert a powerful influence on reading engagement and comprehension. With the rising use of digital literacy platforms aimed at supporting children's reading skills, there is a critical need to accurately measure how children with ASD and/or ADHD engage with and comprehend the texts they read on these platforms.

3. PRESENT STUDY

3.1 Digital Literacy Platform

Amira [1] is an online, AI-powered reading platform that delivers real-time, adaptive literacy instruction. The virtual tutor,named Amira, listens to students as they read aloud, identifies reading miscues or skill gaps, and provides brief, real-time instructional supports (micro-interventions) aligned to the Science of Reading and tailored to the student's ability level and performance history. During comprehension dialogue activities, Amira prompts students with questions designed to assess and deepen skills that are aligned to grade-level Reading Comprehension standards. For students in grades 3 and above—where the cognitive and language demands are appropriate—Amira may initiate a comprehension dialogue micro-intervention rather than using the traditional multiple-choice format. These dialogues use open-ended prompts to probe the student's ability to demonstrate the skill, capturing explanation, reflection, or elaboration. Dialogues are modeled after authentic classroom questioning. The student responds either orally or by typing a response, and Amira uses advanced speech recognition and natural language processing (NLP) to capture and compare student responses against a detailed pre-defined rubric.

Each conversation can extend up to two turns of dialogue. The dialogue follows the structure of an opening prompt from the tutor, student response, a follow-up tutor prompt, student response, and final tutor response. A response timer is activated after the initial prompt is delivered, and students are given 10 seconds to begin their response to that turn. If no response is detected within this timeframe, the system will automatically deliver scaffolded support (a hint and a re-prompting of the same question). If any response attempt is detected, the student has 60 seconds in total to respond to the prompt on each turn. There is a visual timer present during the interaction to convey this per-turn time limit. The follow-up prompt is intended to scaffold the student towards further addressing rubric criteria that they did not adequately cover in their initial answer.

All response data—including ASR transcripts, timing, accuracy of the response based on the skills being taught, and behavioral markers such as hesitation or repetition—is logged for subsequent analysis, enabling researchers to investigate both dialogue engagement and comprehension skill performance within a dialogic reading context. For this study, we focus solely on the audio recordings and transcribed stu-

Table 2: Subset of acoustic features from eGeMAPS [23] use in analysis grouped by category.

Domain	Features
Temporal	Loudness Peaks Per Sec; Voiced segment length (s: mean, std. dev.); Unvoiced segment length (s: mean, std. dev.)
Energy	Harmonic-to-Noise ratio (mean, std. dev.)
Frequency	Fundamental Frequency (F0: mean, std. dev., 20th, 50th, 80th percentiles); F0 mean rising slope; F0 mean falling slope; jitter (mean, std. dev.)
Spectral	Alpha ratio, voiced (mean, std. dev.); Alpha ratio, unvoiced (mean); Hammarberg index, voiced (mean, std. dev.); Hammarberg index, unvoiced (mean, std. dev.); Spectral flux, voiced (mean, std. dev.); Spectral flux, unvoiced (mean); Spectral slope 0–500 Hz (mean, std. dev.); Spectral slope 500–1500 Hz (mean, std. dev.)

dent responses from the comprehension dialogues described above.

3.2 Methods

3.2.1 Participants and Data Collection

Participants who consented to be contacted for research studies were recruited via the Center for Autism Research (CAR) at the Children's Hospital of Philadelphia (CHOP). CHOP's Electronic Health Records (EHR) database. 1000 emails per week were sent out to CHOP patients based on certain selection criteria, including (a) diagnosis of autism spectrum and/or attention deficit/hyperactivity spectrum disorder or not, (b) age 5-12 years, and (c) in the "learning to read" phase according to parent report. 144 children aged 5-12 years with a diagnosis of autism (N=29), ADHD (N=55), autism + ADHD (N=38), or no diagnoses of autism or ADHD (N=22) consented to participate in a 7-week at-home reading intervention pilot study. Once enrolled, participants were instructed to read four to five stories per week on Amira , or for at least 30 minutes. Caregivers completed surveys providing basic demographic information and feedback on their child's experience with Amira. Full demographic details for participants can be seen in Table 1. This study was overseen by the Institutional Review Boards at CHOP and MDRC.

3.2.2 Acoustic Features and their Extraction

For each of the comprehension activity turns, we extract acoustic features based on the extended Geneva Minimalistic Acoustic Parameter set (eGeMAPS) [23] using the OpenSMILE toolkit[25]. eGeMAPS is considered a high quality acoustic feature set for use in emotion recognition tasks [19], which are closely related to the engagement identification question. eGeMAPS consists of 25 low-level acoustic features comprising four key categories related to these tasks: prosody, temporal, pitch, and spectral features. Acoustic features are extracted at regular intervals (e.g, every 10 ms) from speech samples and the resulting feature time series is mapped to fixed-dimensional feature vectors by applying one or more statistical functions (e.g., mean, standard deviation, percentiles), resulting in 88 features[1] for each student response turn. For frequency values, we focus solely on the fundamental frequency F_0 which has been used in prior studies as a measure of distinguishing pitch values between neurodivergent and neurotypical students [53]. This results in a subset of 49 acoustic features that we use for our analysis as seen in Table 2.

Table 3: Breakdown of the study dataset based on values with the parent-reported engagement labels: not engaged (NE), Somewhat Engaged (SE), and Very Engaged (VE). Only students with audio recording files are included in the count.

Clinical Group	Not Engaged	Somewhat Engaged	Very Engaged
ADHD	1	17	26
ASD	0	8	13
ASD+ADHD	1	12	8
Not ASD or ADHD	0	4	8

Prosodic features attempt to characterize the expressive qualities of speech including voice quality, rhythm, and intonation that convey emotion and emphasis; examples include jitter, shimmer, and harmonic noise ratio measures.

Frequency features attempt to characterize how high or low the voice sounds and how it varies over time, capturing both the fundamental frequency (F_0, perceived as pitch) and the resonant frequencies of the vocal tract (formants). As mentioned above, we only analyze fundamental frequency values in this work. These features were crucial in the work of Litman et al. [44] which showed that pitch was statistically significant for discriminating between engaged and disengaged students.

Temporal features attempt to characterize speech timing, including speaking rate and pause frequency. We consider the rate of speech, number of pauses based on loudness values, and variations in voiced and unvoiced segment lengths, which have been used in prior work discriminating between student engagement levels [44].

Spectral features are features that correspond to the spectral shape, such as slope, which are frequently used for affect recognition [23, 61]. These also include the first four Mel-Frequency Cepstral Coefficients (MFCCs), which have a documented association with valence and level of interest [24].

For further details regarding feature extraction (e.g., frame step, window size, MFCC filterbank/DCT parameters, functionals applied to each acoustic feature), consult [19] and [23].

3.2.3 Parent Reported Engagement (PRE) Labels

Following seven weeks of the at-home intervention, parents were given a survey to gather information about their child's

[1]Not all statistical functions are applied to all low-level acoustic features.

experience with the literacy learning system. This survey included a specific question about how engaged their child was while interacting with the tutor with three possible values: *Not Engaged*, *Somewhat Engaged*, and *Very Engaged*. This measure, which we will refer to as the parent-reported engagement (PRE) label, represents a relatively basic measure of engagement, as it does not ask parents to distinguish between the various ways in which engagement might manifest. However, it represents a valuable perspective on engagement from caregivers who are most familiar with the participants' unique focus styles and avoids concerns about the reliability of children's self-report data [21, 28].

PRE was provided by the parents of 139 participants, with 98 of those participants having audio recordings available for comprehension conversations. Labels were highly skewed towards the *Very Engaged* and *Somewhat Engaged* values with only 4 participants reported as *Not Engaged* during the sessions, only two of whom had audio recordings. As such, any patterns observed involving the Not Engaged group should be considered exploratory. The full distribution of clinical diagnosis based on engagement level can be seen in Table 3. We exclude any students who have fewer than two audio recordings present in the data. We apply the parent-reported engagement (PRE) level as a measure of overall engagement propensity to all comprehension activities with audio recording for the student. This is not intended to indicate turn-level engagement, but instead enable us to identify what acoustic features are correlated with children's general engagement statement.

3.2.4 Data Preprocessing

We conducted a series of preprocessing steps. In order to ensure that each participant had enough acoustic data for analysis, we first selected for students who had participated in at least two comprehension activities. We then aggregated all of the comprehension conversation turns for that student and aligned them with the appropriate audio file. Acoustic features were extracted using the OpenSMILE [25] toolkit for each of the student conversational turns. We collected system-generated transcriptions of student responses, which were processed through a simple word count function to generate verbal response counts for each turn. We then align the transcribed student response data, audio files, and the demographic and engagement reports from the parent surveys[2].

3.2.5 Measuring the Effect of Diagnosis x Engagement

To examine effects of clinical diagnosis and PRE levels on acoustic features while accounting for the fact that we have multiple audio recordings for each child, we employed population average models using generalized estimating equations (GEE). GEE models are well-suited for correlated data and provide robust inference without requiring explicit modeling of subject-specific random effects [43, 59, 75]. We also explored the use of mixed-effect models; however, they could not be applied across all the desired acoustic features due to limited within-child variability.

[2]The resulting dataset will be released for future work following the completion of the requisite anonymization process.

Table 4: Number of acoustic features showing significant population-level effects after FDR correction, stratified by feature category and predictor.

Feature Category	Diagnosis	Eng.	Diagnosis × Eng.
Energy	0	3	3
Frequency	0	6	8
Spectral	0	14	27
Temporal	0	2	5
Total	**0**	**25**	**33**

For each acoustic feature mentioned above, we fit a Gaussian GEE of the form:

$$\begin{aligned} Feature_{ij} = \beta_0 + \boldsymbol{\beta}_1^\top \text{Diagnosis}_i + \beta_2 \text{Engagement}_i \\ + \boldsymbol{\beta}_3^\top \left(\text{Diagnosis}_i \times \text{Engagement}_i\right) \\ + \beta_4 \text{Age}_i + \beta_4 \text{Sex}_i + \varepsilon_{ij}. \end{aligned} \quad (1)$$

where i represents children and j represents individual acoustic observations. Individual acoustic observations are represented by a vector of acoustic features from OpenSMILE that were extracted from a single turn in the comprehension dialogue. These models included clinical diagnosis, engagement, and the interaction of diagnosis and engagement as categorical predictors, with child age (mean-centered) and sex (parent-reported assigned sex at birth) as covariates. They were fit using the python *statsmodels* package [60] with the correlation structure set to exchangeable to account for the within-child correlation from the repeated observations [43]. Given the broad range of values for different acoustic features, we normalized the values using *StandardScaler* from *scikit-learn* [55]. We also use the *statsmodels* package to account for possible false-discovery rates (FDR) for diagnosis, engagement, and interaction effects with the Benjamini-Hochberg false discovery rate procedure [8] with adjusted p-values set using the methodology defined by Yekutieli and Benjamini [72].

4. RESULTS

We examine the results of fitting a GEE model across the 49 acoustic features in Table 2 as well as the verbal response count of the student responses. In Table 4, we present the acoustic features that are the most impacted by engagement, diagnosis or the combination of diagnosis and engagement. The parent-reported level of engagement acts as an independent predictor of acoustic features across all the categories. There were no statistically significant diagnosis-related effects present in the dataset. The interaction effects of diagnosis and parent-reported engagement are distributed across all of the feature categories.

4.1 Verbal Response Count

Table 5: GEE Results for Verbal Response Count. This table shows the FDR-adjusted p-values across all model terms. None of the values were considered statistically significant.

Predictor	FDR-adjusted p-value
Diagnosis	0.0627
Engagement	0.2420
Diagnosis × Engagement	0.2998
Age	0.1256
Sex	0.3986

In alignment with prior work in verbal engagement, we created a GEE model for the normalized verbal response count. This is taken as the count of words from the transcribed student response for a single turn. The results are displayed in Table 5. Our findings demonstrate that there was no statistically significant impact of engagement, diagnosis, or the interaction of the two on this metric. This suggests that this feature is not dependent on any of these categories.

4.2 Effect of Age and Sex on Results

We account for variations in age and sex in our results by adding them as intercepts to the GEE model. The results that we report for all engagement, diagnosis, and their interaction are those that did not have a statistically significant influence from these variables. Of the 49 acoustic features evaluated, only one had a statistically significant association age: the standard deviation of jitter. Mean child age had a positive coefficient of 0.101, indicating that as children get older than the mean age (in our case 8.4 years old), their variability in jitter levels increased as well. Child sex (as reported by parents in the demographic survey) did not have a statistically significant association with any of the acoustic features tested.

4.3 Engagement Effects

The GEE model shows that PRE levels have an effect on three *energy*-related features (i.e., shimmer, mean Harmonic-to-Noise Ratio and standard deviation of the Harmonic-to-Noise Ratio). PRE also impacts a number of *frequency* features (e.g., those related to the pitch based on F_0 semitone values and jitter, and changes in consecutive F_0 frames) and *spectral shape* features (e.g., Alpha Ratio, Hammarberg Index, and Spectral Slope values). In addition, PRE affects both the mean values of voiced segment length as well as loudness peaks per second - both of which are associated with speaking rates. These findings are consistent with prior work evaluating fluctuations in acoustic features related to arousal, engagement, and emotion [3, 7, 24, 44]. The majority of the contrast effects for the model appear between the *Not Engaged* and *Very Engaged* groups, however, we do see a meaningful difference between shimmer values for the *Somewhat Engaged* and *Very Engaged* groups. Students in the Somewhat Engaged groups (regardless of clinical diagnosis) exhibited higher mean values of shimmer than those in the Very Engaged group.

4.4 Diagnosis Effects

The GEE analysis of our data did not result in any statistically significant effects for diagnosis as an independent predictor. These findings are surprising given a large body of existing research that demonstrates noticeable differences in prosody, pitch, and voice quality for individuals with autism spectrum disorder vs. neurotypical peers [9, 48, 53, 54]. It is possible that no differences were detected in this study due to high variability in the speech characteristics of individuals with a given diagnosis or the fact that this sample was collected in a highly constrained (and non-social) context. There are still context-dependent diagnostic differences within the dataset that vary based on the child's parent-reported engagement level.

4.5 Diagnosis x Engagement Effects

As shown in Table 4, thirty-three acoustic features exhibited significant interactions between clinical diagnosis and PRE, indicating that diagnostic differences in vocal production depended on PRE. Multiple acoustic categories showed interaction effects, including three associated with *energy*, eight associated with *frequency*, twenty-seven involved with *spectral shape*, and five involved with *temporal values*. Of these features, 18 of them demonstrated an interaction effect without the presence of a statistically significant effect from one of the other predictors. We report the full set of the interaction effects in Table 6. Notably, features related to pitch distribution, spectral variability, and voice quality showed the strongest evidence of context-dependent diagnostic effects, underscoring the importance of considering engagement state when interpreting acoustic markers of clinical diagnosis in educational interactions. The contrast effects referencing Not Engaged in the term should be interpreted as exploratory since there were only 2 students in the analyzed data with audio recordings at this engagement propensity level. Coefficients and patterns reported for the Somewhat Engaged level are representative of a more substantial number of participants and observations in the dataset.

4.5.1 Energy

All three of the features in the energy category demonstrated interaction effects: the mean Harmonics-to-Noise (HNR) ratio and the standard deviation for the shimmer values. However, none of the individual contrast effects between groups registered as statistically significant. There are still some trends that are present in the dataset that should be mentioned for future exploration. Autistic children who were more engaged exhibited higher values of HNR whereas the neurotypical group had the opposite pattern. As seen in Figure 3, For HNR, the ADHD and ASD + ADHD groups more closely resemble the pattern exhibited by neurotypical children, with both showing lower values when Very Engaged vs. Somewhat Engaged. When children in the ADHD and ASD+ADHD groups are rated as Not Engaged, the HNR values are dramatically lower than for the mid-level of engagement, but there is high variability in these groups. This finding is harder to interpret given that there are a very limited number of samples in the dataset with the Not Engaged label. From a practical perspective, when HNR is higher it is indicative of clearer phonation [3] — which means that children with ASD are pronouncing words more clearly when more engaged whilst the other groups are actually reducing their clarity.

We also see opposing patterns between the neurotypical and ASD groups for the shimmer feature. The standard deviation for shimmer is lower for more engaged neurotypical children, whereas it is higher for more engaged ASD children. The more-engaged ADHD group shows a slight increase in shimmer values. Higher shimmer values mean that there is more intensity variability, which is frequently associated with high arousal emotions and higher levels of stress [3].

4.5.2 Frequency

Variations in the frequency characteristics of speech are highly indicative of individual's arousal levels and engagement [3, 23, 44]. In Figure 4, we can see the difference in mean values

Table 6: Acoustic features that exhibit a statistically significant contrast effect (after FDR) based on the interaction of diagnosis × engagement. The reference group for the clinical diagnosis is always *Not ASD or ADHD* and for engagement it is always *Very Engaged*, not engaged is represented as NE and somewhat engaged is represented as SE. Cells show GEE coefficient (Standard Error). * indicates $p < .05$. Dashes indicate non-significant contrasts. † indicates contrast excluded due to anomalous SE = 0. All values rounded to 4 decimals.

Feature Type	Feature	ADHD:NE	ASD:NE	ASD:SE	ASD+ADHD:NE
Spectral	alphaRatioUV sma3nz (mean)	−0.5758 (0.1514)*	—	—	0.4108 (0.1876)*
	hammarbergIndexUV sma3nz (mean)	0.8934 (0.1482)*	—	—	−0.6737 (0.1893)*
	hammarbergIndexV sma3nz (std dev Norm)	−0.1400 (0.0629)*	†	—	—
	mfcc1V sma3nz (std dev Norm)	−0.2257 (0.0614)*	†	−0.1860 (0.0932)*	—
	mfcc2 mean	0.4623 (0.1637)*	—	—	−0.4865 (0.1950)*
	mfcc3 mean	−0.8495 (0.1296)*	—	—	0.8784 (0.1643)*
	mfcc4 mean	0.7354 (0.1316)*	—	0.8278 (0.3660)*	−0.5800 (0.1899)*
	spectralFlux sma3 (std dev Norm)	0.4964 (0.1335)*	—	—	−0.4418 (0.1846)*
	slopeV0_500 sma3nz (std dev Norm)	−0.0945 (0.0404)*	—	—	—
	slopeV500_1500 sma3nz (std dev Norm)	0.0923 (0.0463)*	†	—	—
Frequency	F0semitone mean Rising Slope	−0.2054 (0.0756)*	—	—	—
	jitter Local Mean	0.2886 (0.0955)*	—	—	—
Temporal	Unvoiced Segment Length (mean)	0.5296 (0.0669)*	—	—	−0.4114 (0.1307)*
	Unvoiced Segment Length (std dev)	0.3700 (0.1083)*	—	—	−0.3406 (0.1305)*
	Voiced Segment Length (std dev)	−0.4769 (0.0996)*	†	—	0.4366 (0.1078)*

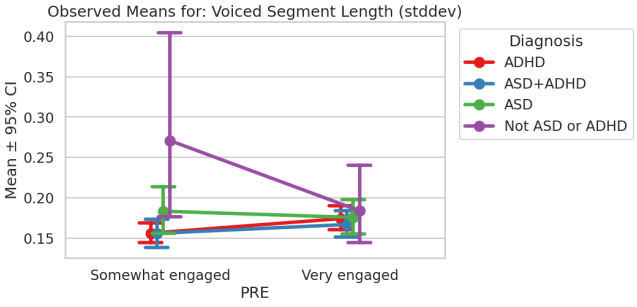


Figure 1: Observed means of the variations of standard deviation in voice segment length across different clinical diagnoses and parent reported levels of engagement.

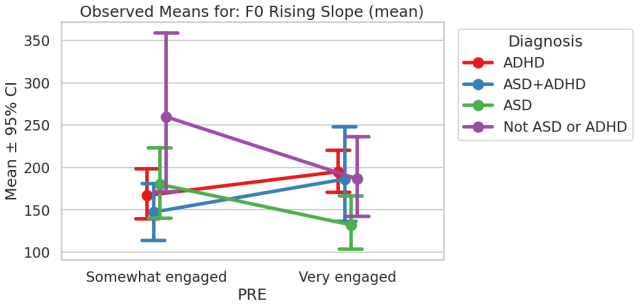


Figure 2: Diagnosis-group level means of the of F0 rising slope across different clinical diagnoses and parent reported levels of engagement.

for the rising slope for F_0 across different clinical diagnoses and engagement values. The neurotypical group exhibits a higher range of values than all of the neurodivergent groups when rated as Somewhat Engaged. These values increase between the Somewhat and Very Engaged groups for individuals with ADHD or ASD+ADHD, but they decrease for neurotypical participants. Essentially, this is indicating that when individuals with ADHD are more engaged they raise their pitch more quickly, whereas the neurotypical students will drop the rate of pitch change when more engaged.

We also notice a stark contrast between the trends of mean values for jitter between the participants with ASD when compared to all other children. Jitter values are higher for Very Engaged ASD children than the Somewhat Engaged group. The pattern is flipped for neurotypical, ADHD, and ASD+ADHD children. Jitter indicates pitch fluctuations in speech and higher values are typically associated with more tremulous or nervous speech [45]. This variation could indicate that individuals with ASD tremble more when they are highly engaged with content than their peers.

For both of these values, the only contrast effect that was statistically significant was for the Not Engaged group with ADHD when compared to the Very Engaged-Neurotypical group. However, given the extremely small sample size of

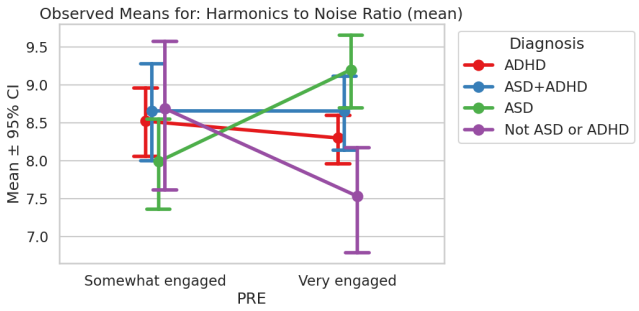


Figure 3: Diagnosis-group level means of the Harmonics-to-noise ratio across different clinical diagnoses and parent reported levels of engagement.

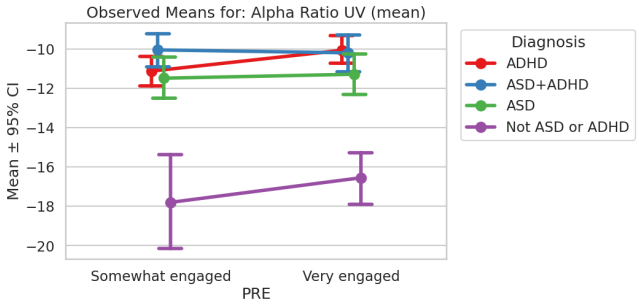


Figure 4: Diagnostic-group level means of the mean Alpha Ratio UV for each of the clinical diagnoses and parent reported levels of engagement.

the Not Engaged group, this result should be verified with a larger dataset.

4.5.3 *Spectral*

Interaction effects are most prevalent for the spectral or vocal shape related features. There are statistically significant effects for the unvoiced Alpha Ratio, voiced and unvoiced Hammarberg Index, MFCC values, spectral Flux and spectral slope values. As seen in Table 9, there are not any statistically significant contrast effects for any of the clinical diagnosis in the Very Engaged group. This means that across the clinical diagnosis, when children were Very Engaged, the spectral shape-related acoustic features appear in a uniform manner in the dataset. On the other hand, there are meaningful contrasts between different clinical diagnosis and the Not Engaged and Somewhat Engaged groups.

In our limited set of Not Engaged participants, we see a number of contrasting acoustic features from the Not ASD or ADHD:Very Engaged reference group. As demonstrated in Figure 4, the ASD+ADHD group had higher values for unvoiced Alpha Ratios than the reference, whereas the ASD group had substantially lower ones. A low Alpha ratio is usually associated with a softer voice. The coefficients for the unoviced Hammarberg Index mirror these results; high Hammarberg indices are associated with a softer voice, whereas lower values would be indicative of a sharper one [62]. This could be indicative of a trend where individuals with ADHD speak in a much softer tone when disengaged, as opposed to disengaged individuals with ASD+ADHD who are talking in sharper tones. Generally speaking, the mean values for the Alpha Ratio of the Not ASD or ADHD groups are considerably lower than they are for all other groups in the study.

We do see significant contrast effects in the Somewhat Engaged groups for some of the MFCC values. This includes lower values for MFCC1V for ADHD children and higher values for MFCC4 for Autistic Students. MFCC1V values are usually associated with variations in spectral tilt, with higher values indicating larger amounts of variation. Instead, for the ADHD children this is saying that there is a limited amount of variation in the spectral tilt when they are Somewhat Engaged.

4.5.4 *Temporal*

All of the temporal features had a statistically significant interaction effect, but only three of them had one without the presence of an Engagement effect. This includes the standard deviation of the voiced segment length as well as the mean and standard deviations of the unvoiced segment length. Lower standard deviations of voiced segment length mean that the speaker is talking with a more regular cadence, usually associated with a confident manner of speaking, whereas higher values can be indicative of hesitant speech [29]. In figure 1, we can see that the neurotypical group shows a much larger decrease in this value between the Very Engaged and Somewhat Engaged participants when compared to the neurodivergent groups. This suggests that neurotypical children will sound more confident when they have higher engagement levels, but this perceived confidence shift may not be reflected in the speech of their neurodivergent peers.

5. DISCUSSION

5.1 Implications for Automatic Detection of Reading Engagement

In highlighting the effects of parent-reported engagement, diagnosis, and their interaction on acoustic features, our ultimate goal is to better inform future models of reading engagement for diverse learner groups. Our results highlight the fact that acoustic features may be a more equitable signal than verbal response count for measuring reading engagement in the home setting. We recommend the use of acoustic features that demonstrate robust effects related to engagement propensity while taking into account interactions between clinical diagnosis and engagement levels. This includes Hammarberg index values, Alpha ratios, shimmer, as well as loudness peaks per second and mean values of voiced segment length. However, all of these values were associated with a statistically significant interaction effect between the diagnosis and PRE. Special consideration should be given when using acoustic features to predict engagement levels for children with ASD. In our data, we notice high variability for a number of acoustic features within this group. This may make it especially difficult to accurately predict engagement levels since there may not be a consistent observable speech pattern.

These findings are intended to be used to create adaptive at-home literacy learning systems. The differences presented in acoustic measures for individuals with ASD or ADHD

are not indicative of a reading comprehension deficit. Nor should they be used as a diagnostic tool. Instead, these should be interpreted as evidence that we need to develop more complex acoustic-feature based predictive models that can account for variations in the presentation of speech-related engagement signals. With effective real-time engagement predictions, it is possible to implement interventions that keep students engaged with reading materials at home, and in turns lead to more opportunities for improving their reading comprehension skills.

5.2 Limitations and Future Directions

The detectors presented here are comparable in performance to a range of others in previous research, but would still benefit from further refinement, particularly from efforts that could expand our data collection efforts. It is important to note that this a sensitivity study on the collected audio data, and we are not evaluating the impact of a particular reading intervention on features related to reading engagement. This is a logical next step in future research studies - leveraging these features as a non-invasive measure of the effect of particular reading interventions on student reading engagement in different subgroups.

Like many studies, this analysis would have benefited from the use of a larger data set. This is particularly challenging because the majority of available speech datasets that might serve as points of comparison–especially those used to identify acoustic features related to engagement–are composed of adult speech samples from extemporaneous speech. In addition to raw frequency differences between adult and child acoustic data, children often exhibit different speech behaviors in extemporaneous speech, and our samples were further constrained by the activities students were asked to perform as part of a task related to learning to reading.

In addition to benefiting generally from a larger sample, we would also benefit from diversifying our sample. Importantly, our current dataset is representative of children with several possible clinical conditions, which allowed us to present findings related to neurodiversity that have been understudied in the EDM community. However, it would still benefit from a more robust sampling of ethnic, socioeconomic, and English language learner status as variation within those diagnoses, as those factors are known to affect students' language patterns.

Future work should also look for ways to improve the granularity of the engagement labels, as the PRE labels were derived from parental evaluations of students' experience with the learning system in its entirety. However, even the most engaged child will not be completely engaged during every lesson, or every turn of that lesson. Ideally, in future experiments we will complement this measure with finer-grained labels, allowing us to develop a more nuanced understanding how students who are highly engaged overall might vary over the course of several weeks while also developing a more nuanced understanding of the acoustic indicators of this engagement. Additionally, we acknowledge that our dataset contained a very small number of students in the not engaged group. Ideally, in a larger sample we would have a less imbalanced set of engagement levels. Such efforts would also substantially improve our predictive modeling of real-time engagement.

Finally, we aim to expand our approach to comprehension conversation data. During a single comprehension activity, the microphone is open for a fixed period of time where the student responds to - at most - two conversational turns. This substantially limits the amount of data that can be collected, as students may be cut off before they are able to fully participate with the task. There may be strategic reasons for the learning system to continue such practices, but it would be good to consider ways in which we might better understand how this constraint affects student engagement, particularly as we seek to use more fine-grained levels than those employed in this study.

Still, the course-grained PRE labels deployed here–in addition to serving as an important step towards better understanding home usage–might also help us to better understand when apparent disengagement at the micro-level may be incidental. For example, they likely indicate a students' overall propensity to re-engage with the learning system, which could be used to help filter more fine-grained, moment-to-moment patterns in order to better understand which students need greater support.

6. CONCLUSION

In this paper, we have explored the use of acoustic features to measure reading engagement during online learning sessions across a diverse learner population. Acoustic features provide a non-invasive manner of capturing critical data related to student behavior and affect while they are working with a digital literacy platform at home. We present the nuances of modeling reading engagement for children with ADHD, ASD, or both based on the unique characteristics of these conditions. Our study includes audio and transcription data collected from 144 children over the course of several weeks who engaged in passage reading and reading comprehension tasks while at home.

Specifically, we use parental reports of engagement (PRE) with the system and clinical diagnosis to create generalized estimating equations (GEEs) for engagement-related acoustic features. We compare the effect of clinical diagnosis, parent report of engagement, child age and sex on a set of 49 acoustic features across 98 students from our dataset. We confirmed the results of prior work demonstrating that certain acoustic features related to frequency, temporal, spectral, and energy characteristics of speech are consistently influenced by engagement levels [44]. However, we discovered that all of the features are also impacted by the interaction between clinical diagnosis and engagement level. This suggests a critical need to develop more robust measures for measuring reading engagement that can account for the unique characteristics associated with different neurodevelopmental conditions. In future work, we hope to collect more refined turn-level engagement annotations that can be used to develop acoustic feature-driven models of reading engagement based on these findings. Furthermore, by integrating these measures of reading engagement into digital literacy systems we can create personalized adaptive learning interventions to increase individual students' reading comprehension gains.

7. ACKNOWLEDGMENTS

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305C240040. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education. This work was made possible by our collaborators at Digital Promise, the Amira Learning team, the Children's Hospital of Philadelphia, and MDRC. The authors would also like to express our gratitude to the families who participated in our study and contributed to this line of research.

APPENDIX

A. PER-FEATURE GEE RESULTS

Table 7 includes all acoustic features organized by high level category as well as the results of fitting the GEE model for that feature.

REFERENCES

[1] Amira Tutor, 2026.

[2] P. Afflerbach and C. Harrison. What Is Engagement, How Is It Different From Motivation, and How Can I Promote It? *Journal of Adolescent & Adult Literacy*, 61(2):217–220, 2017. _eprint: https://ila.onlinelibrary.wiley.com/doi/pdf/10.1002/jaal.679.

[3] F. S. Alamri, A. Rashad Mirdad, T. Saba, M. Amjad Raza, and U. Siddique. Investigating the Impact of Combined Spectral and Prosodic Features on Speech Emotion Recognition. *IEEE Access*, 14:5719–5732, 2026.

[4] A. P. Association. *Diagnostic and statistical manual of mental disorders*. American Psychiatric Publishing, Inc., Washington, DC, 5th ed., text rev. edition, 2022.

[5] R. S. Baker, J. L. Ocumpaugh, and J. M. A. L. Andres. BROMP quantitative field observations: A review. In R. Feldman, editor, *Learning Science: Theory, Research, and Practice*, pages 127–156. McGraw-Hill, New York, NY, 2020.

[6] A. T. Barber and S. L. Klauda. How Reading Motivation and Engagement Enable Reading Achievement: Policy Implications. *Policy Insights from the Behavioral and Brain Sciences*, 7(1):27–34, 2020. _eprint: https://doi.org/10.1177/2372732219893385.

[7] J. E. Beck. Engagement tracing: using response times to model student disengagement. In *Proceedings of the 2005 conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, pages 88–95, NLD, May 2005. IOS Press.

[8] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B (Methodological)*, 57(1):289–300, 1995.

[9] Y. S. Bonneh, Y. Levanon, O. Dean-Pardo, L. Lossos, and Y. Adini. Abnormal speech spectrum and increased pitch variability in young autistic children. *Frontiers in Human Neuroscience*, 4:237, 2011.

[10] N. Bosch, S. D'Mello, R. Baker, J. Ocumpaugh, V. Shute, M. Ventura, L. Wang, and W. Zhao. Automatic Detection of Learning-Centered Affective States in the Wild. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 379–388, Atlanta Georgia USA, Mar. 2015. ACM.

[11] K. Brimo, L. Dinkler, C. Gillberg, P. Lichtenstein, S. Lundström, and J. Åsberg Johnels. The co-occurrence of neurodevelopmental problems in dyslexia. *Dyslexia (Chichester, England)*, 27(3):277–293, Aug. 2021.

[12] M. Bustos-López, N. Cruz-Ramírez, A. Guerra-Hernández, L. N. Sánchez-Morales, N. A. Cruz-Ramos, and G. Alor-Hernández. Wearables for Engagement Detection in Learning Environments: A Review. *Biosensors*, 12(7):509, July 2022.

[13] R. A. Calvo and S. D'Mello. Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications. *IEEE Transactions on Affective Computing*, 1(1):18–37, Jan. 2010.

[14] CDC. Data and Statistics on Autism Spectrum Disorder, May 2025.

[15] M. L. Danielson, A. H. Claussen, R. H. Bitsko, S. M. Katz, K. Newsome, S. J. Blumberg, M. D. Kogan, and R. Ghandour. ADHD Prevalence Among U.S. Children and Adolescents in 2022: Diagnosis, Severity, Co-Occurring Disorders, and Treatment. *Journal of Clinical Child and Adolescent Psychology: The Official Journal for the Society of Clinical Child and Adolescent Psychology, American Psychological Association, Division 53*, 53(3):343–360, 2024.

[16] M. M. Davidson and S. Ellis Weismer. Characterization and prediction of early reading abilities in children on the autism spectrum. *Journal of Autism and Developmental Disorders*, 44(4):828–845, Apr. 2014.

[17] D. Demszky, R. Wang, S. Geraghty, and C. Yu. Does Feedback on Talk Time Increase Student Engagement? Evidence from a Randomized Controlled Trial on a Math Tutoring Platform. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 632–644, Kyoto Japan, Mar. 2024. ACM.

[18] G. Dietz, Z. Pease, B. McNally, and E. Foss. Giggle gauge: a self-report instrument for evaluating children's engagement with technology. In *Proceedings of the Interaction Design and Children Conference*, IDC '20, pages 614–623, New York, NY, USA, June 2020. Association for Computing Machinery.

[19] C. Doğdu, T. Kessler, D. Schneider, M. Shadaydeh, and S. R. Schweinberger. A Comparison of Machine

Table 7: Feature-by-significance table for all acoustic features. Entries are FDR-adjusted p-values. Asterisk (*) indicates $p_{\text{FDR}} < 0.05$.

Feature	Diagnosis	Engagement	Diagnosis × Engagement
Temporal features			
loudnessPeaksPerSec	0.3927	0.0000*	0.0000*
Voiced segment length (mean)	0.9649	0.0094*	0.0000*
Voiced segment length (std. dev.)	0.9758	0.4877	0.0000*
Unvoiced segment length (mean)	0.3782	0.3930	0.0000*
Unvoiced segment length (std. dev.)	0.2547	0.8751	0.0001*
Energy features			
Harmonic-to-noise ratio (mean)	0.4363	0.0000*	0.0000*
Harmonic-to-noise ratio (std. dev.)	0.3927	0.0276*	0.0142*
Shimmer (std. dev.)	0.2547	0.0000*	0.0000*
Frequency features			
F0 mean	0.2547	0.0000*	0.0000*
F0 standard deviation	0.2547	0.0367*	0.0000*
F0 20th percentile	0.2547	0.0000*	0.0000*
F0 50th percentile	0.2547	0.0000*	0.0000*
F0 80th percentile	0.2547	0.0000*	0.0000*
F0 mean rising slope	0.2547	0.0000*	0.0000*
F0 mean falling slope	0.2547	0.0000*	0.0000*
Jitter (mean)	0.2547	0.0038*	0.0000*
Jitter (std. dev.)	0.2547	0.1076	0.0000*
Spectral features			
Alpha ratio, voiced (mean)	0.2547	0.0000*	0.0000*
Alpha ratio, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Alpha ratio, unvoiced (mean)	0.2547	0.4877	0.0000*
Hammarberg index, voiced (mean)	0.2547	0.0000*	0.0000*
Hammarberg index, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Hammarberg index, unvoiced (mean)	0.0053*	0.0000*	0.0000*
Spectral flux, voiced (mean)	0.2547	0.0000*	0.0000*
Spectral flux, voiced (std. dev.)	0.2547	0.0000*	0.0000*
Spectral flux, unvoiced (mean)	0.2547	0.4877	0.0000*
Spectral slope 0–500 Hz (mean)	0.2547	0.0000*	0.0000*
Spectral slope 0–500 Hz (std. dev.)	0.2547	0.0000*	0.0000*
Spectral slope 500–1500 Hz (mean)	0.2547	0.0000*	0.0000*
Spectral slope 500–1500 Hz (std. dev.)	0.2547	0.0000*	0.0000*

Learning Algorithms and Feature Sets for Automatic Vocal Emotion Recognition in Speech. *Sensors (Basel, Switzerland)*, 22(19):7561, Oct. 2022.

[20] S. D'Mello and A. Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, Apr. 2012.

[21] S. D'Mello, B. Lehman, R. Pekrun, and A. Graesser. Confusion can be beneficial for learning. *Learning and Instruction*, 29:153–170, Feb. 2014.

[22] F. El Zein, M. Solis, R. Lang, and M. K. Kim. Embedding perseverative interest of a child with autism in text may result in improved reading comprehension: A pilot study. *Developmental Neurorehabilitation*, 19(3):141–145, June 2016.

[23] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong. The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing. *IEEE Transactions on Affective Computing*, 7(2):190–202, Apr. 2016.

[24] F. Eyben, F. Weninger, and B. Schuller. Affect recognition in real-life acoustic conditions — a new perspective on feature selection. In *Interspeech 2013*, pages 2044–2048. ISCA, Aug. 2013.

[25] F. Eyben, M. Wöllmer, and B. Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, MM '10, pages 1459–1462, New York, NY, USA, Oct. 2010. Association for Computing Machinery.

[26] J. A. Fredricks, P. C. Blumenfeld, and A. H. Paris. School Engagement: Potential of the Concept, State of the Evidence. *Review of Educational Research*, 74(1):59–109, 2004.

[27] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, and M. P. Wenderoth. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415, June 2014.

[28] K. A. Fuller, N. S. Karunaratne, S. Naidu, B. Exintaris, J. L. Short, M. D. Wolcott, S. Singleton, and P. J. White. Development of a self-report instrument for measuring in-class student engagement reveals that pretending to engage is a significant unrecognized problem. *PLOS ONE*, 13(10):e0205828, Oct. 2018.

[29] I. Gliser, C. Mills, N. Bosch, S. Smith, D. Smilek, and J. D. Wammes. The Sound of Inattention: Predicting Mind Wandering with Automatically Derived Features of Instructor Speech. *Artificial Intelligence in Education*, 12163:204–215, June 2020.

[30] Guher Gorgun, Seyma Nur Yildirim-Erbasli, and C. D. Epp. Predicting Cognitive Engagement in Online Course Discussion Forums. July 2022.

[31] J. T. Guthrie, A. Wigfield, et al. Engagement and motivation in reading. *New York*, 2000.

[32] J. T. Guthrie, A. Wigfield, and W. You. Instructional Contexts for Engagement and Achievement in Reading. In S. L. Christenson, A. L. Reschly, and C. Wylie, editors, *Handbook of Research on Student Engagement*, pages 601–634. Springer US, Boston, MA, 2012.

[33] K. He, A. Gastón-Panthaki, D. Zhuo, J. Munsey, M. Zhang, and M. Warschauer. StoryPal: Supporting Young Children's Dialogic Reading with Large Language Models. In *Proceedings of the 24th Interaction Design and Children*, pages 494–511, Reykjavik Iceland, June 2025. ACM.

[34] A. James, Y. Chua, T. Maszczyk, A. Moreno-Núñez, R. Bull, K. Lee, and J. Dauwels. Automated Classification of Classroom Climate by Audio Analysis. pages 41–49. Sept. 2019.

[35] B. W. Jeff Johns. A Dynamic Mixture Model to Detect Student Motivation and Proficiency.

[36] S. Kai, L. Paquette, R. S. Baker, N. Bosch, S. D'Mello, V. Shute, and M. Ventura. A Comparison of Video-based and Interaction-based Affect Detectors in Physics Playground.

[37] M. Kakoulidou, F. Le Cornu Knight, R. Filippi, and J. Hurry. The Effects of Choice on the Reading Comprehension and Enjoyment of Children with Severe Inattention and no Attentional Difficulties. *Research on Child and Adolescent Psychopathology*, 49(11):1403–1417, Nov. 2021.

[38] S. Karumbaiah, J. Ocumpaugh, and R. Baker. The Influence of School Demographics on the Relationship Between Students' Help-Seeking Behavior and Performance and Motivational Measures. June 2019.

[39] K. Khowaja and S. S. Salim. A systematic review of strategies and computer-based intervention (CBI) for reading comprehension of children with autism. *Research in Autism Spectrum Disorders*, 7(9):1111–1121, Sept. 2013.

[40] S. Y. Kim, M. Rispoli, R. A. Mason, C. Lory, E. Gregori, C. A. Roberts, D. Whitford, and D. Wang. The Effects of Using Adapted Science eBooks Within Shared Reading on Comprehension and Task Engagement of Students with Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders*, 55(12):4185–4196, Dec. 2025.

[41] Y. Lee, B. G. Jang, and K. C. Smith. A Systematic Review of Reading Engagement Research: What Do We Mean, What Do We Know, and Where Do We Need to Go? *Reading Psychology*, 42(5):540–576, 2021. _eprint: https://doi.org/10.1080/02702711.2021.1888359.

[42] S. Li, S. P. Lajoie, J. Zheng, H. Wu, and H. Cheng. Automated detection of cognitive engagement to inform the art of staying engaged in problem-solving. *Computers & Education*, 163, 2021.

[43] K.-Y. LIANG and S. L. ZEGER. Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22, Apr. 1986.

[44] D. Litman, H. Friedberg, and K. Forbes-Riley. Prosodic cues to disengagement and uncertainty in physics tutorial dialogues. In *Interspeech 2012*, pages 755–758. ISCA, Sept. 2012.

[45] J. K. MacCallum, Y. Zhang, and J. J. Jiang. Acoustic analysis of the tremulous voice: assessing the utility of the correlation dimension and perturbation parameters. *Journal of communication disorders*, 43(1):35, 2010.

[46] N. S. McIntyre, E. J. Solari, J. E. Gonzales, M. Solomon, L. E. Lerro, S. Novotny, T. M. Oswald, and P. C. Mundy. The Scope and Nature of Reading Comprehension Impairments in School-Aged Children with Higher-Functioning Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders*, 47(9):2838–2860, Sept. 2017.

[47] A. C. Miller, J. M. Keenan, R. S. Betjemann, E. G. Willcutt, B. F. Pennington, and R. K. Olson. Reading comprehension in children with ADHD: cognitive underpinnings of the centrality deficit. *Journal of Abnormal Child Psychology*, 41(3):473–483, Apr. 2013.

[48] A. Mohanta and V. Kumar Mittal. Autism Speech Analysis using Acoustic Features. In D. M. Sharma and P. Bhattacharya, editors, *Proceedings of the 16th International Conference on Natural Language Processing*, pages 85–94, International Institute of Information Technology, Hyderabad, India, Dec. 2019. NLP Association of India.

[49] K. P. Nowell, K. E. Bodner, M. D. Mohrland, and S. M. Kanne. Neurodevelopmental disorders. In *Handbook of rehabilitation psychology, 3rd ed*, pages 357–370. American Psychological Association, Washington, DC, US, 2019.

[50] J. Ocumpaugh, R. S. Baker, and M. M. T. Rodrigo. Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) 2.0 Technical and Training Manual.

[51] J. Ocumpaugh, N. Nasiar, A. F. Zambrano, A. Goslen, J. Vandenberg, J. Esiason, J. Rowe, and S. Hutt. Refocusing the lens through which we view affect dynamics: The Skills, Difficulty, Value, Efficacy and Time Model. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 192–203, Dublin Ireland, Mar. 2025. ACM.

[52] L. Paquette, Z. Li, R. Baker, J. Ocumpaugh, and A. Andres. Who's learning? Using demographics in EDM research.

[53] J. Parish-Morris, M. Liberman, N. Ryant, C. Cieri, L. Bateman, E. Ferguson, and R. Schultz. Exploring Autism Spectrum Disorders Using HLT. In K. Hollingshead and L. Ungar, editors, *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*, pages 74–84, San Diego, CA, USA, June 2016. Association for Computational Linguistics.

[54] R. Paul, L. D. Shriberg, J. McSweeny, D. Cicchetti, A. Klin, and F. Volkmar. Brief report: relations between prosodic performance and communication and socialization ratings in high functioning speakers with autism spectrum disorders. *Journal of Autism and Developmental Disorders*, 35(6):861–869, Dec. 2005.

[55] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[56] A. Ramakrishnan, E. Ottmar, J. LoCasale-Crouch, and J. Whitehill. Toward Automated Classroom Observation: Predicting Positive and Negative Climate. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8, May 2019.

[57] J. Ricketts, C. R. G. Jones, F. Happé, and T. Charman. Reading comprehension in autism spectrum disorders: the role of oral language and social functioning. *Journal of Autism and Developmental Disorders*, 43(4):807–816, Apr. 2013.

[58] Y. Rong, C.-J. Yang, Y. Jin, and Y. Wang. Prevalence of attention-deficit/hyperactivity disorder in individuals with autism spectrum disorder: A meta-analysis. *Research in Autism Spectrum Disorders*, 83:101759, May 2021.

[59] A. Rotnitzky and N. P. Jewell. Hypothesis Testing of Regression Parameters in Semiparametric Generalized Linear Models for Cluster Correlated Data. *Biometrika*, 77(3):485–497, 1990.

[60] S. Seabold and J. Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[61] L. Tamarit, M. Goudbeek, and K. Scherer. Spectral Slope Measurements in Emotionally Expressive Speech. 2008.

[62] B. Tracey, D. Volfson, J. Glass, R. Haulcy, M. Kostrzebski, J. Adams, T. Kangarloo, A. Brodtmann, E. R. Dorsey, and A. Vogel. Towards interpretable speech biomarkers: exploring MFCCs. *Scientific Reports*, 13:22787, Dec. 2023.

[63] N. J. Unrau and M. Quirk. Reading Motivation and Reading Engagement: Clarifying Commingled Conceptions. *Reading Psychology*, 35(3):260–284, 2014. _eprint: https://doi.org/10.1080/02702711.2012.684426.

[64] C. Vukelich. The development of listening comprehension through storytime. *Language Arts*, 53(8):889–891, 1976.

[65] M. Watanabe, M. Imundo, K. Christhilf, T. Arner, and D. S. Mcnamara. Building Reading Comprehension and Knowledge with iSTART: An ITS to Provide Formative Feedback in Reading Instruction at Scale. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, pages 510–513, Atlanta GA USA, July 2024. ACM.

[66] M. F. Westerveld and J. M. A. Roberts. The Oral Narrative Comprehension and Production Abilities of Verbal Preschoolers on the Autism Spectrum. *Language, Speech, and Hearing Services in Schools*, 48(4):260–272, Oct. 2017.

[67] A. Y. Wong, R. L. Bryck, R. S. Baker, S. Hutt, and C. Mills. Using a Webcam Based Eye-tracker to Understand Students' Thought Patterns and Reading Behaviors in Neurodivergent Classrooms. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, LAK2023, pages 453–463, New York, NY, USA, Mar. 2023. Association for Computing Machinery.

[68] Y. Xu, D. Wang, P. Collins, H. Lee, and M. Warschauer. Same benefits, different communication patterns: Comparing Children's reading with a conversational agent vs. a human partner. *Computers & Education*, 161:104059, Feb. 2021.

[69] Y. Xu and M. Warschauer. A content analysis of voice-based apps on the market for early literacy development. In *Proceedings of the Interaction Design*

and Children Conference, pages 361–371, London United Kingdom, June 2020. ACM.

[70] Z. Xu, K. K. Wijekumar, G. Ramirez, X. Hu, and R. Irey. The effectiveness of intelligent tutoring systems on K-12 students' reading comprehension: A meta-analysis. *British Journal of Educational Technology*, 50(6):3119–3137, 2019. _eprint: https://bera-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/bjet.12758.

[71] V. P. Yache, L. Moradbakhti, I. Neuner, and T. Veselinovic. Predicting affective engagement and mental strain from prosodic speech features. *Frontiers in Psychiatry*, 16:1656292, Sept. 2025.

[72] D. Yekutieli and Y. Benjamini. Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics. *Journal of Statistical Planning and Inference*, 82(1):171–196, Dec. 1999.

[73] B. Zablotsky, M. D. Bramlett, and S. J. Blumberg. The Co-Occurrence of Autism Spectrum Disorder in Children With ADHD. *Journal of Attention Disorders*, 24(1):94–103, Jan. 2020.

[74] A. F. Zambrano, N. Nasiar, J. Ocumpaugh, A. Goslen, J. Zhang, J. Rowe, J. Esiason, J. Vandenberg, and S. Hutt. Says Who? How different ground truth measures of emotion impact student affective modeling. pages 211–223, 2024.

[75] S. L. Zeger and K.-Y. Liang. Longitudinal Data Analysis for Discrete and Continuous Outcomes. *Biometrics*, 42(1):121–130, 1986.

[76] Z. Zeng, S. Chaturvedi, and S. Bhat. Learner affect through the looking glass: Characterization and detection of confusion in online courses. 2017.

[77] J. Zhang, K. Wang, and Y. Zhang. Physiological Characterization of Student Engagement in the Naturalistic Classroom: A Mixed-Methods Approach. *Mind, Brain, and Education*, 15(4):322–343, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/mbe.12300.

Exploring Disagreement as a Dialogue Policy

Mariah Bradford
Colorado State University
mbrad@rams.colostate.edu

ABSTRACT

With growing interest in applying artificially intelligent agents to collaborative learning settings, we are challenged to identify meaningful moments for these agents to intervene. I propose moments of disagreement within groups to be pivotal - that is, the interaction can lead to learning gains or disruption depending on how they are handled. With that in mind, I plan to run user studies over a short group task to explore a baseline turn-based intervention policy compared to a disagreement-driven intervention policy. While these moments are nuanced, this study will provide some initial insights into a specific, educationally grounded, and explainable intervention policy during a group task conflict.

Keywords

User study, Disagreement, Collaboration, Dialogue policy

1. INTRODUCTION TO THE PROBLEM

Collaborative learning is a well-studied area in education, with growing opportunity to apply artificially intelligent agents as helpers [1, 6, 7, 19]. However, as we develop the necessary architectures to handle multimodal, multiparty, and in-person interactions, we are still left with a gap between the technology and educationally grounded human-AI interaction [6]. One question we must ground in the learning sciences is when to intervene, if at all. A closely related question is how conflict is handled in small groups. In a previous article, Rummel et al. contrasted two possible extremes for the future of AI in collaborative work: a dystopian future and a utopian future [17]. Notably, the two paths both included group conflict and dialogue policies. The dystopian scenerio described a pair with frequent conflict, frustrating interventions, and unexplainable outcomes from the AI. In the utopian scenerio, however, the dialogue policy was explainable, and conflict was treated as a nuanced situation, with helpful interventions aimed at productive conversation [17]. While both futures are still far off, I aim to explore the intersection of dialogue policy and conflict management in small group tasks. Previous work has explored human-centered dialogue policies for collaborative learning [4, 6], including wizard-of-oz analysis to inform dialogue policy and real-time, state-based dialogue policies. For example, in [6], group states were automatically classified and used to determine interventions. One outcome of this study was showcasing that active negotiations could contribute to unproductive perseverence; that is, when students were addressing task conflict, they could get stuck. While this was one consideration in their dialogue policy, I am interested in isolating this aspect and diving deeper into its impact on collaborative outcomes.

Mariah Bradford. Exploring Disagreement as a Dialogue Policy. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 851–854. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039925

2. PROPOSED FRAMING AND SOLUTION

Collaborative work often requires progress toward common ground [19, 26]. However, if task conflict arises, it can create an obstacle which requires negotiation to resolve [7, 8]. While previous work has shown mixed outcomes, with task conflict sometimes being a positive learning event [8] and sometimes a negative learning event [9, 11], the literature seems to indicate that the difference in these pivotal moments is how the conflict is handled [9, 12, 20, 23]. With this in mind, I propose that disagreement can serve as a signal to intervene. In particular, disagreements which are not being resolved or are escalating may require assistance from an external party [3, 6, 7]. Disagreement, for this paper, refers to a conflict in which we cannot reach common ground with the current task propositions [26]. For example, if one student believes the answer is A, and another student believes the answer is *not* A, they must resolve this disagreement if common ground is needed for the task. To explore this specific dialogue policy, I will implement user studies with a real-time, multimodal common ground tracking system [21] with the addition of interventions.

DPIP Task. For this project, groups of four will complete a *Distributed Partial Information Puzzle* (DPIP) [25], wherein participants work together to recreate a target Lego structure. Following the jigsaw approach used in [6], three of the participants have unique information and must work together to piece their knowledge into one combined solution (in my case, the Lego structure). The remaining participant acts as the builder, constructing the structure but without any information to start with. This task requires collaboration between each team member, as well as negotiating unique perspectives to reach the final outcome. In this task, disagreement naturally arises as participants attempt to orient together and give directions to the builder. An exam-

ple of this can be seen in Fig 2. The group thought they were correctly completing the task, but they came to an impass when there was a conflict between two members' views. While one participant (P4) saw a red block in a specific location, another participant (P2) believed it should have been an orange block. The builder places a red, removes it, and places an orange, at which point P4 rests her head in her hands as a sign of confusion. At this point, the solution is that the participants are actually referring to two separate corners; each person is correct, but their beliefs about their wall orientation needs to be adjusted. I propose moments such as this one to be potentially helpful moments for an agent to intervene.



Dialogue

P4 And there's two of [red].
P2 I have orange on top of the red.
P4 Orange?
P4 Oh no.
P4 That's not right for me.

Figure 1: Participants reach a disagreement while working on the DPIP task. Participants are labeled P1, P2, P3, and P4, from left to right. P4 sees a red square while P2 sees an orange square, and the group must re-negotiate their task beliefs before they can continue.

Intervention Agents. The interventions will be provided as speech and on-screen text to the group. Interventions will come from one of three fine-tuned versions of Meta-Llama-3-8B-Instruct: Identity Preference Optimization (IPO) [2], Proximal Policy Optimization (PPO) [18], or the Frictional Agent Alignment Framework (FAAF) [15]. IPO and PPO are well-known fine-tuning techniques; IPO is an offline contrastive approach, while PPO is an online reinforcement learning approach. FAAF is a more recent approach which fine-tunes the model to act as a collaborator in a frictive state, which makes it more aligned with the goal of the intervention.

Dialogue Policy Conditions. I will compare two dialogue policies. My baseline condition is to provide an intervention every seven utterances. This was based on a qualitative exploration between five and ten utterances, with five being too frequent and ten being too far apart. This baseline provides an explainable but naive comparison, with no meaningful or education-based policy; it is simply rule-based. The second condition provides an intervention when a disagreement is detected in the dialogue by a large language model. This condition explores the specific challenge of conflict management in collaborative learning. It is also grounded in existing literature on collaborative learning, which has found conflict to be impactful - either positively or negatively - on the group [8].

Evaluation. I will evaluate the outcomes based on three dimensions; group speed, correctness of final structure, and user experience. The group speed will simply consider the time taken to complete the task. Correctness of the structure will compare the group's final structure with the target structure. User experience will include a self-report survey [13, 14] after completing the task as well as behavioral signals for user engagement with the agent. Signals will include verbal and nonverbal items such as visual attention toward the agent and speaking over the agent [5].

LLM Disagreement Detection. Besides the user outcomes, I also must evaluate the performance of the system on the detection task. The task will be detecting disagreement in small group dialogues. One additional challenge in this task is that disagreements are often expressed *implicitly* through contradiction rather than directly stating disagreement [24]. Because of this, I aim to leverage the interpretive power in LLMs. To evaluate LLMs on this task, I will test multiple models on various datasets. I will explore the model's performance on the Delidata dataset [10]. This is a dataset of small groups performing the Wason card selection task, a well-known reasoning task [22], and includes labels for explicit group disagreement. I will also label a subset of this dataset for implicit disagreement in order to compare performance on both types of expression. Further, I will use a synthetic version of the DPIP task presented in [16]. These provide some initial evaluation on the LLM performance of detecting disagreement, which will serve as the gate for my proposed dialogue policy exploration.

3. EXPECTED CONTRIBUTIONS

This research will offer insight into the impact of intervening during disagreement as a dialogue policy. This builds off of prior work exploring group state dialogue policy, which considers unproductive negotiations as a group state [6], but isolates this one aspect for further investigation. This also provides a study outline for other specific rule-based and educationally-grounded dialogue policy explorations. As we move toward AI assistants in collaborative learning, it is critical to have explainable and grounded dialogue policy. Disagreement is an especially interesting moment because of its potential to negatively or positively impact the group; this makes it worth investigating further on its own. It is also a timely challenge, as we now have live multimodal and multiparty systems. It's important to bridge the gap between the existence of the system and years of foundational learning sciences outlining the challenges which arise in collaborative settings. I propose a human-centered AI approach, which focuses on one proposed moment of intervention known to be pivotal in collaboration and tests it in a user study with a live agent system.

4. REFERENCES

[1] J. Andrews-Todd and C. M. Forsyth. Exploring social and cognitive dimensions of collaborative problem solving in an open online simulation-based task. *Computers in Human Behavior*, 104:105759, Mar. 2020.

[2] M. G. Azar, Z. D. Guo, B. Piot, R. Munos, M. Rowland, M. Valko, and D. Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

[3] C. De Kock and A. Vlachos. I Beg to Differ: A study of constructive disagreement in online conversations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2017–2027, 2021.

[4] I. Dey, E. Doherty, R. Zhang, N. Hoang, J. B. Bush, L. Hirshfield, and S. Puntambekar. Examining support moments with a conversational ai partner. In *Proceedings of the 18th International Conference on Computer-Supported Collaborative Learning-CSCL 2025, pp. 169-177*. International Society of the Learning Sciences, 2025.

[5] I. Dey, S. Puntambekar, R. Li, D. Gengler, R. Dickler, L. M. Hirshfield, and N. Krishnaswamy. The nice framework: analyzing students' nonverbal interactions during collaborative learning. In *Pre-conference workshop on collaboration analytics at 13th international learning analytics and knowledge conference (lak 2023)*, 2023.

[6] E. Doherty, E. M. Perkoff, S. von Bayern, R. Zhang, I. Dey, M. Bodzianowski, S. Puntambekar, and L. Hirshfield. Piecing together teamwork: A responsible approach to an LLM-based educational jigsaw agent. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–17, 2025.

[7] A. C. Graesser, S. M. Fiore, S. Greiff, J. Andrews-Todd, P. W. Foltz, and F. W. Hesse. Advancing the Science of Collaborative Problem Solving. *Psychological Science in the Public Interest*, 19(2):59–92, Nov. 2018. Publisher: SAGE Publications Inc.

[8] K. A. Jehn. A qualitative analysis of conflict types and dimensions in organizational groups. *Administrative science quarterly*, pages 530–557, 1997. Publisher: JSTOR.

[9] K. A. Jehn, L. Greer, S. Levine, and G. Szulanski. The effects of conflict types, dimensions, and emergent states on group outcomes. *Group decision and negotiation*, 17(6):465–495, 2008. Publisher: Springer.

[10] G. Karadzhov, T. Stafford, and A. Vlachos. DeliData: A dataset for deliberation in multi-party problem solving. *Proceedings of the ACM on Human-Computer Interaction*, 7(CSCW2):1–25, 2023. Publisher: ACM New York, NY, USA.

[11] J. Lahti, A. Eteläpelto, and S. Siitari. Conflict as a challenge to productive learning during long-term collaboration. *Learning to Collaborate: Collaborating to Learn, Nova Scotia Publishers, New York, NY*, 2004.

[12] D. Lee, Y. Huh, and C. M. Reigeluth. Collaboration, intragroup conflict, and social skills in project-based learning. *Instructional science*, 43:561–590, 2015. Publisher: Springer.

[13] S. A. Mannan, P. Hansen, V. P. Vimal, H. N. Davies, P. DiZio, and N. Krishnaswamy. Combating spatial disorientation in a dynamic self-stabilization task using ai assistants. In *Proceedings of the 12th International Conference on Human-Agent Interaction*, pages 113–122, 2024.

[14] B. M. Muir. Trust in automation: Part i. theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics*, 37(11):1905–1922, 1994.

[15] A. Nath, C. Graff, A. Bachinin, and N. Krishnaswamy. Frictional agent alignment framework: Slow down and don't break things. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11042–11089, 2025.

[16] A. Nath, H. VanderHoeven, and N. Krishnaswamy. Craft: Grounded multi-agent coordination under partial information. *arXiv preprint arXiv:2603.25268*, 2026.

[17] N. Rummel, E. Walker, and V. Aleven. Different futures of adaptive collaborative learning support. *International Journal of Artificial Intelligence in Education*, 26(2):784–795, 2016. Publisher: Springer.

[18] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[19] C. Sun, V. J. Shute, A. Stewart, J. Yonehiro, N. Duran, and S. D'Mello. Towards a generalized competency model of collaborative problem solving. *Computers & Education*, 143:103672, 2020.

[20] M.-H. Tsai and C. Bendersky. The pursuit of information sharing: Expressing task conflicts as debates vs. disagreements increases perceived receptivity to dissenting opinions in groups. *Organization Science*, 27(1):141–156, 2016. Publisher: Informs.

[21] H. VanderHoeven, B. Bhalla, I. Khebour, A. C. Youngren, V. Venkatesha, M. Bradford, J. Fitzgerald, C. Mabrey, J. Tu, Y. Zhu, et al. Trace: Real-time multimodal common ground tracking in situated collaborative dialogues. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 40–50, 2025.

[22] P. C. Wason. Reasoning about a Rule. *Quarterly Journal of Experimental Psychology*, 20(3):273–281, Aug. 1968. Publisher: SAGE Publications.

[23] L. R. Weingart, K. A. Jehn, and K. L. Krueger. Manage intrateam conflict through collaboration. *Principles of Organizational Behavior: The Handbook of Evidence-Based Management 3rd Edition*, pages 403–427, 2023. Publisher: Wiley Online Library.

[24] J. Zhou, C. E. Hmelo-Silver, Z. Ryan, C. Stiso, D. Murphy, J. Danish, C. A. Chinn, and R. G. Duncan. Disagreeing softly: Supporting students in managing disagreement in peer critique: Disagreeing

softly: Supporting students in managing disagreement in peer critique. *International Journal of Computer-Supported Collaborative Learning*, 20(2):249–282, 2025.

[25] Y. Zhu, M. Bradford, K. Lai, T. Obiso, V. Venkatesha, J. Pustejovsky, and N. Krishnaswamy. Distributed partial information puzzles: Examining common ground construction under epistemic asymmetry. *arXiv preprint arXiv:2603.05450*, 2026.

[26] Y. Zhu, H. VanderHoeven, K. Lai, M. Bradford, C. Tam, I. Khebour, R. Brutti, N. Krishnaswamy, and J. Pustejovsky. Modeling Theory of Mind in Multimodal HCI. In *International Conference on Human-Computer Interaction*, pages 205–225. Springer, 2024.

Identifying When Support Matters Most: Longitudinal LMS Behaviors To Predict Failure Trajectories

Christopher Steadman
University of Minnesota
Minneapolis, MN, USA
stead032@umn.edu

ABSTRACT

Student retention remains a significant challenge in higher education, with early academic failure frequently serving as an initial indicator of attrition. Yet many models focus on binary outcomes that can mask the severity of academic struggle for some students. Thus, this ongoing research focuses on students who experienced multiple failures (i.e., receiving two or more D or F grades) in their first semester. We are exploring two research questions: 1) how early can we meaningfully identify students at risk of experiencing multiple failures, and 2) what behavioral indicators might signal that a student is more likely to experience multiple failures. Using longitudinal LMS data from the Unizin Data Platform for 5,094 first-year students at a large public university, we applied machine learning to 352 temporal features to identify at-risk students. Preliminary models achieved a peak recall of 0.68, identifying critical intervention windows at Week 5 and Week 11. These findings suggest that while early behavioral markers exist, mid-semester disengagement may represent a critical point for "just-in-time" intervention.

Keywords

Student retention, Academic failure, Temporal modeling

1. INTRODUCTION

The transition to college can be a challenging time for students, both on a personal and institutional level [25, 10]. These challenges can act as barriers to student success, with student retention remaining a persistent challenge for higher education institutions. It represents a critical goal for all stakeholders due to both the economic and social benefits for individuals [12] and the financial and academic stability it signals for institutions [21, 20]. Despite years of effort, retention rates have remained relatively stagnant [8], with recent data indicating that approximately 35% of US students fail to graduate from their program [6]. First-year students are particularly vulnerable to attrition with roughly 18% of students failing to return for their second year [19].

Christopher Steadman. Identifying When Support Matters Most: Longitudinal LMS Behaviors To Predict Failure Trajectories. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 855–859. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039948

Importantly, students' experiences of transition and persistence, as well as their choices about remaining in higher education, are shaped by differences in prior preparation, available support networks, and broader structural inequities [25, 3]. Understanding when and how these transition and persistence challenges emerge has therefore become as important as measuring whether students ultimately persist. Institutions are increasingly seeking ways to identify actionable moments during the semester when timely support can alter a student's trajectory, rather than relying solely on retrospective indicators, such as final grades or withdrawal status. To address these needs, institutions have turned to Learning Analytics and 'Big Data' to identify behavioral patterns that place students at risk. Early efforts in this space proved that data from Learning Management Systems (LMS), such as login frequency and resource engagement, could serve as a viable 'early warning system' [15]. More recent work has used machine learning techniques to assess how administrative, demographic, and behavioral data can be used to accurately identify and predict students' level of success or failure [4, 5]. Furthermore, there is a growing shift toward progressive modeling that leverages the temporal nature of student behavior to identify academic struggle as it unfolds, rather than after the fact [27, 17].

However, a gap remains in understanding how academic difficulty can accumulate to place students on a path toward attrition. Prior work has considered failure and attrition as binary constructs, distinguishing only between students who succeed and those who do not. While analytically convenient, this framing can obscure meaningful variation in how academic struggle unfolds, particularly for students who encounter repeated setbacks within a single term. Experiencing multiple course failures may represent a qualitatively different moment in a student's trajectory [16], one that signals compounding academic and motivational challenges rather than an isolated difficulty. This study investigates this indicator at scale, considering the relationship between students' LMS usage across courses and their risk of experiencing 'multiple failures', here defined as receiving two or more D or F grades in their first semester.

2. RELATED WORK

Early work using LMS behavioral trace data demonstrated their importance in predicting academic risk. For example, volume metrics, such as frequency and duration of logins, were found to be significant predictors of performance, accounting for as much as 31% of the variability in final grades

[18]. Subsequent work using richer datasets demonstrated that combining interaction logs with grade data and other non-LMS enrollment data could reliably be used for early outcome prediction (in some cases as soon as the 8th day of instruction) [23] and successfully used for automated intervention [1]. As data availability has increased, this work has expanded beyond descriptive analytics to include machine learning approaches that integrate demographic, administrative, and behavioral data sources in an effort to improve predictive accuracy [2, 14, 13]. These more complex datasets have allowed us to better model academic risk as a dynamic process rather than a static outcome. However, analyzing LMS behavior at an institutional level has historically been hindered by fragmented data systems that often lacked uniformity both within and between institutions. In response to this, the Unizin Consortium was developed as a collaborative effort among research universities to create a shared, standardized data infrastructure to support the analysis of student engagement at-scale [28].

Despite the substantial progress that has been made in identifying students at risk of dropout or course failure, most studies operationalize academic risk using binary outcome measures or focus on failures that occur within individual courses or programs. Such approaches provide useful indicators of eventual outcomes but may obscure differences in how academic difficulties emerge and accumulate across a student's broader educational experience. Specifically, experiencing multiple course failures in a single academic term represents one such profile. For first-year students in particular, this pattern can trigger institutional probationary policies, financial aid consequences, and reduced opportunities for academic recovery, creating a precarious academic position early in their college careers [26, 22, 7]. Despite the institutional significance, relatively little research has examined how behavioral signals associated with multiple concurrent failures develop over time.

The present study thus extends prior learning analytics work by shifting the analytic focus from predicting whether failure occurs to understanding when and how clusters of academic difficulty become detectable. Recent increases in the availability of large-scale, institutionally integrated educational data have made it possible to examine these patterns with greater temporal and contextual resolution than was previously feasible. Leveraging longitudinal LMS engagement data drawn from the Unizin Data Platform, which links behavioral and administrative records within an authentic undergraduate environment, we (1) model multiple-course failure as a distinct outcome, and (2) examine the temporal emergence of behavioral indicators that may support earlier, more targeted intervention. Guided by this perspective, we consider two research questions:

RQ1: How well can we predict that students will experience multiple failures in their first semester of college?

RQ2: How early can we make meaningful predictions about students' failure rate?

By using behavioral data related to students' LMS engagement, we aim to identify the factors that most meaningfully predict student failure with the long-term goal of providing more timely and equitable institutional intervention.

3. METHODOLOGY

Data for this study were collected from the Unizin Data Platform, which is a repository of LMS records from the 16 university members of Unizin [28]. We initially queried the database to select all students who were first enrolled in their program in Fall 2022 at a large public Midwestern university. The initial sample comprised 8,382 students. To ensure the analysis focused on a consistent first-year undergraduate cohort, a multi-step data cleaning process was applied. First, we excluded students with interaction records prior to the start of the semester to remove continuing or transfer students. Second, students with fewer than 3 graded courses were excluded as a proxy for full-time status. Additionally, we removed students enrolled in graduate-level coursework. The final analytic sample consisted of 5,094 students. Due to student privacy concerns, the original dataset used in this study is not publicly available. However, a synthetic dataset that mirrors the original data structure and the complete analysis pipeline are available on the Open Science Framework [24].

For this analysis, we defined 'failure' as a grade below a C-, based on institutional guidelines. Using this definition, we categorized students into three groups based on their Fall 2022 performance: (1) Multiple Failure (N=160), comprising students with two or more Ds or Fs; (2) Single Failure (N=321), serving as a comparative failure group for students with exactly one D or F; and (3) Non-Failure (N=4613), representing students who met all course requirements. Descriptive results indicate that students who experience multiple instances of failure in their first semester exhibit significantly lower term-to-term persistence, with only 50% continuing past their first year, compared to 92% for non-failure students. This highlights the need to understand what behaviors may be most associated with these failure experiences, so that we can better intervene before these students leave.

3.1 Feature Engineering

Features were extracted from the Unizin Data Platform to capture granular behavioral patterns. The study period covered a 16 week semester, resulting in a total of 240 base features (15 per week).

3.1.1 Base Feature Identification

We identified core behavioral metrics to serve as proxies for student engagement and self-regulation.

1. **Temporal Interaction Patterns**: To assess possible temporal interaction effects, we categorized all LMS events into four distinct time-of-day bins (12:00 AM – 6:00 AM, 6:00 AM – 12:00 PM, 12:00 PM – 6:00 PM, and 6:00 PM – 12:00 AM).

2. **Self-Regulation Proxies**: Engagement with the LMS "To-Do" list and calendar features was used to represent planning and organizational behavior. This aligns with prior research that shows that learners with higher self-regulation skills engage in greater planning behaviors than their peers [11].

3. **Volume and Intensity**: We captured the duration of active sessions, the frequency of specific actions, and the number of distinct 30-minute study sessions.

4. **Performance Indicators**: Unizin compares assignment submission timestamps to due dates to calculate submission latency, and these calculations were used to create an average weekly submission metric.

3.1.2 Feature Engineering and Aggregation

To simulate real-time predictive capabilities, raw event data were aggregated at the weekly level. To account for different dimensions of student effort, we engineered three specific types of derived features:

1. **Time-Binned Counts & Proportions**: For each time-of-day bin, we calculated both the raw count of interactions and the proportion of the week's total activity that occurred within that bin.

2. **Proportional Engagement**: We calculated the proportion of total weekly To-Do list activity relative to overall To-Do list engagement to normalize planning behavior.

3. **Cumulative Features**: To capture longitudinal trends, we created cumulative versions of count and duration metrics. For interaction counts, the four time-binned features were first summed to create a single "Total Interactions" metric for the week, which was then iteratively added to create a single "Cumulative Total Interactions" feature. All other non-proportional metrics (e.g., Student Activity Scores, Sessions) were aggregated individually. This process resulted in an additional 112 cumulative features, yielding a final dataset of 352 potential features.

To ensure the ecological validity of our predictive models, we implemented temporal constraints on the feature sets. At any given point in the semester (Week k), the model was restricted to a "snapshot" of the student's history up to that point. The total number of features available to the model at Week k is calculated as follows:

$$\text{Total Features} = (15 \times k) + (7 \times k) \quad (1)$$

In this calculation, 15 represents the base features generated each week (e.g., interaction counts, time-binned proportions), and 7 represents the longitudinal cumulative features (e.g., running totals of interactions). This approach ensures that the model never "sees" into the future and instead relies only on information that would be available to an instructor or institution in a real-time environment.

To prepare the dataset for modeling, we performed feature selection and normalization independently for each weekly model iteration. Specifically, we removed features that were strongly correlated (r>.90) to address potential multicollinearity, scaled features using a standardized scaler, and applied the Synthetic Minority Over-sampling Technique (SMOTE) [9] to the training folds to address class imbalances.

3.2 Modeling Approach

The dataset was split at the student level into 80% training data (N = 4075) and 20% testing data (N = 1019). We created 5 different randomized training and testing splits and applied a 5-fold cross-validation procedure using a Randomized Search (k=5) to each of the training datasets to tune model parameters and obtain reliable performance estimates. We utilized Random Forest and XGBoost classifiers due to their ability to handle non-linear relationships and high-dimensional data. Given that the Multiple Failure group represents a relatively small proportion of the population (3.1%), we addressed the severe class imbalance by optimizing our models for Recall (sensitivity). Recall reflects the proportion of truly "at-risk" students that the model successfully identifies. In this context, low recall would mean students experiencing multiple failures are overlooked by the system, preventing institutions from offering potential support during critical periods. Because prior research has shown that early academic setbacks can compound and contribute to attrition, missing these students (false negatives) represents the most consequential form of error.

4. CURRENT PROGRESS

Model performance indicates that meaningful detection becomes possible relatively early in the term. By Week 5, recall for the Multiple Failure group exceeded 50%, suggesting that behavioral differences associated with later academic difficulty are already observable within the first month of coursework. Performance continued to improve as more temporal data became available, reaching a peak recall of .68 by Week 15 (Figure 1).

To interpret which behaviors contributed most strongly to model predictions, we examined the SHAPley Additive Explanations (SHAP) values (Figure 2). This analysis revealed two critical windows for intervention. First, features related to late submissions in the first month remained among the strongest predictors of multiple failure even at the end of the semester. This indicates that a pattern of procrastination established in the first few weeks of college is a persistent indicator of risk. Second, interaction behavior during Weeks 8-11 emerged as highly influential. This period immediately follows midterm exams, and suggests that this drop in interaction may reflect academic disengagement following disappointing midterm results. This window may thus be an important window for increased support, especially for students already showing signs of academic difficulty.

Our analysis also revealed that not all instances of academic difficulty reflect the same underlying process. While students in the Multiple Failure group exhibit more generalized patterns of disengagement that became increasingly detectable over time, Single Failure students were frequently misclassified as Non-Failure. This suggests that failing a single course may be an idiosyncratic, localized, or course-specific challenge tied to the demands of a particular course rather than a broad behavioral shift.

5. CONTRIBUTION AND FUTURE WORK

This research offers both theoretical and practical contributions to the Educational Data Mining and Learning Sciences communities. From a learning science perspective, this work offers theoretical insights into how academic disengagement

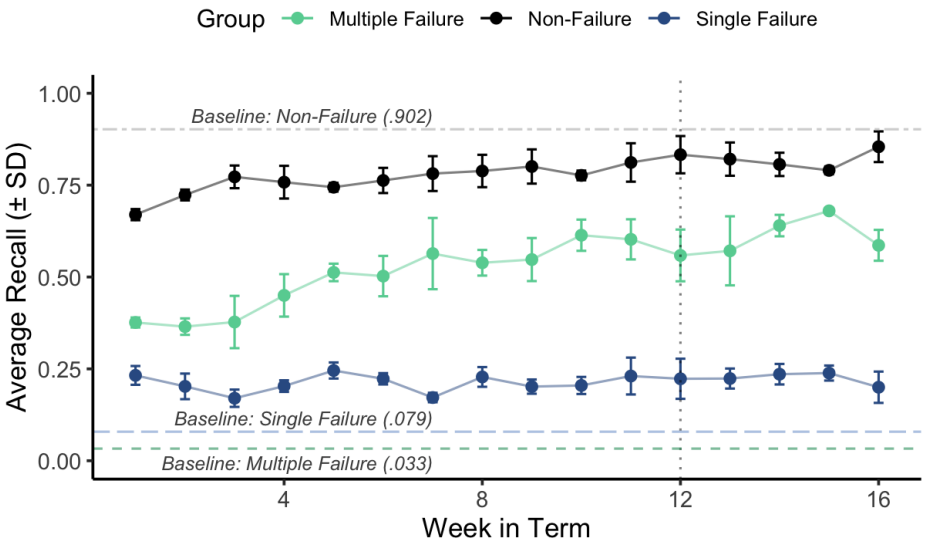


Figure 1: Weekly model recall performance by student failure outcome. Solid lines represent the average recall achieved by the model for each group, while horizontal dashed lines represent the baseline recall for each specific group, established using a stratified dummy classifier. The vertical dotted line at Week 12 marks the Thanksgiving break, corresponding to a period of atypical LMS engagement.

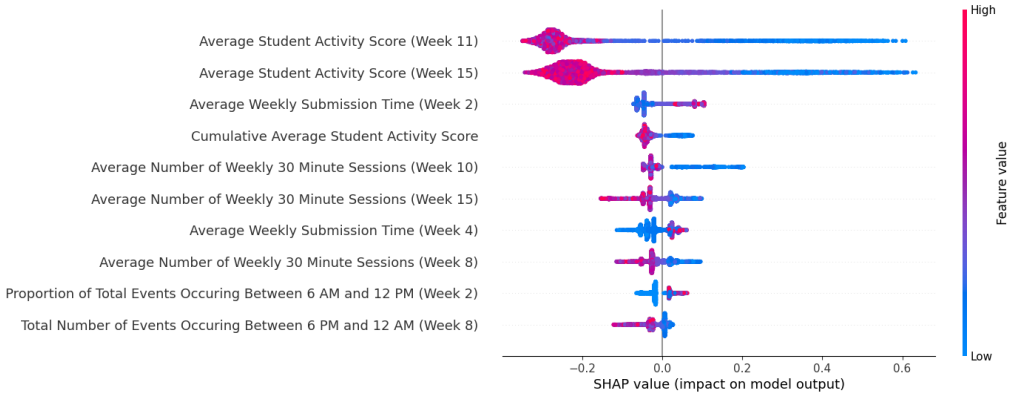


Figure 2: SHAP summary plot for the top 10 features for the Multiple Failure group

unfolds at an institutional level. Additionally, the use of the Unizin Data Platform ensures that the methodology applied to this work is scalable and provides a blueprint for standardized institution-level learning analytics. This type of cross-institutional work will be key for determining whether observed behavioral patterns can generalize to other settings.

With an eye towards generalizability, future research will look to replicate these findings across different first-year cohorts to solidify the timing of key behavioral windows. We would also like to explore additional analyses using explainable AI and deep learning methods to improve model performance and ensure that we are accurately identifying the behavioral patterns that are most predictive of repeated failure. In addition, we will be carrying out a longitudinal analysis to assess how behavioral signatures evolve over time, particularly for students in the Multiple Failure group. Assessing how behaviors change over time for students who persisted despite experiencing significant early academic setbacks could provide insights into the types of behaviors that allowed them to persist. Lastly, we plan to integrate disciplinary data to determine if certain majors or gateway courses exhibit unique engagement signatures and explore how course difficulty influences learning behavior.

The ultimate goal of this research is to support the development of proactive early warning systems. Identifying 'Multiple Failure' trajectories allows institutions to deploy limited advising resources to the struggling students most likely to drop out. By focusing on 'just-in-time' intervention, we can better identify the students who are struggling the most as they transition to college and provide them with the support they need before academic probation or permanent withdrawal occurs.

6. REFERENCES

[1] K. E. Arnold and M. D. Pistilli. Course signals at purdue: Using learning analytics to increase student success. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge (LAK '12)*, pages 267–270, New York, NY, USA, 2012. Association for Computing Machinery.

[2] L. Aulck, D. Nambi, and N. Velagapudi. Mining university registrar records to predict first-year undergraduate attrition. In *Proceedings of The 12th International Conference on Educational Data Mining*, EDM 2019, pages 9–18, Montreal, Canada, 2019. International Educational Data Mining Society.

[3] B. O. Barefoot*. Higher education's revolving door: confronting the problem of student drop out in us colleges and universities. *Open Learning: The Journal of Open, Distance and e-Learning*, 19(1):9–18, 2004.

[4] C. Beaulac and J. S. Rosenthal. Predicting university students' academic success and major using random forests. *Research in Higher Education*, 60(7):1048–1064, 2019.

[5] J. Berens, K. Schneider, S. Gortz, S. Oster, and J. Burghoff. Early detection of students at risk - predicting student dropouts using administrative student data from german universities and machine learning methods. *Journal of Educational Data Mining*, 11(3):1–41, 2019.

[6] B. Berg, B. Randolph, M. Holsapple, and D. Shapiro. Yearly progress and completion. Technical report, National Student Clearinghouse Research Center, Herndon, VA, December 2025.

[7] N. A. Bowman and N. Jang. What is the purpose of academic probation? its substantial negative effects on four-year graduation. *Research in Higher Education*, 63(8):1285–1311, 2022.

[8] G. D. Caruth. Student engagement, retention, and motivation: Assessing academic success in today's college students. *Participatory Educational Research*, 5(1):17–30, 2018.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[10] R. Deil-Amen and R. L. Turley. A review of the transition to college literature in sociology. *Teachers College Record*, 109(10):2324–2366, 2007.

[11] F. S. Kia, S. D. Teasley, M. Hatala, S. A. Karabenick, and M. Kay. How patterns of students dashboard use are related to their achievement and self-regulatory engagement. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge (LAK '20)*, pages 340–349, Frankfurt, Germany, 2020. Association for Computing Machinery.

[12] G. D. Kuh, J. Kinzie, J. H. Schuh, and E. J. Whitt. *Student success in college: Creating conditions that matter*. John Wiley & Sons, Hoboken, New Jersey, 2011.

[13] Y. C. Liew, K. Y. Lim, T. Y. Lim, C. J. Tan, K. K. Chai, and X. Deng. The effect of data transformation techniques on machine learning performance: A case study on student dropout prediction. In *2024 IEEE 5th International Conference on Pattern Recognition and Machine Learning (PRML)*, pages 1–7, Chongqing, China, 2024. IEEE.

[14] O. H. T. Lu, A. Y. Q. Huang, J. C. H. Huang, A. J. Q. Lin, and S. J. H. Yang. Applying learning analytics for the early prediction of students' academic performance in blended learning. *Journal of Educational Technology & Society*, 21(2):220–232, 2018.

[15] L. P. Macfadyen and S. Dawson. Mining lms data to develop an "early warning system" for educators: A proof of concept. *Computers & Education*, 54(2):588–599, 2010.

[16] A. J. Martin and H. W. Marsh. Fear of failure: Friend or foe? *Australian Psychologist*, 38(1):31–38, 2003.

[17] M. V. Martins, L. Baptista, J. Machado, and V. Realinho. Multi-class phased prediction of academic performance and dropout in higher education. *Applied Sciences*, 13(8):1–15, 2023.

[18] L. V. Morris, C. Finnegan, and S.-S. Wu. Tracking student behavior, persistence, and achievement in online courses. *The Internet and Higher Education*, 8(3):221–231, 2005.

[19] National Center for Education Statistics. Undergraduate retention and graduation rates. Condition of Education. U.S. Department of Education, Institute of Education Sciences, 2022. Accessed: 2026-02-10.

[20] P. O'Keeffe. A sense of belonging: Improving student retention. *College Student Journal*, 47(4):605–613, 2013.

[21] P. S. Pruett and B. Absher. Factors influencing retention of developmental education students in community colleges. *Delta Kappa Gamma Bulletin*, 81(4):32–40, 2015.

[22] L. Schudde and J. Scott-Clayton. Pell grants as performance-based scholarships? an examination of satisfactory academic progress requirements in the nation's community colleges. *Research in Higher Education*, 57(8):943–967, 2016.

[23] V. C. Smith, A. Lange, and D. R. Huston. Predictive modeling to forecast student outcomes and drive effective interventions in online community college courses. *Online Learning*, 16(3):51–61, June 2012.

[24] C. Steadman. Predicting failure trajectories using longitudinal lms behaviors. `https://osf.io/ua2q8/overview`, 2026. Accessed: 2026-06-22.

[25] P. T. Terenzini, L. I. Rendon, M. Lee Upcraft, S. B. Millar, K. W. Allison, P. L. Gregg, and R. Jalomo. The transition to college: Diverse students, diverse stories. *Research in higher education*, 35(1):57–73, 1994.

[26] V. Tinto. *Completing College: Rethinking Institutional Action*. University of Chicago Press, Chicago, IL, 2012.

[27] R. Trakunphutthirak and V. C. S. Lee. Application of educational data mining approach for student academic performance prediction using progressive temporal data. *Journal of Educational Computing Research*, 60(3):742–776, 2022.

[28] Unizin Consortium. Unizin Data Platform. Unizin Product Information, 2024. Accessed: 2026-02-10.

Identifying and Modeling Epistemic Emotions in Collaborative Problem-Solving

Sifatul Anindho
Colorado State University
sifatul.anindho@colostate.edu

Nathaniel Blanchard
Colorado State University
nathaniel.blanchard@colostate.edu

ABSTRACT

Epistemic emotions such as confusion and curiosity play a central role in learning. However, they are difficult to identify in collaborative problem-solving. Existing approaches usually rely on behavioral coding or self-reports, both of which results in contradictory empirical results in different populations and environments, raising concerns about generalization, construct validity and what constitutes the gold-standard ground truth. Here, I propose and outline my ongoing investigation of measurement and analysis of epistemic emotions in collaborative problem solving. I adopt a retrospective cued-recall framework to capture self-reported cognitive-affective states, providing a minimally disruptive and ecologically valid method of annotating internal experience. I then characterize the temporal and relational structure of these states using ordered network analysis, demonstrating how epistemic emotions exhibit cooccurrence and transition dynamics in collaborative problem-solving. Finally, I propose to systematically compare multiple representations of affect to examine how annotation methods, temporal alignment, and representational choices influence what is measured, observed, and interpreted as epistemic emotional dynamics.

Keywords

Epistemic Emotions, Collaborative Problem-Solving, Affective Computing, Ordered Network Analysis, Multimodal Modeling

1. INTRODUCTION

Epistemic emotions, such as confusion and boredom, can play a central role in learning, influencing how learner's interpret information and construct knowledge [1], and this influence can sometimes be quite complex. Baker et al. [2] showed that some of these emotions can have completely different effects on learning depending on their incidence and persistence. Confusion, for example, is an epistemic emotion often associated with impasse or difficulty. Researchers find that confusion can actually be beneficial for learning when appropriately resolved, leading to deeper understanding. But when it persists without resolution, it's shown to be detrimental in the same population groups [3]. This highlights that the impact of epistemic emotions depends not only on their presence, but also on their temporal dynamics. Therefore, developing computational models that can automatically detect and track these states in real-world learning environments is of growing interest in the EDM commmunity. Such models have the potential to power intelligent tutoring systems that provide pedagogical interventions which support productive phases of such emotions and break disruptive cycles.

Epistemic emotions often manifest differently across learning environments. Prior work has shown that both the incidence and temporal dynamics of affective states vary across learning contexts and interaction conditions, with the same emotion exhibiting different patterns and consequences depending on the task and environment [2, 3]. A relatively less explored learning environment is collaborative-problem solving (CPS). CPS is a learning paradigm when two or more individuals come together to solve problems with a shared goal [4]. In such settings, identification of epistemic emotions becomes challenging, since learning emerges through interaction, coordination, and shared reasoning among multiple individuals. In CPS, emotions are shaped by social dynamics, including negotiation, disagreement, and joint attention. As a result, these states are often subtle, context dependent, and regulated in ways that differ from individual learning scenarios. This introduces additional challenges for measurement. There is no clear gold-standard ground truth for cognitive-affective states in CPS, and internal experiences must be inferred through self-reports or external observations, both of which are inherently imperfect. In particular, social masking and regulation—where individuals suppress, exaggerate, or modulate their expressions in response to group dynamics—can lead to misalignment between observable signals and true internal states [5]. Conversely, the temporal imprecision and recall bias in self-reports complicates the alignment between subjective labels and multimodal behavioral data [6].

Here, I briefly discuss my ongoing work and outline a research roadmap for investigating how epistemic emotions in collaborative problem-solving can be more reliably measured. The central premise is that cognitive-affective states must be conceptualized as temporally dynamic, context dependent, and methodologically constructed phenomena. To

Sifatul Anindho, and Nathaniel Blanchard. Identifying and Modeling Epistemic Emotions in Collaborative Problem-Solving. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 860–864. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040002

this end, this work is organized into three components. First, I employ retrospective cued recall to capture self-reported cognitive-affective states, providing a non-intrusive and minimally disruptive measure of internal experience. Second, I analyze the temporal and relational structure of these states using ordered network analysis, revealing patterns of co-occurrence and transition that challenge static labeling assumptions. Third, I examine how different temporal resolutions and aggregation strategies influence the interpretation of these states, drawing on recent work on affective chronometry and dynamics. Together, these components aim to characterize what different forms of "ground truth" capture, and how methodological choices shape our understanding of cognitive-affective processes in collaborative learning.

2. THE PROBLEM AND STATE OF THE ART

Research in educational psychology has long argued that emotions are not peripheral to learning, but are tightly coupled with attention, motivation, self-regulation, and knowledge construction. Control-value theory, for example, frames achievement emotions as arising from learners' appraisals of control and value [7]. In parallel, research on epistemic emotions has emphasized that states such as confusion are especially relevant during complex sense-making because they arise from cognitive disequilibrium, uncertainty, and the effort to resolve gaps in understanding [8, 3]. Researchers have also shown that affect in learning could be modeled as dynamic, contextual, and tied to the learner's meaning-making process rather than as a static display of basic emotion [1].

Within EDM, this perspective motivated a substantial body of work on detecting learner affect from observations, interaction traces, and multimodal signals. A particularly influential line of research focused on building reliable observational infrastructures for studying affect in authentic educational settings. The Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) was developed to support minimally disruptive field observation of student affect and behavior, and became one of the field's most widely used protocols for collecting affect labels in classrooms and digital learning environments [9]. This foregrounded a measurement problem that is easy to underestimate in machine learning: affect detectors are only as meaningful as the labels and populations on which they are built. Ocumpaugh et al. [10] demonstrated that affect detectors trained in one population or setting may not generalize well to another, even when the nominal task appears similar. Zambrano et al. [11] compared different ground-truth measures of emotion and showed that the choice of measurement approach can materially shape the conclusions drawn from affective modeling studies.

Previous research also suggests that states such as confusion, frustration, and surprise are meaningful in how they unfold and interact over time [12]. For instance, confusion may support learning when it transitions into productive inquiry, yet hinder progress when it persists or co-occurs with disengagement. However, empirical approaches that aggregate affective experiences across entire activities or treat them as independent labels have struggled to recover consistent evidence for these theoretical dynamics [13, 14]. This disconnect points to a key limitation in current modeling approaches: by collapsing temporal sequences into static representations, they discard the relational and transitional structure that defines epistemic emotions.

These concerns become sharper in collaborative problem-solving. Compared to individual tutoring or solo problem-solving, collaborative settings introduce social regulation and coordination. Learners may mask confusion to preserve competence in front of peers, suppress frustration to maintain group harmony, or display attentiveness strategically while being disengaged internally [5]. This creates a particularly difficult setting for automatic affect detection because the mapping from observable signal to felt state is neither direct nor stable [15, 7]. For this reason, collaborative learning is not simply another application domain for affect models developed elsewhere. It is a setting that stresses the assumptions built into those models.

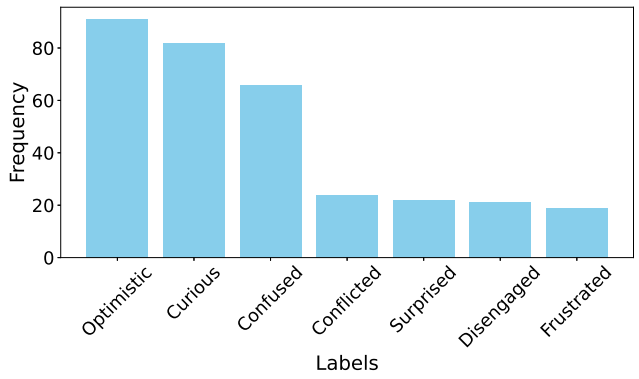
Taken together, this literature points to three key design requirements for the present work. First, measurement must be treated as a primary research problem rather than a preprocessing step, echoing prior work on observational rigor, population validity, and the instability of ground truth in affect modeling [**?**, 10, 11]. Second, epistemic emotions should be conceptualized as dynamic and relational processes rather than independent, static categories, consistent with work on affective dynamics, confusion resolution, and state transitions [3]. Third, collaborative problem-solving should be treated as a distinct affective context, where social regulation, masking, and interactional dependencies complicate the mapping between observable signals and internal states. Building on these insights, I investigate the problem of ground truth for epistemic emotions by combining construct-aligned annotation, analysis of temporal and relational structure, and systematic examination of how different measurements of affect shape our understanding of cognitive-affective processes in collaborative learning.

3. THEORETICAL FRAMING

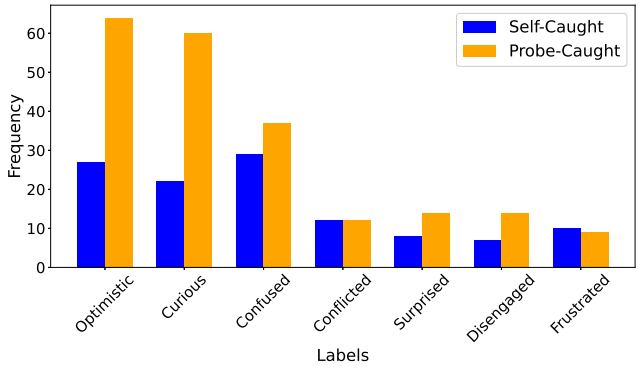
The theoretical framing of my doctoral research is grounded in the premise that epistemic emotions in collaborative problem solving are mutually inclusive and context-dependent processes, rather than static and independently observable states. Drawing from educational psychology, epistemic emotions arise from learners' interactions with uncertainty, task demands, and social context, and their role in learning depends on how they evolve over time and in relation to other states. From this perspective, the notion of a single, fixed ground truth for affect is inherently problematic. Instead, what is commonly treated as "ground truth" should be viewed as a methodological construct shaped by choices in measurement, temporal aggregation, and representation. To adress this, I ground the investigation in a sequence of studies that progressively examine how epistemic emotions can be measured and represented in collaborative problem-solving in a meaningful and valid way. Some of this work has already been carried, and the remaining is in plan.

3.1 Foundations (Completed Work)

Initially, I explored the space of internal states that arise in collaborative contexts. Using a retrospective cued-recall paradigm, I collected data of collaborative problem-solving sessions where participants were asked to verbalize their internal states while reviewing recordings of their group in-



(a) Overall frequency distribution of reported affective states



(b) Distribution across reporting mechanisms

Figure 1: Distribution of reported cognitive-affective states in our dataset

teractions. These internal monologues were transcribed and thematically analyzed, revealing a set of prevalent epistemic states—including labels such as *conflicted* and *reserved*—that are not commonly emphasized in prior work on individual learning. Analysis of n-grams and manually extracted keywords further supported the consistency of these states, and semantic similarity analysis confirmed coherence within and across identified categories. This study established both the diversity and context-specificity of epistemic emotions in collaborative settings [16].

Building on this, I developed a structured self-report framework in which participants labeled their cognitive-affective states using a more structured retrospective protocol that provided a subset of the initial labels, either spontaneously (self-caught) or in response to periodic probes (probe-caught). This enabled the collection of a more formalized dataset of retrospectively reported affective states that are temporally anchored across multiple groups. along with rich metadata including timestamps, reporting type, and participant roles. Analysis of the resulting data showed that frequency distributions of reported states were largely consistent across reporting mechanisms, with subtle but meaningful differences (Figure 1). For example, states such as *confused* were reported more frequently in self-caught conditions, while probe-caught reports tended to capture lower-arousal states, suggesting that participants may use available labels as approximations for a broader "neutral-engaged" condition. These findings highlight how measurement choices influence not only what is captured, but how affective states are interpreted. I further examined the temporal characteristics of these states by analyzing their distribution over normalized task time. Figure 2 shows a scatterplot of affective state reports over normalized collaborative task time. This analysis revealed systematic patterns: disengagement increased over time, likely reflecting fatigue or waning attention, while confusion persisted over longer durations, consistent with its characterization as a sustained epistemic state [3]. Frustration exhibited mid-task peaks, aligning with its role as an intermediate response to prolonged difficulty [1], and surprise appeared as a transient, intermittent state [2]. These results reinforce that epistemic emotions are not static events, but temporally structured processes with distinct dynamics [17].

Next, I investigated the relational structure of these epistemic emotions using ordered network analysis (ONA) (see Figure 3) [18]. This work showed that epistemic emotions form a structured system of co-occurrences and transitions that cannot be recovered from frequency-based summaries alone. In particular, I identified a stable "epistemic core" linking curiosity, optimism, and confusion, while also observing that reporting mechanisms (self-caught vs. probe-caught) and group performance (faster vs. slower groups) emphasize different aspects of this structure. For example, the roles of confusion and disengagement shift in their relationship to conflict depending on group dynamics, suggesting that the meaning of a state is contingent on its position within an interactional sequence [19].

3.2 Proposed Analysis (Future Work)

The next stage of this doctoral research centers on a systematic investigation of ground truth by comparing different annotation methods, temporal resolutions, and data modalities. Specifically, I propose a set of studies organized around three questions. First, how do different annotation methods capture epistemic emotions? I will compare self-reported states obtained through retrospective cued recall with externally observable indicators derived from multimodal behavioral data, including facial action units, vocal prosody, and conversational features. Agreement will be assessed using both categorical metrics (e.g., Cohen's κ, F1) and distributional comparisons, while also examining where and why divergence occurs. Rather than treating disagreement as error, I will analyze systematic patterns of misalignment, such as cases where internal reports indicate confusion but observable signals remain neutral, as potential evidence of social masking or regulation.

Second, how does temporal alignment influence interpretations of affect? I will investigate the temporal fidelity of different signals by comparing self-reported events with behavioral data across varying temporal windows (e.g., $\pm$1s, $\pm$5s, $\pm$10s) and aggregation strategies. This includes evaluating whether certain modalities exhibit leading or lagging relationships with reported states, and whether affective episodes are better represented as point events or temporally diffuse intervals. Building on prior work in affective chronometry, I will quantify differences in persistence, onset, and decay across representations, examining how these properties shift under different temporal assumptions.

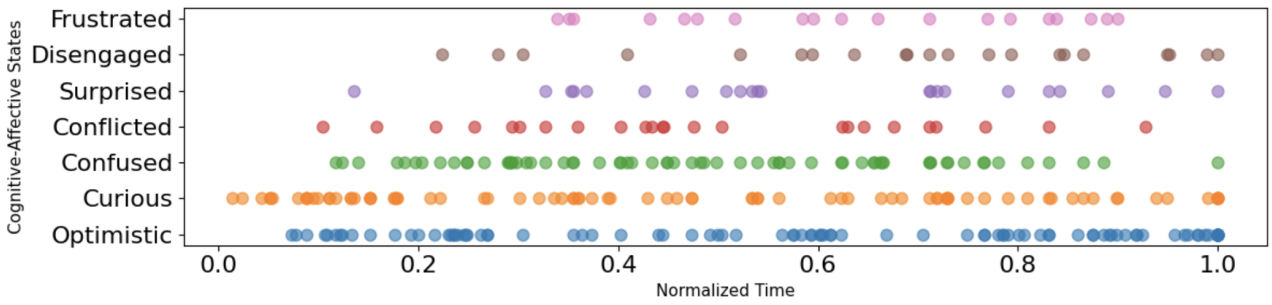


Figure 2: Distribution of affective state reports over normalized collaborative session time

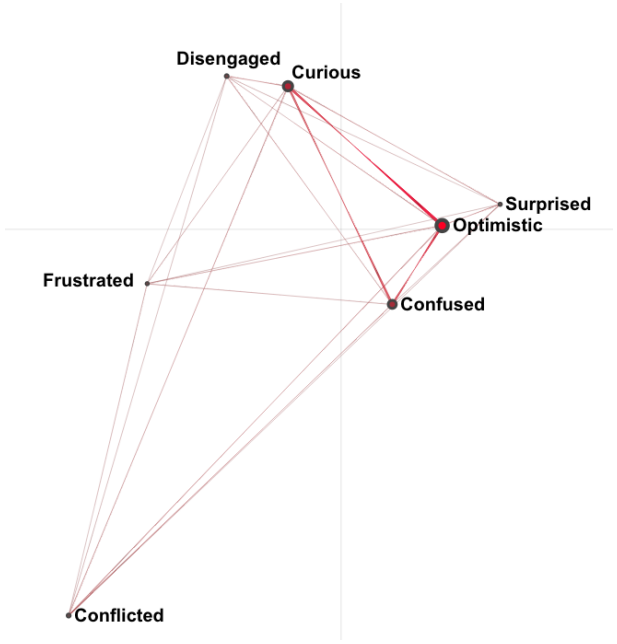


Figure 3: ONA network for affective states across all CPS groups. The first two dimensions explain 34.3% and 21.3% of the variance

Third, how do representational choices shape conclusions about affective dynamics? Using ordered network analysis and related sequence-based methods, I will construct relational representations of affect from each operationalization (e.g., raw self-reports, temporally aggregated labels, multimodal-derived signals) and compare their structural properties. This includes examining differences in cooccurrence patterns, transition probabilities, and higher-order structures such as stable "cores" or hubs of interaction. By comparing these structures across representations, I aim to identify which aspects of affective dynamics are robust to measurement choices and which are artifacts of specific annotation or aggregation strategies.

Across these studies, the goal is not to identify a single "correct" ground truth, but to characterize the space of plausible representations and the conditions under which they converge or diverge. In doing so, this work treats ground truth as a multi-faceted construct that must be interrogated empirically. The outcome is a framework for understanding how methodological decisions—ranging from annotation protocols to temporal segmentation—shape both the analysis and interpretation of cognitive-affective states in collaborative learning.

4. CONTRIBUTION AND IMPACT

This research aims to contribute to both the learning sciences and computer science communities by reframing the problem of affect modeling as a question of measurement, representation, and validity.

From a learning sciences perspective, this work advances the study of epistemic emotions in collaborative settings by emphasizing construct validity and temporal dynamics. By leveraging retrospective cued recall, it provides a minimally disruptive yet theoretically grounded approach to capturing internal states in naturalistic environments. Further, by analyzing the temporal and relational structure of these states, this work contributes to a deeper understanding of how epistemic emotions unfold during collaborative problem-solving, moving beyond frequency-based accounts toward process-oriented representations. More broadly, it contributes to ongoing discussions in EDM regarding the nature of ground truth, the role of measurement in learning analytics, and the interpretation of affective data.

From a computer science perspective, this work frames affect modeling as a data-centric machine learning problem, where the primary challenge lies in the definition and reliability of supervision signals rather than model capacity alone. By systematically examining how different annotation strategies, temporal resolutions, and representations influence downstream analysis, this research provides insights into learning under weak, subjective, and temporally misaligned supervision. These challenges are not unique to affect modeling, but arise in many real-world machine learning settings involving human-centered data. As such, this work contributes to broader efforts in machine learning to understand label uncertainty, temporal misalignment, and the relationship between data collection methods and model behavior.

Finally, this research has implications for the design of affect-aware systems in education. By clarifying what different forms of "ground truth" capture and where they fall short, this work provides a foundation for developing models and systems that are more interpretable, robust, and aligned with the constructs they aim to represent. In doing so, it can

support the development of educational technologies that respond not only to observable behavior, but to the underlying cognitive-affective processes that drive learning.

5. REFERENCES

[1] Sidney D'Mello and Art Graesser. Dynamics of affective states during complex learning. *Learning and Instruction*, 22(2):145–157, 2012.

[2] Ryan SJd Baker, Sidney K D'Mello, Ma Mercedes T Rodrigo, and Arthur C Graesser. Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive–affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies*, 68(4):223–241, 2010.

[3] Sidney D'Mello, Blair Lehman, Reinhard Pekrun, and Art Graesser. Confusion can be beneficial for learning. *Learning and Instruction*, 29:153–170, 2014.

[4] Friedrich Hesse, Esther Care, Juergen Buder, Kai Sassenberg, and Patrick Griffin. A framework for teachable collaborative problem solving skills. In *Assessment and teaching of 21st century skills: Methods and approach*, pages 37–56. Springer, 2014.

[5] Marion K Underwood. Peer social status and children's understanding of the expression and control of positive and negative emotions. *Merrill-Palmer Quarterly (1982-)*, pages 610–634, 1997.

[6] Ulrich W Ebner-Priemer, Janice Kuo, Stacy Shaw Welch, Tanja Thielgen, Steffen Witte, Martin Bohus, and Marsha M Linehan. A valence-dependent group-specific recall bias of retrospective self-reports: A study of borderline personality disorder in everyday life. *The Journal of nervous and mental disease*, 194(10):774–779, 2006.

[7] Reinhard Pekrun. The control-value theory of achievement emotions: Assumptions, corollaries, and implications for educational research and practice. *Educational psychology review*, 18(4):315–341, 2006.

[8] Arthur C Graesser. Emotions are the experiential glue of learning environments in the 21st century. *Learning and Instruction*, 70:101212, 2020.

[9] Jaclyn Ocumpaugh, Ryan S. Baker, and Ma. Mercedes T. Rodrigo. Baker rodrigo ocumpaugh monitoring protocol (bromp) 2.0 technical and training manual. Technical report, Teachers College, Columbia University and Ateneo Laboratory for the Learning Sciences, New York, NY, USA and Manila, Philippines, 2015.

[10] Jaclyn Ocumpaugh, Ryan Baker, Sujith Gowda, Neil Heffernan, and Cristina Heffernan. Population validity for educational data mining models: A case study in affect detection. *British Journal of Educational Technology*, 45(3):487–501, 2014.

[11] Andres Felipe Zambrano, Nidhi Nasiar, Jaclyn Ocumpaugh, Alex Goslen, Jiayi Zhang, Jonathan Rowe, Jordan Esiason, Jessica Vandenberg, and Stephen Hutt. Says who? how different ground truth measures of emotion impact student affective modeling. *International Educational Data Mining Society*, 2024.

[12] Sidney D'Mello and Art Graesser. The half-life of cognitive-affective states during complex learning. *Cognition & Emotion*, 25(7):1299–1308, 2011.

[13] Shamya Karumbaiah, Ryan S Baker, and Jaclyn Ocumpaugh. The case of self-transitions in affective dynamics. In *International conference on artificial intelligence in education*, pages 172–181. Springer, 2019.

[14] Jaclyn Ocumpaugh, Nidhi Nasiar, Andres Felipe Zambrano, Alex Goslen, Jessica Vandenberg, Jordan Esiason, Jonathan Rowe, and Stephen Hutt. Refocusing the lens through which we view affect dynamics: The skills, difficulty, value, efficacy and time model. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 192–203, 2025.

[15] James J Gross. Emotion regulation: Current status and future prospects. *Psychological inquiry*, 26(1):1–26, 2015.

[16] Sifatul Anindho, Videep Venkatesha, Mariah Bradford, Anne M Cleary, and Nathaniel Blanchard. An exploration of internal states in collaborative problem solving. In *International Conference on Human-Computer Interaction*, pages 135–150. Springer, 2025.

[17] Sifatul Anindho, Videep Venkatesha, and Nathaniel Blanchard. A methodological framework for capturing cognitive-affective states in collaborative learning. *arXiv preprint arXiv:2507.01166*, 2025.

[18] Yuanru Tan, Andrew R Ruis, Cody Marquart, Zhiqiang Cai, Mariah A Knowles, and David Williamson Shaffer. Ordered network analysis. In *International conference on quantitative ethnography*, pages 101–116. Springer, 2022.

[19] Sifat Anindho, Videep Venkatesha, Jaclyn Ocumpaugh, and Nathaniel Blanchard. Ordered network analysis of epistemic emotions during collaborative problem solving. In *Artificial Intelligence in Education*, Lecture Notes in Computer Science, 2026. In press.

Modeling Epistemic Vigilance in Collaborative Problem Solving: A Multimodal Approach Using Social Deduction as a Proxy Testbed

Videep Venkatesha
Colorado State University

Nathaniel Blanchard
Colorado State University

ABSTRACT

AI agents that support collaborative learning must detect reasoning failures as they unfold: misconceptions propagating unchecked, weak claims accepted without scrutiny, and participants reasoning actively but reaching wrong conclusions. Building such detection models requires datasets where these failures occur and are labeled and yet existing collaborative datasets capture only idealized cooperation under shared epistemic goals. This paper presents a doctoral research program addressing this gap through three integrated contributions: (1) a multi-layer annotation framework that traces communicative acts through belief updates to epistemic vigilance assessments, operationalizing constructs from cognitive science for behavioral annotation; (2) a proxy testbed using social deduction games where unreliable information enters group reasoning with known provenance, enabling traceable study of how epistemic failures propagate; and (3) multi-person egocentric sensing that captures participant-relative signals unavailable to traditional exocentric setups. This work extends my prior research on propositional extraction, common ground tracking, deliberation chain modeling, and automated annotation in collaborative settings, building toward an integrated system that can monitor the quality of group reasoning in real time. Pilot data from a 6-player session demonstrates that failed vigilance, belief fragmentation, and traceable misinformation effects emerge naturally and are captured by the framework.

Keywords

epistemic vigilance, collaborative problem solving, multimodal learning analytics, Theory of Mind, belief dynamics

1. PROBLEM AND STATE OF THE ART

Collaborative problem solving (CPS) is a core competency in education that involves sharing information, building common ground, reasoning about evidence, forming and revising beliefs, and coordinating action through communication [15, 18]. For collaboration to be productive, participants must evaluate each other's contributions, weigh evidence, update beliefs, and coordinate decisions often under genuine uncertainty, especially when there is no single clear answer and the task is open-ended. The quality of this evaluation process, what Sperber et al. [16] term epistemic vigilance, is what separates productive collaboration from groups that converge on misconceptions or fragment into incompatible understandings.

The challenge for educational data mining is that existing collaborative datasets overwhelmingly capture idealized cooperation. Corpora such as the DELI dataset [8] and frameworks like the General Competency Model [18] operationalize CPS under shared epistemic goals, where all participants work toward a correct answer. Real classrooms are messier where students unknowingly spread misconceptions, assert incorrect solutions with unwarranted confidence, freeload, or dominate discussions unproductively [3]. These dynamics of unreliable contributions, the natural heterogeneity of contributor quality, are pervasive in educational settings [9, 14] yet rare and unlabeled in controlled datasets. Without data where epistemic failures occur with known ground truth, building detection models is intractable.

A second gap concerns sensing modality. Current multimodal learning analytics for collaboration rely predominantly on exocentric sensing with overhead cameras and external microphones [1]. Egocentric sensing offers the potential to capture participant-relative signals that exocentric setups cannot reliably recover: whether a listener is attending to the speaker during a critical claim, or scanning others' reactions before updating a belief. Whether these first-person signals carry additional information about the social-cognitive constructs relevant to CPS, and epistemic vigilance in particular, is an open empirical question that this work aims to investigate.

1.1 Prior Work: Building Toward Epistemic Monitoring

My doctoral research has progressively built the computational infrastructure needed to model collaborative reasoning. This trajectory establishes both the technical foundations and the motivating gaps that the current work addresses.

Extracting what groups discuss. I developed methods for extracting task-relevant propositions from natural speech in collaborative settings, using cross-encoder architectures

Videep Venkatesha, and Nathaniel Blanchard. Modeling Epistemic Vigilance in Collaborative Problem Solving: A Multimodal Approach Using Social Deduction as a Proxy Testbed. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 865–869. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039976

adapted from coreference resolution [22]. This work established that the semantic content of group dialogue can be recovered automatically, even from noisy overlapping speech, and was later extended to handle automated transcription with minimal degradation [21]. These methods became a core component of TRACE, a real-time multimodal common ground tracking system that integrates speech, gesture, gaze, and object detection to model the shared beliefs of collaborating groups as they emerge [19].

Modeling how discussions evolve. I contributed to work on linking deliberation chains in collaborative dialogues, modeling the causal structure from earlier utterances to probing questions that drive group reasoning forward [13]. This work formalized the sequential structure of collaborative discourse, showing that probing interventions and their causal antecedents can be automatically clustered using joint learning frameworks.

Understanding the limits of automation. Most recently, I evaluated 13 LLMs across six model families on the task of annotating pedagogical moves in educational dialogue [20]. A key finding was that models achieve moderate agreement on content-focused categories with clear surface markers but fail on categories defined by pedagogical function, precisely the categories (like CPS facets) that require understanding speaker intent and social dynamics rather than surface text patterns.

The gap this work addresses. Collectively, this prior work can track *what* groups discuss and *whether* they reach agreement, but it cannot assess *whether the reasoning that led there was sound*. A group may converge on a shared belief through careful evidence evaluation or through uncritical acceptance of a confident but wrong claim. Current systems, including those I have helped built, treat convergence as success without evaluating the epistemic quality of the path. The present work addresses this gap directly, introducing a framework and testbed for studying how groups evaluate (or fail to evaluate) the reliability of contributions.

2. THEORETICAL FRAMING AND METHODOLOGY

2.1 Theoretical Foundations

The research integrates three theoretical frameworks. *Epistemic vigilance* [16] provides the overarching lens: the cognitive mechanism for evaluating incoming information by assessing source reliability and content plausibility rather than accepting claims uncritically. In collaborative learning, failed epistemic vigilance manifests as misconceptions spreading through groups, weak arguments being accepted without challenge, and overconfident errors going undetected.

Bayesian Theory of Mind [2] provides the belief-modeling framework, formalizing how observers infer others' mental states from observed behavior. In my framework, players report first-order beliefs (what they believe about others' roles) and second-order beliefs (what they think others believe about them), grounding Theory of Mind measurement in self-report data that can serve as supervision signals for computational models.

Argumentative Theory of Reasoning [12] motivates the focus on persuasion dynamics. Mercier and Sperber argue that reasoning quality in collaborative settings depends critically on the group's argumentative practices: whether claims are challenged, whether evidence is demanded, and whether the group evaluates contributions on epistemic merit rather than social dynamics.

2.2 Testbed: Social Deduction as a CPS Environment

I use the social deduction game Secret Hitler as a proxy testbed where players are secretly assigned cooperative or adversarial roles and engage in structured rounds of discussion, voting, and policy enactment. For the majority of players, the game is collaborative problem solving: pooling incomplete evidence, evaluating reliability of claims, and coordinating decisions under genuine uncertainty [15, 18]. A minority of players are structurally incentivized to introduce unreliable information such as asserting false claims with confidence, and undermining group reasoning. These contributions are functionally equivalent to the misconceptions and overconfident errors in collaboration. The epistemic challenge is shared, even though the underlying intent differs. Crucially, every piece of unreliable information has known provenance (role assignments provide ground truth), and the game enforces the information asymmetry that the hidden profile literature identifies as the hardest aspect of real collaboration [11, 17].

2.3 Multi-Layer Annotation Framework

The framework is hierarchical, with each layer targeting a progressively deeper dimension of group social-cognitive dynamics.

Layer 1: Communicative Acts. Each utterance is classified using the six-category persuasion strategy taxonomy from Lai et al. [10]: Identity Declaration, Accusation, Interrogation, Call for Action, Defense, and Evidence (Krippendorff's $\alpha > 0.6$). This layer draws on my prior work in propositional extraction [22] and epistemic move classification [19], extending those methods to a setting where communicative acts include deliberate misinformation. Categories like Interrogation and Defense parallel the questioning and justification behaviors that characterize productive student reasoning, the moments where a collaborator challenges a claim or defends their position with evidence, rather than accepting contributions uncritically.

Layer 2: Player-Reported Belief States. After each round, players report first-order beliefs about every other player's role (yielding calibration data against ground truth) and second-order beliefs about what others believe about them (enabling direct measurement of Theory of Mind accuracy [2]). These reports serve as supervision signals for models that must eventually infer beliefs from observable behavior alone. In the classroom parallel, this captures whether a collaborator can identify when someone else is not contributing productively to the group, whether through providing misleading information, dominating the discussion, or steering the group in an unproductive direction.

Layer 3: Epistemic Vigilance. Inferred post-hoc using Lay-

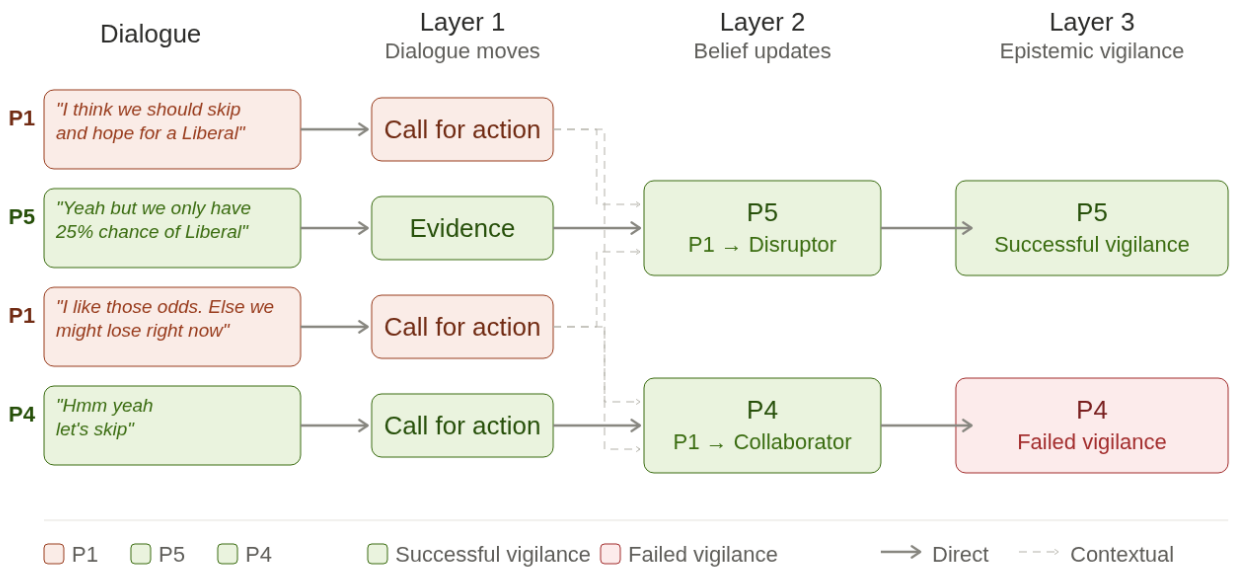


Figure 1: Three-layer annotation of a discussion phase. Utterances are classified by communicative function (Layer 1), linked to belief updates (Layer 2), and assessed against ground truth as successful or failed epistemic vigilance (Layer 3). Color indicates true role (red = adversarial, green = collaborative; hidden during play).

ers 1 and 2 together with ground-truth role assignments [16]. For each communicative act, we assess whether the receiver's belief update was warranted: *successful vigilance* (belief moved toward truth) or *failed vigilance* (belief moved away from truth). The cascading structure of communicative act → belief update → vigilance assessment aims to mirrors what a classroom agent must track, extending the common ground tracking paradigm from my prior work [19]. This layer captures not just whether students can identify unreliable contributions, but whether they actually succeeded in doing so, and by examining which communicative behaviors (Layer 1) precede successful vs. failed vigilance, we can study what kinds of group dynamics lead collaboration to succeed or break down.

Figure 1 illustrates this cascading structure with an excerpt from a pilot session. P1 (true role: Disruptive) advocates for an irrational move of skipping a turn and hoping for a favorable outcome. P5 correctly objects by presenting evidence that the odds are poor, updating their belief that P1 is a Disruptor showing successful vigilance. P4, however, accepts P1's reasoning and agrees to skip, maintaining the belief that P1 is a trustworthy Collaborator despite the weak justification: failed vigilance. The same exchange thus produces both outcomes in different receivers, and the framework captures exactly where and why the group's reasoning diverged.

2.4 Sensing Infrastructure

All players wear Meta Aria Gen 1 research glasses [5], capturing egocentric RGB video, eye-tracking data, and IMU signals. An overhead camera provides a synchronized exocentric reference view, producing $n + 1$ synchronized video streams per session wehre n is the numebr of players. This multi-person egocentric configuration enables gaze-derived features aligned with our annotation layers: gaze target identity at utterance onset, scanpath structure [4] during successful vs. failed vigilance episodes, and group-level attention divergence computed across all players which is a measure unavailable in single-viewpoint datasets [7, 6].

2.5 Research Trajectory

My research proceeds in three phases. *Phase 1 (current): Framework design and pilot validation.* I have designed the annotation framework and conducted pilot sessions to verify that target phenomena emerge naturally. *Phase 2: Data collection and annotation.* I will conduct additional sessions to build a dataset sufficient for computational modeling, adapting the annotation methods from my propositional extraction and deliberation chain work [22, 13] to this new setting. My evaluation of LLM annotation capabilities [20] will inform which layers can be partially automated and which require human coding. *Phase 3: Computational modeling.* Using the annotated multimodal data, I will train models that predict Layer 3 vigilance outcomes from observable behavioral features, systematically comparing egocentric-only, exocentric-only, and fused feature sets, building on the multimodal fusion architecture of TRACE [19].

3. PROGRESS TO DATE

Pilot data from two 6-player, 9-round session demonstrates that the target phenomena emerge naturally. Figure 2(a) tracks four collaborative players' evolving beliefs about P5, whose true role was Hitler/Disruptive. Despite 9 rounds

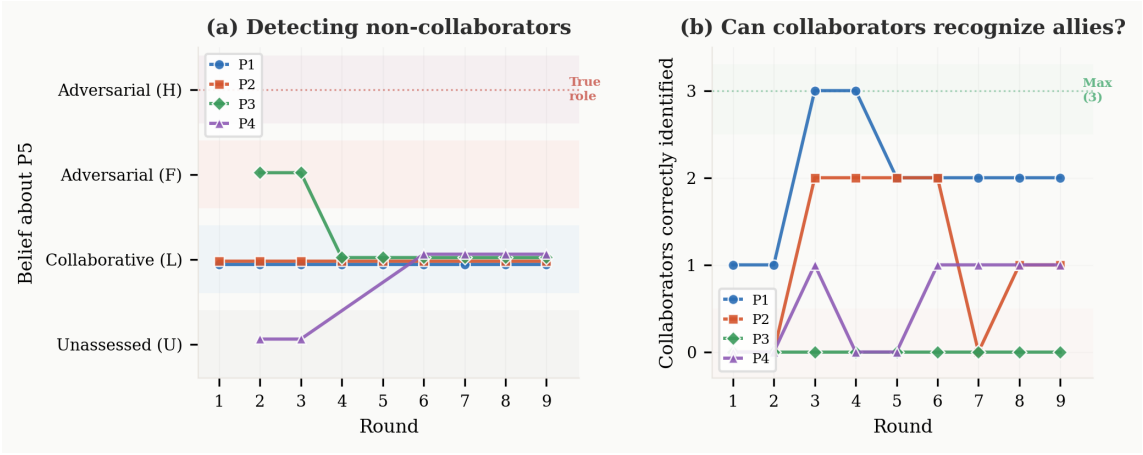


Figure 2: (a) Four collaborative players' beliefs about P5 (true role: Disruptive) across rounds. No collaborator ever correctly classified P5, illustrating failed vigilance. (b) Number of fellow collaborators each player correctly identified per round (max = 3).

of discussion and evidence, no collaborator ever correctly identified P5. P3 initially suspected P5 (Rounds 2–3) but reversed toward a collaborative classification by Round 4, constituting failed vigilance: P3 possessed a partially correct signal but abandoned it, likely in response to P5's communicative acts. Figure 2(b) reveals a complementary pattern where P3 correctly identified zero allies across all 9 rounds, placing every collaborator in the adversarial category which is the profile of a group member who is actively reasoning but systematically misdirected, analogous to a student who is engaged but whose contributions fragment rather than strengthen shared understanding [3]. Distinguishing active-but-counterproductive participation from constructive collaboration is precisely the capability the Layer 3 assessment operationalizes. P1, by contrast, peaked at identifying all three allies (Rounds 3–4) before declining, suggesting group-level belief fragmentation as unreliable information accumulated.

4. EXPECTED CONTRIBUTIONS AND IMPACT

For learning sciences. The annotation framework provides a structured methodology for studying how unreliable information propagates through collaborative groupsa phenomenon widely documented [3, 9] but rarely studied with the temporal granularity and ground truth this testbed provides. The operationalization of epistemic vigilance as a measurable construct in group behavioral data offers a new lens for understanding when and why collaborative reasoning fails.

For computer science. The testbed and dataset enable development of multimodal models for detecting reasoning failures in group settings. The multi-person egocentric sensing configuration provides a template for capturing participant-relative signals in small-group research, and the systematic comparison of egocentric vs. exocentric features will inform sensing design for future learning analytics systems.

For the broader EDM/AIED community. This work addresses a fundamental bottleneck: the scarcity of labeled data capturing collaborative dynamics beyond idealized cooperation. By providing a proxy testbed where epistemic failures occur frequently, with known ground truth, and with rich multimodal capture, the research accelerates model development that would be impractical to iterate on in instructional settings [18]. Combined with my prior work on propositional extraction, common ground tracking, and automated annotation evaluation, this research will produce an integrated pipeline: from sensing raw multimodal signals, through extracting propositional content, to assessing the epistemic quality of group reasoning—the full trajectory a collaboration monitoring agent would need to traverse.

5. REFERENCES

[1] K. Ahuja, D. Kim, F. Xhakaj, V. Varga, A. Xie, S. Zhang, J. E. Townsend, C. Harrison, A. Ogan, and Y. Agarwal. Edusense: Practical classroom sensing at scale. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–26, 2019.

[2] C. L. Baker, J. Jara-Ettinger, R. Saxe, and J. B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4):0064, 2017.

[3] B. Barron. When smart groups fail. *The journal of the learning sciences*, 12(3):307–359, 2003.

[4] R. Dewhurst, M. Nyström, H. Jarodzka, T. Foulsham, R. Johansson, and K. Holmqvist. It depends on how you look at it: Scanpath comparison in multiple dimensions with multimatch, a vector-based approach. *Behavior research methods*, 44(4):1079–1100, 2012.

[5] J. Engel, K. Somasundaram, M. Goesele, A. Sun, A. Gamino, A. Turner, A. Talattof, A. Yuan, B. Souti, B. Meredith, et al. Project aria: A new tool for egocentric multi-modal ai research. *arXiv preprint arXiv:2308.13561*, 2023.

[6] K. Grauman, A. Westbury, L. Torresani, K. Kitani, J. Malik, T. Afouras, K. Ashutosh, V. Baiyya, S. Bansal, B. Boote, et al. Ego-exo4d: Understanding

skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024.

[7] K. Grauman, M. Wray, A. Fragomeni, J. P. Munro, W. Price, P. Arbelaez, D. Crandall, D. Damen, G. M. Farinella, B. Ghanem, et al. Around the world in 3,000 hours of egocentric video. In *Computer Vision and Pattern Recognition (CVPR)*, 2022.

[8] G. Karadzhov, T. Stafford, and A. Vlachos. Delidata: A dataset for deliberation in multi-party problem solving. *Proceedings of the ACM on Human-Computer Interaction*, 7(CSCW2):1–25, 2023.

[9] S. J. Karau and K. D. Williams. Social loafing: A meta-analytic review and theoretical integration. *Journal of personality and social psychology*, 65(4):681, 1993.

[10] B. Lai, H. Zhang, M. Liu, A. Pariani, F. Ryan, W. Jia, S. A. Hayati, J. Rehg, and D. Yang. Werewolf among us: Multimodal resources for modeling persuasion behaviors in social deduction games. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6570–6588, 2023.

[11] L. Lu, Y. C. Yuan, and P. L. McLeod. Twenty-five years of hidden profiles in group decision making: A meta-analysis. *Personality and Social Psychology Review*, 16(1):54–75, 2012.

[12] H. Mercier and D. Sperber. Why do humans reason? arguments for an argumentative theory. *Behavioral and brain sciences*, 34(2):57–74, 2011.

[13] A. Nath, V. Venkatesha, M. Bradford, A. Chelle, A. C. Youngren, C. Mabrey, N. Blanchard, and N. Krishnaswamy. "any other thoughts, hedgehog?" linking deliberation chains in collaborative dialogues. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 5297–5314, 2024.

[14] S. L. Piezon and W. D. Ferree. Perceptions of social loafing in online learning groups: A study of public university and us naval war college students. *International Review of Research in Open and Distributed Learning*, 9(2):1–17, 2008.

[15] J. Roschelle and S. D. Teasley. The construction of shared knowledge in collaborative problem solving. In *Computer supported collaborative learning*, pages 69–97. Springer, 1995.

[16] D. Sperber, F. Clément, C. Heintz, O. Mascaro, H. Mercier, G. Origgi, and D. Wilson. Epistemic vigilance. *Mind & language*, 25(4):359–393, 2010.

[17] G. Stasser and W. Titus. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of personality and social psychology*, 48(6):1467, 1985.

[18] C. Sun, V. J. Shute, A. Stewart, J. Yonehiro, N. Duran, and S. D'Mello. Towards a generalized competency model of collaborative problem solving. *Computers & Education*, 143:103672, 2020.

[19] H. VanderHoeven, B. Bhalla, I. Khebour, A. C. Youngren, V. Venkatesha, M. Bradford, J. Fitzgerald, C. Mabrey, J. Tu, Y. Zhu, et al. Trace: Real-time multimodal common ground tracking in situated collaborative dialogues. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 40–50, 2025.

[20] V. Venkatesha, S. Anindho, E. Seefried, and N. Blanchard. Using LLMs to annotate pedagogical moves: You know what i mean? In *Proceedings of the 27th International Conference on Artificial Intelligence in Education (AIED 2026)*. Springer, 2026. Accepted for publication.

[21] V. Venkatesha, M. Bradford, and N. Blanchard. Dude, where's my utterance? evaluating the effects of automatic segmentation and transcription on cps detection. In *International Conference on Artificial Intelligence in Education*, pages 144–151. Springer, 2025.

[22] V. Venkatesha, A. Nath, I. Khebour, A. Chelle, M. Bradford, J. Tu, J. Pustejovsky, N. Blanchard, and N. Krishnaswamy. Propositional extraction from natural speech in small group collaborative tasks. *International Educational Data Mining Society*, 2024.

Modeling Learner-AI Interaction in Time-Constrained Programming Tasks

Omkar Joshi
Centre for Educational Technology, Indian Institute of Technology Bombay, Mumbai, India
omkar.joshi21@gmail.com

ABSTRACT

The increasing use of AI coding assistants in programming environments is transforming how learners approach problem solving, particularly under time-constrained conditions. While prior research has examined perceptions and performance outcomes, there is limited understanding of how learners interact with AI tools at a behavioral level. This doctoral research adopts a data-driven approach to model learner-AI interaction in programming tasks, grounded in the theoretical framework of Self-Regulated Learning (SRL). The study collects fine-grained interaction logs from undergraduate and early-graduate computer science students completing algorithmic programming tasks with an AI coding assistant under both timed and untimed conditions. By capturing sequences of learner actions and AI queries, the research aims to (1) characterize AI usage patterns using process mining and sequential analysis, (2) model how time pressure alters behavioral transitions between coding, debugging, and AI-assisted activities, (3) identify distinct learner profiles through clustering of interaction features, and (4) develop and validate behavioral proxies for cognitive engagement and help-seeking using convergent analysis with self-report measures and qualitative examination of interaction sessions. Preliminary studies indicate a shift toward outcome-oriented behaviors and varied reliance on AI under time pressure, motivating the need for systematic modeling of interaction processes. The expected contributions include new representations of learner-AI interaction, validated methods for operationalizing learning constructs in AI-mediated environments, and empirical insights into how time constraints shape AI-assisted programming behavior.

Keywords

AI-assisted programming, learner behavior modeling, cognitive engagement, time-constrained programming

1. INTRODUCTION

Omkar Joshi. Modeling Learner-AI Interaction in Time-Constrained Programming Tasks. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 870–874. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039967

The increasing integration of generative artificial intelligence (GenAI) tools into programming environments is transforming how learners approach problem solving. AI coding assistants now provide real-time support for code generation, debugging, and explanation, enabling learners to complete tasks more efficiently [4, 14]. While these tools offer clear productivity benefits, their impact on learning processes remains insufficiently understood, particularly in time constrained contexts such as competitive programming, technical interviews, and timed assessments, where learners must balance speed, correctness, and depth of understanding under pressure. Time pressure interacts with AI availability in ways that may fundamentally alter learner behavior. Preliminary evidence suggests that under time constraints, learners shift toward outcome-oriented strategies, using AI to generate or regenerate complete solutions rather than engaging in systematic reasoning [9]. However, these observations are based primarily on self-reports and outcome-based measures. The fine-grained behavioral dynamics of how learners integrate AI assistance into their problem-solving processes - when they seek help, how they transition between independent work and AI-assisted activity, and how they regulate these decisions, remain largely unexamined. This gap is particularly significant for the Educational Data Mining (EDM) community, where understanding learning processes through behavioral data is a central concern [2]. This doctoral research addresses this gap by developing a data-driven approach to modeling learner-AI interaction in programming tasks, grounded in the framework of SRL [16]. The study collects fine-grained interaction logs from computer science students completing algorithmic programming tasks with an AI coding assistant under both timed and untimed conditions. Using process mining, sequential analysis, and clustering techniques, the research aims to characterize AI usage patterns, model temporal dynamics of assistance-seeking, identify distinct learner profiles, and develop validated behavioral proxies for constructs such as cognitive engagement and help-seeking.

2. RELATED WORK

2.1 Learning Analytics in Programming

EDM research has extensively used fine-grained interaction logs, code edits, compilations, submissions, and error events to model learner behavior in programming environments. Techniques such as sequence mining, process modeling, and clustering have been applied to identify problem solving strategies, detect learning patterns, and predict performance [6, 10, 12, 13]. These studies demonstrate the value of be-

havioral data in uncovering latent learning processes beyond final outcomes. However, this body of work assumes environments where learners interact primarily with compilers, editors, and instructional resources. The emergence of AI coding assistants introduces a fundamentally different form of interaction, where learners can offload core cognitive tasks such as code generation and debugging to an AI system, necessitating new analytical approaches that account for AI-mediated actions within behavioral sequences.

2.2 AI-Assisted Programming and Learner Behavior

Recent studies have examined AI coding assistants in programming education, finding that such tools can reduce task completion time and lower barriers to entry, but may also encourage over-reliance, superficial engagement, and reduced opportunities for deep learning [4, 11, 14]. Learners frequently use AI to generate complete solutions or iterate through alternatives, especially under performance pressure [9]. While valuable, these findings are largely based on self-reports, controlled experiments, or outcome-based measures. Comparatively little work captures and analyzes fine-grained interaction traces between learners and AI systems during programming tasks, leaving the micro-level dynamics of how learners integrate AI assistance into their problem-solving processes insufficiently understood.

A growing body of EDM research has begun addressing learning in AI-mediated environments more broadly, emphasizing challenges in representing complex interaction sequences, distinguishing productive from unproductive AI usage, and operationalizing constructs such as engagement and self-regulation from behavioral data [3, 15]. However, these efforts have not specifically examined learner-AI interaction in time-constrained programming contexts, where the interplay between time pressure, AI availability, and learner strategies may fundamentally alter regulatory behavior.

2.3 Research Gap

Prior work has (i) established log-based behavioral analysis as a productive approach to studying programming learning, and (ii) identified the growing influence of AI assistants on learner behavior. However, a critical gap remains in the data-driven modeling of learner-AI interaction sequences, particularly in contexts where time constraints shape decision-making and strategy use. Existing studies do not provide representations that integrate Integrated Development Environment (IDE) actions and AI interactions into unified behavioral traces, nor do they offer validated operationalizations of learning constructs such as cognitive engagement and help-seeking in AI-mediated environments. This research addresses this gap by applying sequential and process mining methods to model learner-AI interaction under both timed and untimed conditions, identifying distinct learner profiles, and developing behavioral proxies grounded in the SRL framework described in Section 4.

3. RESEARCH OBJECTIVES

The research is guided by four objectives, organized progressively from behavioral characterization to construct operationalization.

Objective 1: Characterize learner-AI interaction behaviors using log data — This objective focuses on identifying how learners integrate AI assistance into their programming workflows. By analyzing interaction logs, the study aims to uncover patterns in AI usage, including frequency, timing, and the sequencing of actions such as code edits, compilation, and AI queries. This objective provides the descriptive foundation upon which subsequent analyses are built.

Objective 2: Model temporal and sequential patterns of AI-assisted problem solving — Building on the behavioral characterization from Objective 1, this objective applies process mining and sequential modeling techniques to understand how learners transition between activities such as coding, debugging, and seeking AI assistance, and how these transitions differ under timed versus untimed conditions. The goal is to identify recurring behavioral structures and to examine how time pressure reshapes the dynamics of AI-assisted problem solving.

Objective 3: Identify distinct learner profiles based on AI usage strategies — Learners may differ significantly in how they use AI assistance, ranging from minimal reliance to heavy dependence. This objective aims to group learners into meaningful profiles based on features derived from interaction sequences and aggregated metrics, enabling a comparative understanding of diverse approaches to AI-assisted problem solving and their associations with task performance.

Objective 4: Develop and validate data-driven proxies for learning-related constructs — Direct measurement of constructs such as cognitive engagement and help-seeking is challenging in programming contexts. This objective focuses on operationalizing such constructs using observable behavioral indicators derived from interaction data, and validating these proxies through convergent analysis with self-report measures and qualitative examination of selected interaction sessions. This objective represents the primary conceptual contribution of the research, bridging learning science constructs with computational analysis of learner-AI interaction.

Together, these objectives form a progressive arc from empirical groundwork (Objectives 1-2) through learner variation (Objective 3) to theoretically grounded construct validation (Objective 4).

4. THEORETICAL FRAMING

This research is situated within the theoretical framework of SRL, which describes how learners plan, monitor, and adapt their strategies during problem solving [16]. SRL provides an integrative lens that encompasses two constructs central to this work: cognitive engagement and help-seeking behavior. Rather than treating these as independent constructs, this research models them as observable manifestations of self-regulatory processes within AI-mediated programming environments. Cognitive engagement , the extent to which learners invest mental effort in understanding problems [5], is typically reflected in iterative refinement and systematic debugging in programming contexts. However, AI assistance may shift effort distribution toward faster but less effortful strategies. From an SRL perspective, engagement reflects the learner's monitoring and control processes: whether they

actively evaluate their understanding or passively accept AI-generated outputs. Help-seeking, a key SRL strategy [1], reflects when and how learners seek external support. Prior research distinguishes adaptive help-seeking (targeted and strategic) from maladaptive patterns such as over-reliance or avoidance [1]. In AI-mediated environments, AI queries represent a form of help-seeking whose cost and accessibility differ fundamentally from traditional sources such as instructors or peers, potentially altering regulatory decisions, particularly under time pressure, where learners must balance speed benefits of AI assistance against learning benefits of independent effort. Time pressure introduces an additional regulatory demand that interacts with both constructs. In time-constrained settings, learners must continuously allocate limited time across competing activities, e.g. coding, debugging, testing, and seeking AI assistance, making SRL processes particularly visible through behavioral traces. The temporal distribution of these activities, and the transitions between them, provide a window into how learners regulate their problem-solving process under constraint. This integrated framing directly informs the methodological approach adopted in this research. Specifically, SRL theory motivates the use of sequence analysis to capture regulatory transitions between activities (Objective 2), the operationalization of cognitive engagement and help-seeking through behavioral indicators rather than self-report alone (Objective 4), and the identification of learner profiles that reflect distinct regulatory strategies (Objective 3). By grounding computational representations in SRL theory, this work bridges learning science constructs with data-driven analysis, enabling interpretable modeling of learner-AI interaction in time-constrained programming tasks.

5. METHODOLOGY

This research adopts a data-driven, mixed-methods approach to model learner-AI interaction behaviors in programming environments. The methodology focuses on collecting fine-grained interaction data and applying analytical techniques to uncover behavioral patterns, temporal dynamics, and learning relevant indicators.

5.1 Data Sources and Instrumentation

Data will be collected from undergraduate and early-graduate computer science students engaged in timed programming tasks at an Indian technical university. Participants will be recruited from students with prior coursework in programming and familiarity with AI coding assistants, ensuring a baseline level of competence. A target sample of approximately 60-80 participants is planned across multiple data collection sessions to support meaningful clustering and sequence analyses. Participants will complete a set of algorithmic programming tasks in Python or C++ within fixed time limits (30-45 minutes per task), designed to reflect the structure and difficulty of competitive programming and timed assessment scenarios. Tasks will span multiple difficulty levels to elicit variation in problem-solving strategies and AI usage. To examine the role of time pressure, a subset of sessions will include a comparison condition where participants complete equivalent tasks without strict time constraints, enabling analysis of how time pressure shapes interaction behaviors. The programming environment will be instrumented using a combination of IDE-level logging and AI interaction capture. Participants will use a standard code editor (e.g., VS Code) integrated with an AI coding assistant (e.g., GitHub Copilot or a GPT-based interface), configured to log all interactions. The primary data sources include: (i) IDE Interaction Logs capturing code edits, compilation attempts, execution events, error occurrences, and associated timestamps; (ii) AI Interaction Logs recording prompts submitted by learners, AI-generated responses, acceptance or rejection of suggestions, and the timing of each exchange; (iii) Task Performance Data including task completion status, correctness of solutions, time-to-completion, and number of submission attempts; and (iv) Self-Report Measures administered before and after sessions, including survey instruments for perceived cognitive load, AI reliance, and self-regulated learning strategies, used both for triangulation and for validating behavioral proxies developed under Objective 4.

5.2 Data Representation

Learner interactions will be represented as time-stamped sequences of discrete events, integrating IDE actions and AI interactions into a unified trace. Each learner session will be modeled as an ordered sequence of state transitions (e.g., edit $\rightarrow$ compile $\rightarrow$ error $\rightarrow$ AI query $\rightarrow$ edit $\rightarrow$ submit), preserving both the order and timing of actions. These sequences serve as the primary input for the process mining and Markov modeling described in Stage 1 of the analysis. In addition, higher-level features will be extracted from these sequences to support the clustering and profiling in Stage 2, including transition entropy (capturing regularity of behavior), dominant n-grams (capturing recurring micro-strategies), temporal features (such as mean interaction intervals and time allocation across activity types), and AI dependency ratios (such as the proportion of debugging episodes that involve AI queries versus independent resolution).

5.3 Data Analysis

The analytical approach is organized around three complementary stages, each aligned with specific research objectives.

Stage 1: Sequence and Process Analysis (Objectives 1 and 2) — Learner interaction sequences will be analyzed using process mining techniques, specifically Directly-Follows Graphs and fuzzy mining algorithms implemented in tools such as ProM or PM4Py, to visualize and compare dominant behavioral pathways across learners [3]. Additionally, first-order Markov models will be fitted to learner action sequences to estimate transition probabilities between states such as coding, compiling, debugging, and AI querying. These transition matrices will serve as compact representations of individual problem-solving processes and will be compared across timed and untimed conditions to examine how time pressure alters behavioral transitions.

Stage 2: Clustering and Learner Profiling (Objective 3) — Feature vectors will be constructed for each learner session, combining sequence-derived features (e.g., transition entropy, most frequent n-grams, proportion of time in each activity state) with aggregated interaction metrics (e.g., total AI queries, mean time between AI requests, ratio of AI-assisted to independent code edits). These feature vectors will be used as input to clustering algorithms. Agglomerative hierarchical clustering will be applied initially to ex-

plore the natural grouping structure, followed by k-medoids clustering for robustness to outliers. Cluster validity will be assessed using silhouette scores and domain-informed interpretation. The resulting profiles will be characterized in terms of their AI reliance patterns, temporal strategies, and association with task performance outcomes.

Stage 3: Construct Operationalization and Validation (Objective 4) — Behavioral proxies for cognitive engagement and help-seeking will be derived from interaction data. Cognitive engagement will be operationalized through indicators such as the duration of sustained independent coding episodes, the ratio of self-initiated to AI-delegated debugging, and the presence of iterative refinement cycles. Help-seeking will be operationalized through the timing of AI queries relative to task progression, the specificity of prompts (classified through manual coding of a subsample), and the dependency of subsequent actions on AI outputs (e.g., accepted verbatim, modified, or discarded). These proxies will be validated through convergent analysis with self-report measures and stimulated recall interviews with a subset of participants.

6. WORK DONE SO FAR

The proposed research builds on three preliminary studies. First, a survey-based study of learner perceptions [7] found that learners view AI tools as performance enhancers while expressing concerns about over-reliance, and that they differentiate between practice and evaluative contexts, directly motivating the inclusion of varied task conditions in the proposed design (Section 5.1). Second, an empirical study of AI interaction behaviors during programming tasks [9] revealed outcome-oriented strategies and iterative AI-dependent debugging under time pressure, providing the behavioral evidence that motivates the sequence analysis under Objectives 1 and 2, and generating preliminary interaction traces that will serve as pilot data for the analytical pipeline. Third, an analysis of competitive programming workflows [8] identified fragmented toolchains as a barrier to capturing coherent behavioral data, directly informing the decision to use a controlled, integrated programming environment rather than observing naturalistic but fragmented settings (Section 5.1). Together, these studies provide the empirical foundation , from perceptions through observed behaviors to environmental constraints, upon which the proposed modeling builds.

7. EXPECTED CONTRIBUTIONS TO EDM

This research aims to contribute to the EDM community across methodological, empirical, and conceptual dimensions.

1. **Unified representations of learner-AI interaction** — Existing learning analytics approaches in programming typically model learner interactions with compilers and editors, treating the programming environment as a closed system. This work contributes new representational approaches that integrate AI interactions, prompts, responses, acceptance and rejection of suggestions into unified behavioral traces alongside IDE actions. These representations extend the analytical vocabulary available to EDM researchers studying any environment where learners interact with intelligent systems, not only programming.

2. **Validated behavioral proxies for learning constructs in AI-mediated contexts** — A persistent challenge in EDM is operationalizing theoretical constructs through behavioral data. This research contributes proxies for cognitive engagement and help-seeking that are specifically designed for AI-mediated environments and validated through convergent analysis with self-report instruments and qualitative examination via stimulated recall. By grounding these proxies in the SRL framework and demonstrating their validity, the work provides a reusable methodological template for researchers seeking to measure learning-relevant constructs in settings where AI tools are embedded in the learning process.

3. **Empirical evidence on the role of time pressure in learner-AI interaction** — By comparing interaction behaviors under timed and untimed conditions, this research provides the first controlled empirical evidence of how time constraints shape AI usage patterns, behavioral transitions, and learner regulatory strategies in programming tasks. These findings contribute to a broader understanding of learning under performance pressure and have direct implications for the design of timed assessments in AI-enabled environments, where the interaction between time constraints and AI availability raises unresolved questions about what such assessments actually measure.

4. **Learner profiles for AI-assisted programming** — The identification of distinct profiles based on interaction patterns and AI usage strategies provides a basis for differentiated instructional approaches, enabling researchers and system designers to respond to meaningful variation in how learners integrate AI assistance.

8. FEEDBACK SOUGHT

This doctoral research seeks feedback from the EDM community on four aspects of the proposed work. (1) Sequential modeling choices: whether first-order Markov models and process mining are sufficient to capture learner-AI interaction complexity, or whether higher-order models (e.g., hidden Markov models) or alternative representations (e.g., epistemic network analysis) would be more appropriate, and how to balance model complexity against interpretability. (2) Construct validation: the adequacy of the proposed convergent validation design combining self-report measures and stimulated recall interviews, including guidance on sample size for stimulated recall and whether additional validation sources (e.g., think-aloud protocols, expert coding of traces) should be considered. (3) Productive versus unproductive AI usage: how to distinguish strategic AI use from dependency under time pressure using behavioral features beyond frequency alone, such as prompt specificity, degree of modification of AI output, and timing relative to independent attempts. (4) Generalizability: strategies for assessing whether findings from a controlled programming environment transfer to naturalistic contexts such as coursework and competitive programming.

9. REFERENCES

[1] V. Aleven, E. Stahl, S. Schworm, F. Fischer, and R. Wallace. Help seeking and help design in interactive learning environments. *Review of educational research*, 73(3):277–320, 2003.

[2] R. S. Baker, K. Yacef, et al. The state of educational data mining in 2009: A review and future visions. *Journal of educational data mining*, 1(1):3–17, 2009.

[3] M. Bannert, P. Reimann, and C. Sonnenberg. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and learning*, 9(2):161–185, 2014.

[4] S. Barke, M. B. James, and N. Polikarpova. Grounded copilot: How programmers interact with code-generating models. *Proceedings of the ACM on Programming Languages*, 7(OOPSLA1):85–111, 2023.

[5] J. A. Fredricks, P. C. Blumenfeld, and A. H. Paris. School engagement: Potential of the concept, state of the evidence. *Review of educational research*, 74(1):59–109, 2004.

[6] M. C. Jadud. Methods and tools for exploring novice compilation behaviour. In *Proceedings of the second international workshop on Computing education research*, pages 73–84, 2006.

[7] O. Joshi and S. Das. Generative ai in coding: Elevating efficiency or eroding skills? insights from an indian user study. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference Companion (LAK '25 Companion)*, pages 200–202. Society for Learning Analytics Research (SoLAR), Mar. 2025. Companion Proceedings.

[8] O. Joshi, K. Mehta, S. Kashyap, A. Purohit, and S. Das. Compete and learn? understanding competitive programming ide workflows and designing context-aware ai assistance to support learning and assessment. In *Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, page 5, Barcelona, Spain, Apr. 2026. ACM. Extended Abstracts, April 13–17, 2026, in press.

[9] O. Joshi, K. Mundra, D. Yadav, and S. Das. Ai coding assistants in competitive programming: Empirical studies from india on human-ai interactions in learning, problem solving, and curiosity development perspectives. In *International Conference on Artificial Intelligence in Education*, pages 181–194. Springer, 2025.

[10] W. Matcha, D. Gašević, N. A. Uzir, J. Jovanović, and A. Pardo. Analytics of learning strategies: Associations with academic performance and feedback. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 461–470, 2019.

[11] S. Peng, E. Kalliamvakou, P. Cihon, and M. Demirer. The impact of ai on developer productivity: Evidence from github copilot. *arXiv preprint arXiv:2302.06590*, 2023.

[12] C. Piech, J. Huang, A. Nguyen, M. Phulsuksombati, M. Sahami, and L. Guibas. Learning program embeddings to propagate feedback on student code. In *International conference on machine Learning*, pages 1093–1102. PMLR, 2015.

[13] T. Sinha, N. Li, P. Jermann, and P. Dillenbourg. Capturing "attrition intensifying" structural traits from didactic interaction sequences of mooc learners. In *Proceedings of the EMNLP 2014 workshop on analysis of large scale social interaction in MOOCs*, pages 42–49, 2014.

[14] P. Vaithilingam, T. Zhang, and E. L. Glassman. Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models. In *Chi conference on human factors in computing systems extended abstracts*, pages 1–7, 2022.

[15] P. H. Winne and R. S. Baker. The potentials of educational data mining for researching metacognition, motivation and self-regulated learning. *Journal of Educational Data Mining*, 5(1):1–8, 2013.

[16] B. J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory into practice*, 41(2):64–70, 2002.

Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory*

Amit Paikrao
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India
amitpaikrao@iitb.ac.in

Ramkumar Rajendran
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India

ABSTRACT

Modern electrical grid systems demand engineers capable of adaptive reasoning, systematic justification, and principled decision-making under uncertainty. However, while many graduates excel technically, they often struggle with these expert-level competencies, revealing gaps in our understanding of expertise development. To address this, we need to identify the granular cognitive and epistemic shifts that occur along the expertise continuum. Drawing on Epistemic Frame Theory, this study examines how epistemic frames shape the developmental trajectory of expertise across second-year, third-year, and final-year undergraduates, compared against field professionals (experts). In the study, participants completed an open-ended electrical grid design task in a computer-based environment. We applied Min-Max normalization to verbal protocols to neutralize verbosity bias and reveal latent structural shifts. Our findings highlight a two fundamental developmental trajectories. An ascending trajectory captures the adoption of a professional lens, wherein students transition from academic idealism to professional pragmatism and from reactive debugging to proactive modeling. Conversely, a descending trajectory reflects the shedding of novice habits, indicating a gradual disengagement from surface-level and trial-and-error strategies. Notably, experts' strategic silence emerged as a marker of proficiency. We observed that experts' silence is a signal of proficiency. The high-frequency behaviors in novices, such as visual monitoring and explicit naming of components, fade away as knowledge becomes tacit. These findings suggest that expertise is defined as much by what is shed (novice habits) as what is acquired (professional lens). These insights suggest that engineering curricula should incorporate earlier authentic practice to bridge the theoretical trap fostering the transition from explicit theoretical rules to implicit professional norms.

Keywords

Epistemic Frame Theory, Trajectory of Expertise, Continuum of Expertise, Min-Max Normalization, Engineering Education

*(Does NOT produce the permission block, copyright information nor page numbering). For use with edm_article.cls

Amit Paikrao, and Ramkumar Rajendran. Multi-dimensional Trajectories of Expertise in Engineering: Perspective from Epistemic Frame Theory. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 875–875. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039972

Quantifying Effects of New Advanced Placement STEM Courses through LLM-Assisted Dataset Linkage at Scale

Daniela Ganelin
Stanford University
dganelin@stanford.edu

ABSTRACT

In this paper, I present a validated crosswalk of large administrative datasets, created using LLM co-design, which enables cross-dataset institutional analytics and causal inference. Specifically, I document the creation of a replicable crosswalk pipeline to link national public datasets on school demographics and Advanced Placement (AP) offerings. Using the resulting detailed panel on AP access at U.S. schools, I quantify changes in AP access over time by student demographics. I then use quasi-experimental methods to investigate the effects of two large-scale curricular interventions of the last decade: new AP courses in computer science and precalculus. I find that both courses have had large effects in increasing access and participation in AP math and computer science, with largest effects for underrepresented groups including Black and low-income students and the schools that serve them. In line with open science principles, I plan to make the crosswalk, construction code, and detailed national dataset public to enable future research.

Keywords

Institutional analytics, causal inference, CS education, human-AI collaboration, educational equity

1. INTRODUCTION

Rigorous secondary math and computer science education play a key role in supporting both the global economy and individuals' success [14, 16]. However, in the U.S. and other countries, challenges persist in ensuring broad STEM participation and success, including disparities in course access and participation by race, socioeconomic status, and gender [23].

A key provider of advanced secondary STEM learning is the nationwide Advanced Placement (AP) program, which serves three million students annually. Students take courses at their high schools and then a standardized end-of-year exam administered by the College Board, with qualifying scores earning students advanced standing or credit at many colleges. Historically, AP calculus and CS participation have mirrored broader disparities. In response to these challenges, the College Board has introduced two new, introductory-level AP courses in recent years: CS Principles in 2016-17 and Precalculus in 2023-24 [11, 25]. Each was designed to attract a broader group of students and has launched at scale, with about 1.5 million combined exams to date.

Daniela Ganelin. Quantifying Effects of New Advanced Placement STEM Courses through LLM-Assisted Dataset Linkage at Scale. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 876–880. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039845

A difficulty in evaluating the effects of these courses and other interventions to increase STEM participation lies in limited national data availability [6]. The College Board has stopped releasing statistics on AP participation, access, and success disaggregated by race and subject [18]. Meanwhile, the last federal data release disaggregating access and participation in AP, calculus, and other advanced courses by subgroup predates the launch of AP Precalculus [23].

In this project, I present a novel LLM-assisted pipeline for joining College Board and federal data from public sources to present a comprehensive look at AP availability in the U.S. by school and subgroup. Validating against administrative data from two states, I show that my crosswalk is more comprehensive, current, and reproducible than the leading existing one. I also join in school-by-subgroup outcome data from Massachusetts, which provides rich publicly available data. In line with open science practices [2], I plan to publicly release the combined data, crosswalk, and generating code, enabling other researchers to study questions related to advanced course access.

Using quasi-experimental causal methods, an emerging area of interest in EDM [3], I evaluate the effects of the two national-level AP curriculum interventions. Nationally, at the institutional level, I find that AP CS Principles and Precalculus had substantial effects on increasing AP math and CS availability, especially for schools serving more Black or low-income students. Additionally, offering CS Principles built a pathway to later offering the more advanced AP CS course. In Massachusetts, I find that both courses substantially increased AP STEM participation, exam passing, and college-preparatory curriculum completion, especially among Black, Hispanic, and socioeconomically disadvantaged students. These results replicate my previous CS Principles results and extend them to the national level, richer outcomes, and a new subject area [13].

I anticipate this project contributing to the EDM community by providing a rich source of open data, a model for AI-assisted dataset construction that enables at-scale institutional analytics using public administrative data, and causal evidence on a national curriculum intervention that is currently improving CS learning outcomes at scale.

2. DATA PIPELINE CONSTRUCTION WITH AI ASSISTANCE

In this section, I describe the process of using LLM assistance to link large-scale administrative datasets, specifically the Common Core of Data and the College Board AP Ledger.

I begin with annual school demographics and directory data for all public schools that ever enroll Grade 9-12 students between 2007-08 and 2024-25 from the federal Common Core of Data (CCD) [22]. The dataset includes 39,609 schools that enroll a total of 270 million Grade 9-12 students.

From the College Board (CB) AP Course Ledger [7], I obtain comprehensive yearly records of audited AP course offerings at 23,435 U.S. schools, including private schools, between 2007-08 and 2025-26. I also obtain CB codes for an additional 26,229 institutions, which I code as never having AP.

Next, using LLM assistance, I construct a CCD-to-CB crosswalk to match schools across the two datasets using school names, addresses, and GPS coordinates, which are included in each data source. For efficiency and reproducibility, the LLM is not used directly to assess each match's quality, as in previous work [19]. Rather, Claude Code (Opus 4.6) co-designs, documents, and implements a fuzzy matching logic algorithm. The approach is designed to capture schools that change names, move addresses, or change CCD codes over time, which can cause one dataset's records to differ from the others.

First, the matching algorithm normalizes school addresses and names (for example, resolving "Middle" vs. "Junior High"), including historical information. Next, it constructs an address index and assesses candidate CCD-to-CB matches using two primary strategies. One strategy considers a CCD and CB school pair to match if they share a normalized address and name similarity, defined by Jaccard and SequenceMatcher, exceeds a threshold. The second strategy defines a match of a pair within a shared geographic block if the combined similarity of name and address, considering both string similarity and geographic distance, exceeds a threshold [21]. Finally, it resolves conflicts where multiple CCD schools share a single CB code by classifying them as "false positives", in which case a worse-matching school loses its assignment; "horizontal composites", involving a sub-school at the same address; and "vertical composites", where a new CCD school takes over enrollment from a previous one for subsequent years.

Matching approaches, thresholds, and implementation details are adjusted through iterative discussion of the 494 schools in Massachusetts until all disagreements with manually verified matchings are resolved. For example, validating against external sources, I identify both correct and incorrect matches among school pairs that share an address but not a name across datasets. To distinguish these cases, Claude designs and constructs a "recent rename bridge" that checks a historical list of names and CB codes to resolve the discrepancy [4]. The AI-assisted process enables rapid construction of a nontrivial matching algorithm that handles a broad variety of special cases and works at national scale, while requiring researcher inspection of only a small sample of each case type. The resulting crosswalk code runs on a MacBook Pro in about two hours. By contrast, the best previous publicly available crosswalk relies in part on a less precise name and ZIP-based matching, and in part on manual searches by Mechanical Turk workers [10, 20]. My crosswalk code is also fully replicable: future researchers can easily recreate all matches and extend to additional data, such as future years, without needing crowdsourced labor, model inference, or state-specific datasets.

My national crosswalk matches 27,866 CCD codes to CB codes, encompassing 97% of total Grade 9-12 enrollment. Comparing my matches to the pre-existing Davenport crosswalk, which provides CB codes for 20,625 of the CCD schools, I find 18,930 cases in which my crosswalk agrees, 914 cases in which Davenport codes do not appear in College Board data (i.e., defunct or mistaken codes), and 781 other cases where my crosswalk disagrees with or excludes a Davenport match. Importantly, my crosswalk introduces matches for 8,817 previously unmatched schools.

I validate the crosswalk by comparison to administrative data available from two states. In Colorado, among 512 schools that are in the CCD universe and have a non-defunct CB code in administrative data, my crosswalk has 481 correct and 6 incorrect matches, while Davenport has 431 correct and 5 incorrect [8, 15]. In Illinois, among 848 schools in the CCD universe with an administrative CEEB code, I have 793 correct and 7 incorrect matches, while Davenport has 697 correct and 11 incorrect. Overall, my crosswalk matches 94% of schools in these two states, with 99% of matches correct. The results sat an LLM-assisted crosswalk can enable dataset linkage with higher efficiency, accuracy, and coverage than human-implemented approaches.

3. PATTERNS OF AP AVAILABILITY

The crosswalk enables construction of a detailed panel dataset of AP availability at U.S. schools by year and subject joined with demographic information, which I use to characterize patterns of AP availability over time. While I focus on in-person math and computer science courses, other researchers could use the same dataset to investigate other subjects or virtual course-taking.

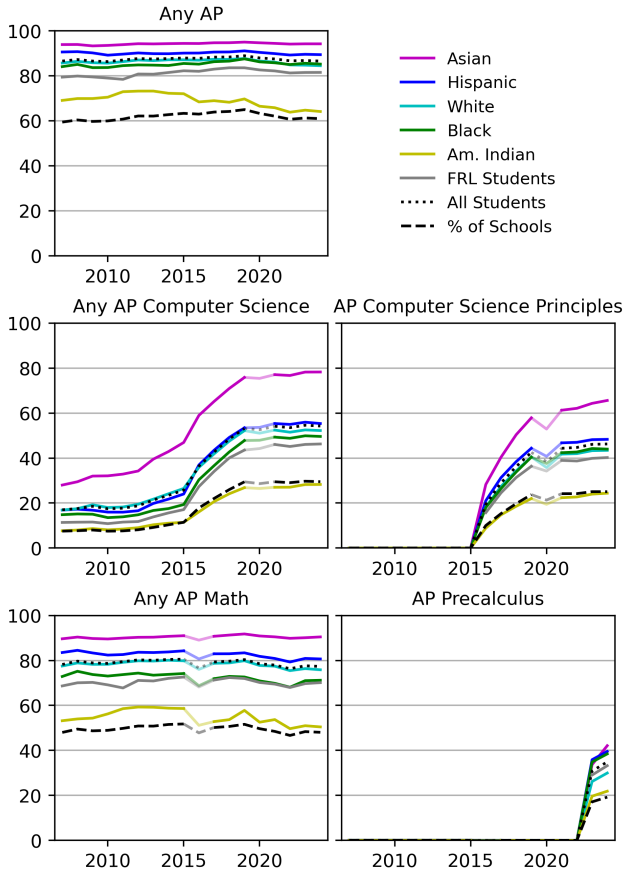


Figure 1. Percent of students nationally who have access to AP, by year (fall-indexed) and subgroup. Colored lines show the portion who attend a school offering each subject, among Grade 9-12 students who attend a regular high school. The gray line shows access among any-grade students who attend a regular high school and qualify for free or reduced-priced lunch. The dashed black line shows the percent of regular high schools that offer AP. Transparent segments mark years in which certain subjects' CB ledger data are incomplete.

I filter the CCD data to 23,487 regular high schools that serve Grades 10-12, merging vertical composites and excluding horizontal sub-schools as well as alternative, special-education, vocational, and virtual schools. These schools enroll a total of 252 million Grade 9-12 students between 2007-08 and 2024-25. Using the crosswalk, which assigns a CB code to 86% of these schools covering 98% of the students, I join school demographic information from CCD with AP records. This allows me to calculate the portion of students nationally who have access to AP by subgroup or school characteristic, subject, and year.

Figure 1 shows patterns of access to CS and math AP over time. While overall AP and AP math access have remained flat over time, access to Precalculus, CS Principles, and CS in general have all grown dramatically. Across all subjects, Asian students have much higher AP access than average and Hispanic students somewhat higher. Meanwhile, poor, Black, and especially American Indian students are less likely to have AP access – except in Precalculus, which disproportionately reaches Black students. Additional results show that AP offerings are higher at larger, urban, and suburban schools. I also find that most of the schools that adopted AP math or CS in recent years used one of the two new courses as their "gateway" course.

4. EFFECTS OF NEW COURSES ON AP AVAILABILITY

4.1 Methods

I demonstrate the applicability of the joined national dataset to policy analysis by examining the effects of schools beginning to offer AP CS Principles or Precalculus. At the national level, I estimate the effects of the new courses on AP offerings across the curriculum. I use a quasi-experimental design that compares changes at treated schools (i.e., schools that take up one of the two new courses) to changes at similar untreated control schools. My primary method is synthetic difference-in-differences (SDID), which weights units and pre-treatment time periods to eliminate bias due to unobservable latent factors [1]. Because SDID requires a balanced panel, I narrow the sample to the 15,799 schools that report Grade 9-12 enrollment in each year between 2007-08 and 2024-25, for a total of 231 million. I verify model fit using event studies and find similar results when varying specifications, e.g. including covariates, using a shorter panel, or using CSDID estimators on the unbalanced panel [5].

4.2 Results

Table 1 shows that both new courses have a substantial effect in opening the door to AP STEM access. Effects are larger for higher-poverty, smaller, and rural schools. Effects are particularly large for CS Principles, which triples the portion of schools offering any AP CS relative to later-treated schools' baseline. Additionally, offering CS Principles increases the chance of a school offering AP CS A, the pre-existing and more advanced, Java-focused AP course. Event study evidence shows that this growth comes primarily in the year *after* CS Principles is first offered, suggesting that CS Principles prepares schools to later build an extended CS pathway. By contrast, Precalculus has minimal impact on Calculus, neither substituting for nor expanding its offerings. However, it has greater effects on expanding AP math access at schools with more Black students.

Table 1: SDID Estimates of Effects on AP Course Offerings

	AP CS Principles		AP Precalculus	
Outcome	**Effect**	**Baseline**	**Effect**	**Baseline**
Any AP	0.06***	0.93	0.05***	0.97
	(0.01)		(0.01)	
Any AP Math or CS	0.10***	0.84	0.10***	0.90
	(0.01)		(0.01)	
Any AP CS	0.50***	0.26	0.05***	0.59
	(0.02)		(0.01)	
AP CS A	0.13***	0.26	0.02***	0.39
	(0.00)		(0.00)	
Any AP Math	0.03***	0.84	0.12***	0.88
	(0.01)		(0.01)	
Any AP Calculus	0.03***	0.82	0.02***	0.84
	(0.00)		(0.01)	
Schools	*15,799*		*15,799*	
Treated	*6,374*		*4,254*	

Notes: Estimated effects of offering new AP STEM courses on AP availability using a national balanced panel, 2007-08 to 2025-26. Baseline columns show the mean value among later-treated schools in the year before first treatment: 2015–16 for CS Principles, 2022–23 for Precalculus. Clustered bootstrap SEs in parentheses. *** $p<0.01$

5. EFFECTS ON STUDENT OUTCOMES

5.1 Methods and Data

Next, I estimate the effect of a school offering the new courses on student outcomes. I extend my data linkage pipeline by further incorporating rich school-level outcome data made publicly available by the Massachusetts Department of Elementary and Secondary Education (DESE) for 2007-08 to 2024-25 [17]. Matching is enabled by state-level school identifiers provided in CCD. Outcomes include by-subject AP exam participation and (partially suppressed) performance records, as well as completion rates of MassCore, a set of college-ready course requirements used for University of Massachusetts admission [24]. All outcomes are disaggregated by school, year, subject, and student subgroup. Critically, these outcomes are college-proximate: passing AP scores and MassCore completion are direct links to college admissions and credit earning.

I use a similar methodology to the national pipeline, using a balanced sample of 283 schools that appear in DESE data and have Grade 9-12 enrollment in each year.

5.2 Results

Preliminary results (Table 2) show the new courses substantially increase AP participation, each causing an annual increase of over 20 exams in the relevant subject area. Subgroup-specific results show that effects are particularly large relative to baseline for Black, Hispanic, and High Needs (e.g., low-income or English Learner) students, as well as female students for CS Principles. For these subgroups, CS Principles more than quintuples CS participation over time relative to baseline, and Precalculus expands math participation by over 70%.

Neither course clearly expands or substitutes for participation in the corresponding more advanced course. However, both courses have some positive effects on exam passing: Precalculus increases AP Calculus passing, suggesting that the course increases students' mathematical preparation. Meanwhile, CS Principles triples a school's probability of having at least 10 passing scores on AP CS exams, relative to a baseline of 13%. Additionally, AP CS Principles has a large and increasing effect on growing AP participation across the curriculum, while Precalculus seems to instead partially

substitute for non-STEM AP courses. Both courses also seem to increase MassCore completion, although these results are not always robust against alternative specifications (e.g., shorter panels). I plan to investigate patterns of heterogeneity and alternative estimation methods, as well as techniques for handling suppressed data.

Table 2: SDID Estimates of Effects on Student Outcomes

	AP CS Principles			AP Precalculus		
Outcome	**Exam Count**	**Base**	**10+ Pass**	**Exam Count**	**Base**	**10+ Pass**
All AP	51.7***	365	-0.00	18.9*	390	-0.02
	(15.2)		(0.02)	(10.3)		(0.02)
All AP CS	21.4***	7	0.25***	-2.9	24	0.06
	(2.1)		(0.03)	(2.6)		(0.04)
AP CS A	0.1	7	0.04*	-0.1	7	0.01
	(0.8)		(0.02)	(0.7)		(0.03)
All AP Math	5.0*	64	0.02	26.8***	55	0.21***
	(2.9)		(0.03)	(3.6)		(0.06)
All AP Calculus	3.3	37	0.01	-2.5*	29	0.10**
	(2.2)		(0.03)	(1.4)		(0.04)
MassCore Finishers	16.2**	179		24.0***	187	
	(6.3)			(8.5)		
Schools	283			283		
Treated	184			88		

Notes: Estimated effects of offering new AP STEM courses on student outcomes using a Massachusetts balanced panel, 2007-08 to 2025-26. MassCore starts in 2008-09 and uses a smaller subsample of 266 schools which have reported values in all years. The measure of exam passing is binary - i.e., at least 10 students earned a 3, 4, or 5 – to avoid small-count data suppressions. Baseline columns show the mean value among later-treated schools in the year before first treatment: 2015–16 for CS Principles, 2022–23 for Precalculus. Clustered bootstrap SEs in parentheses. *** $p<0.01$, ** $p<0.05$, * $p<0.10$.

6. CONCLUSIONS

In this paper, I demonstrate a process for using LLMs to build replicable, validated linkage pipelines across large, public administrative educational datasets. A similar process would allow discovery of connections across additional education domains [12].

Using the pipeline, I construct a novel dataset covering AP offerings at nearly all U.S. regular public high schools from 2007-08 to the present. I use this dataset, as well as rich student outcome data from Massachusetts, to examine the effects of two new courses: AP CS Principles and AP Precalculus. Using causal methods, I show that both courses have had large effects in expanding and broadening AP math and CS access and participation, helping build a diverse STEM pipeline for the future.

The results also suggest the power of a new opportunity in shifting school culture towards advanced learning in CS and other subjects. CS Principles was explicitly designed to broaden participation in computing as a large NSF-funded initiative including teacher training, extensive partner-led curriculum development, and a creative project component in assessment [9]. My results suggest that when schools introduce it, they do more than register students for the course: they build a pipeline to offering more advanced CS learning opportunities in the future, bring in previously underrepresented groups, grow AP participation across the curriculum, shift to becoming schools with high AP CS pass counts, and encourage completion of a college-ready curriculum.

AP Precalculus may yet achieve similar effects. I hope that the crosswalk, AI-assisted methodology, and compiled contextualized, granular administrative data that I share through this project will help researchers in studying the effects of this policy and other large-scale educational interventions.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] Arkhangelsky, D., Athey, S., Hirshberg, D. A., Imbens, G. W., and Wager, S. 2021. Synthetic difference-in-differences. Amer. Econ. Rev. 111, 12, 4088–4118. DOI=https://doi.org/10.1257/aer.20190159.

[2] Baker, R. S.,

Hutt, S., Brooks, C. A., Srivastava, N., and Mills, C. 2024. Open science and educational data mining: Which practices matter most? In Proceedings of the 17th International Conference on Educational Data Mining. International Educational Data Mining Society.

[3] Botelho, A. F., Closser, A., Sales, A., Heffernan, N., and Vanacore, K. 2024. Causal inference in educational data mining. In Proceedings of the 17th International Conference on Educational Data Mining (EDM 2024 Tutorial).

[4] Brevard College. 2020. CEEB Lookup Master List. https://brevard.edu/wp-content/uploads/2020/08/ceeb-lookup-masterlist.pdf.

[5] Callaway, B. and Sant'Anna, P. H. C. 2021. Difference-in-differences with multiple time periods. J. Econometrics 225, 2, 200–230. DOI=https://doi.org/10.1016/j.jeconom.2020.12.001.

[6] Chatterji, R., Campbell, N., and Quirk, A. 2021. Closing Advanced Coursework Equity Gaps for All Students. Center for American Progress.

[7] College Board. 2025. AP Course Ledger. https://apcourseaudit.inflexion.org/ledger/ (accessed 2025).

[8] Colorado Department of Higher Education (CDHE). 2026. i3 Lookup Field Values — High School. https://highered.colorado.gov/Data/More/LookupTables.aspx?type=highschool (accessed 2026-03-04).

[9] Cuny, J. 2012. The CS10K project: Mobilizing the community to transform high school computing. ACM Inroads 3, 2, 32–36.

[10] Davenport, M. 2025. CEEB-NCES crosswalk. Presented at the 2025 NCAIR Conference. https://nc-air.org/2025-ncair-conference-presentations/.

[11] Ewing, M., Wyatt, J., Iarrapino, M., and Jacklin, A. 2025. AP Precalculus: Who Participated and What We Learned in Launch Year. College Board Research.

[12] Figlio, D., Karbownik, K., and Salvanes, K. 2017. The promise of administrative data in education research. Educ. Finance Policy 12, 2, 129–136. DOI=https://doi.org/10.1162/EDFP_a_00229.

[13] Ganelin, D. and Dee, T. S. 2025. New Advanced Placement course designed to broaden access promotes participation and demographic diversity in computer science education. Proc. Natl.

Acad. Sci. 122, 17, e2422298122. DOI=https://doi.org/10.1073/pnas.2422298122.

[14] Goodman, J. 2019. The labor of division: Returns to compulsory high school math coursework. J. Labor Econ. 37, 4, 1141–1182. DOI=https://doi.org/10.1086/703135.

[15] Illinois Board of Higher Education (IBHE). 2022. ACT Code to RCDTS Code Crossref. https://www.ibhe.org/iheis.html (accessed 2026-03-04).

[16] Liu, J., Conrad, C., and Blazar, D. 2024. Computer science for all? The impact of high school computer science courses on college majors and earnings. EdWorkingPaper No. 24-904. Annenberg Institute at Brown University. DOI=https://doi.org/10.26300/z517-fw07.

[17] Massachusetts Department of Elementary and Secondary Education (DESE). 2025. Statewide Reports. https://profiles.doe.mass.edu/statereport/ (accessed 2025).

[18] Najarro, I. and Pendharkar, E. 2022. The case of the missing data on AP students. Education Week (Jul. 29, 2022).

[19] Narayan, A., Chami, I., Orr, L., and Ré, C. 2022. Can foundation models wrangle your data? Proc. VLDB Endow. 16, 4, 738–746. DOI=https://doi.org/10.14778/3574245.3574258.

[20] Office of Data Analytics, University of Colorado Boulder. 2023. CEEB-NCES Crosswalk. GitHub. https://github.com/UC-Boulder/ceeb_nces_crosswalk.

[21] Shah, N. et al. 2025. Efficient record linkage in the age of large language models: The critical role of blocking. Algorithms 18, 11, 723. DOI=https://doi.org/10.3390/a18110723.

[22] U.S. Department of Education, National Center for Education Statistics. 2025. Common Core of Data (CCD), 2007–08 through 2024–25. https://nces.ed.gov/ccd/.

[23] U.S. Department of Education, Office for Civil Rights. 2024. 2021–22 Civil Rights Data Collection: A First Look.

[24] Voices for Academic Equity. 2024. All Over the Map: Massachusetts High School Graduation Requirements. MassCore Data Brief, Fall 2024.

[25] Wyatt, J., Feng, J., and Ewing, M. 2020. AP Computer Science Principles and the STEM and Computer Science Pipelines. College Board Research.

Real-Time Scaffolding of Critical Thinking: AI Agents for Cognitive Presence Detection in Discussions

Fengjiao Tu
University of North Texas
FengjiaoTu@my.unt.edu

Haihua Chen
University of North Texas
Haihua.Chen@unt.edu

Ji Hyun Yu
University of North Texas
Jihyun.Yu@unt.edu

ABSTRACT

This study designs, implements, and evaluates an AI-based discussion facilitation agent that promotes critical thinking in graduate courses by detecting cognitive presence (CP) in real time. Grounded in the Community of Inquiry (CoI) framework, the agent embeds a fine-tuned RoBERTa transformer model (trained on 17,627 human-coded MOOC sentences; macro-F1 ≈ 0.70) into live graduate-level online discussions. The system classifies each student post into one of four CP phases—Triggering, Exploration, Integration, or Resolution—and immediately generates theory-informed, formative prompts to scaffold learners toward deeper inquiry. A design-based research methodology is used to iteratively refine the model and evaluate its pedagogical impact in instructional design courses. Three research questions address model accuracy, student and instructor perceptions of utility, and technical and pedagogical feasibility. Expected contributions include an interpretable NLP approach to CP detection, a replicable blueprint for real-time AI scaffolding, and empirical evidence that automated nudges can elevate online discourse beyond the Exploration plateau commonly observed in CoI studies.

Keywords

Cognitive presence, Community of Inquiry, AI agent, natural language processing, real-time scaffolding, online discussion

1. INTRODUCTION

Cognitive presence (CP) is defined as the extent to which learners construct and confirm meaning through sustained reflection and discourse, and is a central component of the Community of Inquiry (CoI) framework [1]. CP develops through four phases of practical inquiry: Triggering Event, Exploration, Integration, and Resolution. Together, these phases describe how critical thinking can emerge in online and hybrid learning environments.

Across two decades of CoI research, a consistent pattern has been reported: most student posts remain at the Exploration phase, and few reach Integration or Resolution [3, 4]. The pedagogical implication is that online discussions often underdeliver on their critical-thinking promise. Prior work suggests, however, that this pattern is responsive to instructor facilitation. When instructors actively guide students to synthesize and resolve, learners do advance to higher CP phases [2]. The difficulty is that such facilitation is hard to sustain in courses with many discussions and many students. Instructors have limited time, and most posts do not receive immediate, individualized guidance in the moment of writing.

In parallel, recent advances in natural language processing (NLP) have made automated CP classification feasible. Transformer-based models can label CP phases with accuracy that approaches human coders [3, 5]. Our prior work fine-tuned RoBERTa on 17,627 human-coded MOOC sentences and reached macro-F1 ≈ 0.70. These classifiers, however, have been used mainly for retrospective analysis of completed courses rather than for supporting learners as they write. The pedagogical potential of CP detection—using it to actually influence discourse in the moment—remains largely untested.

This study addresses that gap. It embeds a validated CP classifier in a live AI discussion agent that monitors graduate-level discussion forums, classifies each new post in real time, and returns immediate, gentle, theory-grounded prompts intended to encourage deeper inquiry. The study examines whether such real-time scaffolding can move student posts toward Integration and Resolution, and under what technical and pedagogical conditions this is feasible.

This study is guided by three research questions:

- **RQ1 (Accuracy):** How reliably can the AI agent classify discussion posts into CP stages compared to human coders? We will benchmark the model against expert-coded data using F1 and Cohen's κ.
- **RQ2 (Utility):** How do students and instructors perceive the agent's real-time prompts? Do they find the feedback clear, timely, and helpful for deepening discussion? Surveys and interviews will gauge acceptability and usefulness.
- **RQ3 (Feasibility):** What technical and pedagogical challenges arise in deploying this real-time agent? We

Fengjiao Tu, Ji Hyun Yu, and Haihua Chen. Real-Time Scaffolding of Critical Thinking: AI Agents for Cognitive Presence Detection in Discussions. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 881–883. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21040044

will examine integration with learning platforms, system responsiveness, instructor workload, and any unintended effects on discourse dynamics.

This project makes three interrelated contributions to the EDM and AIED communities. Theoretically, it extends the CoI framework from a retrospective coding scheme into an automated real-time in- tervention, demonstrating how a well-established learning theory can be operationalized in an AI system. Methodologically, it demonstrates an explainable transformer NLP approach—aug- mented with SHAP analyses and CoI-theoretic features—that makes model decisions transparent and trustworthy for educational stakeholders. Instructionally, it offers a concrete blueprint for deploying real-time AI scaffolding in online and hybrid graduate courses, with documented lessons on integration, usability, and pedagogical impact. By anticipating that real-time CP nudges will increase the propor- tion of Integration/Resolution posts and that students will perceive the agent positively, this research also builds empirical evidence for the promise of AI-driven formative feedback in higher education. Findings will be disseminated to both technical (EDM/AIED) and educational (Learning Sciences, instructional design) communities, reinforcing the interdisciplinary spirit of the EDM conference

2. RELATED WORK

This study draws on two lines of prior research: empirical CoI work on cognitive presence in online discussions, and NLP work on automated CP detection. We review each below before identifying the gap that motivates this study.

2.1 The Exploration Plateau in CoI Research

The CoI framework conceptualizes meaningful online learning as the interaction of social, teaching, and cognitive presence [1]. Empirical CP research consistently reports that discussion contributions concentrate at the Exploration phase, with relatively few posts reaching Integration or Resolution [4]. Facilitation is identified as the primary driver of advancement: when instructors explicitly guide students to resolve problems, participants do progress to higher CP phases [2]. Two implications follow. First, the Exploration plateau is not absolute; it responds to deliberate scaffolding. Second, scaffolding of this kind has so far been documented mainly in small, instructor-intensive settings, where post-by-post facilitation is feasible.

2.2 Automated Detection of Cognitive Presence

A parallel line of work uses NLP to automate CP classification. Transformer architectures, such as BERT and RoBERTa, have achieved strong performance on educational text classification tasks. [3] introduced MOOC-BERT for automated phase labeling. [5] extended this line with annotation-guideline-based knowledge augmentation for educational text classification. [2] report that GPT-class large language models can approximate human coders on CP content analysis. Our prior work fine-tuned RoBERTa with embedded CoI features and reached macro-F1 ≈ 0.70; the same analysis also revealed a correlation between higher-order CP posts and downstream outcomes including grades and self-reported knowledge transfer.

2.3 The Gap This study Addresses

Although both lines of work are well developed, they have rarely been combined. The CoI facilitation literature shows that scaffolding can move learners beyond Exploration but does not scale to the post level. The NLP literature shows that machines can detect CP reliably but has used detection almost entirely as a post-hoc analytic tool. To our knowledge, no prior study has embedded a CP classifier in a live discussion environment to provide real-time, theory-grounded prompts to students as they write. This study is positioned at exactly this intersection: it asks whether automated, in-the-moment CP scaffolding can support the kind of advancement toward Integration and Resolution that human facilitation can produce but cannot sustain at scale.

3. METHODOLOGY

This project employs a design-based research (DBR) approach, iteratively developing the AI agent and testing it in authentic classroom settings, then refining the model and prompts based on real classroom feedback. The work comprises two interlocking components.

3.1 Model Refinement

We begin with an existing transformer classifier (RoBERTa fine-tuned on MOOC data with embedded CoI features). To adapt it to graduate seminar discussions, we apply transfer learning: initializing from MOOC-trained weights, then retraining on a smaller set of manually coded graduate-level posts. Approximately 20% of collected posts will be double-coded by experts to create a gold-standard test set. Model performance will be evaluated using precision, recall, and F1 for each CP category, along with inter-rater agreement (Cohen's κ) between model and human coders. Shapley Additive Explanations (SHAP) analyses [6] will be used to identify the linguistic features driving classification decisions, rendering the model interpretable to both technical and educational audiences.

3.2 AI Agent Development

The refined classifier is embedded into an interactive agent within the learning management system (LMS) or chat platform. During weekly forum activities, the agent listens for new posts. When a student submits a contribution, the agent immediately classifies it into a CP phase and generates a tailored, formative prompt. These prompts are co-designed with instructors and grounded in CoI theory. For example, if a post is rated as Exploration, the agent might respond: *"You have shared some great ideas. How might these connect to other concepts we have covered?"* If a post is already at the Integration level, it might offer: *"Nice synthesis of ideas. Can you take this further by applying it to a real-world problem?"* Crucially, prompts are phrased as open questions, not evaluations, and students are free to revise their post or ignore the suggestion with no grading penalty. Instructors have oversight through a dashboard that logs each classification and can disable or customize feedback as needed.

3.3 Evaluation Design

The AI agent will be deployed in several graduate instructional design courses over a semester (approximately six discussions per course, targeting roughly 300 posts in total).

Data collection from Fall 2025 (without the CP-supported agent) will serve as a baseline comparison. The agent-supported implementation is planned for Fall 2026. Both quantitative and qualitative data will be collected: system logs will capture processing time, uptime, and integration issues (RQ3); post-semester surveys will assess student perceptions of clarity and helpfulness (RQ2); and semi-structured interviews will probe student and instructor experiences in depth (RQ2–RQ3). Model accuracy metrics will be computed against expert-coded samples (RQ1).

4. REFERENCES

[1] D. Randy Garrison, Terry Anderson, and Walter Archer. 2001. Critical thinking, cognitive presence, and computer conferencing in distance education. *American Journal of Distance Education* 15, 1 (2001), 7–23.

[2] Daniela Castellanos-Reyes, Larisa Olesova, and Ayesha Sadaf. 2025. Transforming online learning research: Leveraging GPT large language models for automated content analysis of cognitive presence. *The Internet and Higher Education* 65 (2025), 101001.

[3] Zhi Liu, Xi Kong, Hao Chen, Sannyuya Liu, and Zongkai Yang. 2023. MOOC-BERT: Automatically identifying learner cognitive presence from MOOC discussion data. *IEEE Transactions on Learning Technologies* 16, 4 (2023), 528–542.

[4] Yuanyuan Hu, Claire Donald, and Nasser Giacaman. 2023. Can multi-label classifiers help identify subjectivity? A deep learning approach to classifying cognitive presence in MOOCs. *International Journal of Artificial Intelligence in Education* 33, 4 (2023), 781–816.

[5] Shiqi Liu, Sannyuya Liu, Lele Sha, Zijie Zeng, Dragan Gašević, and Zhi Liu. 2025. Annotation guideline-based knowledge augmentation: Towards enhancing large language models for educational text classification. *IEEE Transactions on Learning Technologies* (2025).

[6] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* 30 (2017), 4765–4774.

Scaling, Demonstrating and Personalizing: the Benefits of LLMs in Educational Contexts

Eamon Worden
Worcester Polytechnic Institute
100 Institute Road
Worcester, Massachusetts
elworden@wpi.edu

Neil Heffernan
Worcester Polytechnic Institute
100 Institute Road
Worcester, Massachusetts
nth@wpi.edu

ABSTRACT

The benefits of explanations and feedback on learning are well established. However, delivering effective feedback at scale remains a substantial challenge due to various logistical and cost-related constraints. This paper reports on my randomized controlled trials (RCTs) evaluating the effectiveness of AI-generated content for middle school mathematics. My first studies include RCTs comparing AI-generated explanations and feedback with business-as-usual (no support), as well as AI-generated content with teacher-written content. Both used a similar pipeline to generate feedback, going through an iterative prompt engineering process with experienced educators, followed by LLM-as-a-judge and a limited human review. One RCT found that students receiving AI-generated feedback were 16% more likely to correct their current answer and 7% more likely to succeed on the subsequent problem. I then moved to live content generation, with LLMs providing feedback to short-answer math problems. I used a cross-over design to find the impact of using GPT-4o to provide short, long, or affective feedback, and also analyzed a model fine-tuned on teacher-written feedback. I found affective feedback was the only feedback that was consistently helpful in both the short- and medium-term transfer learning for students. This informs my future research, aiming to generate and cache various styles of content using LLMs, then personalize it to students' learning characteristics. My future work builds on this by aiming to personalize the style, length, and tone of feedback to students at scale based on both student and misconceptions features at scale, which no prior work has managed to achieve.

Keywords

Large Language Models, AI-generated feedback, Randomized Controlled Trial, Mathematics Education, Learning at Scale, Automated Content Generation

1. INTRODUCTION

Eamon Worden, and Neil T. Heffernan. Scaling, Demonstrating and Personalizing: the Benefits of LLMs in Educational Contexts. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 884–888. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039954

Large Language Models (LLMs) are increasingly deployed in educational contexts for tasks such as chatbots [21], feedback generation [24], and question generation [13]. However, equally important to creating these systems is evaluating their effectiveness and fairness. LLM-generated content needs to be grounded in pedagogy and delivered effectively, otherwise it risks decreasing critical thinking by removing the opportunity for productive struggle [4]. However, if implemented effectively, it can personalize content such as on-demand support, feedback, and more.

Given the risks and potential of LLMs, my focus has been on using established pedagogical techniques and advanced prompt engineering to generate content at scale before empirically demonstrating the effectiveness. Working with the ASSISTments platform [8], I have conducted RCTs using AI-generated content with tens of thousands of students in grades 6-8, Illustrative Math. Using techniques such as RAG [12], few-shot prompt [2], LLM-as-a-judge [27], and chain-of-thought [22], I have generated on-demand supports to aid students, which had previously been an expensive and time-consuming effort requiring the recruitment of teachers, who themselves are limited in time. This has informed my research into how LLMs can be used to scale content in digital learning platforms (DLPs) at an affordable and unprecedented rate. I am now working to generate various styles of content, which, with the correct bandits algorithm, aims to provide personalized content on a new level.

My works so far has scaled on-demand explanations and feedback. Explanations are a form of instructional support that provides all the steps necessary to walk through a solution to a problem, often presented as step-by-step solutions. They are well-established as effective tools for student learning [20, 17]. Notably, explanations are more effective for students with less expertise than their higher-skilled peers as described by the "expertise reversal effect" [9]. One major theory for why explanations are helpful for all students, but particularly helpful for low-performing students, is cognitive load theory [18], which posits that explanations reduce the mental effort required to process information, thereby allowing students to focus their cognitive resources on acquiring and remembering mathematical knowledge.

The other content I've been working to generate, feedback, is among the most effective instructional interventions documented in the learning sciences [6, 7]. Meta-analyses consistently show that well-designed corrective feedback supports

error detection, improves problem-solving strategies, and promotes self-regulation, particularly for lower-achieving students [23, 10]. Despite this evidence, delivering personalized feedback at scale remains a persistent challenge in digital learning platforms (DLPs). Prior approaches using crowdsourcing have shown promise but suffer from quality variability [15, 5]. Recent advances in large language models (LLMs) present a new opportunity. LLMs can generate coherent, context-sensitive feedback at low marginal cost. However, whether AI-generated feedback improves learning outcomes over no feedback—and for which students—remains an open empirical question with limited large-scale evidence in authentic classroom settings.

This informs my three main research questions for my last studies. **Research Plan 1** investigates "Under what conditions does feedback style moderate learning outcomes in middle school mathematics?" This aims to analyze the impact of various feedback styles inspired by Hattie & Timperley [6] and which works best by looking at student characteristics and classifying wrong answers. With sufficient data, this will enable **Research Plan 2**, which asks "Can a bandit or similar algorithm effectively personalize feedback style for students in a live DLP, and what student features are necessary to do so?" This investigates personalizing at scale within ASSISTments, which requires understanding students and constantly measuring learners' characteristics and updating latent student traits [19]. Lastly, moving to a more dynamic setting, **Research Plan 3** asks "Does personalized LLM-generated feedback on open-ended responses improve learning outcomes compared to no feedback, and does it do so sustainably over a full academic year?" Open-ended problems are very important as they promote deeper and more critical thinking, but scaling personalized feedback for tens of thousands of students can be expensive.

2. PUBLISHED WORKS

During my PhD, I have published several works on my various studies, aiming to generate and evaluate the effectiveness of content. I have first-author submissions to both Learning at Scale [25] and AIED [24] and an under-review submission at Learning at Scale this year.

Studies 1 & 2: Scaling Static Explanations and Feedback [25, 26]. In these works, I have empirically evaluated the effectiveness of LLM-generated explanations and static feedback messages through RCTs in ASSISTments. For these RCTs, I collaborated with former teachers, curriculum designers at ASSISTments, as well as other PhD students within my lab, to engineer effective prompts for explanations and feedback, before caching them in the ASSISTments database and engineering the middleware needed to randomly select the LLM-generated content, which was displayed to students as shown in figures 1 and 2. These studies have been run *en vivo* in ASSISTments by randomly selecting to deliver nothing, LLM-generated content, or teacher-authored content, and comparing their effectiveness, with tens of thousands of students. To analyze our results, I used multiple mixed-effects linear and logistic regression models with various outcome metrics. Our results were consistent across the studies: LLM-generated content was better than nothing, and there was no reliable difference between LLM-generated content and teacher-authored content. Also consistent with the expertise reversal effect, I found these supports were more effective for lower-knowledge students. Given the incredibly cheap cost of scaling this content with LLMs (about 5 cents per content), I feel this confirms that LLMs can aid students by scaling content without harming student performance.

Study 3: Scaling Feedback Generation in a Live Setting [24]. While scaling static content is valuable as done in **studies 1 & 2**, providing dynamic feedback to short answers is equally valuable. For **study 3** I worked with ASSISTments to engineer a student interface that could deliver feedback when they submit short-answer responses to open-ended problems. Prior to our work, of the 50 million instances of students submitting a short-answer, non-computer-gradeable response in ASSISTments, only 1 million (2%) actually received feedback, and often 2-3 days later. As such, I am laying the groundwork for ASSISTments to provide short-answer feedback at scale.

In this study, along with collaborators at WPI and the ASSISTments foundation, I recruited 11 teachers with a total of 322 students, and employed a crossover design to evaluate the effectiveness of LLM-generated feedback of various styles, short feedback, long feedback, affective (self-level) feedback, or feedback from a model which was fine-tuned on teacher-written feedback in my prior work [1]. Our findings suggested that only affective feedback was useful to students in both the next-problem and medium transfer range, whereas other feedback styles had minimal help, or harmed student learning due to giving away excessive information and removing the opportunity for productive struggle. This was a surprising result, given that affective supports are generally thought to be ignored by students in DLPs. However, it also inspires my future work on providing personalized feedback to all students, so that students who benefit more from a specific feedback style receive that personalized feedback.

3. ONGOING WORK

I intend for this to be the main component of my dissertation. Having shown LLMs can generate effective content, I aim to generate personalized content at scale. This involves three studies, the first of which is an RCT aiming to explain when different styles of feedback are useful, and the second, which aims to personalize in a live setting using a bandit's algorithm [16], and the last, which repeats this for open-response questions.

Research Plan 1 & 2. This study, which has already started, aims to build upon my prior work generating wrong answer feedback for students with LLMs. I found LLM-generated feedback was effective; however, inspired by Hattie & Timperley's styles of feedback [6], I have generated 6 styles of feedback: Feedback; Feed-up; Feedback & Feed-up; Feedback & Self-regulation; Feedback, Feed-up & Self-regulation; and Self-level feedback. These feedback styles differ in the type of information provided, such as whether a small hint is given, whether self-regulatory behavior is encouraged, or whether the feedback is affective. The goal is to determine which type of feedback works best, and answer "when?" and "why?".

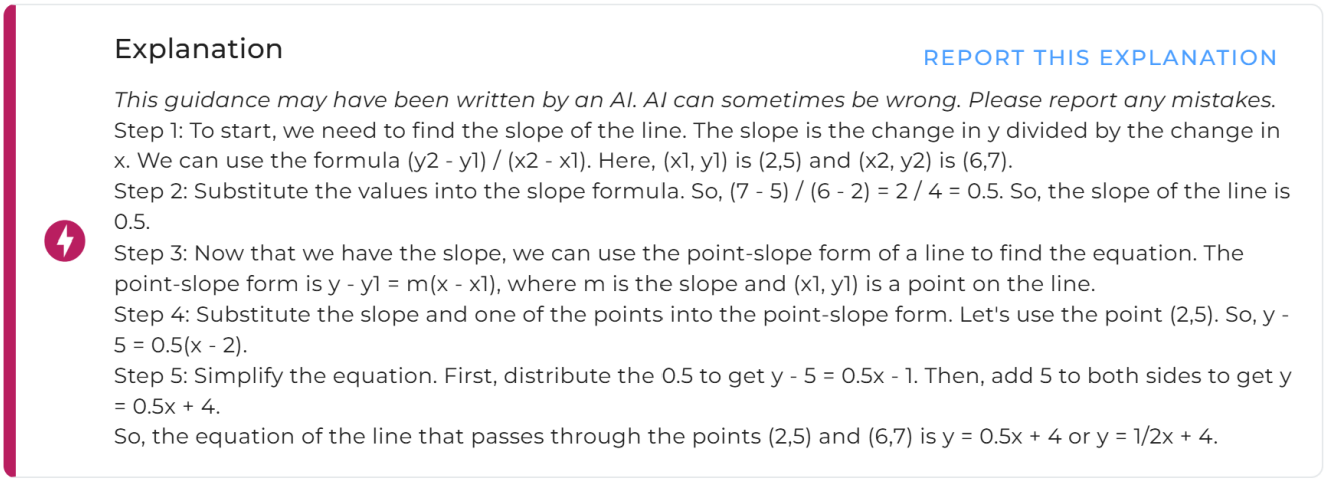


Figure 1: An example explanation with the report button and warning message.

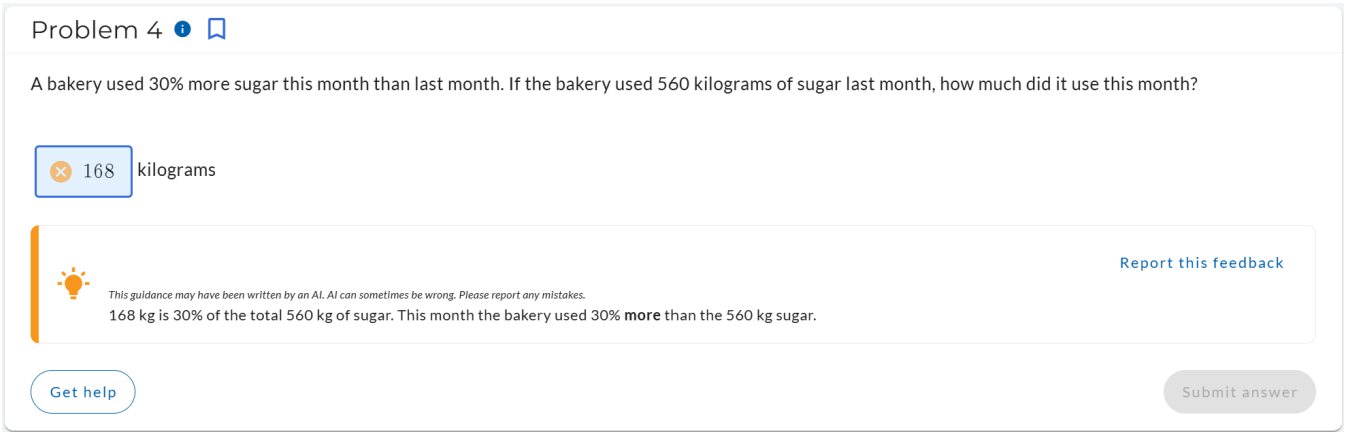


Figure 2: An example feedback message as it appears in ASSISTments.

The "when?" refers to different types of wrong answers and student characteristics. For instance, for the problem **"A bakery used 30% more sugar this month than last month. The bakery used 560kg of sugar last month. How much did it use this month?"**, I hypothesize that the best style of feedback for a student who said "168" (forgot to add it to the original amount) will be very different from the best type of feedback for a student who said "590" (does not understand percentages). For the student who answered "168", I hypothesize self-regulatory feedback, which emphasizes that reading the problem carefully will best help the student, whereas for the student who answered "590", a combination of feedback and feed-up will best guide them towards a better understanding of percentages. The same can be true for student characteristics. Reinforcement learning has shown that incorperating student features into a bandit model can improve learning outcomes, which I intend to do once I have collected sufficient data [11].

As mentioned, my first research project comparing the styles of feedback is currently running, however, I could use guidance on the analysis plan. I hypothesize that the key to understanding which styles of feedback are helpful, and when, is categorizing styles of misconceptions into different types. Categories such as "forgot last step", "conceptual misunderstanding", "procedural error" or more may reveal deeper insights into when each style of feedback may be more or less helpful. Determining which categories to include and how to classify them is a challenge I am facing. However, there are other approaches I have considered. Using Bayesian Knowledge Tracing or a similar algorithm to classify a student's knowledge of various topics, or even their individual slip rate, appears feasible based on prior research [14], which could inform both analysis and bandits models, which is my second project relating to this work. However, defining the best methodological approach and analysis plan has been a difficult task for me, and I would appreciate the insights of experienced researchers in the field to improve and solidify this methodology.

Research Plan 3. Building on my feedback research, my other work involves generating dynamic feedback using LLMs for open-response problems. These are problems that require students to type 1-3 sentences to answer a question,

as opposed to fill-in problems, which require a number or short expression. This makes caching feedback infeasible, and before LLMs, attempts to scale feedback proved expensive and difficult [3]. However, in a live, undefined setting, I have more options for how I might generate feedback. I may generate feedback, feed-up, self-regulation, and affective feedback as in **Study 1**. However, by being able to see, rather than just infer, a student's work, I can go beyond basic feedback. I may provide targeted interventions, such as providing sentence starters or clarifying specific portions of answers. I have worked with ASSISTments and we intend to enable this feature for 50% of teachers beginning in the fall of 2026 and lasting for the complete 2026-2027 school year. This will include around 500 teachers with tens of thousands of students in the study.

This work will include building on my prior work (**Study 3**)by opening new opportunities for longitudinal analysis. This will analyze the effect of repeated interactions over the course of the school year. I will analyze whether there are diminishing returns, increased disengagement, or increased unproductive behavior (such as gaming) as a result of adding LLM-feedback to ASSISTments. Alternatively, I will be able to investigate whether there is a cumulative advantage or other sustained learning improvements for students who receive feedback on their open-ended questions. Last, I hope to gather sufficient data to fine-tune a custom LLM to generate feedback for students in ASSISTments, as I have done in prior work [1].

4. CONCLUSION

Personalizing feedback for students at scale remains a consequential and open problem in educational data mining. While the learning sciences have long established that feedback is among the most effective instructional interventions, we have lacked both the tools to deliver it affordably at scale and the empirical understanding of when different feedback styles are most effective. My completed work addresses this using three large-scale RCTs across 250,000 observations and 50,000 students, which demonstrate that LLM-generated content is effective and cost-effective. My remaining work addresses a harder yet more impactful problem. Categorizing student misconceptions is a prerequisite for understanding the causal structure of feedback effectiveness. If we cannot distinguish a student who made a procedural slip from one with a conceptual misunderstanding, we cannot know which feedback style to deliver, nor why it worked when it did. This represents a contribution to the EDM and learning science community beyond ASSISTments: a framework for misconception-aware feedback personalization that can inform how any DLP reasons about learner state. Combined with a bandit-based personalization system and longitudinal data from tens of thousands of students, this work aims to produce a large-scale empirical study of personalized feedback in middle school mathematics.

5. REFERENCES

[1] S. Baral, E. Worden, W.-C. Lim, Z. Luo, C. Santorelli, A. Gurung, and N. Heffernan. Automated feedback in math education: A comparative analysis of llms for open-ended responses. *arXiv preprint arXiv:2411.08910*, 2024.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[3] J. A. Erickson, A. F. Botelho, S. McAteer, A. Varatharaj, and N. T. Heffernan. The automated grading of student open responses in mathematics. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 615–624, 2020.

[4] G. P. Georgiou. Chatgpt produces more" lazy" thinkers: Evidence of cognitive engagement decline. *arXiv preprint arXiv:2507.00181*, 2025.

[5] A. Gurung, S. Baral, M. P. Lee, A. C. Sales, A. Haim, K. P. Vanacore, A. A. McReynolds, H. Kreisberg, C. Heffernan, and N. T. Heffernan. How common are common wrong answers? crowdsourcing remediation at scale. In *Proceedings of the Tenth ACM Conference on Learning@ Scale*, pages 70–80, 2023.

[6] J. Hattie and H. Timperley. The power of feedback. *Review of educational research*, 77(1):81–112, 2007.

[7] K. Haughney, S. Wakeman, and L. Hart. Quality of feedback in higher education: A review of literature. *Education Sciences*, 10(3):60, 2020.

[8] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International journal of artificial intelligence in education*, 24(4):470–497, 2014.

[9] S. Kalyuga. The expertise reversal effect. In *Managing cognitive load in adaptive multimedia learning*, pages 58–80. IGI Global, 2009.

[10] A. N. Kluger and A. DeNisi. The effects of feedback interventions on performance: a historical review, a meta-analysis, and a preliminary feedback intervention theory. *Psychological bulletin*, 119(2):254, 1996.

[11] M. P. Lee and N. T. Heffernan. Improving student support personalization with historical data and theoretically informed feature choice. In *International Conference on Artificial Intelligence in Education*, pages 421–426. Springer, 2025.

[12] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[13] S. S. Mucciaccia, T. M. Paixão, F. W. Mutz, C. S. Badue, A. F. de Souza, and T. Oliveira-Santos. Automatic multiple-choice question generation and evaluation systems based on llm: A study case with university resolutions. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2246–2260, 2025.

[14] Z. A. Pardos and N. T. Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *International conference on user modeling, adaptation, and personalization*, pages 255–266. Springer, 2010.

[15] T. Patikorn and N. T. Heffernan. Effectiveness of crowd-sourcing on-demand assistance from teachers in

online learning platforms. In *Proceedings of the Seventh ACM Conference on Learning@ Scale*, pages 115–124, 2020.

[16] E. Prihar, A. Haim, A. Sales, and N. Heffernan. Automatic interpretable personalized learning. In *Proceedings of the Ninth ACM Conference on Learning@ Scale*, pages 1–11, 2022.

[17] A. Rourke and J. Sweller. The worked-example effect using ill-defined problems: Learning to recognise designers' styles. *Learning and Instruction*, 19(2):185–199, 2009.

[18] J. Sweller. Cognitive load theory, 2011.

[19] L. Tetzlaff, F. Schmiedek, and G. Brod. Developing personalized education: A dynamic framework. *Educational psychology review*, 33(3):863–882, 2021.

[20] P. W. Van Gerven, F. G. Paas, J. J. Van Merriënboer, and H. G. Schmidt. Cognitive load theory and aging: Effects of worked examples on training efficiency. *Learning and instruction*, 12(1):87–105, 2002.

[21] R. E. Wang, A. T. Ribeiro, C. D. Robinson, S. Loeb, and D. Demszky. Tutor copilot: A human-ai approach for scaling real-time expertise. *arXiv preprint arXiv:2410.03017*, 2024.

[22] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[23] B. Wisniewski, K. Zierer, and J. Hattie. The power of feedback revisited: A meta-analysis of educational feedback research. *Frontiers in psychology*, 10:487662, 2020.

[24] E. Worden, M. Lee, A. Siedahmed, A. Sales, J. Zhang, R. Shraga, and N. Heffernan. Short, long, or affective: Evaluating llm-generated feedback styles for student learning. In *International Conference on Artificial Intelligence in Education*, 2026.

[25] E. Worden, K. Vanacore, A. Haim, and N. Heffernan. Scaling effective ai-generated explanations for middle school mathematics in online learning platforms. In *Proceedings of the Twelfth ACM Conference on Learning@ Scale*, pages 40–49, 2025.

[26] E. e. a. Worden. Under review. In *Conference on Learning@Scale*, 2026.

[27] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Toward Vision-Language Models as AI Tutors in Physical STEM Manipulation

Changsoo Jung
Computer Science
Colorado State University
Changsoo.Jung@colostate.edu

Nathaniel Blanchard
Computer Science
Colorado State University
Nathaniel.Blanchard@colostate.edu

ABSTRACT

Collaborative physical manipulation tasks are widely used in STEM education to develop spatial reasoning and collaborative problem-solving. AI tutoring systems for these settings remain limited because they cannot reliably perceive student actions or evaluate in 3D whether those actions are spatially correct. This study investigates how Vision-Language Models (VLMs) can serve as perception and judgment modules in AI tutoring systems for physical block construction. Preliminary work has revealed a fundamental dissociation in current VLMs: they can identify what a student did but cannot evaluate whether it is spatially correct. Our proposed work extends this foundation toward a full AI tutoring pipeline that integrates multimodal signals and is validated through a learning outcomes study.

Keywords

vision-language models, AI tutoring, spatial reasoning, STEM education, collaborative learning

1. INTRODUCTION

This dissertation investigates the use of Vision-Language Models (VLMs) as perception and judgment modules in AI tutoring systems for collaborative physical STEM tasks. The central premise is that effective tutoring in such tasks requires automated systems that can both interpret a learner's actions on shared physical artifacts and evaluate whether those actions move the task toward its goal. Preliminary work indicates that current VLMs can identify what a student did but struggle to evaluate whether it is spatially correct, exposing a fundamental capability gap between perception and judgment. The remainder of this paper details the open problem and research challenges, the theoretical framing of the work, preliminary findings, the proposed research plan, and the expected contributions of the dissertation.

2. PROBLEM STATEMENT AND RESEARCH CHALLENGES

Changsoo Jung, and Nathaniel Blanchard. Toward Vision-Language Models as AI Tutors in Physical STEM Manipulation. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 889–891. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039960

Collaborative physical manipulation is a cornerstone of STEM education because it develops spatial reasoning and procedural problem-solving [4, 10]. When students build a structure together, a tutor must monitor what is being built, detect deviations from the goal, and deliver corrective feedback in real time. Existing intelligent tutoring systems have focused on language-based feedback [7, 9] and symbolic domain reasoning [11]. These approaches do not transfer to physical manipulation where collaborative group dynamics directly shape how students perform actions on shared artifacts. Recent VLMs are a natural candidate for this role given their ability to process images and language jointly, but several challenges remain before VLMs can be reliably applied as AI tutors in classroom environments.

Visual complexity. Real manipulation sessions involve hands occluding blocks, inconsistent lighting, and camera perspectives that compress depth. Unlike clean rendered scenes used in prior spatial reasoning benchmarks [5, 6], a VLM processing a real video frame must reason about a noisy and partially observable 3D scene.

Incremental state change. A construction task unfolds one block at a time over 20 to 40 minutes. An AI tutor must detect a single block addition, removal, or repositioning against a growing background structure. Detecting absence (a block that was removed) is especially difficult because it requires comparing two states at a location that now appears empty.

Spatial correctness judgment. Identifying what a student did is not the same as evaluating whether it is correct. A tutor must compare the student's result against a goal configuration and determine whether the spatial relationship between blocks matches the target. This requires a form of 3D reasoning that goes beyond object recognition.

Multimodal collaboration context. In multi-party manipulation sessions, directors give spoken instructions to guide the builder [12]. An AI tutor that can ground these instructions spatially could use them to better interpret ambiguous student actions. It is unclear how much this conversational context helps VLM judgment.

These challenges together define the open problem that our study addresses: *can VLMs be made reliable enough to serve as the perception and judgment backbone of an AI tutoring system in a real physical manipulation task, and if not, what is missing?*

3. THEORETICAL FRAMING

In this study, we explore two complementary frameworks. The *model-tracing* paradigm from intelligent tutoring systems [11] holds that an effective tutor must maintain a model of the student's current state and compare it against a target. In a physical manipulation task, this means tracking a 3D block configuration and detecting deviations from a goal structure at each step. We also explore on *grounded language understanding* [2] which requires that language and vision be jointly anchored in physical space. Together, these frameworks motivate our central research question: can VLMs acquire sufficient 3D spatial grounding to support accurate tutoring feedback in physical STEM lab experiments?

4. PRELIMINARY STUDY

As the first phase of the study, we designed and ran an experiment to evaluate whether current VLMs can judge student manipulations in a physical construction task. As illustrated in Figure 1, each case provides four images: a block reference sheet, an initial state, a builder result, and a goal state. Models must identify what manipulation occurred and judge whether the builder's result matches the goal. We evaluated four frontier VLMs on 421 cases drawn from 19 real sessions [12] spanning place, move, and delete actions.

The central finding is a fundamental dissociation. Open-weight 7-8B models correctly identify what manipulation a student performed in up to 93.8% of cases when given structured state contexts. However, their judgment of whether it achieves the spatial goal remains at chance even with the same context. Proprietary frontier models (GPT-5.2 [8], Claude Sonnet 4.6 [1], Gemini 3 Flash [3]) achieve 80-93% judgment accuracy. This result shows that *knowing what happened* and *evaluating whether it is correct* are distinct capabilities in current VLMs and that the second does not follow from the first. This is the core problem we aim to resolve in the proposed studies.

5. PROPOSED RESEARCH

Study 2: Fine-Grained Spatial Error Analysis. The study 1 (Section 4) measures correctness judgment as a binary outcome. Our next step is to decompose errors into fine-grained spatial error types including position offset, stacking level error, and shape confusion. This will identify precisely where VLM spatial reasoning breaks down and inform what targeted representational support is needed for each error type.

Study 3: Multimodal Context Integration. The underlying dataset [12] contains synchronized speech and gesture annotations alongside 3D structure states. Director instructions such as "put the red block to the left of the blue one" provide explicit spatial context that may help a VLM resolve ambiguous context. We will investigate whether integrating transcribed speech and gesture improves judgment accuracy and at what cost in system complexity. This experiment extends our study to multimodal learning analytics.

Study 4: AI Tutor Prototype and Learning Outcomes. The final phase will integrate findings from Studies 1-3 into an AI tutor prototype and evaluate it in a controlled user study. Participants will complete a block construction task with or without AI tutor correctness feedback. The primary outcome is task accuracy, and the secondary measure is spatial reasoning transfer assessed through a post-task evaluation. This study will provide the ecological validity and learning science evidence.

6. EXPECTED CONTRIBUTIONS

Our current study and future research will produce four contributions to the AI and education communities. First, a benchmark for evaluating VLMs as AI tutors in real collaborative physical manipulations with reconstructable 3D ground-truth annotations. Second, empirical evidence that action identification and spatial correctness judgment are distinct VLM capabilities with clear implications for tutoring system design. Third, design guidelines for representing 3D spatial information to VLMs covering input modality and multimodal context integration. Fourth, a controlled study connecting VLM-based tutoring feedback to student learning outcomes in a physical construction task. Taken together, these contributions bridge the gap between VLM capability evaluation and practical AI tutoring design in physical STEM learning environments.

7. CONCLUSION

Physical STEM manipulation is a rich and underexplored setting for AI tutoring research at the intersection of computer vision and learning sciences. This study addresses a concrete open problem: current VLMs can recognize student actions but cannot reliably evaluate whether those actions are spatially correct. Bridging this gap requires better spatial reasoning models and a clearer understanding of what representational support AI tutors need in physical environments. Addressing this gap will advance both the theoretical understanding of VLM spatial reasoning and the practical development of AI tutoring systems for education.

8. REFERENCES

[1] Anthropic. System card: Claude Sonnet 4.6. Technical report, Anthropic, February 2026. Accessed: 2026-03-17.

[2] B. Chen, Z. Xu, S. Kirmani, B. Ichter, D. Sadigh, L. Guibas, and F. Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14455–14465, 2024.

[3] G. DeepMind. Gemini 3 flash technical report. Technical report, Google DeepMind, 2025.

[4] I. Khebour, R. Brutti, I. Dey, R. Dickler, K. Sikes, K. Lai, M. Bradford, B. Cates, P. Hansen, C. Jung, et al. When text and speech are not enough: A multimodal dataset of collaboration in a situated task. *Journal of Open Humanities Data*, 10:1–14, October 2024.

[5] F. Liu, G. Emerson, and N. Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023.

[6] W. Ma, H. Chen, G. Zhang, Y.-C. Chou, J. Chen, C. de Melo, and A. Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International*

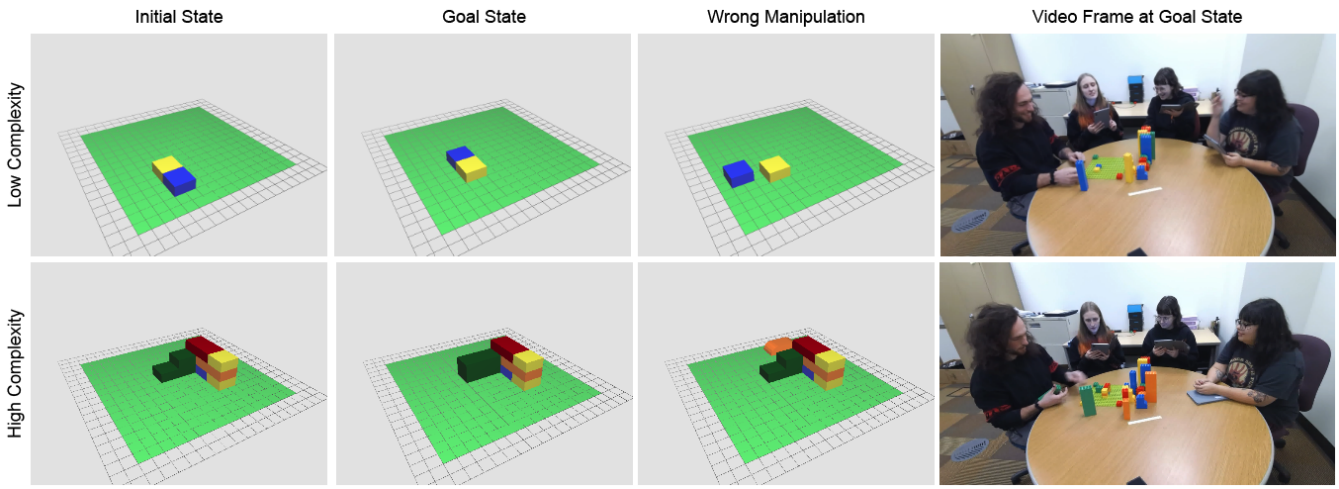


Figure 1: Examples used in the preliminary study to illustrate manipulation judgment across two complexity levels. The top row shows a low complexity structure, and the bottom row shows a high complexity structure. The four columns show the Initial State, Goal State, Wrong Manipulation, and Video Frame at Goal State. In the low complexity example, the required action is placing a square blue block next to the yellow block, but the blue block is placed in the wrong location. In the high complexity example, the required action is to place the square green block at level 2, but an orange double-sided curve block is placed at the back of the structure.

Conference on Computer Vision, pages 6924–6934, 2025.

[7] J. Macina, N. Daheim, S. Chowdhury, T. Sinha, M. Kapur, I. Gurevych, and M. Sachan. Mathdial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. In H. Bouamor, J. Pino, and K. Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5602–5621, Singapore, December 2023. Association for Computational Linguistics.

[8] A. Singh, A. Fry, A. Perelman, A. Tart, A. Ganesh, A. El-Kishky, A. McLaughlin, A. Low, A. Ostrow, A. Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

[9] A. Tack and C. Piech. The ai teacher test: Measuring the pedagogical ability of blender and gpt-3 in educational dialogues. *arXiv preprint arXiv:2205.07540*, 2022.

[10] H. VanderHoeven, M. Bradford, C. Jung, I. Khebour, K. Lai, J. Pustejovsky, N. Krishnaswamy, and N. Blanchard. Multimodal design for interactive collaborative problem-solving support. In *International Conference on Human-Computer Interaction*, pages 60–80. Springer, 2024.

[11] K. VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46(4):197–221, 2011.

[12] Y. Zhu, C. Jung, K. Lai, V. Venkatesha, M. Bradford, J. Fitzgerald, H. Jamil, C. Graff, S. K. G. Kumar, B. Draper, N. Blanchard, J. Pustejovsky, and N. Krishnaswamy. Multimodal common ground annotation for partial information collaborative problem solving. In *Proceedings of the 21st Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-21)*, pages 85–91, Düsseldorf, Germany, Sept. 2025. Association for Computational Linguistics.

Towards Modeling Affecto-Cognitive-Metacognitive Interactions in Machine Learning Education

Aditya Rajmane*
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India
aditya.rajmane@iitb.ac.in

Ramkumar Rajendran
Centre for Educational Technology
Indian Institute of Technology Bombay
Mumbai, India
ramkumar.rajendran@iitb.ac.in

Kshitij Sharma
Department of Computer Science
Norwegian University of Science and Technology
Trondheim, Norway
kshitij.sharma@ntnu.no

ABSTRACT

This doctoral research investigates the interplay between cognition, affect and metacognition (CAM) in students' engagement with machine learning (ML) tasks in computer-based learning environments. While prior work has examined these CAM processes, limited research has explored how they dynamically interact during complex, data-driven problem-solving. To address this gap, this work proposes the design and development of a tutoring system (ML-Tutor) to teach Machine Learning. This system captures learners' interactions and identifies the affective and metacognitive processes while learning. The research conceptualizes metacognition as embedded within the learning activity and operationalizes it through planning, monitoring, and control processes. A preliminary paper-and-pencil study has informed the design of tasks, prompts, and data collection strategies, demonstrating the feasibility of eliciting multimodal traces of learner activity. This work is ongoing, with current efforts focused on developing the CBLE, refining the analytical framework, and identifying suitable units of analysis for studying temporally unfolding learning processes. Feedback is sought on the conceptual framing, operationalization of metacognition, and methodological approach for analyzing multimodal data.

Keywords

Affective Computing, Cognitive Processes, Metacognition, Multimodal Learning Analytics, Intelligent Tutoring Systems, Computer-based Learning Environment, Personalization

1. INTRODUCTION

Student engagement is widely recognized as a core component of learning outcomes, particularly in complex, open-ended domains that require sustained effort and self regulation. Some perspectives conceptualize engagement as a multidimensional construct encompassing behavioral, cognitive, and affective components, each contributing uniquely to learning processes ([12, 5, 13]). In recent years, this view has been extended to emphasize the dynamic and situated nature of engagement, where learners' involvement fluctuates over time in response to task demands, contextual factors, and internal states ([20]). This shift toward process-oriented understandings of engagement has become especially relevant in computer-based learning environments, where data enables the study of engagement as it unfolds in real time [2].

Affective Engagement

Affective engagement [19] has emerged as a critical dimension of learning, particularly in computer-based learning environments where learners interact with complex, open-ended tasks. Affective states are not merely byproducts of learning but play a constitutive role in influencing attention, persistence, and strategy use [7]. As such, understanding affective engagement is essential for designing learning environments that can support productive learning trajectories. With the integration of machine learning techniques, [17] detecting affect is possible in more authentic environments. Within the context of educational data mining, affect detection has relied on multimodal learning analytics, enabling richer representations of learners' experiences across cognitive and behavioral dimensions [16]. Current affective computing systems largely treat affect as an observable signal to be detected and responded to, with limited consideration of its interaction with cognitive processes during problem solving [17].

Affect, Cognition and Metacognition

The relationship between affect and cognition is central to understanding self-regulated learning [6]. Efklides' Metacognitive and Affective Model of Self-Regulated Learning [8] (MASRL) conceptualizes this interaction by linking cognitive processing, metacognitive monitoring, and affective experiences such as effort, difficulty, and confidence. Within this framework, affect actively shapes learners' judgments, strategy use, and regulation during task engagement. Recent work has further emphasized the importance of capturing these processes through fine-grained, multimodal data,

*corresponding author

Aditya Rajmane, Ramkumar Rajendran, and Kshitij Sharma. Towards Modeling Affecto-Cognitive-Metacognitive Interactions in Machine Learning Education. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 892–895. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039859

demonstrating that understanding self-regulated learning requires integrating trace data, self-reports, observable interaction data [11, 9, 10]. There also is a dynamic and temporal nature of affect and metacognitive processes during learning. Studies have shown that learners' emotional states evolve over time and are closely tied to regulation and task performance [15]. However, there is limited research that integrates cognition, metacognition, and affect to provide a holistic understanding of learning processes. One key challenge lies in the multimodal nature of data collection. Affective states are often captured using modalities such as facial expressions, while cognitive and metacognitive processes are typically inferred from interaction data such as clickstreams. Integrating these heterogeneous data sources remains complex.

> Addressing the challenge of integrating the multimodalities can enable a more comprehensive modeling of learner behavior. Such integrated models have the potential to better capture the dynamics of learning and support the design of more effective, personalized scaffolding strategies in computer-based learning environments.

2. OUR WORK

Our work is situated in the context of machine learning (ML) education for school students. The focus is on developing a foundational understanding of core ML concepts such as classification, features, and similarity-based reasoning through accessible, real-world analogies and interactive activities. The learning tasks are designed to progressively move from intuitive understanding to structured reasoning. For instance, students engage with classification problems using everyday categories (e.g., fruit, objects) before transitioning to simplified representations of algorithms such as K-Nearest Neighbors (KNN). These tasks are intentionally designed to surface cognitive challenges and uncertainty, creating opportunities to observe how learners respond to difficulty in emerging domains such as ML.

The following research goals (RG's) were formulated:

1. To investigate how pedagogical approaches for teaching classification, feature identification, and similarity-based reasoning can be translated into a CBLE.

2. To design and develop a CBLE that embeds opportunities for eliciting and capturing learners' cognitive, metacognitive, and affective processes during task engagement.

Paper-Pencil Based Intervention

A preliminary paper-and-pencil study was conducted to study the feasibility of creating a CBLE aligning with RG 2. The study involved school students in the age group of 11 - 14, engaging with introductory machine learning (ML) concepts through structured activities focused on classification and similarity-based reasoning.

Pedagogical Design and Methodology

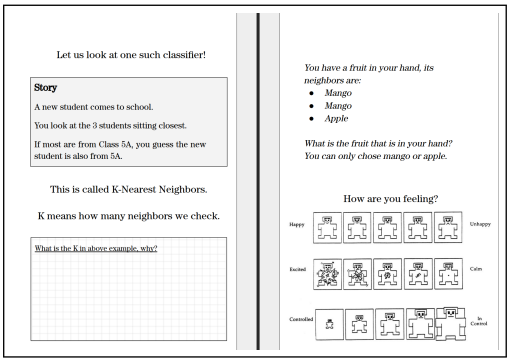


Figure 1: Excerpts from Paper-and-Pencil Intervention

The instructional content was adapted from the CBSE Artificial Intelligence curriculum for grades 6-8, ensuring alignment with age-appropriate learning objectives and prior exposure. Interactive tools such as Teachable Machine [4] were incorporated to enable students to train and test simple models, making abstract processes visible and manipulable. Students then engaged in structured activities like solving classification problems using tabular patient data, requiring them to identify relevant features, compare instances, and make decisions using nearest-neighbor logic. These activities were followed with reflective prompts (e.g., "How are you feeling?") and self-report checklists in the end to encourage monitoring of effort, difficulty, and persistence.

The affective self-report items were adapted from established instruments, the Self-Assessment Manikin (SAM Manikin) and Semantic Differential Scales [3], which have been widely used to capture affective responses in learners of comparable age groups (11 to 14 years). The study was conducted with eight ($N = 8$) participants aged 11 to 14. This design intentionally integrates cognitive engagement with opportunities for metacognitive reflection and affective expression, while also serving as a preliminary exploration of how such task structures and measures may be adapted for implementation within a computer-based learning environment.

Data Collection Strategy

Prior to data collection, formal ethics approval was obtained from the Institute's Ethics and Review Board. Informed consent was obtained from guardians, and assent was obtained from participating students. A multi-source data collection approach was adopted to capture learners' cognitive and affective experiences during task engagement. Data sources included students' responses to structured worksheets (Ref. 1) and embedded self-report measures capturing affective states such as effort, difficulty, and persistence. The instructional material was organized into discrete activities, with affective self-reports using the Self-Assessment Manikin (SAM) [3] administered at the end of each activity to capture affect at multiple points during the learning process 3 In addition to self-reports, video recordings of participants were collected to enable analysis of behavioral and physiological signals. These recordings allow for the extraction of heart rate using remote photoplethysmography (rPPG), a non-contact method for estimating physiological signals from facial videos [21, 18, 1]. To support temporal align-



Figure 2: Student Participant

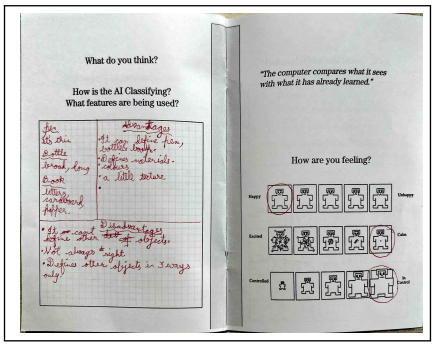


Figure 3: Student Work

ment, the teacher maintained timestamps corresponding to each activity, enabling synchronization of video, self-report, and task data. This multimodal and temporally structured design provides a foundation for examining how affective states, cognitive processes, and physiological indicators co-occur during different phases of task engagement.

Preliminary Observations

While no formal analysis has been conducted, the collected multimodal data provides an initial basis for exploring relationships between cognitive processes, affective experiences, and physiological indicators such as heart rate (rPPG), informing the design of a subsequent computer-based learning environment.

3. CBLE AND EMBEDDED METACOGNITION

The next phase of our research involves the design of an adaptive computer-based learning environment (CBLE) that embeds metacognitive processes within task interaction. Rather than treating metacognition as an external construct measured through post-hoc reports or think-aloud protocols, this work conceptualizes metacognition as an activity that is enacted during learning.

In this work, metacognition is operationalized through three core processes: planning, monitoring, and control [14]. The CBLE will incorporate explicit support for planning by prompting learners to articulate their approach before attempting a task. For example, *learners may be asked to identify which features are relevant for a classification problem or to select a strategy for solving it.* These prompts are designed to provide observable indicators of how learners prepare to engage with machine learning tasks. Such interactions allow planning to be captured as part of the learning process rather than inferred retrospectively. Monitoring processes will be embedded through periodic prompts and checkpoints during task engagement. *These include self-reports of perceived difficulty, effort, and confidence, as well as affective indicators captured through instruments such as the Self-Assessment Manikin. Artifacts for monitoring such as an interactive clock and progress bar will also be available.* By integrating these prompts within task flow, the system captures learners' moment-to-moment awareness of their cognitive and affective states. Control processes will be inferred from learners' interactions with the system, including actions such as *revisiting problems, modifying responses, switching strategies, and persisting after incorrect attempts.* These behaviors provide observable evidence of how learners regulate their learning in response to perceived difficulty or feedback. By capturing these actions alongside planning and monitoring data, the system enables the examination of how metacognitive regulation unfolds during task engagement.

These processes will be integrated within the CBLE consisting of a domain module, a pedagogical module, and a learner module. The domain module will structure machine learning tasks such as classification and, in future iterations, clustering. The pedagogical module will orchestrate prompts and scaffolds that elicit planning, monitoring, and control. Content will be in the form of text, videos, and interactive animations. The assessment will include MCQ's, MSQ's and descriptive questions. The learner module will capture multimodal data, including task performance, self-reports, and physiological signals, enabling the construction of temporally aligned traces of learner activity. Together, this design provides a foundation for studying how metacognitive processes are enacted and how they relate to cognitive and affective dynamics during learning.

4. FEEDBACK SOUGHT

a. Theoretical–Operational Alignment: The adequacy of the proposed conceptualization of affect, cognition, and metacognition, particularly the alignment between the MASRL inspired cognitive–affective loops and their operationalization through multimodal indicators. Whether the mapping from latent constructs (e.g., perceived difficulty, confidence, regulatory control) to observable features is sufficiently well-specified and theoretically defensible.
b. Unit of Analysis and Temporal Segmentation: The validity and analytical utility of defining these affecto-metacognitive episodes as the primary unit of analysis in computer-based learning environments, including the proposed criteria for segmenting continuous interaction data into episodes and the implications of segmentation choices for modeling temporal dynamics and cognitive–affective coupling.

5. CONCLUSION

This work outlines a research trajectory for studying the dynamic interplay between affect, cognition, and metacog-

nition in ML learning environments through the design of an adaptive tutoring system. The preliminary study informs this direction, with future work focusing on refining the analytical framework and leveraging multimodal data from CBLEs to examine learning processes at a finer temporal level.

6. ACKNOWLEDGMENTS

The work is part of Enhancing AI Education between India and Norway (INNOAIED) project (INCP2-2024/10367). The authors would like to thank the support provided by members of the Learning Analytics Lab at IIT Bombay.

7. REFERENCES

[1] Y. Benezeth, D. Krishnamoorthy, D. J. Botina Monsalve, K. Nakamura, R. Gomez, and J. Mitéran. Video-based heart rate estimation from challenging scenarios using synthetic video generation. *Biomedical Signal Processing and Control*, 96:106598, Oct. 2024.

[2] M. Bond, O. Viberg, and N. Bergdahl. The current state of using learning analytics to measure and support K-12 student engagement: A scoping review. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, LAK2023, pages 240–249, New York, NY, USA, Mar. 2023. Association for Computing Machinery.

[3] M. M. Bradley and P. J. Lang. Measuring emotion: The self-assessment manikin and the semantic differential. *Journal of Behavior Therapy and Experimental Psychiatry*, 25(1):49–59, Mar. 1994.

[4] M. Carney, B. Webster, I. Alvarado, K. Phillips, N. Howell, J. Griffith, J. Jongejan, A. Pitaru, and A. Chen. Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–8, Honolulu HI USA, Apr. 2020. ACM.

[5] S. L. Christenson, A. L. Reschly, and C. Wylie, editors. *Handbook of Research on Student Engagement*. Springer US, Boston, MA, 2012.

[6] S. de Mooij, J. Lämsä, L. Lim, O. Aksela, S. Athavale, I. Bistolfi, F. Jin, T. Li, R. Azevedo, M. Bannert, D. Gašević, S. Järvelä, and I. Molenaar. A Systematic Review of Self-Regulated Learning through Integration of Multimodal Data and Artificial Intelligence. *Educational Psychology Review*, 37(2):54, June 2025.

[7] S. K. D'mello and J. Kory. A Review and Meta-Analysis of Multimodal Affect Detection Systems. *ACM Computing Surveys*, 47(3):1–36, Apr. 2015.

[8] A. Efklides. Affect, Epistemic Emotions, Metacognition, and Self-Regulated Learning. *Teachers College Record: The Voice of Scholarship in Education*, 119(13):1–22, Dec. 2017.

[9] Y. Fan, L. Lim, J. van der Graaf, J. Kilgour, M. Raković, J. Moore, I. Molenaar, M. Bannert, and D. Gašević. Improving the measurement of self-regulated learning using multi-channel data. *Metacognition and Learning*, 17(3):1025–1055, Dec. 2022.

[10] Y. Fan, M. Rakovic, J. van der Graaf, L. Lim, S. Singh, J. Moore, I. Molenaar, M. Bannert, and D. Gašević. Towards a fuller picture: Triangulation and integration of the measurement of self-regulated learning based on trace and think aloud data. *Journal of Computer Assisted Learning*, 39(4):1303–1324, 2023.

[11] Y. Fan, J. van der Graaf, L. Lim, M. Raković, S. Singh, J. Kilgour, J. Moore, I. Molenaar, M. Bannert, and D. Gašević. Towards investigating the validity of measurement of self-regulated learning based on trace data. *Metacognition and Learning*, 17(3):949–987, Dec. 2022.

[12] J. A. Fredricks, P. C. Blumenfeld, and A. H. Paris. School Engagement: Potential of the Concept, State of the Evidence. *Review of Educational Research*, 74(1):59–109, 2004.

[13] S. Heikkinen, M. Saqr, J. Malmberg, and M. Tedre. A longitudinal study of interplay between student engagement and self-regulation. *International Journal of Educational Technology in Higher Education*, 22(1):21, Apr. 2025.

[14] J. E. Jacobs and S. G. Paris. Children's Metacognition About Reading: issues in Definition, Measurement, and Instruction. *Educational Psychologist*, 22(3-4):255–278, June 1987. _eprint: https://doi.org/10.1080/00461520.1987.9653052.

[15] S. P. Lajoie, J. Zheng, S. Li, A. Jarrell, and M. Gube. Examining the interplay of affect and self regulation in the context of clinical reasoning. *Learning and Instruction*, 72:101219, Apr. 2021.

[16] K. Mangaroska, K. Sharma, D. Gasevic, and M. Giannakos. Multimodal Learning Analytics to Inform Learning Design: Lessons Learned from Computing Education. *Journal of Learning Analytics*, 7(3):79–97, Dec. 2020.

[17] L. Mathur, M. Mataric, and L.-P. Morency. Expanding the Role of Affective Phenomena in Multimodal Interaction Research. In *Proceedings of the 25th International Conference on Multimodal Interaction*, ICMI '23, pages 253–260, New York, NY, USA, Oct. 2023. Association for Computing Machinery.

[18] D. McDuff, S. Gontarek, and R. Picard. Remote measurement of cognitive stress via heart rate variability. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, 2014:2957–2960, 2014.

[19] R. W. Picard. *Affective computing*. MIT Press, Cambridge, Mass., 1. paperback ed edition, 2000.

[20] G. M. Sinatra, B. C. Heddy, and D. Lombardi. The Challenges of Defining and Measuring Student Engagement in Science. *Educational Psychologist*, 50(1):1–13, Jan. 2015. _eprint: https://doi.org/10.1080/00461520.2014.1002924.

[21] W. Verkruysse, L. O. Svaasand, and J. S. Nelson. Remote plethysmographic imaging using ambient light. *Optics express*, 16(26):21434–21445, Dec. 2008.

When and How Does LLM Support Help Learning? Toward Data-Driven Design Principles for AI-Enhanced Educational Interfaces

Eason Chen
Carnegie Mellon University
eason.tw.chen@gmail.com

ABSTRACT

Large language models are rapidly deployed in educational systems, yet fundamental questions remain about when LLM support helps learning and when it may hinder it. This dissertation develops design principles for LLM-enhanced educational interfaces through three studies: (1) an experiment showing that expertise paradoxically reduces AI collaboration effectiveness in educational decision-making; (2) a classroom deployment revealing that conversational chatbot usage is negatively associated with exam performance while structured proof-review is not; and (3) a controlled experiment demonstrating that open-ended self-explanation with LLM feedback significantly enhances transfer in calculus ($p = .030$) while requiring $3.5\times$ fewer practice problems. Drawing on learning analytics and educational data mining perspectives, this work contributes methods for automated assessment of student reasoning ($\kappa = 0.78$ vs. human consensus), behavioral pattern detection from interaction logs, and adaptive feedback design informed by data-driven student models.

Keywords

Large language models, Self-explanation, Automated assessment, Learning analytics, Learner behavior modeling, Human-AI collaboration, Educational data mining

1. PROBLEM AND MOTIVATION

The integration of large language models into education promises scalable, personalized support that was previously possible only through human tutoring or expensive rule-based systems [9]. However, emerging evidence suggests that the *form* of LLM support matters as much as its availability. Unconstrained conversational AI can reduce student-generated reasoning and encourage answer outsourcing [15], while structured interfaces that anchor feedback in student work may better preserve productive struggle [1]. Yet we lack a systematic understanding of *which* design choices matter and *why*.

Self-explanation, the process of articulating the reasoning behind a solution, is one of the most robust learning strategies in educational research [3, 12]. However, it faces longstanding practical barriers: it is time-costly [4], explanation quality varies widely [2], and benefits may disappear when learning time is equated with additional practice [10]. LLMs could provide scalable feedback on student-generated explanations, but whether this preserves the generative processing that makes self-explanation effective remains unclear.

These challenges point to a fundamental gap: while LLMs are being widely adopted in education, we lack principled design guidelines grounded in empirical evidence about which forms of AI support enhance learning and which undermine it. My dissertation addresses this gap through the following research question:

How can LLM-supported self-explanation be designed to maximize learning transfer, and what design principles govern the effectiveness of AI feedback on student reasoning?[1]

2. THEORETICAL FRAMING

I draw on three complementary perspectives.

Self-explanation theory establishes that articulating reasoning promotes learning by helping learners identify gaps and integrate new knowledge [3]. Decades of work in intelligent tutoring systems has shown that prompting students to explain solution steps improves learning, particularly when combined with tutorial dialogue that ensures explanation quality [1, 2]. The open question is whether LLM feedback preserves these generative benefits or short-circuits them.

The ICAP framework [5] predicts that learning outcomes improve as engagement moves from passive to active to constructive to interactive. Self-explanation is constructive; when LLM feedback responds adaptively to student explanations, the interaction may approach the interactive level, where each partner's contribution triggers further knowledge-building. This predicts that open-ended self-explanation with adaptive AI feedback should outperform menu-based selection.

The assistance dilemma [6] and the Knowledge-Learning-Instruction (KLI) framework [7] provide a principled basis for understanding when instructional support helps versus hin-

Eason Chen. When and How Does LLM Support Help Learning? Toward Data-Driven Design Principles for AI-Enhanced Educational Interfaces. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 896–900. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040097

[1]To preserve double-blind review, we do not cite the author's own publications in this submission. Full citations will be included in the camera-ready version.

ders. The assistance dilemma captures the tension between providing information (reducing errors but potentially reducing learning) and withholding it (increasing productive struggle). The KLI framework specifies that optimal instruction depends on the type of knowledge: sense-making knowledge benefits from constructive activities such as self-explanation, while routine knowledge benefits from practice [8]. In AI-assisted contexts, this manifests as over-reliance (answer outsourcing) versus under-reliance (dismissing valuable feedback) [14]. Interface design and learner characteristics both moderate these patterns [13, 11].

Together, these predict that effective LLM-supported self-explanation requires interfaces that promote generative processing before feedback, provide quality-ensuring feedback without bypassing reasoning, and calibrate support to the type of knowledge and learner characteristics [8].

3. RESEARCH PROGRESS

My research has progressed from understanding how humans interact with AI in educational contexts (Study 1) to examining how LLM interface design shapes learning outcomes in classrooms (Study 2) to directly testing LLM-supported self-explanation as a mechanism for transfer (Study 3). Each study builds on insights from the previous one, progressively narrowing from broad human-AI interaction patterns to specific design principles for self-explanation feedback.

3.1 Study 1: Expertise and AI Reliance in Educational Decision-Making (Published)

Motivation. Before designing LLM tools for learners, I wanted to understand how people use AI recommendations in educational contexts and how expertise moderates reliance. If we cannot predict who benefits from AI assistance, we cannot design effectively for diverse learners.

Method. A within-subjects study ($N = 95$) examined how experienced tutors (experts) and non-tutors (novices) evaluated tutor-praise responses under three AI-assisted conditions: no AI, AI with textual reasoning explanations, and AI with inline highlighting. The task involved judging the pedagogical appropriateness of tutor praise for student work, requiring subject-matter expertise. I decomposed errors into over-reliance (accepting incorrect AI recommendations) and under-reliance (rejecting correct AI advice), treating these as distinct behavioral indicators rather than collapsing them into a single accuracy metric.

Results. AI assistance significantly improved accuracy over unassisted performance ($p = .009$), but human-AI collaboration consistently underperformed the AI-only baseline (88% accuracy), quantifying the "cost of human oversight." A paradox of expertise emerged: novices outperformed experts when AI was present ($p = .021$) because they followed AI suggestions more closely, while experts frequently overrode correct advice. Textual reasoning reduced under-reliance when AI was correct ($p < .05$) but increased over-reliance when AI was wrong, revealing a fundamental trade-off in explanation design. A repeated-measures ANOVA on time cost showed significant main effects for both highlighting ($p < .001$) and reasoning ($p < .001$), with a significant interaction ($p = .001$): users substituted between explanation types rather than simply adding reading time.

Implications. Expertise moderates AI support effectiveness, and explanation design involves fundamental trade-offs between over-reliance and under-reliance. Through the lens of the KLI framework [7], the expertise paradox can be understood as a knowledge-type mismatch: experts' deep domain knowledge leads them to override AI on the basis of sense-making processes, even when the AI's pattern-matching is more accurate for this specific task. These insights motivated me to examine how interface *constraints*, not just explanations, shape the effectiveness of LLM support for learning.

This study demonstrates a key methodology for learning analytics: decomposing aggregate accuracy into process-level indicators (over-reliance, under-reliance, time cost) that reveal the mechanisms behind human-AI interaction effects. This contrasts with prior work that treats AI assistance as a binary treatment and overlooks the quality of human-AI coordination.

3.2 Study 2: Conversational vs. Structured LLM Support in a Course (Under Review)

Motivation. Study 1 showed that interface design affects AI reliance in a controlled task. Study 2 tests whether these dynamics generalize to authentic classroom settings with real learning goals.

Method. We deployed an LLM-powered tool in an undergraduate discrete mathematics course ($N = 148$) using a staggered-access design. The tool integrated two LLM-powered components: a chatbot for conversational math support and a proof-review tool that provides structured, localized feedback on students' written proof attempts. The proof-review tool requires students to first produce their own proof before receiving feedback, while the chatbot allows open-ended conversation. We analyzed adoption patterns, outcome associations, and mediation pathways using log data combined with course performance records.

Results. Earlier access improved homework scores ($B = 2.71$, $p = .026$) but not exam scores. After controlling for self-efficacy and prior performance, chatbot usage negatively predicted exam performance ($B = -0.030$, $p = .014$), while proof-review showed no such association ($p = .480$). To understand this divergence, we coded chatbot transcripts for answer-seeking behaviors (directly requesting solutions rather than conceptual explanations). Students classified as answer-seekers scored significantly lower ($d = 0.55$), and a serial mediation model revealed that chatbot usage partially mediated the self-efficacy $\rightarrow$ exam pathway.

Implications. The proof-review tool, which requires students to generate their own work first (analogous to self-explanation), showed no negative association, while the open-ended chatbot did. This motivated Study 3: if anchoring LLM feedback in student-generated reasoning is key, then explicitly scaffolding self-explanation with LLM feedback should enhance learning. However, this study is observational; usage-outcome associations may reflect selection effects (struggling students may use the chatbot more). A

controlled experiment was needed to test the causal mechanism.

This study exemplifies learning analytics methodology: mining interaction logs to discover behavioral patterns (answer-seeking) and testing predictive models (regression, mediation) for learning outcomes. The answer-seeking classifier represents a potentially deployable tool for real-time intervention in AI-assisted learning environments.

3.3 Study 3: LLM-Supported Self-Explanation in Calculus (Under Review, Central to Thesis)

Motivation. Studies 1 and 2 converged on the hypothesis that LLM interfaces promoting generative processing should yield better learning outcomes. Study 3 directly tests this through a controlled experiment.

Method. A between-subjects experiment ($N = 92$) compared three conditions in a calculus learning environment: no self-explanation (control, $n = 29$), menu-based self-explanation ($n = 35$), and open-ended self-explanation with LLM feedback ($n = 28$). All participants had a fixed 60-minute practice session. Transfer was measured via "Not Enough Information" (NEI) items requiring metacognitive judgment about whether a problem provides sufficient information. Participants were recruited via Prolific and screened for algebra proficiency without advanced calculus background.

The automated evaluation pipeline scored student explanations against a four-level rubric (0, 0.3, 0.7, 1.0) capturing completeness and conceptual accuracy. Two human graders achieved $\kappa = 0.70$ (quadratic-weighted); agreement between the LLM and human consensus was $\kappa = 0.78$ (substantial). The system provided color-coded quality indicators (red/yellow/green) after each attempt, with corrective explanations after two consecutive low-quality responses.

Results. The open-ended condition significantly outperformed control on NEI transfer questions ($\beta = +11.9$ percentage points, $p = .030$, $d = 0.44$; adjusted $R^2 = .204$), while completing 3.5× fewer practice problems (16.9 vs. 58.9) with comparable overall learning gains (+9.3% vs. +9.4%). Menu-based self-explanation did not differ from control ($p = .343$). This directly addresses the concern that self-explanation benefits disappear when learning time is equated [10].

Notably, open-ended participants completed 3.5× fewer problems yet achieved comparable learning gains, suggesting that the time spent explaining was at least as productive as additional practice. They also spent more time on the post-test (20.6 vs. 14.9 minutes, $p \approx .09$), writing more detailed explanations, a behavioral indicator that the intervention shifted their problem-solving approach toward more generative engagement. For NEI multiple-choice items (identifying insufficient information without requiring explanation), the open-ended condition showed a non-significant advantage ($\beta = +9.3\%$, $p = .183$, $d = 0.26$), suggesting that the self-explanation effect is strongest when assessed through explanation quality rather than answer accuracy alone.

Open questions. The sample was not drawn from a classroom. The transfer effect was specific to NEI items; whether it extends to other transfer forms is unknown. The feedback design was fixed; which components are essential has not been isolated.

This study makes a computational contribution to educational data mining: an LLM-based automated assessment pipeline achieving substantial inter-rater reliability with human graders, enabling scalable feedback provision and data collection. This pipeline could be applied to other domains requiring assessment of open-ended student reasoning.

4. PROPOSED THESIS DIRECTION

Study 3's finding that LLM-supported open-ended self-explanation enhances transfer forms the central contribution of my thesis. The finding that open-ended self-explanation with LLM feedback enhances transfer while requiring dramatically fewer practice problems opens several questions: which feedback components are responsible, whether fixed-format feedback can be improved through personalization, and whether these effects hold in authentic classrooms. My planned work addresses these through three directions:

Feedback mechanism decomposition. Study 3 bundles rubric-based scoring, color-coded indicators, and corrective explanations into a single treatment. Which components drive the transfer effect? I plan factorial experiments that systematically vary feedback type (score only vs. score + hint vs. full explanation) and feedback timing (immediate vs. delayed) to isolate causal effects. The automated evaluation pipeline from Study 3 provides the infrastructure to maintain consistency across conditions while scaling to larger samples. This addresses a gap in the self-explanation literature, where feedback designs are typically studied as monolithic treatments rather than decomposed into their constituent mechanisms.

Adaptive feedback intensity. A recurring pattern across my studies is that many comparisons yielded null effects: explanation styles did not improve accuracy in Study 1, proof-review showed no positive association in Study 2, and menu-based self-explanation did not differ from control in Study 3. One interpretation is that fixed-format support is too rigid: the same feedback may over-assist strong students (reducing productive struggle) while under-assisting weak ones (failing to correct misconceptions). I plan to investigate feedback that adapts to explanation quality in real time: minimal acknowledgment for strong explanations (preserving autonomy), targeted hints for partial ones (scaffolding without giving away the answer), and corrective feedback only after repeated failures (preventing frustration). This directly operationalizes the assistance dilemma [6] through data-driven student models, and the KLI framework [7] provides principled criteria for when sense-making support should be intensified versus withdrawn.

Classroom deployment. Study 3 used Prolific participants in a single session. Validating these effects in an authentic course would strengthen ecological validity and test whether benefits persist over longer periods, including across exam performance. The staggered-access methodology from Study 2 provides a template for quasi-experimental deployment. Longitudinal interaction data would also enable trajectory analysis of explanation quality over time, connecting to the EDM

community's work on early warning systems and detector-based interventions. Building on Study 2's answer-seeking coding, I plan to develop automated classifiers from interaction logs that could trigger just-in-time interventions when students shift toward unproductive usage patterns.

4.1 Research Timeline

My timeline prioritizes the highest-impact extensions first. In the near term (next 6 months), I will conduct the feedback mechanism decomposition experiment, as this directly builds on the existing Study 3 infrastructure and addresses the most pressing open question. Concurrently, I will begin developing the answer-seeking classifier using coded transcripts from Study 2 as training data. In the medium term (6–12 months), I will implement the adaptive feedback system, informed by both the decomposition results and trajectory analysis from Study 3 data. The classroom deployment will follow (12–18 months), integrating the adaptive system into an authentic course using the staggered-access design from Study 2.

5. EXPECTED CONTRIBUTIONS

- **Empirical evidence** that LLM-supported open-ended self-explanation enhances transfer ($p = .030$) with 3.5× fewer problems, directly addressing concerns that self-explanation benefits disappear when learning time is equated with additional practice [10].
- **Automated assessment and behavioral detection** pipelines: an LLM-based explanation evaluator ($\kappa = 0.78$ vs. human consensus) enabling scalable feedback, and an answer-seeking classifier from interaction logs enabling just-in-time interventions. These address key bottlenecks in both self-explanation research and EDM early warning systems.
- **Error decomposition methodology** for human-AI interaction, decomposing outcomes into over-reliance, under-reliance, and time cost as process-level indicators, grounded in the KLI framework's distinction between knowledge types [7]. This provides diagnostic information beyond aggregate accuracy for designers of AI-enhanced educational tools.
- **Design guidelines for LLM integration in education**, synthesizing across all three studies: (1) anchor AI feedback in student-generated work; (2) design for generative processing before assistance; (3) monitor and intervene on unproductive usage patterns; (4) calibrate support intensity to knowledge type, learner characteristics, and explanation quality, operationalizing the assistance dilemma [6] through data-driven student models.

6. FEEDBACK SOUGHT

My dissertation is at the transition between completed empirical work and planned extensions. I would particularly value feedback on:

1. **Thesis scope:** Should I focus narrowly on self-explanation feedback design (decomposing which feedback components drive transfer), or maintain the broader framing that connects interface design, reliance patterns, and self-explanation across all three studies?
2. **Transfer measurement:** My current transfer measure is specific (recognizing insufficient information). What additional measures of deep understanding would strengthen the claim that LLM-supported self-explanation promotes genuine conceptual transfer?
3. **Addressing null effects:** Many comparisons across my studies were non-significant. Is adaptive feedback intensity the most promising direction, or are there other design levers more likely to produce robust learning gains?
4. **Theoretical framing:** Is the ICAP-based argument (that LLM feedback elevates self-explanation from constructive to interactive) well-supported, or are there alternative explanations for the transfer effect that the EDM community would find more compelling?
5. **Ecological validity:** What study designs would most convincingly demonstrate that these effects matter in real classrooms, beyond single-session Prolific experiments?

7. REFERENCES

[1] V. Aleven and K. R. Koedinger. An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. *Cognitive Science*, 26(2):147–179, 2002.

[2] V. Aleven, K. R. Koedinger, and O. Popescu. The need for tutorial dialog to support self-explanation. 2000.

[3] M. T. Chi, M. Bassok, M. W. Lewis, P. Reimann, and R. Glaser. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, 13(2):145–182, 1989.

[4] M. T. Chi, N. De Leeuw, M.-H. Chiu, and C. LaVancher. Eliciting self-explanations improves understanding. *Cognitive Science*, 18(3):439–477, 1994.

[5] M. T. Chi and R. Wylie. The ICAP framework: Linking cognitive engagement to active learning outcomes. *Educational Psychologist*, 49(4):219–243, 2014.

[6] K. R. Koedinger and V. Aleven. Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review*, 19:239–264, 2007.

[7] K. R. Koedinger, A. T. Corbett, and C. Perfetti. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, 36(4):757–798, 2012.

[8] K. R. Koedinger, J. C. Stamper, E. A. McLaughlin, and T. Nixon. Testing theories of transfer using error rate learning curves. *Topics in Cognitive Science*, 8(3):589–609, 2016.

[9] X. Li et al. Bringing generative AI to adaptive learning in education. In *Proceedings of AIED 2024*, 2024.

[10] K. L. McEldoon, K. L. Durkin, and B. Rittle-Johnson. Is self-explanation worth the time? a comparison to additional practice. *British Journal of Educational Psychology*, 83(4):615–632, 2013.

[11] K. Morrison et al. The impact of AI explanations on human decision-making. In *Proceedings of CHI 2023*, 2023.

[12] B. Rittle-Johnson, A. M. Loehr, and K. Durkin. Promoting self-explanation to improve mathematics

learning: A meta-analysis and instructional design principles. *ZDM Mathematics Education*, 49:599–611, 2017.

[13] H. Vasconcelos et al. Explanations can reduce overreliance on AI systems during decision-making. In *Proceedings of the ACM on Human-Computer Interaction*, 2023.

[14] O. Vereschak, G. Bailly, and B. Caramiaux. How to evaluate trust in AI-assisted decision making? a survey of empirical methodologies. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2):1–39, 2021.

[15] C. Zhai et al. The effects of over-reliance on AI dialogue systems on students' cognitive abilities. *Smart Learning Environments*, 11:28, 2024.

Assessing Early Literacy at Scale: Format Effects in Unsupervised Digital Assessment

Harriet Crisp
EIDU, Berlin
harriet.crisp@eidu.com

Amar Lalwani
EIDU, Berlin
amar.lalwani@eidu.com

ABSTRACT

Assessing foundational literacy at scale in low-resource settings is challenging due to the cost of assessor-led testing. Digital platforms offer a scalable alternative, but the performance of different item formats under unsupervised conditions, particularly for young learners, remains underevidenced.

This paper analyses a large-scale deployment of digital assessments delivered through EIDU's platform-based education programme in pre-primary classrooms in Kenya, comprising multiple-choice questions (MCQs) and very short answer questions (VSAQs). Across 230,000 learners and 1.8 million item responses, MCQs achieved significantly higher completion and scores than VSAQs, with item format explaining 31% of variance in completed-assessment scores. However, clustering incorrect VSAQ responses revealed highly structured patterns of partial knowledge, response-format confusion, and emerging phonics understanding that binary scoring entirely obscures.

Binary correctness scoring of both MCQs and VSAQs showed no significant association with an independent literacy assessment (IDELA; linked sample $n = 384$), but classifying incorrect VSAQ responses by degree of understanding uncovered a significant directional signal, though whether this generalises beyond the small validation sample remains an open question. The results highlight a trade-off between MCQ engagement and the richer learner signal in VSAQ errors, motivating further validation of their diagnostic value.

Keywords

Digital assessment, Foundational literacy, Very short answer questions (VSAQs), Unsupervised learning environments, Low-resource settings

1. INTRODUCTION & MOTIVATION

It is widely recognised that foundational literacy and numeracy skills are essential predictors of future academic success [6, 22], therefore accurate assessment of these skills is essential. This is of particular importance to learners in low- and middle-income countries (LMICs), who consistently demonstrate substantial gaps in these critical areas. National assessments in a Sub-Saharan African country indicate that 60% of Grade 3 learners fail to achieve minimum proficiency levels in reading [29], compared to less than 10% in high income countries [13].

Many widely used child assessment instruments were developed and normed in high-income settings, raising concerns about contextual fit, feasibility, and interpretability when transferred to low-resource environments [7, 8, 14]. In response, LMIC-focused tools have emerged for early learning measurement, including population- and program-evaluation instruments such as IDELA [18] and MELQO [21], and foundational skill assessments such as the Early Grade Reading Assessment (EGRA) and Early Grade Mathematics Assessment (EGMA) [5, 23]. Both EGRA and EGMA have been widely adopted in low- and middle-income contexts for system diagnosis and program evaluation.

However, because these assessments are typically assessor-led, one-to-one, and time-intensive, they are logistically infeasible at scale, constraining their use for continuous learning measurement [3]. This has motivated the development of technology-mediated assessments, including tablet- and computer-based approaches that automate administration and reduce logistical burden [2, 15, 19].

While self-administered digital assessments such as SA-EGRA and SA-EGMA can align well with enumerator-led tools under supervision [26], evidence suggests that assessment validity depends strongly on task format. Although evidence on whether open-ended formats outperform MCQs is mixed [32], spelling tasks using very short answer questions (VSAQs), which require learners to actively type brief responses (e.g., a letter or short word), have shown stronger alignment with non-digital counterparts than MCQs, despite yielding lower raw scores [25]. This advantage is partly attributable to the ability to capture partial correctness, which carries meaningful signal about emerging literacy skills beyond binary scoring [16, 24]. However, given young children's sensitivity to interaction design [4], it remains unclear whether these diagnostic benefits persist in fully unsupervised deployments—an important requirement for scalable assessment in low-resource contexts.

Harriet Crisp, and Amar Lalwani. Assessing Early Literacy at Scale: Format Effects in Unsupervised Digital Assessment. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 901–908. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21040036

This uncertainty mirrors findings from a broader assessment literature showing that short-answer formats elicit deeper cognitive processing than MCQs, which primarily measure recognition and are more susceptible to guessing [1, 20]. VSAQs better reflect retrieval processes and reduce cueing bias; experimental evidence from digital platforms shows that fill-in questions can lead to better learning outcomes than MCQs through productive struggle, albeit at the cost of lower raw performance [9, 10, 31]. However, this evidence comes largely from older learners in higher-education settings with stable literacy and test familiarity. Whether VSAQs retain diagnostic value for young learners in unsupervised, low-income digital environments, where more difficult tasks increase the risk of disengagement, remains an open question.

This paper addresses this gap by examining how VSAQs function in a large-scale, unsupervised early-grade digital assessment deployment in low-resource settings. Analysis of item-level constructed responses beyond binary correctness reveals systematic patterns of partial knowledge and response-format confusion, which are aggregated into response-aware attempt-quality features. These features yield the only significant directional association with independent literacy outcomes in a linked validation sample, where binary-scored MCQs and VSAQs do not, though cross-validation was inconclusive at the available sample size. The results characterise a trade-off between the accessibility of MCQs and the diagnostic richness of VSAQs, and motivate further validation of richer scoring approaches for unsupervised early-grade assessment.

2. STUDY CONTEXT AND DEPLOYMENT

The analyses in this paper are drawn from EIDU, a large-scale platform-based education programme deployed in low-resource pre-primary and primary school settings. The platform delivers structured daily literacy and numeracy lesson plans aligned to local curricula through low-cost touchscreen Android phones and is used by teachers during regular classroom instruction. The system operates entirely offline, with content and learner interactions stored locally on the device.

Typically, each class has access to a single device. Teachers use the phone to guide lessons, after which it is shared among learners, who take turns interacting individually with the software to complete digital personalised learning (DPL) games without one-to-one adult supervision. Learners spend approximately five minutes each on literacy and numeracy activities daily, with all interaction data anonymised and collected for analysis. The platform is deployed at scale, reaching over 700,000 learners across multiple low- and middle-income countries in Sub-Saharan Africa; prior large-scale experimental work has shown that this delivery model substantially improves foundational literacy and numeracy outcomes over a one-year period [11].

Within the platform, short assessment units were embedded into learners' individual sessions to measure progress against weekly literacy learning objectives. These assessments were delivered alongside DPL games, rather than as a distinct testing activity. As a result, learners were typically unaware that they were being assessed, and all assessment interactions occurred under the same unsupervised conditions as routine learning activities. This design reduces reactivity and testing effects, a persistent challenge in early childhood assessment, while enabling data collection at scale with minimal disruption to classroom routines.

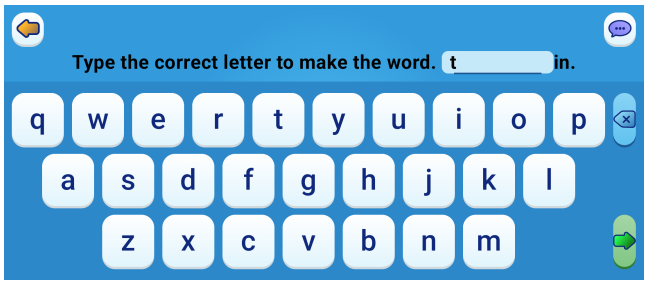


Figure 1: Example of a Blending item (VSAQ).

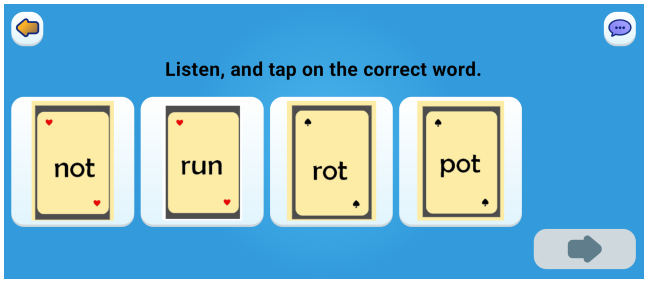


Figure 2: Example of a Word Recognition item (MCQ).

2.1 Assessment Design

A literacy assessment unit was designed for each lesson week during the rollout, aligned to that week's learning objective. Each unit comprised five items and used a single response format: either very short answer input via an on-screen keyboard (VSAQ) or multiple-choice selection (MCQ).

Three item types were used. **Blending** items (Week 1, 3, 6, and 8 assessments) required learners to identify the initial phoneme of a spoken three-letter word by typing the first letter (VSAQ; Figure 1). **Word Spelling** items (Weeks 4 and 5) required learners to type the full three-letter word after hearing the audio prompt (VSAQ). **Word Recognition** items (Weeks 2 and 7) required learners to select the written word matching the audio from four options (MCQ; Figure 2).

2.2 Serving and Interaction Conditions

Assessment units were unlocked once the teacher marked the fifth lesson of the corresponding week as complete, and served to each learner as the first unit in their next DPL session. Learners could exit units at any point, proceeding directly to standard DPL games.

All assessments were completed under unsupervised conditions. Teachers did not receive real-time feedback during assessment attempts, and learners received no guidance beyond the on-screen and audio instructions provided by the software. As a result, assessment outcomes reflect not only learners' underlying knowledge, but also interaction behaviour, including misunderstanding of task requirements, typing errors, and other interface-related effects.

2.3 Study Sample and Data

The assessments were rolled out to Pre-Primary 2 (PP2) learners in Kenya, typically around five years old, over six weeks during the final term of the 2025 school year (September–October). Of approximately 420,000 PP2 learners active on the platform during the roll-out window, around 230,000 learners (55%) were served at least one assessment; the remaining learners were in classes that did not reach the lesson-completion thresholds required to unlock assessments. Across these learners, approximately 1.8 million item responses were collected, averaging 1.6 completed assessments per learner.

In addition to the digital assessment data, assessor-administered IDELA assessments were conducted in October 2025 with a randomly selected sample of PP2 learners across 60 schools in a single programme county. Of the 477 learners assessed, 384 also completed at least one digital assessment and are included in subsequent analyses. Mean IDELA literacy scores were similar for the full sample (63.9%) and the linked subsample (64.3%). IDELA is delivered orally in one-to-one settings and aggregates performance across foundational subtasks including letter–sound knowledge, print awareness, and expressive vocabulary, providing a holistic independent benchmark for early literacy.

3. METHODOLOGY

In the operational system, assessment items were scored using binary correctness, based on whether the learner's response exactly matched the expected answer. For completed assessments, unit-level scores were computed as the average proportion of correct responses. The analysis proceeds in three stages:

1. An initial descriptive analysis of assessment outcomes and raw responses.
2. Item-level analysis and clustering of incorrect VSAQ inputs.
3. Predictive modelling to evaluate how different representations of learner performance align with external outcomes.

The descriptive findings from Stage (1) are reported in the Results section; this section describes the response processing and feature construction steps used in Stages (2) and (3).

3.1 Response Processing

While the operational system scored items as binary correct or incorrect, prior research shows that partially correct constructed responses can contain meaningful learning signal beyond correctness alone [17, 27, 28]. To capture this signal, incorrect VSAQ responses were examined to identify systematic patterns in learner input.

Unique incorrect response strings were grouped using hierarchical agglomerative clustering over character n-gram Term Frequency-Inverse Document Frequency (TF-IDF) representations, with a distance threshold used to induce clusters. Inspection of these clusters revealed recurring structural response forms and their relative frequency in the data. These insights informed the definition of a fixed set of twenty interpretable *response pattern* categories (e.g., partial phoneme matches, character repetitions, reversals, or copying visible text). All incorrect responses were subsequently assigned to one of these categories through manual labelling, with ambiguous cases resolved through author discussion.

Importantly, all response clustering and category assignment was conducted independently of, and prior to, the statistical analyses linking digital features to external outcomes; categories were defined based on interpretation of learner cognition, not optimised for downstream predictive performance.

3.2 Feature Construction

Each granular response pattern category was assigned to one of five ordered *attempt-quality levels*, reflecting increasing degrees of task engagement and correctness:

- **Non-Meaningful Attempt**: Inputs showing no evidence of meaningful item engagement (e.g., *qwerty*).
- **Low-Quality Attempt**: Responses indicating limited engagement with the item but no clear evidence of item-relevant understanding (e.g., entering the letters displayed on the screen).
- **Partially Correct Attempt**: Responses demonstrating partial or emerging understanding of the target concept, but which are incomplete or incorrect (e.g., only writing the first two letters of the word in a Word Spelling assessment).
- **Conceptually Correct Attempt**: Responses that reflect correct conceptual understanding but fail to meet the formatting expected by the interface (e.g. writing the full word in a Blending assessment where only the first letter is required).
- **Fully Correct Attempt**: Responses that satisfy formal correctness under the operational scoring rules.

For each learner, the sum of items in each attempt-quality level was computed and normalised by the total number of items attempted, meaning both complete and incomplete assessments could be included in analysis. These aggregated representations serve as the primary features for evaluating alignment with external learning outcomes.

3.3 Baselines

To assess the added value of response-aware representations, the proposed features were compared against two baselines: correctness-only VSAQ scores (proportion of Fully Correct attempts under binary scoring) and correctness-only MCQ scores.

3.4 Ethical Considerations

This study used anonymised data collected as part of routine platform operations, conducted with the consent and support of participating counties and local stakeholders. Personally identifiable information is encrypted locally on devices, and gatekeeper consent is obtained for all learners.

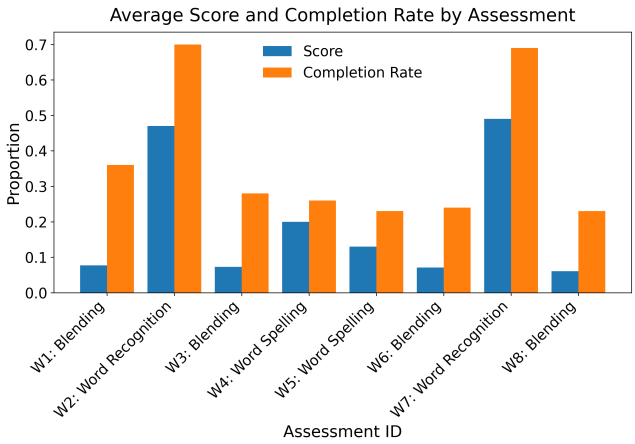


Figure 3: Completion Rate and Score by Assessment ID.

4. RESULTS

4.1 Completion and Performance

Completion rate reflects the proportion of learners who submit responses to all five items in an assessment, while score reflects correctness conditional on completion. As shown in Figure 3, both metrics varied strongly by assessment format. MCQ-based assessments (Weeks 2 and 7) achieved substantially higher completion and scores than VSAQ-based assessments (all other weeks). Assessment format alone explained 31% of the variance in completed-assessment scores ($R^2 = 0.313$), with MCQ assessments yielding scores 38 percentage points higher on average ($p < 0.001$). It should be noted that since both MCQ assessments had four response options, random guessing would yield an expected score of 25%, partly contributing to the higher raw scores observed relative to VSAQs. Format effects were also evident for completion: a logistic regression predicting assessment completion from format alone showed that MCQ assessments were over three times more likely to be completed (odds ratio ≈ 3.1, $p < 0.001$; pseudo-$R^2 = 0.047$).

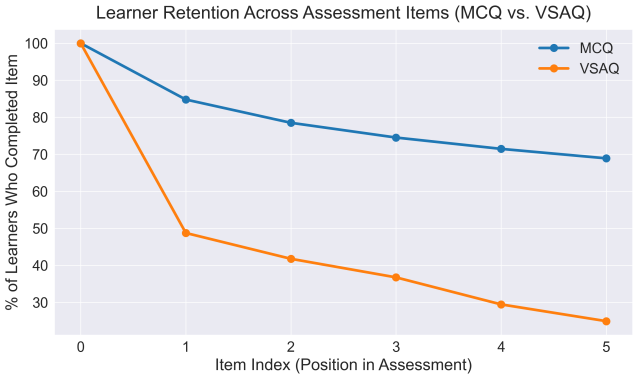


Figure 4: Learner retention across items for MCQ- vs VSAQ-based assessments.

Figure 4 shows the proportion of learners who attempted each item for MCQ vs VSAQ-based assessments, normalised to the introductory screen. All learners shown completed the introductory item, which required only tapping *"Start"* to proceed. As both formats comprised five scored items, differences in dropout rates cannot be attributed to assessment length. Approximately 85% of learners who reached the introduction attempted the first scored MCQ item, compared to only 49% for VSAQs, indicating an immediate response-format barrier rather than gradual disengagement. Among those who attempted the first item, subsequent attrition remained higher for VSAQs (24% drop-off vs. 16% for MCQs). This pattern is consistent with sustained difficulty in responding to typed-input items relative to recognition-based formats.

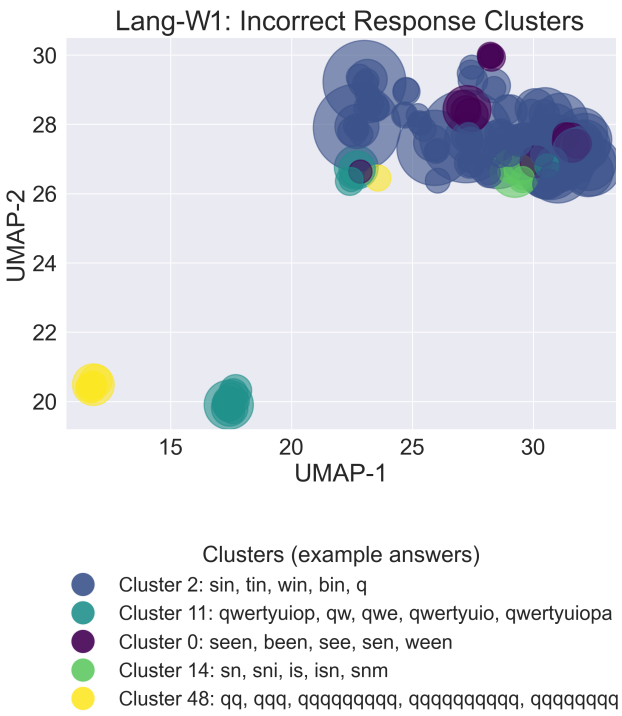


Figure 5: Largest Incorrect Response Clusters for Week 1 Blending assessment.

4.2 Systematic Patterns in Item Responses

Item-level analysis of incorrect VSAQ responses revealed highly structured and interpretable patterns rather than random error.

Figure 5 shows a two-dimensional Uniform Manifold Approximation and Projection (UMAP) visualisation of the five largest clusters of incorrect responses for the Week 1 Blending assessment, in which learners were prompted to enter the initial letter of a spoken three-letter word with the rime *in*. The clustering reveals various recurring response forms, including full word entry, rime substitution, letter reversal, character repetition, and keyboard-swipe inputs. Despite reflecting very different levels of understanding, all are marked as incorrect for failing the formal constraint of providing only the first letter.

Similar response forms were observed consistently across items and weeks, indicating systematic rather than item-

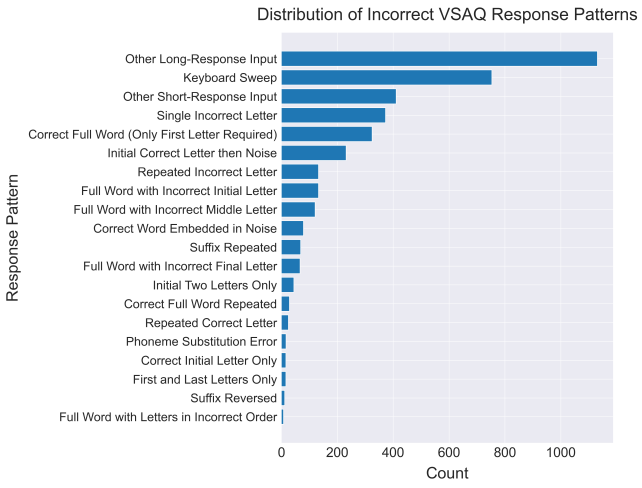


Figure 6: Distribution of response pattern categories across all VSAQ incorrect items.

specific behaviour. Across the four Blending weeks, the most common incorrect response for 90% of items was typing the full word rather than the initial letter, indicating recognition of the target phoneme but misunderstanding of the response requirement. Conversely, in Word Spelling assessments, where the full word was required, a substantial proportion of incorrect responses consisted of only the initial letter. This asymmetry suggests confusion between visually similar task formats that place different demands on learner input. More generally, these response patterns reflect structured partial understanding rather than random error, highlighting signal that binary scoring obscures.

Figure 6 shows the distribution of incorrect VSAQ item responses across the twenty response pattern categories. The three largest categories, accounting for 57% of attempts, reflect disengaged inputs: keyboard sweeps and other unstructured long or short responses. Beyond these, the most frequent patterns indicate partial or misdirected task engagement, such as single incorrect letters, entry of the full word and provision of the correct initial letter followed by unstructured text.

Figure 7 shows the distribution of item responses across the five higher-level attempt-quality categories. While fully correct responses account for only 7% of attempts, a further 22% fall into intermediate categories that reflect partial or conceptual correctness constrained by response format. This structure motivates the use of response-aware representations rather than binary correctness when modelling early learning outcomes.

4.3 Alignment with External Outcomes

To evaluate the extent to which digital assessment signals reflect underlying literacy ability, digital learner representations are compared against assessor-administered IDELA literacy scores, a widely used measure of early literacy skills.

Linear regression models were used to compare three representations of digital learner performance against IDELA literacy scores. In all models, the dependent variable was the

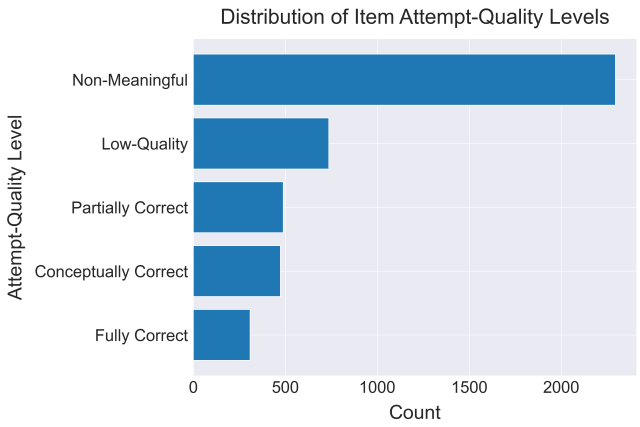


Figure 7: Distribution of item responses across attempt-quality levels.

learner's IDELA literacy score, and predictors were standardised learner-level features derived from digital assessments. The following representations were compared:

1. **MCQ correctness-only**: the proportion of MCQ items answered correctly.
2. **VSAQ correctness-only**: the proportion of VSAQ items answered strictly correctly.
3. **Response-aware VSAQ features**: the distribution of attempt-quality levels across VSAQ items.

Regression coefficients therefore represent the expected percentage point (pp) change in IDELA literacy score associated with a one standard deviation (SD) increase in the corresponding digital assessment feature.

Table 1: Comparison of digital assessment representations for predicting IDELA literacy scores.

Model	R^2	CV R^2	Model p-value
MCQ (correctness)	0.005	-0.064	0.283
VSAQ (correctness)	0.018	-0.062	0.009
VSAQ (response-aware)	0.095	0.008	< 0.001

As shown in Table 1, predictive alignment was weak for both binary-correctness representations: correctness-only VSAQ models explained little variance ($R^2 = 0.018$), while MCQ-only models explained almost none ($R^2 = 0.005$) and were not statistically significant. In contrast, the response-aware VSAQ model achieved higher, though still modest, explanatory power in sample ($R^2 = 0.095$) with monotonic associations with IDELA literacy scores. However, 10-fold cross-validated R^2 was near-zero or negative across all three models, with the response-aware model the only one to retain a positive R^2_{CV} (0.008). With each fold containing only approximately 38 learners, stable out-of-sample estimation is difficult at this sample size [30], and larger-scale external validation is needed.

Despite the correctness-only models explaining little variance in external literacy outcomes, Table 2 shows that cor-

Table 2: Learner-level linear regressions predicting IDELA literacy score: correctness-only models.

Model	Variable	Mean	SD	Coef.	p
MCQ	Intercept	–	–	66.36	< 0.001
	Correct (prop.)	0.43	0.50	1.10	0.283
VSAQ	Intercept	–	–	64.33	< 0.001
	Correct (prop.)	0.07	0.16	2.31	0.009

Table 3: Learner-level linear regression predicting IDELA literacy score: response-aware VSAQ model. (Per-item effects assume five items per assessment, i.e., one item corresponds to 0.2 of the proportion scale)

Attempt Type	Mean	SD	Coef.	p	pp per Item
Intercept	–	–	64.33	0.000	–
Low-Quality	0.17	0.18	-0.00	1.000	ns
Partially Correct	0.12	0.16	2.78	0.001	3.48
Conceptually Correct	0.11	0.17	3.92	0.000	4.61
Fully Correct	0.07	0.16	2.62	0.002	3.28

rectness coefficients were positive in both cases, indicating directional association with higher literacy scores, with a larger effect for VSAQs (2.31 vs. 1.10), though both were weak and, for MCQs, not statistically significant.

In contrast, Table 3 shows that response-aware VSAQ features recover richer signal. In this model, Non-Meaningful attempts serve as the reference category. Partially and conceptually correct attempt-quality features both retain independent predictive power even when controlling for fully correct responses. Conceptually correct attempts show the largest association (3.92pp per SD), exceeding that of fully correct attempts (2.62pp), suggesting that responses reflecting correct understanding but incorrect formatting carry meaningful signal. As noted above, these are in-sample associations and larger-scale validation is needed to establish whether they generalise.

5. DISCUSSION AND CONCLUSIONS

This study examines how pre-primary learners engage with different digital assessment formats in large-scale, unsupervised, low-resource settings, and what this implies for scalable learning measurement. Consistent with prior work, MCQ-based assessments were substantially easier for learners to complete: scores were 38 percentage points higher and completion odds were more than three times those of VSAQ-based assessments. In contrast, VSAQs exhibited sharp early attrition, with around half of learners exiting before attempting the first scored item. This pattern points to an immediate interaction or response-format barrier rather than gradual disengagement. While such barriers are typically mitigated in supervised settings, where an adult ensures task completion, in unsupervised contexts increased task difficulty directly reduces data coverage. This gap may be partially addressed through improved task scaffolding to clarify response expectations.

A central finding of this study is that incorrect VSAQ responses can contain highly structured and interpretable signals. Learners frequently produced responses reflecting partial phonological knowledge, confusion between task formats (e.g., writing the full word when only the initial letter was required, and vice versa), and emerging but incomplete literacy skills. These patterns were consistent across items and weeks, indicating systematic rather than idiosyncratic behaviour. The attempt-quality taxonomy developed here captures a gradient of learner understanding that binary scoring collapses entirely.

These response-aware features yielded the only statistically significant in-sample association with IDELA literacy scores, where correctness-only representations did not. However, cross-validated performance was weak across all models. Low explained variance is common when linking digital process data to external benchmarks in low-income early-childhood settings [12, 33], and in the present study, noise is compounded by fully unsupervised administration on shared classroom devices, a narrow sub-skill focus relative to the holistic IDELA composite [18], and a small single-county validation sample ($n = 384$). The key question for future work is therefore not whether alignment is strong in absolute terms, but whether the directional signal observed here can be confirmed with larger and more diverse validation samples.

Together, the findings suggest that moving beyond binary scoring of constructed responses may provide diagnostic value in unsupervised early-grade digital assessment. MCQs sustain engagement, but VSAQ errors carry structured learner signal that conventional scoring discards. Hybrid designs combining both formats, alongside richer scoring and improved scaffolding to mitigate early attrition, offer a promising direction for scalable learning measurement in these settings.

6. LIMITATIONS AND FUTURE WORK

This study has several limitations. The external validation sample is small ($n = 384$), drawn from a single county, and insufficient to support stable out-of-sample predictive validation. Digital assessment data are observed only for learners in classes that progressed far enough to unlock assessments, introducing potential selection bias, and response pattern categories were assigned through potentially subjective manual labelling. The analyses are limited to PP2 learners and a single language and interface design, limiting generalisability to other age groups or contexts. MCQ and VSAQ formats were confounded with item type (word recognition vs. blending/spelling), limiting causal attribution of observed differences to response format alone.

Future work should prioritise larger-scale external validation with independently sampled learners across multiple regions. Validating the attempt-quality framework across additional grades, subjects, languages, and deployment settings will be essential for establishing its broader applicability. Scalable and automated approaches to response classification should also be explored, alongside examination of how response-aware signals can be incorporated into adaptive instruction or feedback. The sharp initial attrition observed for VSAQs points to the need for improved task scaffolding, such as clearer instructions, worked examples, or tighter input constraints, to reduce response-format confusion while preserving diagnostic value.

7. REFERENCES

[1] M. G. Carneiro Queiroz, F. C. Specian Junior, P. T. Hamamoto Filho, T. M. Santos, S. K. Schauber, A. M. Woltman, and D. Cecilio-Fernandes. Comparison of cognitive workload between very short answer questions and multiple-choice questions: an eye-tracking experiment. *Medical Education Online*, 31(1):2621434, 2026.

[2] K. Carson, G. Gillon, and T. Boustead. Computer-administrated versus paper-based assessment of school-entry phonological awareness ability. *Asia Pacific Journal of Speech, Language and Hearing*, 14(2):85–101, 2011.

[3] M. Clarke and D. Luna-Bazaldua. Primer on large-scale assessments of educational achievement. Technical report, World Bank, 2021. © World Bank. CC BY 3.0 IGO.

[4] L. W. Drozdick, K. Getz, S. E. Raiford, and O. Zhang. WPPSI-IV: Equivalence of Q-interactive and paper formats. Q-interactive technical report 14, Pearson, 2016.

[5] M. M. Dubeck and A. Gove. The early grade reading assessment (egra): Its theoretical foundation, purpose, and limitations. *International Journal of Educational Development*, 40:315–322, 2015.

[6] G. J. Duncan, C. J. Dowsett, A. Claessens, K. Magnuson, A. C. Huston, and P. Klebanov. School readiness and later achievement. *Developmental Psychology*, 43(6):1428–1446, 2007.

[7] P. L. Engle, L. C. H. Fernald, H. Alderman, J. R. Behrman, C. O'Gara, A. Yousafzai, M. C. de Mello, M. Hidrobo, N. Ulkuer, I. Ertem, and S. Iltus. Strategies for reducing inequalities and improving developmental outcomes for young children in low-income and middle-income countries. *The Lancet*, 378(9799), 2011.

[8] L. C. H. Fernald, P. Kariger, P. L. Engle, and A. Raikes. Examining early child development in low-income countries: A toolkit for the assessment of children in the first five years of life. Technical report, 2009.

[9] A. Gurung, K. Vanacore, A. A. McReynolds, K. S. Ostrow, E. Worden, A. C. Sales, and N. T. Heffernan. Multiple choice vs. fill-in problems: The trade-off between scalability and learning. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, pages 507–517, 2024.

[10] C. F. Herrmann-Abell, J. Hardcastle, and G. E. DeBoer. Exploring the comparability of multiple-choice and constructed-response versions of scenario-based assessment tasks. *Grantee Submission, NARST Annual International Conference*, 2022.

[11] L. Major, R. Daltry, M. Otieno, K. Otieno, A. Zhao, C. Sun, J. Hinks, A. Friedberg. Digital personalised learning to improve literacy and numeracy outcomes: a randomised controlled trial in kenyan pre-primary classrooms. *Research Papers in Education, 1–36*, 2026.

[12] M. S. McHenry, D. Mukherjee, S. Bhavnani, A. Kirolos, J. D. Piper, M. M. Crespo-Llado, and M. J. Gladstone. The current landscape and future of tablet-based cognitive assessments for children in low-resourced settings. *PLOS Digital Health*, 2(2):e0000196, 2023.

[13] I. V. S. Mullis, M. O. Martin, P. Foy, and M. Hooper. Pirls 2016 international results in reading, 2017.

[14] S. Nag. Assessment of literacy and foundational learning in developing countries: Final report. Technical report, Health & Education Advice and Resource Team (HEART), UK Department for International Development (DFID), 2017.

[15] M. Neumann and D. Neumann. Validation of a touch screen tablet assessment of early literacy skills and a comparison with a traditional paper-based assessment. *International Journal of Research and Method in Education*, 42:1–14, 2018.

[16] G. Ouellette and M. Sénéchal. Pathways to literacy: A study of invented spelling and its role in learning to read. *Child Development*, 79(4):899–913, 2008.

[17] R. Pelánek and T. Effenberger. Beyond binary correctness: Classification of students' answers in learning systems. *User Modeling and User-Adapted Interaction*, 30:867–893, 2020.

[18] L. Pisani, I. Borisova, and A. J. Dowd. Developing and validating the international development and early learning assessment (idela). *International Journal of Educational Research*, 91:1–15, 2018.

[19] N. J. Pitchford and L. A. Outhwaite. Can touch screen tablets be used to assess cognitive and motor skills in early years primary school children? a cross-cultural study. *Frontiers in psychology*, 7:1666, 2016.

[20] T. Puthiaparampil and M. M. Rahman. Very short answer questions: a viable alternative to multiple choice questions. *BMC medical education*, 20(1):141, 2020.

[21] A. Raikes, R. Sayre, D. Davis, K. Anderson, M. Hyson, E. Seminario, and A. Burton. The measuring early learning quality & outcomes initiative: purpose, process and results. *Early Years*, 39(4):360–375, 2019.

[22] E. Romano, L. Babchishin, L. S. Pagani, and D. Kohen. School readiness and later achievement: Replication and extension using a nationwide survey. *Developmental Psychology*, 46(5):995–1007, 2010.

[23] RTI International. Early grade mathematics assessment (egma) toolkit. Technical report, United States Agency for International Development (USAID), 2009.

[24] RTI International. Spelling, reading comprehension, and oral language subtasks: Complements to the early grade reading assessment. Technical Brief, 2017.

[25] RTI International. Self-administered egra/egma: Results of pilot test. Technical report, 2022.

[26] RTI International. Self-administered egra/egma: Results of pilot test in chichewa. Technical report, 2023.

[27] A. H. Sam, R. Westacott, M. Gurnell, R. Wilson, K. Meeran, and C. Brown. Comparing single-best-answer and very-short-answer questions for the assessment of applied medical knowledge in 20 UK medical schools: cross-sectional study. *BMJ Open*, 9(9):e032550, 2019.

[28] L. W. T. Schuwirth and C. P. M. van der Vleuten. Programmatic assessment: From assessment of

learning to assessment for learning. *Medical Teacher*, 33(6):478–485, 2011.

[29] Uwezo. Are all our children learning? uwezo 7th learning assessment report, 2021.

[30] A. Vabalas, E. Gowen, E. Poliakoff, and A. J. Casson. Machine learning algorithm validation with a limited sample size. *PLOS ONE*, 14(11):e0224365, 2019.

[31] E. V. van Wijk, R. J. Janse, B. N. Ruijter, J. H. Rohling, J. van der Kraan, S. Crobach, M. d. Jonge, A. J. d. Beaufort, F. W. Dekker, and A. M. Langers. Use of very short answer questions compared to multiple choice questions in undergraduate medical students: an external validation study. *PLoS One*, 18(7):e0288558, 2023.

[32] X. Wang, C. Rose, and K. Koedinger. Seeing beyond expert blind spots: Online learning design for scale and quality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2021.

[33] H. Yuan, M. Ocansey, S. Adu-Afarwuah, M. Sheridan, A. Hamoudi, H. Okronipa, S. M. Kumordzie, B. M. Oaks, and E. L. Prado. Evaluation of a tablet-based assessment tool for measuring cognition among children 4–6 years of age in Ghana. *Brain and Behavior*, 12(10):e2749, 2022.

Decoding Chinese Pinyin Keystrokes by Comparing Transcription and Composition Tasks

Qizhi Xu[1], Zedi Lin[1], Zongting Ge[1], Mo Zhang[2], MengXiao Zhu[1]*
1 University of science and technology of China, 2 Educational Testing Service
fred@ustc.edu.cn, lin1635559009@mail.ustc.edu.cn, gezongting@mail.ustc.edu.cn, mzhang@ets.org, mxzhu@ustc.edu.cn

ABSTRACT
This study investigates the cognitive and behavioral characteristics of Chinese digital writing by modeling Pinyin-mediated keystroke processes. Prior keystroke research has largely focused on alphabetic languages, where keystrokes map directly onto text, limiting transferability to Chinese input methods that involve phonological entry followed by candidate selection. To address this gap, we propose a Dual-layer Rhythm Model (DLRM) that separates physical keystroke execution from cognitive-level selection and commitment processes. Using fine-grained keystroke logs from 34 graduate students completing both a transcription task and an argumentative writing task, we extracted process features related to input rhythm, candidate-selection latency, pause structure, and revision behavior. Within-subject comparisons showed that argumentative writing produced significantly greater rhythm variability, longer selection latencies, and higher revision rates than transcription, indicating increased cognitive load. Correlation analyses further revealed that cognitive-layer pause indicators, especially within-character pauses and post-commit pauses, were significant negative predictors of argument quality, whereas physical-layer speed measures were not. These findings demonstrate that cognitively grounded timing features derived from Pinyin input provide meaningful indicators of writing proficiency and support the development of process-aware models for Chinese writing assessment and intelligent writing support systems.

Keywords
Chinese writing, keystroke logging, Pinyin input, cognitive load, writing process analytics

1. INTRODUCTION
Writing involves complex cognitive processes such as planning, organizing, expressing ideas, and revising[6]. These processes require the coordinated operation of higher-order functions, including working memory, attentional control, and knowledge retrieval[6, 25]. Therefore, inferring writers' cognitive abilities solely from the final text provides an incomplete and potentially biased view, process-level evidence is also needed[8]. A deeper understanding of the cognitive mechanisms underlying the writing process is essential for strengthening the validity of writing assessment, improving writing instruction, and developing automated, process-sensitive interventions[10].

Traditional process-tracing methods such as think-aloud protocols are limited by verbalization costs, incomplete introspective access, and poor scalability[13, 20].

Qizhi Xu, Zedi Lin, Zongting Ge, Mo Zhang, and Mengxiao Zhu. Decoding Chinese Pinyin Keystrokes by Comparing Transcription and Composition Tasks. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 909–916. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.
https://doi.org/10.5281/zenodo.21039903

Keystroke logs provide a window into cognitive processes that are not visible in the final written product [20]. They capture fine-grained behaviors such as typing speed, pauses, revisions, and bursts of text production[7, 20]. Moreover, the high temporal resolution, non-intrusive nature, and ecological validity of keystroke data make it well suited for large-scale studies of writing processes and for developing automated, process-aware writing assessment systems[8, 20].

However, research on keystroke-log analysis for Chinese writing remains far less developed than in English[11]. Chinese writers typically rely on Pinyin input methods as a proxy for text production[34]. Most Chinese input methods involve a two-stage process: writers first enter Pinyin letters that represent the intended Chinese characters, and then identify and select the target character from a list of candidates[21]. This workflow introduces additional cognitive steps, making analytical approaches developed for alphabetic-language keystroke data difficult to transfer to Chinese. Consequently, advancing keystroke-based methods for studying Chinese writing requires process-analytic techniques that explicitly model the Pinyin-mediated input stream[23].

Empirical research on Chinese writing with Pinyin input remains limited. Existing studies have primarily examined how Pinyin input affects lexical processing, while paying less attention to how Pinyin-mediated input unfolds during authentic writing tasks[6]. Systematic analyses that link input-process features to writing quality are especially scarce. Against this backdrop, the present study proposes a method for analyzing keystroke logs of Pinyin-mediated Chinese input, characterizes behavioral features of the input process, and investigates their underlying cognitive mechanisms. We further model the relationship between input processes and writing quality. Specifically, we address the following research questions:

RQ1: What are the key characteristics of Pinyin-mediated Chinese input, and how should these features be interpreted?

RQ2: How does input during writing differ from input during simple typing, and what cognitive activities might these differences reflect?

RQ3: Which input features are associated with writing proficiency, and what are their plausible cognitive interpretations?

2. RELATED WORK
2.1 Chinese Pinyin Keystroke Analysis
Keystroke-log research in English writing has established relatively mature theoretical and methodological foundations[20]. In English, keystroke events map relatively transparently onto the language production process. Accordingly, English keystroke-log studies have focused on physical-level properties of letter input[9], focusing on physical-level properties such as IKI, pause frequency, revision rate, and burst length [7, 20].

However, entering Pinyin letters externalizes only phonological encoding; the intended Chinese characters are not committed to the

text until the writer identifies and selects the target orthographic form from a candidate list (see Figure 1)[16, 26, 28, 38]. This adds a cognitive step from sound-based encoding to character-form selection that does not exist in alphabetic keyboard input.

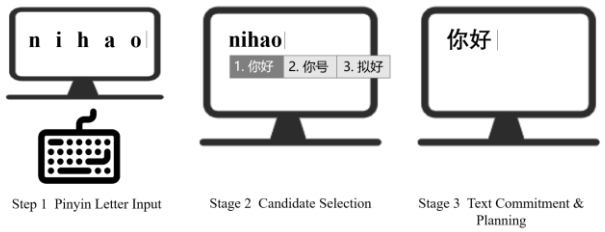


Figure 1. Chinese Pinyin Input Process

Consider intra-word pauses as an example. In Chinese keystroke data, there is no direct counterpart to the English operational definition[16, 29, 30]. Some studies have imported English criteria and defined intra-word pauses as pauses when the cursor is positioned between two Chinese characters within a "word" in the text box[36]. Theoretically, however, such pauses are difficult to interpret as spelling retrieval or orthographic encoding, making the validity of these analyses questionable. More importantly, these definitions ignore pauses at the Pinyin-entry stage and pauses during candidate selection[35].

In Pinyin-mediated input, physical keystrokes represent only the first stage of language production. The cognitive translation from Pinyin to Chinese characters has its own temporal dynamics and cognitive demands[15]. The interval between completing Pinyin entry and confirming a character (i.e., selection latency) is a Pinyin-specific temporal indicator that reflects the cost of orthographic recognition and decision making. Such cognitively meaningful indicators may contain critical information about Chinese writing processes, yet prior Chinese keystroke analyses have not systematically accounted for them[32]. Moreover, relying on cursor position in the text box introduces an additional challenge: accurate Chinese word segmentation, which is itself non-trivial and can propagate error into feature extraction. Together, these issues create substantial technical and theoretical challenges for keystroke-based research on Chinese writing.

More importantly, prior work rarely uses designs that disentangle the cognitive load induced by transcription from the cognitive load of the writing task itself[18, 19]. Keystroke logs reflect an intertwining of input and composition processes: inter-keystroke intervals are shaped by typing proficiency, but they also vary with writers' ongoing planning, formulation, and revision[20, 33]. In Pinyin-mediated Chinese input, transcription itself involves multistage cognitive processing and substantial individual differences. Without controlling for, or directly measuring, the cognitive demands of the input system, it is difficult to estimate the cognitive characteristics of the writing process accurately[14, 37].

2.2 Working-Memory Model of Writing

Baddeley's multicomponent model of working memory provides a useful framework for explaining the cognitive mechanisms involved in writing[3]. The model comprises three subsystems: the phonological loop, the visuospatial sketchpad, and the central executive[4, 5]. These subsystems interact to enable written composition.

Pinyin-mediated input places qualitatively different demands on working memory than handwriting[24]. During phonological encoding, Pinyin input requires writers to activate phonological representations of the intended words and maintain them in the phonological loop[27]. In contrast, during handwriting, many writers rely more on visuospatial memory or directly map orthographic forms to motor actions[17]. During candidate selection, Pinyin input requires maintaining a visual representation of the candidate list in the visuospatial sketchpad while the central executive supports feature comparison and choice decisions. During strategy management, the central executive must monitor the input method's predictions, evaluate the efficiency of the current strategy, and switch strategies when needed[12]. Consequently, especially when the process is not fully automatized, Pinyin input may impose higher working-memory load and reduce the resources available for other components of writing.

Skill acquisition theory posits that complex skills shift from controlled to automatic processing[1, 2, 22] When Pinyin spelling and candidate selection are highly automatized, transcription costs are minimized, freeing cognitive resources for content development[25] . In contrast, non-automatized input competes with composing processes, disrupting idea flow[19, 31]

Building on the cognitive theories reviewed above, we propose the DLRM (Dual-layer Rhythm Model) of Pinyin input as a theoretical framework for analyzing Chinese digital writing processes. The model decomposes Pinyin-mediated input into two relatively independent temporal dimensions: a physical layer and a cognitive layer. The physical layer refers to inter-keystroke intervals between Pinyin letters, which reflect the automatization of Pinyin encoding and the fluency of finger-motor execution. These temporal patterns are primarily shaped by input proficiency; once a sufficient level of skill is reached, Pinyin entry becomes a highly automatized motor program whose rhythm is relatively stable across writing tasks and less sensitive to fluctuations in task-related cognitive load.

In contrast, the cognitive layer captures the interval from completing Pinyin entry to confirming the target Chinese character, as well as intervals between successive character confirmations. These time points mark key cognitive transitions from phonological activation to orthographic retrieval, while also carrying signals of higher-level composing processes such as lexical access, syntactic assembly, and semantic integration. Consequently, cognitive-layer timing is expected to be highly sensitive to writing-task demands. The relationship between the two layers reflects writers' resource allocation strategies: skilled writers can maintain physical-layer fluency while allocating sufficient time at the cognitive layer to support high-quality decisions.

Based on this framework, we propose the following hypotheses:

H1: Compared with a transcription task, an essay-writing task will impose higher cognitive load, reflected in (a) greater variability in physical-layer rhythm, (b) longer cognitive-layer processing time, and (c) more frequent revisions and strategic input behaviors.

H2: Cognitive-layer indicators (e.g., selection latency, inter-character confirmation intervals, and pause patterns during selection) will predict argumentative writing quality more strongly than physical-layer indicators.

3. METHOD

3.1 Participants and Experimental Design

We used a within-subject experimental design to characterize Pinyin-mediated Chinese input behaviors associated with idea

generation under different writing-task conditions. Participants were 39 native Chinese-speaking graduate students, all of whom had extensive experience using Pinyin input methods for computer-based writing. The study was conducted in a controlled, in-person setting. Keystroke data were recorded using the logging module integrated into the CLASS platform.

The experiment included two writing tasks designed to induce different levels of cognitive load during Chinese Pinyin input (see appendix A and B).

The first task was a transcription task, in which participants were asked to accurately copy a provided Chinese philosophical passage within five minutes. The text discussed the nature of logic and knowledge and was approximately 173 characters long. This task served as a low–cognitive-load baseline condition, primarily reflecting transcription and input-rhythm characteristics under minimal idea-generation demands.

The second task was a timed argumentative writing task. Participants were given a structured argument prompt adapted from a standardized analytical writing assessment and were asked to compose a full written response within 30 minutes. The prompt required participants to analyze the assumptions underlying a recommendation and evaluate its logical validity. This task was designed to elicit high-level cognitive activities, including planning, reasoning, and argument construction, under authentic composition conditions.

3.2 Data Collection and Preprocessing

Before feature computation, we cleaned and standardized the raw keystroke logs. We excluded one participant with extremely low task completion and four participants who did not produce valid Pinyin-mediated input, yielding a final analytic sample of 34 participants. We then harmonized potential logging differences caused by input-methods by retaining only events directly related to text production. Finally, we reconstructed each participant's continuous input stream by ordering events chronologically within each task. After cleaning, Task 1 contained 21,975 valid keystroke records and Task 2 contained 99,554 records.

3.3 Modeling Pinyin Input Behaviors

Because Pinyin-mediated Chinese input does not directly produce Chinese characters but proceeds through Pinyin letter entry followed by candidate selection, we explicitly modeled the Pinyin-mediated input process. Using the keystroke logs, we reconstructed each participant's stream of Pinyin letter entry by tracking insertions and deletions at the end of the Pinyin buffer and assigning a timestamp to each letter event. We then computed inter-key intervals (IKIs) at the Pinyin-letter level and extracted summary statistics to characterize input rhythm and its stability. In addition, we quantified deletion behavior as an indicator of online monitoring during transcription.

In Pinyin-mediated input, the commitment of a Chinese character to the final text is treated as a key boundary of text production. End-of-text generation typically reflects the immediate externalization of ideas: writers must concurrently translate concepts into linguistic form, retrieve lexical items, and perform syntactic encoding. Its temporal properties therefore provide a more direct indicator of cognitive load during content generation.

Accordingly, we focused on incremental generation events that increased the text at the end. After each keystroke, if the number of final characters at the end of the text increased relative to the previous event, we labeled the event as an end-of-text generation (i.e., an incremental commit). Events occurring in the middle of the text or involving deletions were excluded from this analysis. For each end-of-text commit, we computed (a) the interval between the final Pinyin-letter keystroke and the appearance of the Chinese character, as an indicator of cognitive effort in candidate recognition and decision making, and (b) the interval between consecutive commits, to capture ideation or planning processes following completion of a production unit.

A data-driven pause threshold of 1,411 ms ($\mu + 1\sigma$ of the pooled inter-event interval distribution, excluding task-unrelated interruptions >10 s) was applied consistently across both tasks.

A key challenge in Pinyin analysis is how to segment Pinyin input in a way that captures individual input habits. To address this, we developed a Pinyin–character alignment model that maps the Pinyin input stream to the final committed Chinese characters(Figure 2). The alignment uses character-commit events as boundaries. For each commit, the model traces backward to retrieve the most recent contiguous sequence of Pinyin letters and treats it as a candidate Pinyin string, which is then enumeratively matched to the committed character(s).

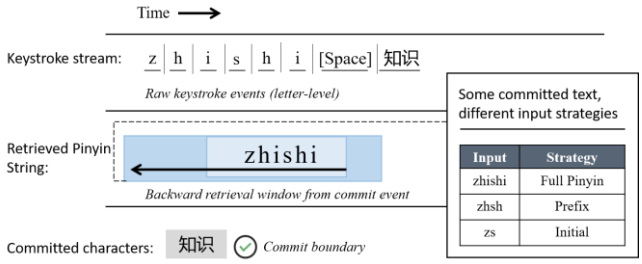


Figure 2. Illustration of the Pinyin–Character Alignment Procedure Based on Commit Boundaries

Based on the selected alignment, we identify and count the use of different Pinyin input strategies (e.g., full Pinyin, initial-letter abbreviation, and prefix-based abbreviations, see Table 1). To control for differences in task duration and text length, strategy usage, pause frequency, and related indicators are normalized by the total number of committed characters, yielding proportion-based measures.

Table 1: Illustration of Pinyin Input Strategies

Strategy Type	Target Characters	Input Letters	Description
Full Pinyin	知识	zhishi	All letters of the full Pinyin spelling for each syllable are entered.
Prefix-based abbreviation	知识	zhsh	A reduced prefix form is entered, keeping multiple leading letters from each syllable.
Initial-letter abbreviation	知识	zs	Only the first letter of each Pinyin syllable is entered.

The alignment model successfully resolved 87.75% of all commit events (alignment coverage rate), yielding a mean of 2.35 committed characters per commit event. The remaining 12.25% of events, in which no valid Pinyin string could be retrieved within the backward-search window or in which the constraint-based filtering could not uniquely resolve the alignment, were excluded from strategy-level analyses but retained for timing-based analyses (e.g.,

inter-key intervals and pause rates), which do not require Pinyin segmentation.

Among successfully aligned commits, the strategy classification yielded the following aggregate distribution: full Pinyin (M = 77.69%), initial-letter abbreviation (M = 12.86%), prefix-based abbreviation (M = 7.83%), and grouped input with backspace correction (M = 1.62%). The last category comprised events in which the Pinyin buffer was interrupted by mid-buffer deletions prior to character commitment; because such sequences could not be unambiguously attributed to a single input strategy, they were excluded from subsequent strategy-level comparisons. No events fell into the residual "other" category (M = 0.00%), indicating that the constraint-based filtering algorithm exhaustively accounted for all strategy types within the aligned portion of the data.

Based on their location in the input stream, we categorized pauses into three types. (1) Within-character pauses are long intervals occurring during Pinyin letter entry and may reflect difficulty in phonological encoding or momentary shifts of attention. (2) Selection pauses are long intervals between completing Pinyin entry and selecting a candidate character, capturing cognitive load associated with orthographic recognition and decision making. (3) Post-commit pauses are long intervals after a character is committed and before the next input event, which may signal planning at the lexical, syntactic, or discourse level. For all three pause types, occurrence rates were normalized by the total number of committed characters (i.e., computed per character).

Beyond backspace rate (immediate deletions), we also computed a broader edit rate, defined as the proportion of all input operations that involved revision actions in the middle of the text (e.g., deletions and substitutions). This measure was used to capture monitoring and revision activity during writing.

For time-based measures (e.g., inter-key interval, selection latency, and post-commit gap), we report both the mean and the median to better represent their typically skewed distributions. For inter-key interval, we additionally computed the standard deviation and the coefficient of variation ($CV = \frac{\sigma}{\mu}$) to quantify the stability and relative variability of input rhythm.

3.4 Comparative Analysis

After extracting Pinyin input process features, we compared the two writing tasks on each metric. All analyses were conducted within participants to control for stable individual differences.

For each metric, we first tested the normality of the paired difference scores using the Shapiro-Wilk test. If the normality test was not significant ($p > .05$), we used a paired-samples t test to compare task conditions, otherwise Wilcoxon signed-rank test was applied. We controlled the inflation of Type I error due to multiple comparisons using the Benjamini-Hochberg (BH) procedure. Statistical significance was determined based on BH-adjusted *p* values.

3.5 Correlating Analysis

The writing quality of Task 2 was assessed using Nussbaum's holistic argumentation rubric, with scores ranging from 0 to 7. All essays were scored independently by two trained raters. Inter-rater reliability was evaluated using Kendall's tau (τ) and linearly weighted Cohen's kappa (κ). Agreement was acceptable to high ($\tau = 0.840$, $\kappa = 0.671$), indicating that the scoring procedure was reliable. The holistic argumentation scores derived from Task 2 yielded a mean of 4.51 (SD = 1.31, range = 1.50–6.50). For subsequent analyses, we used the mean of the two raters' scores.

To examine the predictive value of Pinyin input features for writing quality, we related 14 key process metrics extracted from Task 2 to the final argumentation scores. A Shapiro-Wilk test indicated that the score distribution did not significantly deviate from normality ($W = 0.953$, $p = 0.151$), supporting the use of Pearson correlation analysis.

4. RESULTS

4.1 Comparative Analysis of Pinyin Input Behaviors Across Tasks

The study first compared the characteristics of participants' Pinyin input behaviors between the transcription task (Task 1) and the argumentative writing task (Task 2). As shown in Table 2, the shift in task type exerted a significant impact on participants' input rhythm, cognitive processing latency, and input strategies, providing preliminary support for Hypothesis 1 (H1).

Table 2: Task Comparison Results for Pinyin Input Features

Metric	Test Type	p_{adj}	*d*	*r*	z
Full Input Ratio	T-test	0.214	-0.233		
Initial Input Ratio	T-test	0.077	-0.334		
Prefix Input Ratio	Wilcoxon	0.030		0.459	-2.676
Mean IKI	Wilcoxon	<0.001		0.872	-5.086
SD of IKI	Wilcoxon	<0.001		0.872	-5.086
CV of IKI	T-test	<0.001	1.577		
Backspace Rate	Wilcoxon	0.027		0.682	-3.975
Mean Selection Latency	T-test	<0.001	-1.252		
Median Selection Latency	T-test	<0.001	-1.477		
Median Post-commitment Gap	Wilcoxon	<0.001		0.849	-4.949
Global Edit Rate	Wilcoxon	0.030		0.394	-2.299
Within-character Pause Rate	Wilcoxon	0.454		0.216	-1.257
Selection Pause Rate	T-test	<0.001	-0.798		
Post-commit Pause Rate	T-test	0.370	0.165		

At the physical keystroke level, a systemic decline in input fluency was observed when participants transitioned from simple character transcription to complex argumentative writing. Statistical results indicated that the mean Inter-Key Interval (Mean IKI) and its standard deviation (SD of IKI) were significantly higher in Task 2 than in Task 1 ($Z = -5.086$, $p_{adj} < .001$, $r = .872$). More importantly, the Coefficient of Variation of IKI (CV of IKI), which measures the relative stability of input rhythm, increased drastically in the writing task ($p_{adj} < .001$, Cohen's $d = 1.577$).

Changes in cognitive-layer metrics further revealed the cognitive load imposed by the writing task. During the candidate character recognition phase, both the mean and median Selection Latency were significantly longer in the writing task compared to the transcription task ($p_{adj} < .001$). Simultaneously, the Selection Pause Rate increased significantly ($p_{adj} < .001$). Furthermore, the Median Post-commit Gap was significantly extended in Task 2 ($Z = -4.949$, $p_{adj} < .001$, $r = .849$).

Regarding input strategies and monitoring, participants exhibited distinct compensatory and regulatory tendencies. While the ratios of full-pinyin and initial-pinyin strategies showed no significant changes, the Prefix Input Strategy Ratio rose significantly in the writing task ($Z = -2.676$, $p_{adj} = .030$, $r = .459$). Meanwhile, both the Backspace Rate ($p_{adj} = .027$) and the Global Edit Rate ($p_{adj} = .030$) were significantly higher in Task 2 than in Task 1. This indicates that the writing task triggered more frequent evaluation and revision activities\.

4.2 Correlation Between Pinyin Input Features and Argumentative Writing Quality

To address RQ3, we conducted Pearson correlation analyses between the 14 key process features extracted from the creative writing task (Task 2) and the holistic argumentation scores. The results revealed that cognitive-layer pause metrics were the most robust predictors of writing quality, while physical-layer fluency and input strategies showed no significant linear relationship with the final product's quality (see Table 3).

Table 3: Pearson Correlations Between Pinyin Input Features and Argumentation Scores (p < .01)**

Metrics	*r*	*p*	95%CL
Full Input Ratio	-0.127	0.480	[-0.451, 0.226]
Initial Input Ratio	0.249	0.162	[-0.103, 0.546]
Prefix Input Ratio	-0.204	0.255	[-0.511, 0.150]
Mean IKI	-0.123	0.496	[-0.447, 0.230]
SD of IKI	-0.058	0.750	[-0.393, 0.291]
CV of IKI	0.016	0.928	[-0.329, 0.358]
Backspace Rate	0.036	0.844	[-0.311, 0.374]
Mean Selection Latency	-0.036	0.842	[-0.375, 0.311]
Median Selection Latency	-0.114	0.528	[-0.440, 0.239]
Mean Post-commit Gap	0.018	0.922	[-0.328, 0.359]
Global Edit Rate	0.175	0.330	[-0.179, 0.489]
Within-character Pause Rate	-0.482**	0.005	[-0.708, -0.166]
Selection Pause Rate	-0.030	0.870	[-0.369, 0.317]
Post-commit Pause Rate	-0.465**	0.006	[-0.697, -0.145]

Statistical analysis identified two critical cognitive-layer indicators that were significantly and negatively correlated with argumentation scores. Specifically, the Within-character Pause Rate ($r = -.482$, $p = .005$, 95% CI [−.708, −.166]) and the Post-commit Pause Rate ($r = -.465$, $p = .006$, 95% CI [−.697, −.145]) showed moderate negative correlations with writing quality.

Notably, physical-layer metrics such as Mean IKI ($r = -.123$, $p = .496$, 95% CI [-0.447, 0.230]) and Median Selection Latency ($r = -.114$, $p = .528$, 95% CI [-0.440, 0.239]) failed to reach statistical significance. This decoupling indicates that the mere speed of finger movement or character selection does not inherently translate to superior intellectual output in Chinese writing. Furthermore, input strategy ratios, such as the Initial Input Ratio ($r = .249$, $p = .162$, 95% CI [-0.103, 0.546]) and Prefix Input Ratio ($r = -.204$, $p = .255$, 95% CI [-0.511, 0.150]), were also not significantly associated with scores.

Table 4 provides a compact summary of hypothesis testing results. H1a and H1b were fully supported, showing greater timing variability and longer cognitive processing intervals in writing than copying. H1c received partial support. Revision-related behaviors (backspace rate and global edit rate) were significantly higher in the writing task, while only one of the strategic input indicators (prefix-based input ratio) showed a significant increase. Other strategy ratios did not differ reliably across tasks. H2 was supported: only cognitively grounded pause indicators were significantly associated with writing quality, whereas low-level motor timing metrics were not.

Table 4: Compact summary of hypothesis testing results. Detailed statistics are reported in Tables 2–3.

	Indicators	Key evidence	Verdict
H1a	CV of IKI	$p_{adj} < .001$, d = 1.577	Supported
H1b	Selection latency	$p_{adj} < .001$, d = -1.252	Supported
H1b	Post-commit gap	$p_{adj} < .001$, r = 0.849	Supported
H1c	Backspace rate	$p_{adj} = .027$, r = 0.682	Supported
H1c	Global edit rate	$p_{adj} = .030$, r = 0.394	Supported
H1c	Full Input Ratio	$p_{adj} = .214$, d = -0.233	Not Supported
H1c	Initial Input Ratio	$p_{adj} = .077$, d = -0.334	Not Supported
H1c	Prefix Input Ratio	$p_{adj} = .030$, r = 0.459	Supported
H2	Within-character pause rate vs. Score	r = −.48, p < .01	Supported
H2	Post-commit pause rate vs. Score	r = −.47, p < .01	Supported

5. DISCUSSION

The present study investigated the characteristics of Chinese Pinyin input behaviors and their relationship with writing quality through the lens of a Dual-layer Rhythm Model. Utilizing fine grained keystroke logging and feature engineering techniques, we captured the nuanced shifts in input patterns as participants transitioned from transcription to argumentative writing. The results confirmed that task induced cognitive load significantly alters physical rhythm, cognitive latency, and input strategies. Most crucially, the correlation analysis revealed that cognitive layer pauses metrics, specifically the Within-character Pause Rate and the Post commit Pause Rate, serve as robust predictors of argumentation quality, whereas physical fluency metrics such as Mean Inter Key Interval do not.

A central finding of this study is the significant negative correlation between the Within-character Pause Rate, the Post-commit Pause Rate, and holistic writing scores. From the perspective of feature engineering, these two metrics constitute diagnostic markers for identifying cognitive bottlenecks during the writing process.

The Within-character Pause Rate reflects difficulties during the phonological encoding phase. In Pinyin-mediated writing, the writer must transform abstract mental concepts into specific phonetic sequences. High quality writers exhibit a high degree of automaticity in this phase. Conversely, frequent prolonged pauses during Pinyin spelling among lower scoring writers suggest that lower-level phonology to orthography conversion consumes excessive cognitive resources. Crucially, this relationship may be bidirectional: the intensive cognitive load inherent in high-level argumentative planning can interfere with the execution of motor

sequences, causing even potentially automated Pinyin input to stutter. According to Working Memory Theory, such non automated lower-level processing creates cognitive competition, thereby depleting the mental resources available for high level argumentative planning and reasoning.

The Post-commit Pause Rate marks the transition from text generation to higher-level macro-planning. Higher-quality writers typically demonstrate stronger pre-planning capacity or more fluent syntactic production, resulting in fewer disruptions between linguistic units. This finding suggests that typing speed alone is not a reliable indicator of writing quality. Consistent with writing-process and keystroke-logging research, cognitively grounded timing features—such as pause distribution—provide more informative signals of composing quality than raw physical execution speed, especially in Pinyin-mediated Chinese writing.

The observed decoupling between physical layer metrics such as Mean Inter Key Interval and writing scores provides significant insights for educational data mining. This phenomenon suggests that in a population of graduate students, physical execution of Pinyin input has reached a stage of high automaticity, essentially becoming background noise in the writing process.

The true variance in writing expertise is concealed within the decision latencies and pause patterns that occur after the physical keystroke sequences. This validates the necessity of our proposed DLRM. Only by isolating the interference of physical execution can researchers accurately capture the cognitive features truly associated with the quality of thought and argumentation.

From the standpoint of Educational Data Mining, the identification of these two critical pause features provides an empirical foundation for developing process oriented intelligent writing systems. Future automated writing evaluation tools could move beyond product-based scoring to provide real time diagnostics. By monitoring the Within-character Pause Rate and the Post commit Pause Rate, a system could distinguish whether a writer is struggling with spelling retrieval or conceptual planning and offer personalized scaffolding accordingly.

Despite these insights, several limitations should be noted. First, the sample size of 34 and the focus on graduate students limit the generalizability of the findings to younger or less proficient populations. Future research should include diverse groups, such as elementary students or second language learners, to test the model's robustness.

Second, this study primarily focused on the incremental generation phase. Macro level editing behaviors, such as structural reorganization or cross paragraph revisions, were not explored in depth. Future studies could integrate eye tracking technology with sequential pattern mining to investigate the recursive nature of the writing process, thereby constructing a more comprehensive feature library for Chinese digital writing.

6. CONCLUSION

This study proposed and validated a Dual-layer Rhythm Model to analyze the cognitive mechanisms underlying Chinese Pinyin input during writing tasks. By leveraging fine grained keystroke logging and data mining techniques, we successfully decoupled the physical execution layer from the cognitive processing layer in digital writing. The results demonstrate that while high level writing tasks induce a systemic decline in physical typing fluency, it is the cognitive layer pauses, specifically those occurring within Pinyin character construction and after character commitment, that serve as significant indicators of writing quality.

The theoretical contribution of this research lies in identifying the unique phonological to orthographical translation costs inherent in Pinyin input, which have been largely overlooked in traditional alphabetic keystroke research. Practically, the identified diagnostic features provide a foundation for developing next generation intelligent writing environments. These systems can transition from static text evaluation to dynamic process monitoring, offering timely interventions when cognitive bottlenecks are detected. Future work will focus on integrating these process features into machine learning models to enhance the predictive accuracy of automated writing assessment systems across diverse learner populations.

7. APPENDIX

7.1 Appendix A. Transcription Text

假如说逻辑一般被认为是思维的科学，那么，人们对于它的了解是这样的，即：好像这种思维只构成知识的单纯形式；好像逻辑抽去了一切内容，而属于知识的所谓第二组成部分，即质料，必定另有来源；好像完全不为这种质料所依赖的逻辑，因而只能提供真正知识的形式条件，而不能包含实在的真实本身，也不能是达到实在的真理的途径，因为真理的本质的东西，内容，恰恰在逻辑以外。

7.2 Appendix B. Argument Prompt

The following is a memorandum from the business manager of a television station. "Over the past year, our late-night news program has devoted increased time to national news and less time to weather and local news. During this period, most of the complaints received from viewers were concerned with our station's coverage of weather and local news. In addition, local businesses that used to advertise during our late-night news program have canceled their advertising contracts with us. Therefore, in order to attract more viewers to our news programs and to avoid losing any further advertising revenues, we should expand our coverage of weather and local news on all our news programs."

Write a response in which you examine the stated and/or unstated assumptions of the argument. Be sure to explain how the argument depends on these assumptions and what the implications are for the argument if the assumptions prove unwarranted.

8. ETHICS AND CONSENT TO PARTICIPATE

Formal ethical approval was not required for this study in accordance with national and institutional guidelines, as determined by the Institutional Review Board of the University of Science and Technology of China. The study involved anonymous data collection and posed no foreseeable risk to adult participants. Participation was voluntary, and informed consent was obtained from all participants prior to data collection. If required, documentation confirming the exemption from IRB review can be provided.

9. ACKNOWLEDGMENTS

This research was also supported by the advanced computing resources provided by the Supercomputing Center of the USTC.

10. REFERENCES

[1] Anderson, J.R. 1982. Acquisition of cognitive skill. *Psychological review*. 89, 4 (1982), 369. https://doi.org/10.1037/0033-295X.89.4.369.

[2] Anderson, J.R. 2014. *Rules of the mind*. Psychology Press.

[3] Baddeley, A. 2003. Working memory: looking back and looking forward. *Nature reviews neuroscience*. 4, 10 (2003), 829–839. https://doi.org/10.1038/nrn1201.

[4] Baddeley, A.D. 2017. Modularity, working memory and language acquisition. *Second language research*. 33, 3 (2017), 299–311. https://doi.org/10.1177/0267658317709852.

[5] Baddeley, A.D. 2015. Working memory in second language learning. *Working memory in second language acquisition and processing*. (2015), 17–28.

[6] Chen, J., Luo, R. and Liu, H. 2017. The Effect of Pinyin Input Experience on the Link Between Semantic and Phonology of Chinese Character in Digital Writing. *Journal of Psycholinguistic Research*. 46, 4 (Aug. 2017), 923–934. https://doi.org/10.1007/s10936-016-9470-y.

[7] Chenoweth, N.A. and Hayes, J.R. 2001. Fluency in Writing: Generating Text in L1 and L2. *Written Communication*. 18, 1 (Jan. 2001), 80–98. https://doi.org/10.1177/0741088301018001004.

[8] Conijn, R., Cook, C., van Zaanen, M. and Van Waes, L. 2022. Early prediction of writing quality using keystroke logging. *International Journal of Artificial Intelligence in Education*. 32, 4 (2022), 835–866. https://doi.org/10.1007/s40593-021-00268-w.

[9] Conijn, R., Roeser, J. and Van Zaanen, M. 2019. Understanding the keystroke log: The effect of writing task on keystroke features. *Reading and Writing*. 32, 9 (2019), 2353–2374. https://doi.org/10.1007/s11145-019-09953-8.

[10] Correnti, R., Matsumura, L.C., Wang, E.L., Litman, D. and Zhang, H. 2022. Building a validity argument for an automated writing evaluation system (eRevise) as a formative assessment. *Computers and Education Open*. 3, (2022), 100084. https://doi.org/10.1016/j.caeo.2022.100084.

[11] Crotti, R., Denaro, G., Du, Z. and MartÃn, R.M. 2026. Hylog: A Hybrid Approach to Logging Text Production in Non-alphabetic Scripts. *arXiv preprint arXiv:2601.17753*. (2026).

[12] Diamond, A. 2013. Executive functions. *Annual review of psychology*. 64, 1 (2013), 135–168. https://doi.org/10.1146/annurev-psych-113011-143750.

[13] Ericsson, K.A. and Simon, H.A. 1980. Verbal reports as data. *Psychological review*. 87, 3 (1980), 215. https://doi.org/10.1037/0033-295X.87.3.215.

[14] Guan, C.Q., Liu, Y., Chan, D.H.L., Ye, F. and Perfetti, C.A. 2011. Writing strengthens orthography and alphabetic-coding strengthens phonology in learning to read Chinese. *Journal of Educational Psychology*. 103, 3 (Aug. 2011), 509–522. https://doi.org/10.1037/a0023730.

[15] Ho, C.S.-H. and Bryant, P. 1997. Learning to read Chinese beyond the logographic phase. *Reading research quarterly*. 32, 3 (1997), 276–289. https://doi.org/10.1598/RRQ.32.3.3.

[16] Huang, C.-R., Lin, Y.-H., Chen, I.-H. and Hsu, Y.-Y. 2022. *The Cambridge handbook of Chinese linguistics*. Cambridge University Press.

[17] Kandel, S. and Perret, C. 2015. How do movements to produce letters become automatic during writing acquisition? Investigating the development of motor anticipation. *International Journal of Behavioral Development*. 39, 2 (2015), 113–120. https://doi.org/10.1177/0165025414557532.

[18] Kellogg, R.T. 1996. A MODEL OF WORKING MEMORY IN WRITING. *The science of writing: Theories, methods, individual differences, and applications*. Lawrence Erlbaum Associates, Inc. 57–71.

[19] Kellogg, R.T. 2008. Training writing skills: A cognitive developmental perspective. *Journal of Writing Research*. 1, 1 (June 2008), 1–26. https://doi.org/10.17239/jowr-2008.01.01.1.

[20] Leijten, M. and Van Waes, L. 2013. Keystroke Logging in Writing Research: Using Inputlog to Analyze and Visualize Writing Processes. *Written Communication*. 30, 3 (July 2013), 358–392. https://doi.org/10.1177/0741088313491692.

[21] Li, T., Yu, L., Zhou, L. and Wang, P. 2023. Using less keystrokes to achieve high top-1 accuracy in Chinese clinical text entry. *Digital Health*. 9, (2023), 20552076231179027. https://doi.org/10.1177/20552076231179027.

[22] Logan, G.D. 1988. Toward an instance theory of automatization. *Psychological review*. 95, 4 (1988), 492. https://doi.org/10.1037/0033-295X.95.4.492.

[23] Lu, X. and Révész, A. 2021. Revising in a non-alphabetic language: The multi-dimensional and dynamic nature of online revisions in Chinese as a second language. *System*. 100, (Aug. 2021), 102544. https://doi.org/10.1016/j.system.2021.102544.

[24] Lyu, B., Lai, C., Lin, C.-H. and Gong, Y. 2021. Comparison studies of typing and handwriting in Chinese language learning: A synthetic review. *International Journal of Educational Research*. 106, (2021), 101740. https://doi.org/10.1016/j.ijer.2021.101740.

[25] McCutchen, D. 2000. Knowledge, Processing, and Working Memory: Implications for a Theory of Writing. *Educational Psychologist*. 35, 1 (Jan. 2000), 13–23. https://doi.org/10.1207/S15326985EP3501_3.

[26] Oxley, J. and Ma, Y. 2020. Considerations for Chinese text input methods in the design of speech generating devices: a tutorial. *Clinical Linguistics & Phonetics*. 34, 4 (2020), 366–387. https://doi.org/10.1080/02699206.2019.1652934.

[27] Perfetti, C.A., Liu, Y. and Tan, L.H. 2005. The lexical constituency model: some implications of research on Chinese for general theories of reading. *Psychological review*. 112, 1 (2005), 43. https://doi.org/10.1037/0033-295X.112.1.43.

[28] Perfetti, C.A. and Tan, L.H. 1998. The time course of graphic, phonological, and semantic activation in Chinese character identification. *Journal of Experimental Psychology: Learning, Memory, and Cognition*. 24, 1 (1998), 101. https://doi.org/10.1037/0278-7393.24.1.101.

[29] Shen, H.H. and Bear, D.R. 2000. Development of orthographic skills in Chinese children. *Reading and Writing*. 13, 3 (2000), 197–236. https://doi.org/10.1023/A:1026484207650.

[30] Sun, X. and Huo, X. 2022. Word-Level and Pinyin-Level Based Chinese Short Text Classification. *IEEE Access*. 10, (2022), 125552–125563. https://doi.org/10.1109/ACCESS.2022.3225659.

[31] Sweller, J. 2011. Cognitive load theory. *Psychology of learning and motivation*. Elsevier. 37–76.

[32] Talebinamvar, M. 2022. Clustering students' writing behaviors using keystroke logging: a learning analytic approach in EFL writing. *Language Testing in Asia*. 12, 1 (2022), 6.

[33] Van Waes, L. and Leijten, M. 2015. Fluency in Writing: A Multidimensional Perspective on Writing Fluency Applied to L1 and L2. *Computers and Composition*. 38, (Dec. 2015), 79–95. https://doi.org/10.1016/j.compcom.2015.09.012.

[34] Wang, J., Zhai, S. and Su, H. 2001. Chinese input with keyboard and eye-tracking: an anatomical study. *Proceedings of the SIGCHI conference on Human factors in computing systems* (2001), 349–356.

[35] Wang, M., Koda, K. and Perfetti, C.A. 2003. Alphabetic and nonalphabetic L1 effects in English word identification: A comparison of Korean and Chinese English L2 learners. *Cognition*. 87, 2 (2003), 129–149.

[36] Wang, X. and Wang, J. 2024. Comparing Chinese L2 writing performance in paper-based and computer-based modes: Perspectives from the writing product and process. *Assessing Writing*. 61, (2024), 100849. https://doi.org/10.1016/j.asw.2024.100849.

[37] Zhai, S. and Kristensson, P.O. 2012. The word-gesture keyboard: reimagining keyboard interaction. *Communications of the ACM*. 55, 9 (2012), 91–101. https://doi.org/10.1145/2330667.2330689.

[38] Zhou, X. and Marslen-Wilson, W. 1999. Phonology, orthography, and semantic activation in reading Chinese. *Journal of Memory and Language*. 41, 4 (1999), 579–606. https://doi.org/10.1006/jmla.1999.2663.

Early Prediction of Student Dropout in a Large-Scale Cram School Using Longitudinal Test Performance Data

Masanori Yamada
Kyushu University
mark@mark-lab.net

Kouki Nakao
Eishinkan, Co. Ltd.
hi-nakao@eishinkan.co.jp

Masaki Miyazono
Eishinkan, Co. Ltd.
m-miyazono@eishinkan.co.jp

Xuewang Geng
Sojo University
geng@mark-lab.net

Hiroyuki Iwashita
Eishinkan, Co. Ltd.
hr-iwashita@eishinkan.co.jp

Hiroshi Kamio
Eishinkan, Co. Ltd.
h-kamio@eishinkan.co.jp

Ryosuke Oogushi
Eishinkan, Co. Ltd.
r-oogushi@eishinkan.co.jp

Koji Takano
Eishinkan, Co. Ltd.
k-takano@eishinkan.co.jp

Toshihide Tsutsui
Eishinkan, Co. Ltd.
t-tutui@eishinkan.co.jp

Yuji Hirata
Eishinkan, Co. Ltd.
y-hirata@eishinkan.co.jp

ABSTRACT

Student dropout is a major challenge in large-scale supplementary education, where early identification of at-risk learners is essential. Although prior educational data mining studies have explored dropout prediction using learning logs, long-term test-based data common in offline settings remain underexplored. This industry paper reports a real-world application of dropout prediction using longitudinal standardized test data from over 3,000 junior high school students in a large Japanese cram school. We engineered interpretable features capturing achievement dynamics, including recovery efficiency after score declines and trend consistency. Random Forest models trained on a previous cohort were validated on an independent cohort. The best-performing model achieved approximately 89% accuracy and a dropout F1-score of 0.80, enabling prediction before the final academic year. Feature importance analysis showed that recovery from performance declines was a stronger indicator of dropout risk than absolute achievement. These results demonstrate that longitudinal test data can support accurate and interpretable dropout prediction at scale, offering practical value for early intervention.

Keywords

Dropout prediction, Early warning, Longitudinal assessment, Cram school

1. INTRODUCTION

Student dropout is a persistent challenge in large-scale supplementary education, where learners and families invest substantial time and financial resources over multiple years. For providers, dropout not only disrupts learning continuity but also limits the effectiveness of long-term curricular planning and student support. Accordingly, there is growing interest in early-warning systems that can identify at-risk learners early enough to enable timely, targeted interventions rather than reactive responses after disengagement becomes irreversible [1][3]. In this study, a cram school refers to a private supplementary education institution that provides after-school academic instruction, primarily aimed at supporting students' preparation for high school entrance examinations.

Educational Data Mining (EDM) and Learning Analytics (LA) research has produced a broad body of work on dropout prediction using machine learning, often leveraging digital trace data such as LMS activity logs, clickstream behavior, and online engagement metrics (e.g., [1][8]).

In many offline or hybrid educational settings, however, these fine-grained behavioral logs are limited or unavailable. Instead, institutions frequently maintain longitudinal assessment records—standardized tests or periodic mock exams—collected consistently over multiple years. While dropout prediction has also been studied with administrative and academic records (e.g., transcripts, credits, grades), evidence remains comparatively thinner regarding how to transform repeated test performance trajectories into actionable early warning signals in real-world, offline-dominant contexts[8].

While various ensemble learning techniques, such as Random Forests [2] and Gradient Boosting Decision Trees like XGBoost [4] and GBM [6], have demonstrated high predictive accuracy in these domains, their application to offline longitudinal records requires careful adaptation. Specifically, beyond achieving high accuracy, it is critical to ensure model interpretability—a growing focus in recent EDM research [7]—so that predictions can directly inform pedagogical interventions. In this study, we employ a Random Forest approach [2] not only for its robust performance but also to validate the effectiveness of our interpretable feature engineering framework.

This gap is particularly relevant for supplementary education providers (e.g., cram schools), where periodic test-based evaluation is central to instructional decision-making. Prior EDM work has shown the potential of predictive modeling in Japanese cram school contexts—for example, using teacher observation reports and exam scores for performance prediction[5]. Building on this line of industry-relevant EDM, we focus specifically on dropout prediction using data that many supplementary education providers already possess at scale: multi-year standardized test records.

Masanori Yamada, Xuewang Geng, Ryosuke Oogushi, Kouki Nakao, Hiroyuki Iwashita, Koji Takano, Yuji Hirata, Masaki Miyazono, Hiroshi Kamio, and Toshihide Tsutsui. Early Prediction of Student Dropout in a Large-Scale Cram School Using Longitudinal Test Performance Data. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 917–922. International Educational Data Mining Society (2026).

© 2026 Copyright is held by the author(s). This work is distributed under the Creative Commons Attribution NonCommercial NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.

https://doi.org/10.5281/zenodo.21039984

In this industry paper, we investigate how longitudinal standardized test data can be utilized to support early identification of students at risk of dropout in large-scale supplementary education. We focus on a real-world setting in which repeated test-based assessments are routinely collected but fine-grained digital learning logs are limited or unavailable. To address this context, we design a feature engineering framework that captures students' achievement dynamics over time, including responses to performance declines, stability of learning trajectories, and distributional patterns of subject performance. We then examine how such interpretable temporal indicators can be integrated into machine learning models for practical early-warning applications. Through cohort-based model development and independent cohort validation, this study aims to evaluate the feasibility, generalizability, and operational relevance of dropout prediction using offline longitudinal assessment records. Ultimately, we seek to clarify whether existing assessment infrastructures in supplementary education can be adapted into data-driven early-warning systems that support timely and targeted student interventions.

2. METHODS

2.1 Context, Data, and Outcome Definition

This study was conducted in collaboration with a large-scale Japanese cram school that prepares students for high school entrance examinations. We analyzed longitudinal standardized test records collected routinely as part of instructional practice.

To ensure consistent longitudinal trajectories, we restricted the cohort to students who participated in the first Grade 7 diagnostic test (N = 3,316). Students subsequently took up to 15 standardized diagnostic and mock examinations from Grade 7 to Grade 9. Participation decreased over time, with 2,200 students remaining at the final Grade 9 test, corresponding to a 33.6% reduction from the initial cohort. An overview of the dataset, observation period, and evaluation design is summarized in Table 1.

Table 1. Overview of the Dataset

Item	Description
Educational setting	Large-scale Japanese cram school preparing students for high school entrance examinations
Initial cohort	3,316 students who participated in the first Grade 7 diagnostic test
Observation period	Grade 7 (Year 1) to Grade 9 (Year 3)
Number of test occasions	Up to 15 standardized diagnostic and mock examinations
Subjects analyzed	Native language, Mathematics, English, overall average
Label definition	Graduates: students with a graduation flag; Dropouts: test participants without the flag
Final test participants	2,200 students at the final Grade 9 test (33.6% decrease from initial cohort)
Minimum tests required	3 test participations (for feature computation)
Evaluation sample size	~3,100 students (varies by model due to minimum test requirement)
Validation design	Train on 2024 cohort, validate on independent 2025 cohort

Students were labeled as *graduates* if they were flagged as having completed enrollment through the final examination. Test participants without this flag were labeled as *dropouts*. In addition, for each test occasion, we recorded whether that test became the student's final attempt, enabling analysis of withdrawal timing. We utilized anonymized data provided by the cram school. The use of this data underwent internal review and approval by the partnering cram school and the institution and was handled in strict compliance with relevant privacy guidelines.

2.2 Assessment Timeline and Subjects

The assessment sequence consisted of four Grade 7 diagnostic tests, transitional mock examinations at grade boundaries (e.g., "new Grade 8" and "new Grade 9"), and multiple Grade 8–9 high school entrance mock tests. Each test included subject-level scores and standardized deviation scores for Native language, Mathematics, and English, as well as overall averages.

This repeated assessment structure enabled us to model not only performance levels but also longitudinal achievement dynamics across critical transitional periods in lower secondary education.

2.3 Feature Engineering

Rather than relying solely on raw test scores, we engineered interpretable features designed to capture students' longitudinal achievement dynamics. Feature design was conducted in close consultation with practitioners to ensure pedagogical relevance and operational usability in real educational settings.

In total, 98 features were engineered from subject-wise scores and standardized deviation scores across test occasions. These features fall into several conceptual categories, summarized with representative examples in Table 2.

Key feature categories include:

1). Recovery-related indicators, measuring how efficiently a student's performance rebounds after a decline (e.g., score recovery efficiency).
2). Trend-based indicators, capturing the direction, slope, and instability of longitudinal change (e.g., linear regression slopes and directional change ratios).
3). Consistency indicators, representing the proportion of changes occurring in the same direction over time.
4). Distributional indicators, such as proportions of high scores (>80), passing scores (>60), and low scores (<40).
5). Relative strength indicators, quantifying subject-wise performance relative to a student's overall average.

For example, recovery efficiency quantifies how effectively a student returns to their individual mean performance following a score decrease, while directional change ratio reflects instability in achievement trajectories caused by frequent switches between improvement and decline.

Table 2. Categories of Engineered Features and Representative Examples

Category	Description	Representative examples
Recovery-related	Ability to rebound after performance decline	Score recovery efficiency (Native language, Mathematics)
Trend-based	Direction and stability of longitudinal change	Linear slope, directional change ratio

Category	Description	Representative examples
Consistency	Stability of improvement or decline	Trend consistency ratio
Distributional	Performance level distribution	High-score ratio (>80), low-score ratio (<40)
Relative strength	Subject-wise relative performance	Subject mean / overall mean

2.4 Modeling Approach

We formulated dropout prediction as a supervised binary classification task. Specifically, we employed the Random Forest algorithm [2], which is widely used in educational data mining for its robustness to overfitting and interpretability. The Random Forest model was implemented using standard hyperparameter settings, including 100 decision trees (n_estimators = 100). Hyperparameters were selected through grid search on the training cohort to optimize predictive performance while avoiding overfitting. To examine the trade-off between prediction timing and performance, we constructed three models that differed in the length of available longitudinal data:

Model 1: Grade 7 Test 1 to Grade 9 Test 6

Model 2: Grade 7 Test 1 to Grade 9 Test 4

Model 3: Grade 7 Test 1 to Grade 9 Test 3

These models correspond to different potential intervention points, with earlier models providing more time for support but less complete information.

Models were trained using the 2024 graduating cohort and evaluated using accuracy and F1-scores for both graduates and dropouts. Because dropout detection is of primary operational importance and the classes are imbalanced, particular emphasis was placed on the dropout F1-score when comparing models.

2.5 Validation Procedure and Evaluation Metrics

To assess generalizability, all models were validated on an independent 2025 graduating cohort. Because feature computation required a minimum of three test participations, the evaluation sample size varied slightly across models, as summarized in Table 1.

Performance was evaluated using accuracy, precision, recall, and F1-scores for both classes. For the selected model, we additionally report a confusion matrix to clarify precision–recall trade-offs relevant to operational deployment.

2.6 Model Selection Criterion

Model selection was based on three criteria:

1). predictive performance, particularly dropout F1-score;

2) prediction timing, enabling intervention before the final academic year; and

3) operational interpretability for practitioners.

Based on these criteria, Model 2 was selected as the most suitable for real-world deployment, achieving strong predictive performance while enabling sufficiently early identification of at-risk students.

3. RESULTS

3.1 Model Comparison and External Validation

Table 3 summarizes the performance of the three dropout prediction models evaluated on the independent 2025 cohort. All models achieved high overall accuracy, ranging from 88.2% to 89.5%, demonstrating that longitudinal test performance data can support reliable dropout prediction at scale.

Table 3. Performance Comparison of Dropout Prediction Models

Model	Data used	Accuracy	Graduate F1	Dropout F1
Model 1	Grade 7 Test 1 – Grade 9 Test 6	88.2%	0.91	0.81
Model 2	Grade 7 Test 1 – Grade 9 Test 4	89.5%	0.93	0.80
Model 3	Grade 7 Test 1 – Grade 9 Test 3	89.1%	0.93	0.78

Among the three models, Model 2, which used test data from Grade 7 Test 1 to Grade 9 Test 4, achieved the best balance between predictive performance and early identification. Specifically, Model 2 attained an accuracy of 89.5%, a graduate F1-score of 0.93, and a dropout F1-score of 0.80. While Model 1 achieved comparable performance, it relied on a longer observation period extending to the final Grade 9 test, limiting its usefulness for early intervention. In contrast, Model 3 enabled earlier prediction but exhibited a lower dropout F1-score, indicating reduced sensitivity for identifying at-risk students.

Figure 1 visualizes this trade-off between prediction timing and predictive performance across the three models. The results indicate that incorporating test data up to Grade 9 Test 4 is sufficient to achieve strong performance while still allowing institutions to intervene before the final academic year.

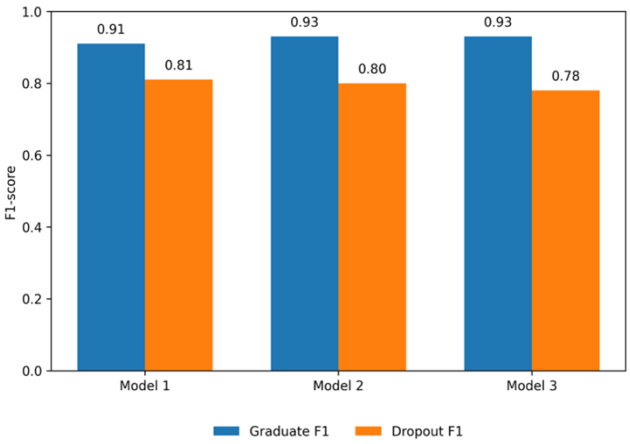


Figure 1. Model Comparison

3.2 Operational Performance of the Selected Model

To examine the practical implications of deployment, we further analyzed the operational performance of Model 2, focusing on precision–recall trade-offs relevant to real-world intervention. As shown in Table 3, Model 2 achieved a dropout precision of approximately 81% and a dropout recall of approximately 80%, indicating that the majority of students identified as high risk indeed withdrew, while a substantial proportion of actual dropouts were successfully detected.

From an operational perspective, this balance suggests that the model can be used to prioritize support resources toward students most likely to disengage, without overwhelming instructors with excessive false positives. Importantly, the model maintained high graduate recall, ensuring that students likely to remain enrolled were not systematically misclassified.

3.3 Feature Importance and Interpretability

To better understand why the models achieved strong predictive performance, we examined feature importance and group-level differences between graduates and dropouts. Across all models, recovery-related indicators consistently ranked among the most influential predictors.

In particular, score recovery efficiency in Native language and Mathematics emerged as the top-ranked features in Model 2. Graduates showed substantially higher recovery efficiency than dropouts, indicating that their performance was more likely to rebound after temporary declines. For example, the mean recovery efficiency in Native language was 0.84 for graduates compared to 0.60 for dropouts, with a similar pattern observed in Mathematics. Differences across the top-ranked features were statistically significant ($p < .01$, independent samples t-test). Assumptions of normality were considered acceptable given the large sample size.

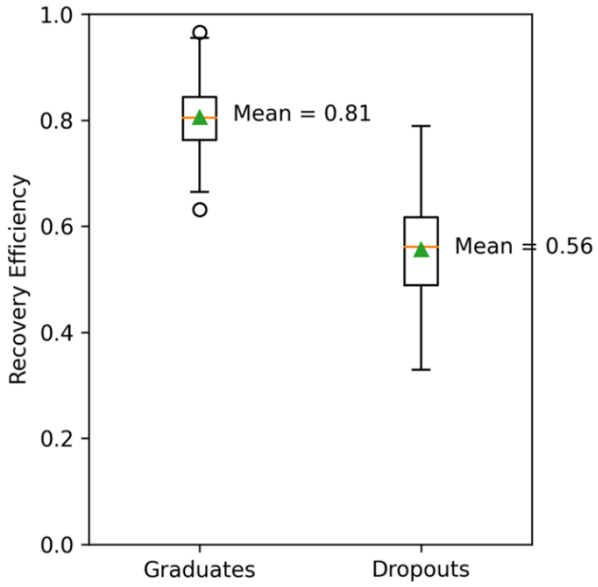


Figure 2. Mathematics test score recovery efficiency

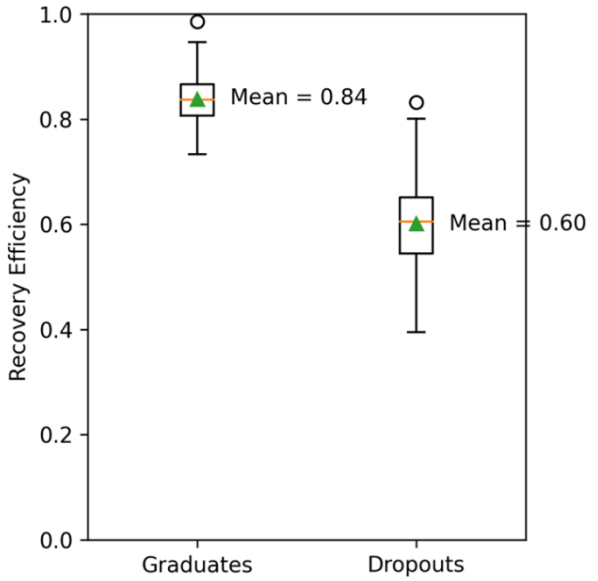


Figure 3. Native language test score recovery efficiency

Figures 2, 3, and 4 presents the distributions of score recovery efficiency for graduates and dropouts across three core subjects. In Mathematics (Figure 2), graduates showed higher recovery efficiency (Mean = 0.81) compared with dropouts (Mean = 0.56). Similarly, In Native language (Figure 3), graduates exhibited substantially higher recovery efficiency (Mean = 0.84) than dropouts (Mean = 0.60). The same pattern was observed in English(Figure 4), where graduates again demonstrated higher recovery efficiency than dropouts. These group differences indicate that the ability to recover after declines is strongly associated with continued enrollment.

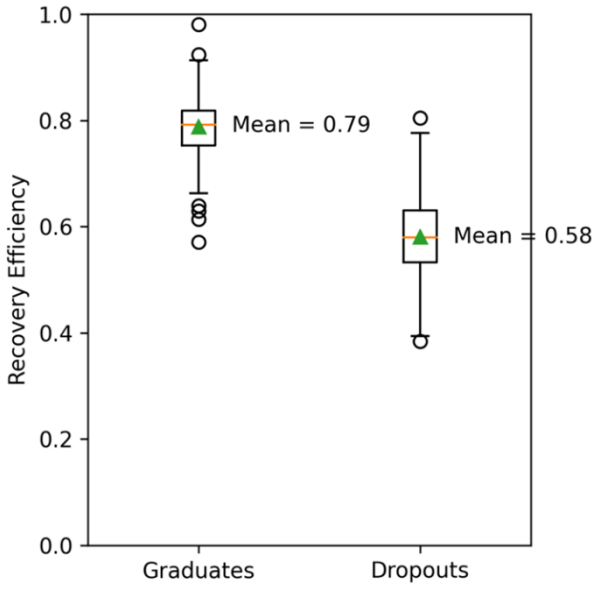


Figure 4. English test score recovery efficiency

3.4 Beyond Absolute Performance Levels

Notably, absolute achievement levels alone were less predictive than longitudinal performance dynamics. Several trend-based features, such as continuous negative deviation slopes and high directional change ratios, were also associated with increased dropout risk. These findings suggest that instability and sustained decline, rather than low scores per se, play a critical role in student disengagement. This emphasis on achievement dynamics aligns with practitioners' observations that students who struggle to recover from setbacks are more likely to lose motivation and eventually withdraw, even if their overall performance remains near average.

3.5 Summary of Key Findings

In summary, the results show that:

(1) dropout prediction using longitudinal test data generalizes well across cohorts (Table 3, Figure 1)

(2) Model 2 provides an effective balance between early prediction and dropout detection performance (Table 3, Figure 1)

(3) interpretable indicators capturing recovery from performance declines consistently differentiate graduates from dropouts across Native language, Mathematics, and English (Figures 2, 3, and 4). Together, these findings support the practical applicability of the proposed approach for early identification and intervention in large-scale supplementary education settings.

4. DISCUSSION

4.1 Practical Implications for Supplementary Education Providers

This study demonstrates that longitudinal standardized test data, which are routinely collected in large-scale supplementary education settings, can be leveraged to build accurate and interpretable dropout prediction models. Unlike prior educational data mining studies that primarily rely on digital learning logs, our approach utilizes offline test records, making it directly applicable to institutions where learning activity logs are limited or unavailable.

The selected model (Model 2) achieved strong predictive performance while enabling identification of at-risk students before the final academic year (Table 3, Figure 1). From an operational perspective, this timing is critical: interventions initiated after Grade 9 Test 4 still allow sufficient time for academic counseling, motivational support, or tailored instructional planning. The achieved balance between dropout precision and recall indicates that the model can guide resource allocation without overwhelming instructors with excessive false positives.

These results suggest that supplementary education providers can integrate predictive analytics into existing assessment infrastructures with minimal additional data collection burden. In practice, this supports data-driven early-warning systems that prioritize students requiring attention, thereby improving retention and educational outcomes.

4.2 Interpretability and Educational Meaning of Recovery Efficiency

Beyond predictive performance, an important contribution of this study lies in the interpretability of the proposed features. Across all subjects, recovery efficiency consistently distinguished graduates from dropouts (Figures 2–4), indicating that resilience to academic setbacks is a central factor associated with continued enrollment. Notably, absolute achievement levels alone were less informative than performance dynamics, aligning with practitioners' observations that disengagement often follows repeated failures to rebound from declines rather than single low test scores.

This finding offers actionable insights for educators. Students with persistently low recovery efficiency may benefit from targeted interventions focusing on study strategies, self-regulation, or motivational support. Thus, the model does not merely predict dropout risk but also provides interpretable indicators that inform the design of individualized support measures.

4.3 Contribution to Industry-Oriented Educational Data Mining

From an industry-oriented EDM perspective, this work contributes a real-world case study demonstrating how predictive modeling can be embedded into commercial educational operations at scale. The dataset spans over 3,000 students across three academic years, and the models were validated on an independent graduating cohort, addressing a common limitation of prior studies that rely on single-cohort evaluation.

Furthermore, the feature engineering strategy—summarized in Table 2—offers a reusable framework for institutions seeking to derive meaningful predictors from longitudinal assessment data. The emphasis on recovery-related and trend-based features highlights the value of modeling temporal learning trajectories, extending beyond traditional snapshot-based performance indicators.

5. LIMITATIONS AND FUTURE WORK

Several limitations should be acknowledged. First, the present analysis focuses primarily on longitudinal test performance data and does not incorporate behavioral or attitudinal measures such as attendance records, homework completion, or survey-based motivation indicators. Integrating multimodal data sources, including digital learning logs and other behavioral traces, may further enhance predictive performance and provide richer explanatory insights into student disengagement processes. In addition, the dataset includes students whose final outcomes were not fully

observed at the time of analysis. These cases may be considered as right-censored data, which could introduce uncertainty in the definition of dropout status.

Second, while the current models generalize across cohorts within a single institution, external validation across different supplementary education providers remains an important direction for future research. Institutional differences in curricula, testing schedules, or instructional policies may influence feature distributions and model transferability. In addition, ensuring the reproducibility of the proposed feature engineering procedures and modeling pipeline across diverse institutional contexts will be essential for broader deployment and comparative evaluation[9].

Finally, the present study examines prediction at discrete test intervals. Future work may explore continuous risk monitoring frameworks that update predictions as new assessments become available, enabling more adaptive intervention strategies [10]. Such extensions could also incorporate fine-grained learning logs [11][12], where available, to complement test-based indicators and support more responsive early-warning systems.

6. CONCLUSION

This industry paper presented a real-world application of dropout prediction in large-scale supplementary education using longitudinal standardized test data. By engineering interpretable features capturing achievement dynamics and validating models on an independent cohort, we demonstrated that early identification of at-risk students can be supported using routinely collected assessment records. The selected model achieved stable predictive performance while enabling intervention before the final academic year.

Beyond predictive accuracy, the analysis indicated that recovery efficiency after performance declines is consistently associated with student persistence across Native language, Mathematics, and English. This insight offers practical guidance for designing targeted academic support.

While several limitations remain and further validation across institutions and data modalities is needed, the proposed approach illustrates how existing assessment infrastructures can be adapted into practical early-warning systems, contributing operational value for educational service providers and methodological insights for industry-oriented educational data mining.

7. ACKNOWLEDGMENTS

This research is supported by Eishinkan, Co. Ltd, and Smart Learning Environment Design Research Unit at Kyushu University.

8. REFERENCES

[1] Gökhan Akçapınar, Adnan Altun, and Petek Aşkar. 2019. Using Learning Analytics to Develop Early Warning Systems for At-Risk Students. International Journal of Educational Technology in Higher Education 16, 40 (2019). https://doi.org/10.1186/s41239-019-0172-z

[2] Leo Breiman. 2001. Random Forests. Machine Learning 45, 1 (2001), 5–32. https://doi.org/10.1023/A:1010933404324

[3] Jae Young Chung and Sunbok Lee. 2019. Dropout early warning systems for high school students using machine learning. Children and Youth Services Review 96 (2019), 346–353. https://doi.org/10.1016/j.childyouth.2018.11.030

[4] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). ACM, 785–794. https://doi.org/10.1145/2939672.2939785

[5] Menna Fateen and Tsunenori Mine. 2021. Predicting Student Performance Using Teacher Observation Reports. In Proceedings of the 14th International Conference on Educational Data Mining (EDM 2021). International Educational Data Mining Society, pp.481-486.

[6] Jerome H. Friedman. 2001. Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics 29, 5 (2001), 1189–1232. https://doi.org/10.1214/aos/1013203451

[7] Scott M. Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In Advances in Neural Information Processing Systems 30, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Long Beach, CA, USA. Curran Associates, Inc., pp. 4768 - 4777.

[8] Matti Vaarma and Hongxiu Li. 2024. Predicting student dropouts with machine learning: An empirical study in Finnish higher education. Technology in Society 76 (2024), Article 102474. https://doi.org/10.1016/j.techsoc.2024.102474

[9] Misato Oi, Masanori Yamada, Fumiya Okubo, Atsushi Shimada, and Hiroaki Ogata. 2017. Reproducibility of findings from educational big data: a preliminary study, Proceedings of the seventh international learning analytics & knowledge conference, 2017, p536-537.

[10] Fumiya Okubo, Sachio Hirokawa, Misato Oi, Chengjiu Yin, Atsushi Shimada, Kentaro Kojima, Masanori Yamada, and Hiroaki Ogata. 2016. Learning Activity Features of High Performance Students, In Proceedings of the 1st International Workshop on Learning Analytics Across Physical and Digital Spaces (Cross-LAK 2016), pp. 28–33.

[11] Masanori Yamada, Xuewang Geng, Min Lu, and Yuta Taniguchi. 2024. Exploring the Role of Metacognition in Enhancing Learning Outcomes through Learning Analytics Dashboard. In eLearn: World Conference on EdTech. Association for the Advancement of Computing in Education (AACE). pp. 294-301

[12] Masanori Yamada, Xuewang Geng, Yoshiko Goda, and Stephanie D. Teasley. 2024. Investigating Metacognitive Behaviors with Online Learning Support Tools, In Proceedings of IEEE ICALT 2024, pp. 280-284