Ranking-Based At-Risk Student Prediction Using Federated Learning and Differential Features

Shunsuke Yoneda
Kyushu University
yoneda.shunsuke.860@s.kyushu-u.ac.jp

Valdemar Švábenský
Kyushu University
valdemar@kyudai.jp

Gen Li
Kyushu University
gen.li@limu.ait.kyushu-u.ac.jp

Daisuke Deguchi
Nagoya University
ddeguchi@nagoya-u.ac.jp

Atsushi Shimada
Kyushu University
atsushi@limu.ait.kyushu-u.ac.jp

ABSTRACT

Digital textbooks are widely used in various educational contexts, such as university courses and online lectures. Such textbooks yield learning log data that have been used in numerous educational data mining (EDM) studies for student behavior analysis and performance prediction. However, these studies have faced challenges in integrating confidential data, such as academic records and learning logs, across schools due to privacy concerns. Consequently, analyses are often conducted with data limited to a single school, which makes developing high-performing and generalizable models difficult. This study proposes a method that combines federated learning and differential features to address these issues. Federated learning enables model training without centralizing data, thereby preserving student privacy. Differential features, which utilize relative values instead of absolute values, enhance model performance and generalizability. To evaluate the proposed method, a model for predicting at-risk students was trained using data from 1,136 students across 12 courses conducted over 4 years, and validated on hold-out test data from 5 other courses. Experimental results demonstrated that the proposed method addresses privacy concerns while achieving performance comparable to that of models trained via centralized learning in terms of Top-n precision, nDCG, and PR-AUC. Furthermore, using differential features improved prediction performance across all evaluation datasets compared to non-differential approaches. The trained models were also applicable for early prediction, achieving high performance in detecting at-risk students in earlier stages of the semester within the validation datasets.

Keywords

grade prediction, early prediction, risk ranking, privacy protection, generalizability, educational data mining

Shunsuke Yoneda, Valdemar Svabensky, Gen Li, Daisuke Deguchi, and Atsushi Shimada. Ranking-Based At-Risk Student Prediction Using Federated Learning and Differential Features. In Caitlin Mills, Giora Alexandron, Davide Taibi, Giosuè Lo Bosco, and Luc Paquette (eds.) Proceedings of the 18th International Conference on Educational Data Mining, Palermo, Italy, July, 2025, pp. 289–302. International Educational Data Mining Society (2025).

https://doi.org/10.5281/zenodo.15870193

1. INTRODUCTION

Digital textbooks are widely used due to their capability to not only allow students to access learning materials on personal devices but also collect records of their interactions as learning logs. These digital textbooks are now implemented in many educational institutions [43, 13, 8, 7], leading to the accumulation of vast amounts of learning logs.

This development has motivated research on students' learning behavior and the prediction of academic performance using learning logs [24, 51, 50]. For example, studies have developed systems that provide instructors with real-time visualizations of students' learning progress, such as the percentage of students keeping pace with the lecture or those remaining on a previous page [42]. Other studies have utilized learning log data to predict final exam scores and classify students into higher- and lower-performing [9].

However, these studies have faced major difficulties in integrating academic performance data and learning logs across schools, which introduces privacy concerns [5, 4, 25]. Consequently, creating generalizable and high-performance models is challenging due to the limited availability of data [1, 44]. Thus, developing methods for performance prediction without directly integrating data is crucial to advance EDM research and strengthen learning support in educational practice.

In conventional machine learning (ML), data stored in separate locations must be centralized on a single server for model training, which raises concerns about privacy – an important topic within the EDM community [5, 4, 25]. Prior work has focused on this issue from the perspective of privacy-preserving EDM infrastructures, such as MORF [19], which allow to train ML models without direct access to the data. *Federated learning*, which has gained attention as a privacy preserving ML approach in non-EDM contexts [29, 37, 22, 32], supports privacy from a different perspective. It is based on distributing model parameters to data owners (hereinafter, referred to as "clients"), who train the model locally on their data. The locally trained parameters are then aggregated on a central server, enabling training without transferring raw data. Compared to approaches like MORF, the advantage of federated learning is that it supports privacy without the need to establish a centrally managed infrastructure. Since data can be processed in a decentralized manner, no management cost applies, and security measures associated with centralized data management are also eliminated.

However, when applying federated learning, discrepancies in the feature distributions among clients can arise due to differences in context, such as e-book usage frequencies or course schedules. These differences can bias model training and degrade the model performance [21, 52, 30]. Prior studies have explored methods focused on improving aggregation to prevent these differences from degrading the models performance [31, 15]. Our study proposes an approach that addresses these discrepancies through data preprocessing using *differential features* [46, 6]. By utilizing relative feature values instead of absolute ones, this approach mitigates disparities in feature distributions among clients. As a result, it improves both generalizability and model performance – two key aspects of many EDM studies [39, 2, 48, 10].

In summary, federated learning with differential features has shown promise for preserving privacy and enhancing ML model properties in non-EDM contexts. However, to the best of our knowledge, this approach has not been validated in the context of EDM research. Therefore, our study aims to *evaluate ML models for at-risk student prediction using the unique approaches of federated learning with differential features*. Furthermore, we investigate whether accurate predictions can be achieved in the early stages of a course—specifically, using learning log data collected up to the halfway point of the lecture sessions—as timely identification of at-risk students is essential for practical interventions in real-world educational settings.

Our study makes the following contributions:

- We enable accurate prediction of students' academic performance using learning log data, with a particular focus on identifying *at-risk students*, while preserving privacy through *federated learning*.
- We enhance model generalizability and performance by utilizing *differential features*, which capture relative differences in students' learning behaviors and academic performance instead of absolute values, thereby mitigating distributional disparities across datasets.
- We demonstrate the applicability of our approach to *early prediction*, showing that at-risk students can be accurately identified in the early stages of a course based on partial learning log data.

2. RELATED WORK

We aim to develop a generalized, high-performance ML model for predicting grades while preserving privacy using federated learning with differential features. Therefore, this section reviews prior studies related to this research across two themes: *grade prediction* and *federated learning*.

2.1 Prediction of Students' Grades

Predicting grades is a crucial area of EDM research aimed at supporting learning activities and enabling personalized educational interventions. Various methods have been proposed, including classification models that categorize students based on their predicted performance [33, 41, 12] and regression models that estimate continuous grade values [18, 28].

For example, Chen et al. [9] constructed a model to classify university students into "higher-score students" or "lower-score students" by leveraging various features, including learning behaviors within digital textbooks (e.g., turning pages, adding/deleting markers, and editing/removing memos). Ong et al. [38] examined whether incorporating "instructor-related features" improves the performance of students' grade prediction using both regression and classification. Altabrawee et al. [3] developed a model to predict academic performance in computer science courses at a university. Their model used features such as frequency of using the internet for studying, time spent on social media, and previous semester grades to classify students as either "Good" or "Weak".

Overall, research on grade prediction encompasses a wide range of approaches, with each study employing different features and methods. However, many studies have focused on predicting grades for specific courses, which limits their applicability. The impact of data changes throughout a single course [27] as well as across various courses. Yet, the models' generalizability to different course contents has not been sufficiently investigated. Therefore, this study aims to develop and evaluate a model capable of generalizing across courses to detect students with poor final grades. To achieve this, federated learning was utilized to address privacy concerns, and differential features were introduced to improve the model's generalizability and performance.

2.2 Federated Learning

Federated learning has gained considerable attention in recent years for enabling privacy-preserving model training across various fields outside EDM [29]. Oldenhof et al. [37] demonstrating federated learning in the medical drug discovery process. Federated learning was employed to train predictive models collaboratively across multiple pharmaceutical companies without centralizing sensitive data. This approach yielded a shared predictive model while maintaining data confidentiality. Kanamori et al. [22] applied federated learning in detecting fraudulent financial transactions. They developed a model in collaboration with five banks, enabling data analysis without centralizing confidential information. The federated learning system outperformed models trained solely on individual bank data, achieving higher performance in detecting fraudulent transactions. This system enabled the detection of fraudulent accounts before actual monetary losses occurred. Liu et al. [32] proposed a federated learning approach for traffic flow prediction, allowing multiple organizations to collaboratively build predictive models without sharing raw data. By leveraging information collected locally, organizations could develop more accurate traffic prediction models while preserving data privacy. Each organization used its own traffic data to train models and shared only the resulting parameters, ensuring the protection of confidential information. Experiments using real-world traffic data demonstrated that this method achieved superior predictive performance compared to traditional approaches.

As demonstrated in the aforementioned studies, federated learning shows strong promise in other fields, enabling multiple organizations to collaboratively train ML models while preserving data privacy. Moreover, this approach increases the amount of data available for training, thereby potentially also enhancing predictive performance. However, federated learning remains underexplored in educational contexts. No publication at the EDM conference in the past five years (2020–2024) has focused on federated learning (based on examining the titles in the proceedings). Since the effectiveness of federated learning has not thoroughly validated – particularly in addressing challenges posed by heterogeneous data distributions – our study is unique because it focuses on these aspects.

To the best of our knowledge, only one recent paper has employed federated learning in education. In 2024, Haastrecht et al. [47] investigated federated learning for educational analytics by comparing different approaches across multiple prediction tasks. While their study demonstrated the feasibility of federated learning, it did not address the impact of feature distribution discrepancies across clients, which can substantially affect model performance. By mitigating the issues caused by feature distribution discrepancies using differential features, our study explores the potential of federated learning in EDM and validates its effectiveness.

3. THE PROPOSED MODELING METHOD

This section explains our method for creating a model that preserves privacy through federated learning and leverages differential features. The overall framework is illustrated in Figure 1. The following subsections focus on *federated learning* and *differential features*, detailing how these approaches are used to conduct learning and prediction.

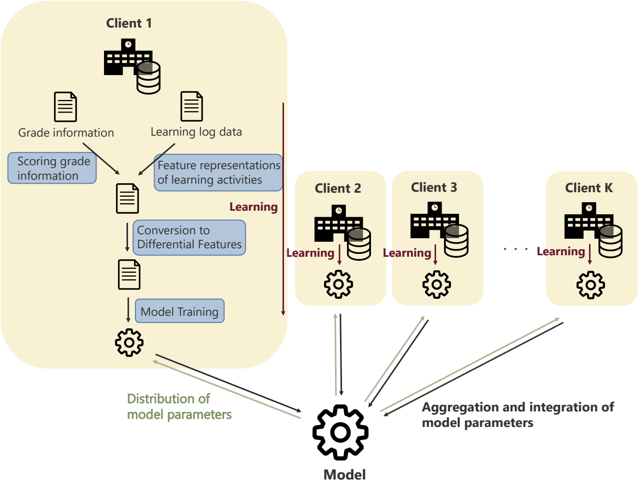

Client 1
Grade information
Learning log data
Scoring grade information
Feature representations of learning activities
Learning
Conversion to Differential Features
Model Training
Client 2
Client 3
Client K
Learning
Learning
· · ·
Learning
Distribution of model parameters
Aggregation and integration of model parameters
Model

Figure 1: The overall framework of our proposed method

3.1 Federated Learning

Figure 2 illustrates the learning and prediction processes of federated learning utilized in this study. This approach enables model training and prediction without the need to aggregate data on server, ensuring data privacy.

In the learning phase, server distributes model parameters to each client. Each client trains the model using its locally held data. Subsequently, the trained model parameters and number of data samples held by clients are sent back to the server, where the model parameters are aggregated. This process is repeated multiple times to train the model on the server.

For the prediction phase, the trained model parameters are distributed to client requiring predictions. The client applies the model to its locally held data to perform predictions.

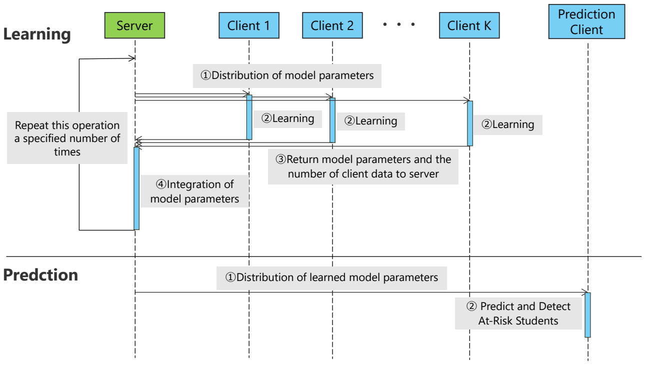

Learning
Server
Client 1
Client 2
· · ·
Client K
Prediction Client
①Distribution of model parameters
②Learning
②Learning
②Learning
Repeat this operation a specified number of times
③Return model parameters and the number of client data to server
④Integration of model parameters
Predction
①Distribution of learned model parameters
② Predict and Detect At-Risk Students

Figure 2: The sequence diagram of learning and prediction in federated learning

3.1.1 Parameter Integration Method

This study employs FedAvg [34] as the method for integrating model parameters. FedAvg performs weighted averaging of model parameters trained by each client, where the weights are determined by the number of data samples held by each client.

Specifically, let ω_k^t denote the model parameters trained by client k at epoch t and n_k represent the number of data samples held by that client. The integrated model parameters ω^t after applying FedAvg are expressed as follows:

$$\omega^t = \sum_{k=1}^{K} \frac{n_k}{N} \omega_k^t \tag{1}$$

N is the total number of data samples across all clients.

As shown in Figure 2, the integrated model parameters ω^t from epoch t are distributed to all clients at the beginning of epoch $t+1$. Each client then uses these parameters to train the model locally during the next epoch.

3.1.2 Overview of Client-Side Learning

Figure 3 illustrates the part of the proposed method focused on client-side learning, which is extracted from the overall framework shown in Figure 1. Here, we briefly explain the client-side learning process.

First, learning log data are transformed into feature representations, while academic performance data are converted into scores. Subsequently, differential features are applied to these datasets to create relative data. Training on these relative data allows to obtain a generalized model.

The following sections provide a detailed explanation of each aspect.

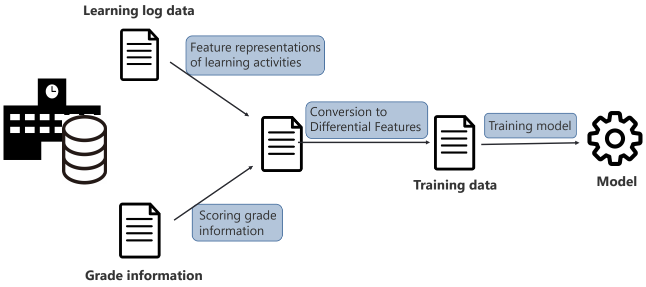

Learning log data
Feature representations of learning activities
Conversion to Differential Features
Training model
Training data
Model
Scoring grade information
Grade information

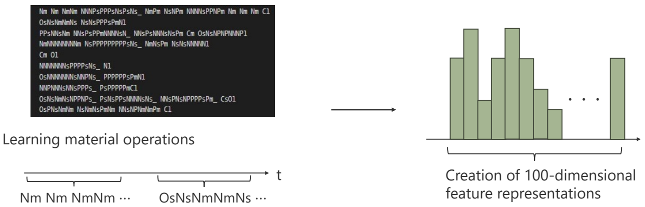
Figure 3: Overview of client-side learning

3.1.3 Learning Log Data in Digital Textbooks

Digital textbook systems are educational platforms that allow learners to access and interact with learning materials through personal devices. While viewing the materials, learners can perform various actions, such as navigating forward or backward through pages, adding notes, or using markers. These interactions are recorded as learning log data and stored in a database.

3.1.4 Feature Representation of Learning Activities

Learning activities cannot be directly used to train ML models; thus, preprocessing the activities of each student into feature representations is necessary. To create feature representations, we adopted the distributed representation of learning material operations, E2Vec [35]. Figure 4 shows an overview of the feature representation creation process.



```
Nm Nm NmNm NNNPsPPPsNsPsNs_ NmPm NsNPm NNNNsPPNPm Nm Nm Nm C1
OsNsNmNmNs NsNsPPPsPmN1
PPsNNsNm NNsPsPPmNNNNsN_ NNsPsNNNsNsPm Cm OsNsNPNPNNNP1
NmNNNNNNNNm NsPPPPPPPPPsNs_ NmNsPm NsNsNNNNN1
Cm O1
NNNNNNNsPPPPsNs_ N1
OsNNNNNNNsNNPNs_ PPPPPPsPmN1
NNPNNNsNNsPPPs_ PsPPPPPmC1
OsNsNmNsNPPNPs_ PsNsPPsNNNNsNs_ NNsPNsNPPPPsPm_ CsO1
OsPNsNmNm NsNmNsPmNm NNsNPNmNmPm C1
```

Learning material operations
t
Nm Nm NmNm ···
OsNsNmNmNs ···
. . .
Creation of 100-dimensional feature representations

Figure 4: Feature representation creation using E2Vec

This method generates feature representations of learners by creating a sequence of symbols from learning logs while preserving the temporal order and time intervals between learning actions. E2Vec defines primitives, units, and actions corresponding to characters, words, and sentences in natural language processing. Learner's log data are expressed as multiple actions with their distributed representations. These representations of actions are then aggregated using a method inspired by Bag of Visual Words [11], resulting in 100-dimensional feature representations for each learner.

In the original EDM study by Miyazaki et al. [35], the feature representations of each student were L2-normalized. However, in this study, normalization was omitted to account for the number of actions generated, enabling the features to be directly used for training.

3.1.5 Scoring of Student Grades

As described further in Section 3.1.6, this study employs a regression model for at-risk students' prediction. Students' academic performance is represented by standard letter grades as points on a five-level scale: F, D, C, B, and A. To make these data compatible with the regression model, the grade points must be converted into numerical scores.

For each grade, let x_1, x_2, x_3, x_4, and x_5 denote the number of students with grades F, D, C, B, and A, respectively, and let X represent the total number of students in the client. The converted grade value G_m for grade m is defined as:

$$G_m = \text{MaxScore} \times \frac{\sum_{j=1}^{m} x_j}{X} \quad (m = 1, 2, 3, 4, 5) \tag{2}$$

Additionally, considering the general criterion that students with total scores between 90% and 100% are assigned grade A, we adopt 0.95 as the value for MaxScore in this study.

3.1.6 Ranking-Based Prediction Using Regression

This study employs a regression model to detect at-risk students. The feature representations are input into the regression model to obtain predicted academic performance values for each student. These prediction values are then sorted in ascending order to identify high-risk students in a ranking format, referred to as a "risk ranking" hereinafter.

An example of the ranking creation using the regression model is shown in Figure 5. The prediction values in the figure are hypothetical and used for illustrative purposes.

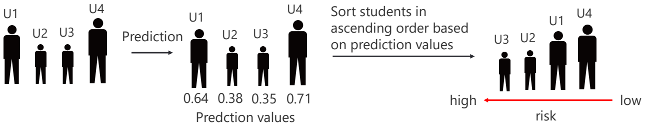


Figure 5: Ranking creation method using a regression model

The risk-ranking-based prediction using a regression model was adopted for the following three key reasons:

1. **Compatibility with Differential Features**
   As described further in Section 3.2, the use of differential features enables to establish higher/lower relationships between students' grades. By employing a regression model, the extent of these differences can be explicitly learned, allowing the model to capture and utilize them.
2. **Limitations of Classification Models**
   Classification models divide students into *at-risk* and *no-risk* groups. However, this complicates identifying no-risk students who are close to being at-risk or at-risk students who are closer to no-risk. This limitation has been observed in previous EDM studies [49]. In contrast, this study utilizes a regression model to estimate predicted performance values. Ranking students based on these prediction values enables to better identify students near the boundary between the two groups.
3. **Improved Generalization of the Predictions**
   The proposed method enables to create highly generalizable models for prediction. For example, consider a scenario where a course consists of 16 lecture sessions, and the goal is to detect at-risk students based on the

data available until the 6th lecture for early prediction. If a model trained on data from all 16 lecture sessions is applied to early prediction data, classification models may struggle because of shifts in data distribution. Since classification relies on decision boundaries, the limited availability of data in early stages can lead to decreased confidence scores and increased misclassification rates. In contrast, a regression model applied in a ranking format mitigates this issue. While the absolute prediction values may fluctuate when trained on full-course data but applied to partial-course data, if this fluctuation occurs uniformly, the ranking order among students remains unaffected. This ensures that high-risk students can still be accurately identified. By leveraging a ranking-based approach with regression models, the proposed method eliminates the need to build separate models for early prediction, allowing models trained on complete data to be directly applied for detecting at-risk students at any stage of the course.

3.2 Differential Features

This study employs *differential features* to enhance the model's performance and generalizability. Such features are created by calculating the differences between feature representations and grade information of two students within the data held by a client. This approach generates data that represent the relative differences between students.

As an example, Figure 6 illustrates the application of differential features to a client that holds data for five students. The combinations shown in the figure represent only a subset of possible combinations for illustrative purposes and do not cover all potential pairings.

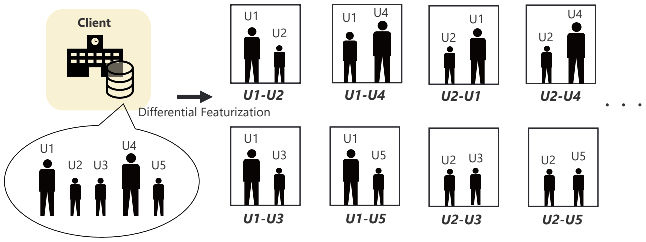


Figure 6: Example of differential feature creation

Specifically, let the set of students in a client be denoted $\mathbf{S}$, the feature representation of student i ($i \in \mathbf{S}$) be v_i, and the grade of student i after scoring based on Equation 2 be g_i. Then, the feature representation d_{ij} and grade information e_{ij} after applying differential features are expressed as:

$$d_{ij} = v_i - v_j \ (i \neq j, \ i, j \in \mathbf{S}) \tag{3}$$

$$e_{ij} = g_i - g_j \ (i \neq j, \ i, j \in \mathbf{S}) \tag{4}$$

3.2.1 Advantages of Differential Features

Differential features have two main benefits:

1. **Increase in Training Data**
   If a client holds data for n students, the use of differential features expands the number of data points to $n(n-1)$. This expansion helps mitigate overfitting and bias when training the local model on clients with limited data because the increased data volume provides a richer dataset for training.
2. **Improved Generalization by Utilizing Relative Values**
   Figure 7 illustrates that introducing differential features enables the use of relative values. Without differential features, absolute feature values are used for training. This can bias the server model because of differences in feature distributions among clients. For instance, clients with different course structures (e.g., semester-based vs. quarter-based lectures) may exhibit substaintial differences in the features' absolute values due to varying interactions with learning materials.

   Instead, differential features use relative values, reducing the impact of such discrepancies and enabling the construction of a more generalized model. This approach improves the model's robustness and adaptability across clients with varying feature distributions.

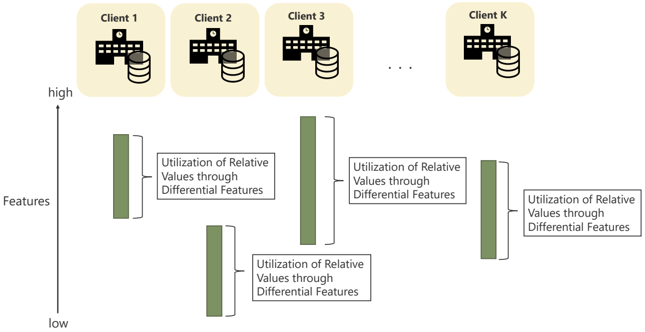

Client 1
Client 2
Client 3
. . .
Client K
high
Features
low
Utilization of Relative Values through Differential Features
Utilization of Relative Values through Differential Features
Utilization of Relative Values through Differential Features
Utilization of Relative Values through Differential Features

Figure 7: Use of relative values in differential features (the graphs represent the distribution of feature values across different clients)

3.2.2 At-Risk Student Predction Using Differential Features

As described in Section 3.1.6, this study employs a regression model to rank students in terms of their risk level (in order from the highest to the lowest risk). However, with differential features, the regression model no longer outputs individual predictions for students but instead provides prediction values for the differences between two students (hereinafter, referred to as "pairwise difference scores"). Therefore, the method for obtaining individual prediction values when using differential features is described below.

Let $\mathbf{S}$ be the set of students in a client. Using the feature representation with differential features, d_{ij}, in the regression model yields a prediction value p_{ij}. However, since p_{ij} represents the prediction value for the difference between two students, it cannot directly be used for at-risk detection. The individual prediction value q_i for a student i ($i \in \mathbf{S}$) is derived as follows:

$$q_i = \sum_{j \in \mathbf{S}, j \neq i} p_{ij} \ (i \in \mathbf{S}) \tag{5}$$

An example of calculating the individual prediction value for a specific student in a client with data for five students is illustrated in Figure 8. (The values are synthetic for illustrative purposes.)

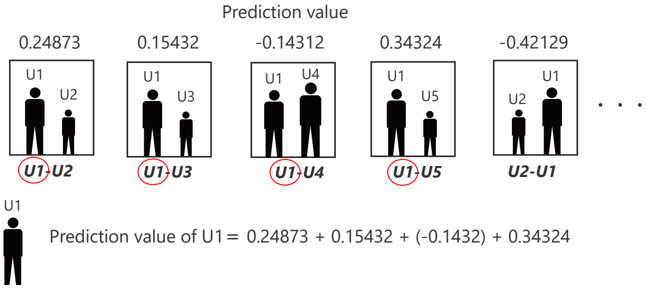


Figure 8: Example of deriving individual prediction values for a student U1 from differential-feature-based predictions

After calculating the individual prediction values for all students, as described in Section 3.1.6, the students are sorted in ascending order of their individual prediction values q_i. This creates a risk ranking, which can then be used to identify at-risk students based on their relative ranks.

It is important to note that the pairwise difference scores obtained from the regression model are used solely for internal model computation and are not provided to users (instructors and/or students, depending on the specific application). Instead, the output available to users is limited to a risk ranking of individual students. Therefore, users are not required to interpret the pairwise scores themselves, and the use of differential features does not hinder the interpretability of the system from the users' perspective.

4. EXPERIMENTAL EVALUATION

This section presents the comparison between the proposed method and baseline methods (defaults without the experimental condition). Additionally, it discusses the proposed method's early prediction capabilities and examines the corresponding results.

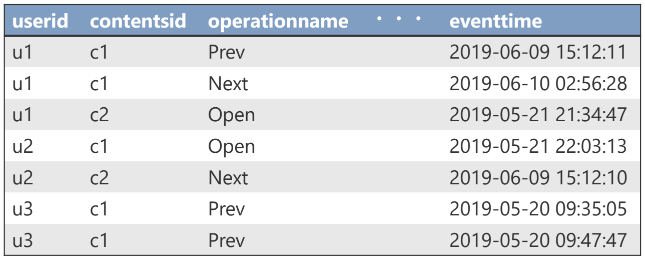
4.1 Experimental Setup

4.1.1 Learning Log Data in the E-book Platform

BookRoll [36, 14] is a widely used system that allows students to access learning materials registered by instructors through their individual devices. When learners view the materials, control buttons are displayed alongside the content of the opened pages. These buttons have various functions, such as moving between pages and adding notes or markers. All these actions are recorded as log data and stored in a database. An example of the recorded log is shown in Figure 9.

The log data include the following information: IDs to identify the learner performing the operation and the material being accessed, the type of operation performed (denoted as "operation name"), and the timestamp of performing the operation (denoted as "event time"). Our study uses these learning log data for model training and prediction.

4.1.2 Data and Clients in Federated Learning



userid	contentsid	operationname	· · ·	eventtime
u1	c1	Prev		2019-06-09 15:12:11
u1	c1	Next		2019-06-10 02:56:28
u1	c2	Open		2019-05-21 21:34:47
u2	c1	Open		2019-05-21 22:03:13
u2	c2	Next		2019-06-09 15:12:10
u3	c1	Prev		2019-05-20 09:35:05
u3	c1	Prev		2019-05-20 09:47:47

Figure 9: Format of the learning logs in the BookRoll system

We collected and used a substantial dataset from *four years* (eight semesters) of undergraduate courses at Kyushu University. A total of *seven courses* were utilized for training and prediction, labeled from A to G. These courses span a diverse range of topics, learning formats, and academic terms, as detailed in Table 1. In this study, data collection was conducted with the informed consent of the students. To ensure privacy protection, all collected data were anonymized and handled to prevent individual identification. Additionally, this study was approved by the institutional ethics committee.

Table 1: Details of courses used for training and prediction

Course	Topic	Format	Academic term
A	Scheme	Lecture+Exercise	Quarter
B	Security	Lecture	Quarter
C	Information and Communication	Lecture	Semester
D	Signal Processing	Lecture	Semester
E	Programming	Exercise	Semester
F	Artificial Intelligence Technology	Lecture	Semester
G	Fortran	Lecture+Exercise	Semester

The grade distribution of the training data used in this study is shown in Table 2. This table presents the number of students receiving each grade (A, B, C, D, F) across different courses, along with the total number of students and the number of lecture weeks for each course. The training dataset includes *1,136 students*, covering a wide range of courses and academic terms. In the "Course" column, the letter represents the course name, while the following number indicates the academic year in which the course was conducted.

Similarly, Table 3 shows the grade distribution for the holdout test data used for prediction evaluation. In addition to student grades, this table includes the number of students classified as "At-risk" and "No-risk" based on their final grades. In this study, students are classified as at-risk if their grade is less than or equal to the grade of the student ranked 15th[1] from the bottom in their actual final grads, while the remaining students are classified as no-risk. This boundary is set arbitrarily for evaluation and is not used to differentiate between at-risk and no-risk during model training.

Finally, we note that the number of no-risk students (264) is more than twice that of at-risk students (127), resulting in a slightly imbalanced dataset. Nevertheless, this distribution

[1] The threshold of 15 was chosen to ensure that the number of at-risk students is sufficient for evaluating Top-n precision ($n = 15$), see the next section.

Table 2: Grade distribution and the number of lecture weeks in the *training data*

Course	A	B	C	D	F	Students	Lectures
A-2019	15	9	6	10	12	52	8
A-2020	22	23	5	3	7	60	7
A-2021	9	11	10	18	6	54	8
B-2019	30	103	28	1	1	163	8
C-2021-1	9	53	32	7	6	107	15
C-2021-2	15	88	37	26	9	175	15
D-2020	61	7	1	2	34	105	14
D-2021	60	3	6	4	33	106	15
E-2020-1	17	23	12	8	13	73	14
E-2020-2	0	2	8	21	25	56	15
F-2021	71	13	4	3	3	150	14
G-2021	26	3	3	0	3	35	16
Total	335	338	152	103	152	1136	

Table 3: Grade distribution and the number of lecture weeks in the separate hold-out *test data* for prediction evaluation

Course	A	B	C	D	F	No-risk	At-risk	Lectures
A-2022	17	6	5	22	2	28	24	8
B-2020	37	38	12	2	4	75	18	7
C-2022-1	17	37	34	4	4	54	42	15
D-2022	50	10	8	8	17	76	17	16
E-2021	3	16	8	4	26	31	26	16
Total	124	107	67	40	53	264	127	

is expected, since we assume to have more students who are not at risk.

In this study, the data in each row of Table 2 (except the summary row "Total") were treated as a client, resulting in 12 clients for training. Subsequently, the trained model was applied to the 5 courses serving as hold-out test data (shown in Table 3) to perform at-risk student prediction.

4.1.3 *Evaluation Metrics*

We utilize a ranking-based approach where students are ordered in ascending order of their prediction values to create a risk ranking. To evaluate the propose method's performance, we use three ranking-specific evaluation metrics: Top-n precision, nDCG, and PR-AUC:

1. **Top-n Precision**
   Top-n precision indicates the proportion of students who are actually at-risk among the top-n students predicted to incur the highest risk. In this study, we evaluate Top-n precision with four different settings: $n = \{5, 10, 15, \text{At-risk}\}$. Here, the "At-risk" refers to the actual number of students classified as at-risk in the test data.

2. **Normalized Discounted Cumulated Gain (nDCG)**
   As described in the reference paper [20], nDCG is a ranking-specific metric used to compare the predicted ranking order based on prediction values with the actual ranking order based on the students' grades. To compute nDCG, each student must be assigned a value. Since higher-risk students are ranked higher in this study, the values assigned to students must increase as their risk level increases. Therefore, we utilize the score G_m derived from Equation 2 and assign each student a value of $1 - G_m$ $(m = 1, 2, 3, 4, 5)$ to calculate nDCG.

3. **Area Under the Precision-Recall Curve (PR-AUC)**
   The Precision-Recall (PR) curve is commonly used in EDM research [17, 26] to evaluate predictive model performance. It plots Top-n precision (vertical axis) against Top-n recall (horizontal axis) as n varies, illustrating their relationship. The area under this curve (i.e., PR-AUC) quantifies model performance, with higher values indicating better balance between precision and recall.

For a robust evaluation, a single evaluation result from one model would be insufficient. Therefore, we conducted training 10 times and calculated the average of the evaluation metrics obtained from the models. This approach provides a more reliable assessment of the model's performance.

4.1.4 *Regression Model*

A neural network was employed as the regression model during training. This neural network consists of two hidden layers: the first layer comprises 50 nodes and the second layer 10 nodes. To prevent overfitting, we applied dropout between the hidden layers, with a dropout rate of 20%. The activation function in each hidden layer is ReLU, commonly used in similar EDM contexts [48, 45].

4.1.5 *Comparisons of Experimental and Baseline Conditions*

We conducted two types of comparative experiments:

1. **Performance comparison between "Federated Learning" and "Centralized Machine Learning"**
   Centralized ML is the baseline method in which the data from all clients are collected in a single location, with potential privacy concerns. The model is trained using the aggregated data.

2. **Performance comparison between "With Differential Features" and "Without Differential Features"**
   Without differential features, applying the regression model to the feature representations yields individual prediction values for each student. As described in Section 3.1.6, students can be easily ranked in order of risk by sorting them in ascending order of their individual prediction values.

4.2 Experimental Results

4.2.1 *Federated vs. Centralized Learning*

First, we compared "Federated Learning" and "Centralized Machine Learning". To ensure stable and uniform conditions for comparison, all evaluations used differential features. The comparison results are shown in Table 4.

These results demonstrate that the Proposed Method (using federated learning) achieves almost the same performance as Baseline Method 1 (using centralized machine learning) across all test data. Specifically, in terms of nDCG, the

Table 4: Proposed Method (using Federated Learning) vs. Baseline Method 1 (using Centralized ML)

Test Data	Method	Top-n precision				nDCG	PR-AUC
		$n = 5$	$n = 10$	$n = 15$	$n =$ At-risk		
A-2022	**Proposed Method**	0.96	0.83	0.80	**0.63**	0.83	**0.75**
	Baseline Method 1	**1.00**	**0.85**	**0.81**	**0.63**	**0.84**	**0.75**
B-2020	**Proposed Method**	**0.72**	**0.56**	**0.43**	**0.41**	**0.72**	**0.46**
	Baseline Method 1	0.64	0.53	0.41	0.37	0.69	0.43
C-2022-1	**Proposed Method**	**1.00**	0.93	**0.79**	**0.72**	**0.87**	**0.78**
	Baseline Method 1	**1.00**	**0.95**	**0.79**	0.70	0.85	**0.78**
D-2022	**Proposed Method**	**0.80**	**0.90**	**0.84**	**0.79**	**0.95**	**0.83**
	Baseline Method 1	**0.80**	**0.90**	0.83	**0.79**	**0.95**	**0.83**
E-2021	**Proposed Method**	**1.00**	**0.97**	**0.83**	**0.64**	**0.86**	**0.78**
	Baseline Method 1	0.98	0.96	**0.83**	0.63	0.85	0.77

Table 5: Proposed Method (With Differential Features) vs. Baseline Method 2 (Without Differential Features)

Test Data	Method	Top-n precision				nDCG	PR-AUC
		$n = 5$	$n = 10$	$n = 15$	$n =$ At-risk		
A-2022	**Proposed Method**	**0.96**	**0.83**	**0.80**	**0.63**	**0.83**	**0.75**
	Baseline Method 2	0.88	0.83	0.73	0.61	0.80	0.70
B-2020	**Proposed Method**	**0.72**	**0.56**	**0.43**	**0.41**	**0.72**	**0.46**
	Baseline Method 2	0.58	0.47	0.35	0.31	0.62	0.37
C-2022-1	**Proposed Method**	**1.00**	**0.93**	0.79	**0.72**	**0.87**	**0.78**
	Baseline Method 2	0.96	0.92	**0.85**	0.65	0.82	0.75
D-2022	**Proposed Method**	0.80	**0.90**	**0.84**	**0.79**	**0.95**	**0.83**
	Baseline Method 2	**0.86**	0.85	0.79	0.76	0.93	0.82
E-2021	**Proposed Method**	**1.00**	**0.97**	**0.83**	**0.64**	**0.86**	**0.78**
	Baseline Method 2	0.92	0.80	0.71	0.60	0.81	0.68

performance degradation of the Proposed Method is at most 0.01 in A-2022, while maintaining comparable or superior performance in other test data. Furthermore, for PR-AUC and Top-n precision ($n =$ At-risk), the Proposed Method consistently demonstrates performance that is equal to or better than Baseline Method 1 across all test data.

Therefore, federated learning enables at-risk student prediction with performance comparable to that of centralized machine learning, with the added benefit of more strongly preserved privacy.

4.2.2 With vs. Without Differential Features

Next, we compared "With Differential Features" and "Without Differential Features". To ensure stable and uniform conditions for comparison, all evaluations used federated learning. The comparison results are presented in Table 5.

The Proposed Method (with differential features) consistently outperformed Baseline Method 2 (without differential features) across all test data in terms of both nDCG and PR-AUC metrics. This indicates that introducing differential features enables more accurate detection of at-risk students, at least in the context of e-book log data.

This improvement is primarily due to two key advantages of differential features: (1) data augmentation, which increases the amount of training data per client and stabilizes learning, reducing overfitting; and (2) relative value utilization, which mitigates differences in feature distributions across clients, improving generalization. These benefits allow the model to maintain stable performance even in heterogeneous educational environments. However, while differential features substantially contribute to these improvements, other potential factors might have also influenced the results.

The findings suggest that introducing differential features enhances not only the performance but also the generalizability of the prediction model, enabling it to achieve stable performance across various test data.

4.2.3 Application to Early Prediction

In the previous sections, predictions were conducted using data collected after the completion of all lecture sessions. However, in practice, the early prediction of at-risk students is crucial [27]. Therefore, we investigated the prediction performance when using learning logs obtained from lecture sessions up to a certain point. The results are shown in Figures 10 to 14.

In these figures, the horizontal axis represents the number of completed lecture sessions, while the vertical axis indicates the PR-AUC evaluation score. The figures illustrate how the prediction performance improves as the number of lecture sessions used in the prediction increases. The blue points represent the results obtained using the proposed method; the yellow points represent the results of Baseline Method 1; the green points represent the results of Baseline Method 2, and the red points represent the results obtained by arranging the students in a random order.

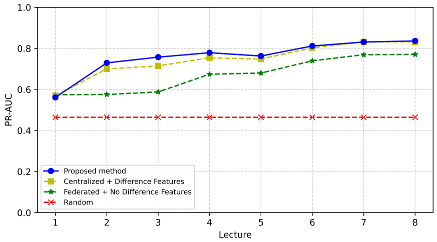


Figure 10: Relationship between lecture sessions and PR-AUC in course A-2022

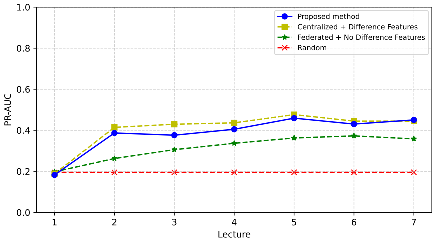


Figure 11: Relationship between lecture sessions and PR-AUC in course B-2020

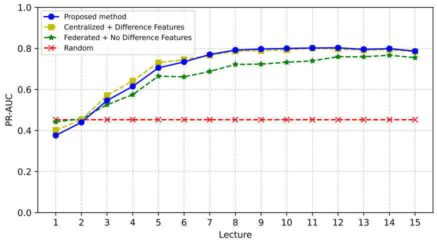


Figure 12: Relationship between lecture sessions and PR-AUC in course C-2022-1

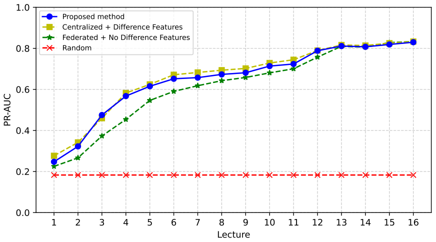


Figure 13: Relationship between lecture sessions and PR-AUC in course D-2022

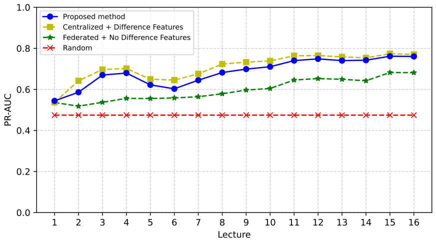


Figure 14: Relationship between lecture sessions and PR-AUC in course E-2021

Table 6: Evaluation of Risk Ranking in Early Prediction

Test Data	**Method**	**Top-n precision**				**nDCG**	**PR-AUC**
		$n = 5$	$n = 10$	$n = 15$	$n =$ At-risk		
A-2022	**Proposed Method**	1.00	0.82	0.77	0.71	0.81	0.78
B-2020		0.64	0.48	0.41	0.37	0.68	0.40
C-2022-1		1.00	0.97	0.92	0.73	0.84	0.79
D-2022		0.78	0.82	0.73	0.69	0.87	0.67
E-2021		0.90	0.80	0.71	0.58	0.80	0.68

The results show that for the quarter-based lectures, A-2022 and B-2020, predictions at the end of the 4th lecture achieved performance almost equivalent to that of predictions made after the final lecture. For semester-based lectures, C-2022-1 achieved comparable performance to the final prediction at the end of the 8th lecture, while D-2022 showed continuous improvement in prediction performance as lecture sessions used for prediction increased. For E-2021, while performance temporarily declined, it improved as lecture sessions used in the prediction increased.

In summary, for A-2022, B-2020, and C-2022-1, detection performance after half of the total lectures was comparable to performance after all lectures. Therefore, for early prediction, the evaluation was conducted using data up to the 4th lecture for A-2022 and B-2020 and up to the 8th lecture for C-2022-1, D-2022, and E-2021. The results are summarized in Table 6. These results indicate that high performance in at-risk prediction can still be achieved in early prediction scenarios.

For all test data except B-2020, Top-n precision ($n = 15$) exceeded 0.7. Thus, when extracting the top 15 high-risk students from the risk ranking, at least 10 of them were accurately classified as at-risk in their final grades. These findings demonstrate the effectiveness of the proposed method for early prediction and its ability to accurately identify high-risk students early in the lecture series.

4.2.4 *Risk Rankings in Early Prediction*

The visualized risk rankings in early prediction using the proposed method are shown in Figures 15 to 19.

In these figures, the horizontal axis represents the students' grades (F, D, C, B, A), while the vertical axis represents their ranks in the risk ranking. The dots are color-coded: students identified as at-risk (corresponding to the number of at-risk students in Section 4.1.2) are marked red, while all other students are marked blue. This color coding enables an intuitive understanding of the relationship between students' grades and their ranks in the risk ranking.

In most test data, students with lower grades tend to be represented by red dots, which indicates a higher proportion of at-risk students among those with poor academic performance. These results suggest that the proposed method effectively captures the relationship between academic performance and risk, consistently identifying students with lower grades as high-risk. This demonstrates that the generated risk rankings align well with the expected academic trends, reinforcing the validity of the model's predictions.

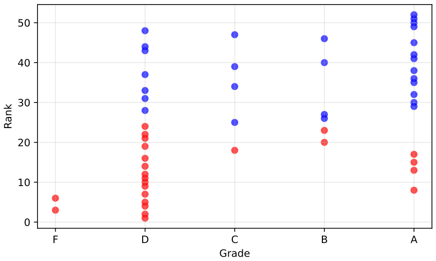


Figure 15: Relationship of grades and rankings in early prediction for A-2022

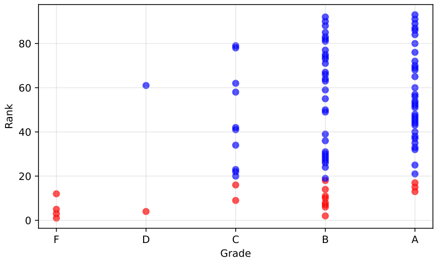


Figure 16: Relationship of grades and rankings in early prediction for B-2020

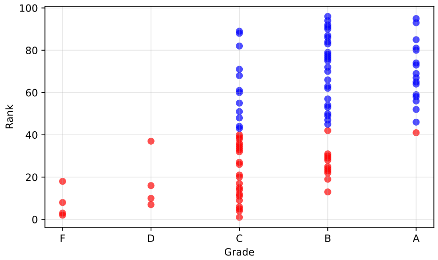


Figure 17: Relationship of grades and rankings in early prediction for C-2022-1

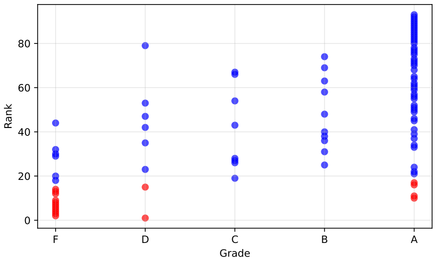


Figure 18: Relationship of grades and rankings in early prediction for D-2022

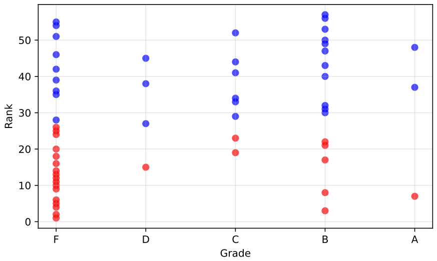


Figure 19: Relationship of grades and rankings in early prediction for E-2021

4.3 Discussion and Limitations

4.3.1 Data Source and Generalizability

Although the proposed approach has been evaluated with learning logs only from BookRoll, our method is applicable in other learning management systems. We publicly provide our modeling code (see Section 5) for others to adopt or extend to other types of learning logs. As a result, further research may adapt our methods to other systems and datasets. However, since all data in this study originates from a single institution and platform, the generalizability of the proposed method to different institutional contexts remains an open question.

4.3.2 Communication and Computational Constraints

Regarding federated learning, this study did not consider the network communication overhead or the client devices' computational constraints. However, federated learning inherently introduces challenges related to network communication costs and device capabilities. These limitations must be considered for real-world deployment, and future work should explore lightweight models and communication-efficient strategies.

4.3.3 Model Interpretability

While our results show that federated learning achieves prediction performance comparable to that of centralized learning (Section 4.2.1), the interpretability of the resulting model has not been fully explored in this study. In particular, it remains unclear which specific input features contribute the most to the risk predictions. Investigating the influence of individual features within the differential features will be an important direction for future work, especially for improving the model's transparency in practical educational settings.

4.3.4 Robustness to Non-IID Data

An important factor to consider in federated learning is whether the model can perform well under non-independent and identically distributed (non-IID) data. This study introduced differential features to address discrepancies and skew in feature distributions across clients—one aspect of non-IID data in federated learning. Notably, differential features transform both the input features and student outcome labels (i.e., grade information) into pairwise differences between students. While our method was not originally designed to address label distribution skew—another important aspect of non-IID data—it may incidentally help mitigate this issue. Although we did not evaluate this effect, we recognize it as a potential secondary benefit of using differential features and consider it an important direction for future work.

4.3.5 Assumptions Behind Differential Features

While our results show improved performance with differential features, we implicitly assume that these features mitigate discrepancies in feature distributions across clients. However, this assumption has not yet been explicitly validated for a general setting, and further investigation in other educational contexts is needed to confirm whether the observed improvements are indeed attributable to reduced inter-client variability in feature distributions.

4.4 Implications for Educational Practice

Our approach offers practical value for both instructors and students. From the instructor's perspective, the risk rankings can inform individualized support strategies, such as prioritizing interventions for high-risk students or organizing differentiated instruction. From the students' perspective, understanding their relative risk status may help them reflect on their learning behavior and take proactive steps toward improvement.

5. CONCLUSION AND FUTURE WORK

This study proposed a method that (1) applies federated learning in EDM to enable privacy-preserving prediction modeling and (2) leverages differential features to use relative values between clients, resulting in a generalizable and high-performing model. Additionally, to effectively utilize differential features, we proposed a method that scores grades, employs regression, and calculates individual prediction values from pairwise difference scores to generate risk rankings.

The evaluation of the proposed method demonstrated the following novel contributions:

- Federated learning achieves at-risk student prediction performance comparable to that of centralized ML, while benefiting from increased privacy protection.
- Introducing differential features improves the performance of at-risk student prediction compared to the baseline without differential features.
- Even when using data from only half of the lecture sessions, the proposed method achieves high performance in at-risk student prediction, which demonstrates its applicability to early prediction scenarios.

5.1 Open Research Challenges

As a result of this novel study, there are several open research challenges in exploring federated learning and differential features in EDM. Specifically, the following topics are identified as areas for future work:

1. **Developing a More Generalizable Model**
   As discussed in Section 4.3.1, the proposed approach is currently designed for learning log data from BookRoll. To develop a more generalizable and robust model, further work could investigate its applicability to different types of learning logs and educational datasets, exploring methods for adapting the model to diverse data sources and system architectures.
2. **Alternative Integration Methods in Federated Learning**
   This study employed the commonly used method called FedAvg for federated parameter integration. While FedAvg is a widely adopted baseline, other integration methods such as FedOpt [40] and SCAFFOLD [23] may offer advantages in terms of convergence speed, stability, or model performance. Future work could explore the impact of these alternative methods on the effectiveness of the proposed approach.
3. **Identifying Actions Influencing Risk Levels**
   Since the risk rankings are generated based on the learning logs, further investigation is required to identify which specific student actions contribute to lower or higher risk, as discussed in Section 4.3.3.

5.2 Availability of Research Code

To support the replicability of the findings—an aspect valued by the EDM community [16]—we have made the code used to produce the results in this paper publicly available at:

https://github.com/limu-research/2025-EDM-FL.

6. ACKNOWLEDGMENTS

This work was supported by JST CREST Grant Number JPMJCR22D1 and JSPS KAKENHI Grant Number JP22H00551, Japan.

7. REFERENCES

[1] A. Abdulraheem, R. Abdullah Arshah, and H. Qin. Evaluating the effect of dataset size on predictive model using supervised learning technique. *International Journal of Software Engineering & Computer Sciences (IJSECS)*, 1(1):75–84, 02 2015. https://doi.org/10.15282/ijsecs.1.2015.6.0006.

[2] H. Almoubayyed, S. Fancsali, and S. Ritter. Generalizing predictive models of reading ability in adaptive mathematics software. In M. Feng, T. Käser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 207–216, Bengaluru, India, July 2023. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115782.

[3] H. Altabrawee, O. Ali, and S. Qaisar. Predicting students' performance using machine learning techniques. *Journal of University of Babylon for pure and applied sciences*, 27(1):194–205, 04 2019. https://doi.org/10.29196/jubpas.v27i1.2108.

[4] R. Baker. Getting past the current trade-off between privacy and equity in educational technology. *The Economics of Equity in K-12 Education: Connecting Financial Investments with Effective Programming*, 123, 2023. https://scholar.google.com/citations?view_op=view_citation&citation_for_view=hvs8PEoAAAAJ:1Aeql8wG3wEC.

[5] R. S. Baker, S. Hutt, C. A. Brooks, N. Srivastava, and C. Mills. Open science and educational data mining: Which practices matter most? In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 279–287, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.12729816.

[6] M. K. Belaid, M. Rabus, and E. Hüllermeier. Pairwise difference learning for classification. In D. Pedreschi, A. Monreale, R. Guidotti, R. Pellungrini, and F. Naretto, editors, *Discovery Science - 27th International Conference, DS 2024, Pisa, Italy, October 14-16, 2024, Proceedings, Part II*, volume 15244, pages 284–299. Cornell University, 6 2024. https://doi.org/10.48550/arxiv.2406.20031.

[7] D. Boulanger and V. Kumar. An overview of recent developments in intelligent e-textbooks and reading analytics. In S. A. Sosnovsky, P. Brusilovsky, R. G. Baraniuk, R. Agrawal, and A. S. Lan, editors, *iTextbooks@AIED*, volume 2384, pages 44–56. CEUR-WS.org, 2019. https://api.semanticscholar.org/CorpusID:195693855.

[8] P. Brusilovsky, S. Sosnovsky, and K. Thaker. The return of intelligent textbooks. *AI Magazine*, 43(3):337–340, 8 2022. https://doi.org/10.1002/aaai.12061.

[9] C.-H. Chen, S. J. H. Yang, J.-X. Weng, H. Ogata, and C.-Y. Su. Predicting at-risk university students based

on their e-book reading behaviours by using machine learning classifiers. *Australasian Journal of Educational Technology*, 37(4):130–144, Jun. 2021. https://doi.org/10.14742/ajet.6116.

[10] J. M. Cock, M. Marras, C. Giang, and T. Käser. Generalisable methods for early prediction in interactive simulations for education. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, volume abs/2207.01457, pages 183–194, Durham, United Kingdom, July 2022. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.6852968.

[11] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. https://www.cse.unr.edu/~bebis/CS773C/ObjectRecognition/Papers/Dance04.pdf.

[12] M. Delianidi, K. I. Diamantaras, G. Chrysogonidis, and V. Nikiforidis. Student Performance Prediction Using Dynamic Neural Models. In S. I. Hsiao, S. S. Sahebi, F. Bouchet, and J. Vie, editors, *Proceedings of the 14th International Conference on Educational Data Mining*, volume abs/2106.00524, Massachusetts, USA, 6 2021. International Educational Data Mining Society. https://doi.org/10.48550/arxiv.2106.00524.

[13] A. M. Embong, A. M. Noor, H. M. Hashim, R. M. Ali, and Z. H. Shaari. E-books as textbooks in the classroom. *Procedia - Social and Behavioral Sciences*, 47:1802–1809, 2012. https://doi.org/10.1016/j.sbspro.2012.06.903.

[14] B. Flanagan and H. Ogata. Learning analytics platform in higher education in japan. *Knowledge Management and E-Learning*, 10:469–484, 11 2018. https://doi.org/10.34105/j.kmel.2018.10.029.

[15] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, and C.-Z. Xu. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10112–10121. IEEE, June 2022. https://doi.org/10.1109/cvpr52688.2022.00987.

[16] 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 M. Feng, T. Käser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 114–124, Bengaluru, India, July 2023. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115651.

[17] M. Hlosta, Z. Zdrahal, and J. Zendulka. Ouroboros: early identification of at-risk students without models based on legacy data. In M. Hatala, A. Wis, P. Winne, G. Lynch, X. Ochoa, I. Molenaar, S. Dawson, S. Shehata, and J. P.-L. Tan, editors, *Proceedings of the Seventh International Learning Analytics & Knowledge Conference*, LAK '17, page 6–15, New York, NY, USA, 2 2017. Association for Computing Machinery. https://doi.org/10.1145/3027385.3027449.

[18] M. Hoq, P. Brusilovsky, and B. Akram. Analysis of an explainable student performance prediction model in an introductory programming course. In M. Feng, T. Käser, and P. Talukdar, editors, *Proceedings of the 16th International Conference on Educational Data Mining*, pages 79–90, Bengaluru, India, July 2023. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115693.

[19] S. Hutt, R. S. Baker, M. M. Ashenafi, J. M. Andres-Bray, and C. Brooks. Controlled outputs, full data: A privacy-protecting infrastructure for mooc data. *British Journal of Educational Technology*, 53(4):756–775, 5 2022. https://doi.org/10.1111/bjet.13231.

[20] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 10 2002. https://doi.org/10.1145/582415.582418.

[21] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. Nitin Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. El Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konecný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, H. Qi, D. Ramage, R. Raskar, M. Raykova, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao. Advances and open problems in federated learning. *Found. Trends Mach. Learn.*, 14(1–2):1–210, June 2021. https://doi.org/10.1561/2200000083.

[22] S. Kanamori, T. Abe, T. Ito, K. Emura, L. Wang, S. Yamamoto, T. P. Le, K. Abe, S. Kim, R. Nojima, et al. Privacy-preserving federated learning for detecting fraudulent financial transactions in japanese banks. *Journal of Information Processing*, 30(0):789–795, 2022. https://doi.org/10.2197/ipsjjip.30.789.

[23] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh. SCAFFOLD: stochastic controlled averaging for on-device federated learning. *CoRR*, abs/1910.06378, 10 2019. http://arxiv.org/abs/1910.06378.

[24] S. Keskin and H. Yurdugül. E-learning experience: Modeling students' e-learning interactions using log data. *Journal of Educational Technology and Online Learning*, 5(1):1–13, 01 2022. https://doi.org/10.31681/jetol.938363.

[25] M. Klose, V. Desai, Y. Song, and E. F. Gehringer. Edm and privacy: Ethics and legalities of data collection, usage, and storage. In A. N. Rafferty, J. Whitehill, C. Romero, and V. Cavalli-Sforza, editors, *Educational Data Mining*. International Educational Data Mining Society, 7 2020. https://api.semanticscholar.org/CorpusID:220928650.

[26] C. Koutcheme, S. Sarsa, A. Hellas, L. Haaranen, and J. Leinonen. Methodological considerations for predicting at-risk students. In J. Sheard and P. Denny, editors, *Proceedings of the 24th Australasian Computing Education Conference*, pages 105–113. ACM, 02 2022.

https://doi.org/10.1145/3511861.3511873.

[27] S. Leelaluk, C. Tang, V. Švábenský, and A. Shimada. Knowledge distillation in rnn-attention models for early prediction of student performance. *arXiv preprint arXiv:2412.14526*, abs/2412.14526, 12 2024. https://doi.org/10.48550/arxiv.2412.14526.

[28] J. Li, S. Supraja, W. Qiu, and A. W. H. Khong. Grade prediction via prior grades and text mining on course descriptions: Course outlines and intended learning outcomes. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, pages 446–453, Durham, United Kingdom, July 2022. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.6853171.

[29] L. Li, Y. Fan, M. Tse, and K.-Y. Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 11 2020. https://doi.org/10.1016/j.cie.2020.106854.

[30] Q. Li, Y. Diao, Q. Chen, and B. He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978. IEEE, 5 2022. https://doi.org/10.1109/ICDE53745.2022.00077.

[31] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, abs/2102.07623, 2 2021. https://arxiv.org/pdf/2102.07623.

[32] Y. Liu, J. James, J. Kang, D. Niyato, and S. Zhang. Privacy-preserving traffic flow prediction: A federated learning approach. *IEEE Internet of Things Journal*, 7(8):7751–7763, 8 2020. https://doi.org/10.1109/jiot.2020.2991401.

[33] Z. Liu, X. Jiao, C. Li, and W. Xing. Fair prediction of students' summative performance changes using online learning behavior data. In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, pages 686–691, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.12729918.

[34] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In A. Singh and X. J. Zhu, editors, *Artificial intelligence and statistics*, volume 54, pages 1273–1282. PMLR, JMLR.org, 4 2017. https://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf.

[35] Y. Miyazaki, V. Švábenský, Y. Taniguchi, F. Okubo, T. Minematsu, and A. Shimada. E2vec: Feature embedding with temporal information for analyzing student actions in e-book systems. In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, volume abs/2407.13053, pages 434–442, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.12729854.

[36] H. Ogata, C. Yin, M. Oi, F. Okubo, A. Shimada, K. Kojima, and M. Yamada. e-book-based learning analytics in university education. In *Doctoral Student Consortium (DSC) - Proceedings of the 23rd International Conference on Computers in Education, ICCE 2015*, pages 401–406. Asia-Pacific Society for Computers in Education, 2015. https://mark-lab.net/wp-content/uploads/2012/06/Ogata_et_al_ICCE2015_1.pdf.

[37] M. Oldenhof, G. Ács, B. Pejó, A. Schuffenhauer, N. Holway, N. Sturm, A. Dieckmann, O. Fortmeier, E. Boniface, C. Mayer, et al. Industry-scale orchestrated federated learning for drug discovery. In B. Williams, Y. Chen, and J. Neville, editors, *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15576–15584. Association for the Advancement of Artificial Intelligence (AAAI), 6 2023. https://doi.org/10.1609/aaai.v37i13.26847.

[38] N. Ong, J. Zhu, and D. Mosse. Towards including instructor features in student grade prediction. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, pages 239–250, Durham, United Kingdom, July 2022. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.6853063.

[39] B. Radmehr, A. Singla, and T. Käser. Towards generalizable agents in text-based educational environments: A study of integrating rl with llms. In B. Paaßen and C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, volume abs/2404.18978, pages 181–193, Atlanta, Georgia, USA, July 2024. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.12729794.

[40] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan. Adaptive federated optimization. *CoRR*, abs/2003.00295, 2 2020. https://doi.org/10.48550/arxiv.2003.00295.

[41] N. Rohani, K. Gal, M. Gallagher, and A. Manataki. Early prediction of student performance in a health data science mooc. In M. Feng, T. Käser, 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. https://doi.org/10.5281/zenodo.8115721.

[42] A. Shimada and S. Konomi. A lecture supporting system based on real-time learning analytics. In *14th International Conference On Cognition And Exploratory Learning In The Digital Age (CELDA 2017)*, pages 197–204, 10 2017. https://files.eric.ed.gov/fulltext/ED579463.pdf.

[43] J. Staiger. How e-books are used: A literature review of the e-book studies conducted from 2006 to 2011. *Reference and User Services Quarterly*, 51(4):355–365, 6 2012. https://doi.org/10.5860/rusq.51n4.355.

[44] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 843–852. IEEE, 10 2017. https://doi.org/10.1109/ICCV.2017.97.

[45] T. Trask, D. N. Lytle, M. Boyle, D. D. Joyner, and D. A. Mubarak. A comparative analysis of student performance predictions in online courses using heterogeneous knowledge graphs. In B. Paaßen and

C. D. Epp, editors, *Proceedings of the 17th International Conference on Educational Data Mining*, volume abs/2407.12153, pages 915–920. International Educational Data Mining Society, July 2024. https://doi.org/10.5281/zenodo.12729997.

[46] M. Tynes, W. Gao, D. J. Burrill, E. R. Batista, D. Perez, P. Yang, and N. Lubbers. Pairwise difference regression: A machine learning meta-algorithm for improved prediction and uncertainty quantification in chemical search. *Journal of Chemical Information and Modeling*, 61(8):3846–3857, 8 2021. https://doi.org/10.1021/acs.jcim.1c00670.

[47] M. van Haastrecht, M. Brinkhuis, and M. Spruit. Federated learning analytics: Investigating the privacy-performance trade-off in machine learning for educational analytics. In A. M. Olney, I.-A. Chounta, Z. Liu, O. C. Santos, and I. I. Bittencourt, editors, *International Conference on Artificial Intelligence in Education*, volume 14830, pages 62–74. Springer, Springer Science+Business Media, 2024. https://doi.org/10.1007/978-3-031-64299-9_5.

[48] V. Švábenský, R. 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*, volume abs/2307.07614, pages 302–309, Bengaluru, India, July 2023. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115790.

[49] V. Švábenský, K. Tkáčik, A. Birdwell, R. Weiss, R. S. Baker, P. Čeleda, J. Vykopal, J. Mache, and A. Chattopadhyay. Detecting Unsuccessful Students in Cybersecurity Exercises in Two Different Learning Environments. In *Proceedings of the 54th Frontiers in Education Conference*, volume abs/2408.08531 of *FIE '24*, New York, NY, USA, 2024. IEEE. https://doi.org/10.1109/FIE61694.2024.10893135.

[50] K. D. Wang, J. M. Cock, T. Käser, and E. Bumbacher. A systematic review of empirical studies using log data from open-ended learning environments to measure science and engineering practices. *British Journal of Educational Technology*, 54(1):192–221, 1 2023. https://doi.org/10.1111/bjet.13289.

[51] Q. Zheng, H. He, T. Ma, N. Xue, B. Li, and B. Dong. Big log analysis for e-learning ecosystem. In Y. Li, X. Fei, K.-M. Chao, and J.-Y. Chung, editors, *2014 IEEE 11th International Conference on e-Business Engineering*, pages 258–263. IEEE, 11 2014. https://doi.org/10.1109/ICEBE.2014.51.

[52] H. Zhu, J. Xu, S. Liu, and Y. Jin. Federated learning on non-iid data: A survey. *Neurocomputing*, 465:371–390, 9 2021. https://doi.org/10.1016/j.neucom.2021.07.098.