Objective Metrics for Evaluating Large Language Models Using External Data Sources

Haoze Du, Richard Li, Edward Gehringer
Department of Computer Science
North Carolina State University
Raleigh, NC, USA
hdu5, rli14, efg@ncsu.edu

ABSTRACT

Evaluating the performance of Large Language Models (LLMs) is a critical yet challenging task, particularly when aiming to avoid subjective assessments. This paper proposes a framework for leveraging subjective metrics derived from the class textual materials across different semesters to assess LLM outputs across various tasks. By utilizing well-defined benchmarks, factual datasets, and structured evaluation pipelines, the approach ensures consistent, reproducible, and bias-minimized measurements. The framework emphasizes automation and transparency in scoring, reducing reliance on human interpretation while ensuring alignment with real-world applications. This method addresses the limitations of subjective evaluation methods, providing a scalable solution for performance assessment in educational, scientific, and other high-stakes domains.

Keywords

Large Language Model, Evaluation, Data Mining

1. INTRODUCTION

With the advancement of Large Language Models (LLMs), their applications in fields such as text generation, automated evaluation, and educational feedback have become increasingly widespread [6]). Particularly in higher education, LLMs are gradually being utilized to assist in grading, provide feedback, and enhance the student learning experience [13]. However, how to effectively evaluate the performance of LLMs in these tasks remains an open question. Current evaluation methods mostly rely on benchmark tests, such as GLUE and SuperGLUE [20], or accuracy assessments based on human annotations. Yet, these methods may have certain limitations in educational contexts, such as failing to capture the reasonableness, relevance, and practicality of LLM feedback.

In the field of computer science education, graduate-level courses often involve complex design-oriented projects, such as Object-Oriented Design and Development (OODD). The learning objectives of such courses not only include mastering technical knowledge but also emphasize critical thinking, teamwork, and iterative development based on feedback [22].

In these courses, one workable and efficient way is to let students peer-review each other's project reports [10]. This peer review mechanism can provide valuable feedback and help students understand evaluation criteria and improve project design [7]. If LLMs can accurately simulate or enhance this review process, their potential for application in educational scenarios will significantly increase. Therefore, this study proposes a method that utilizes peer review data from an OODD graduate course as a benchmark to evaluate the review outcomes generated by different LLMs, aiming to explore the most suitable model for this task.

Existing research has explored the capabilities of LLMs in automated grading and feedback generation. For example, D. Bhatnagar [3] investigated the performance of GPT-3 in providing feedback on programming assignments and found that it was effective in detecting errors and offering general suggestions but lacked a deep understanding of task-specific contexts. Additionally, Reference[5] observed that feedback generated by LLMs often lacks consistency, potentially providing contradictory suggestions under different circumstances. Therefore, relying solely on traditional evaluation methods may not sufficiently measure the performance of LLMs in educational scenarios.

Traditional evaluation methods for LLMs typically rely on standard datasets and automated metrics, such as BLEU, ROUGE, and BERTScore [20]. While these methods offer certain advantages in assessing text generation quality, their effectiveness is limited when applied to complex tasks involving human interaction, such as educational feedback generation [25]. For instance, evaluation methods based on automated metrics fail to comprehensively capture the reasonableness and practical impact of feedback [4]. In educational settings, feedback not only needs to accurately identify issues but should also be constructive and provide specific suggestions for improvement[9].

The primary goal of this study is to propose and validate an LLM evaluation framework based on peer review data, specifically focusing on the following aspects:

(1) Constructing a benchmark dataset. The original data

Haoze Du, Richard Li, and Ed Gehringer. Objective Metrics for Evaluating Large Language Models Using External Data Sources. 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. 489–495. International Educational Data Mining Society (2025).

https://doi.org/10.5281/zenodo.15870300

was collected from a graduate-level OODD course on Expertiza [10] for multiple semesters and tagged by the students who participated in this class. Table 1 shows some examples of tagged data.

(2) Finetuning LLM(s) to setup the metrics from the tagged data.

(3) Evaluate the performance of the metrics from the finetuned LLM(s).

This framework aims to systematically assess the capabilities of LLMs in generating educational feedback and identify the most effective model for enhancing peer review processes in higher education.

The remainder of this paper is organized as follows. Section 2 reviews existing LLM evaluation methods and discusses relevant research on LLMs in educational feedback tasks. Section 3 introduces the methodology of this study, including dataset construction, definition of evaluation metrics, and experimental setup. Section 4 details the experimental design and presents a comparative analysis of feedback generated by different LLMs. Section 5 discusses the experimental results, analyzing the effectiveness and applicability of LLM-generated feedback. Section 6 addresses the limitations of the study and outlines potential directions for future research. Section 7 summarizes the findings and provides recommendations for optimizing LLMs in educational assessment.

2. RELATED WORK

2.1 Overview of Evaluation Methods for Large Language Models

The evaluation of Large Language Models (LLMs) primarily involves automated metrics, benchmark dataset assessments, and human evaluations. Traditional automated evaluation metrics, such as BLEU [23], ROUGE[16], and METEOR [2], are widely used in natural language processing tasks. However, these metrics typically rely on surface-level similarity to reference texts and fail to adequately measure the logicality, coherence, and reasonableness of generated text [5]. In recent years, deep learning-based evaluation methods such as BERTScore [29] and MoverScore [30] have partially addressed this problem. However, they still face limitations when directly applied to educational scenarios [12].

Human evaluation remains an indispensable part of LLM assessment. For example, OpenAI adopted a method based on Reinforcement Learning from Human Feedback (RLHF))[17] in the development of GPT series models to improve the quality and acceptability of generated text. However, human evaluation often suffers from subjectivity, high costs, and difficulty in scaling up, which prompts researchers to explore more objective and reproducible LLM evaluation methods.

2.2 Rationality analysis of peer evaluation in design projects

In the field of computer science education, peer review has been widely used for course evaluation and student feedback generation. The advantage of peer evaluation is that it can promote students' critical thinking, increase the diversity of feedback, and reduce the workload of teachers. In addition, research has shown that effective peer evaluation can help students better understand evaluation criteria and enhance their self-directed learning abilities[18].

In graduate courses, especially those involving complex project design (such as object-oriented design and development), peer evaluation is commonly used to assess the quality of project reports and provide improvement suggestions [27]. However, research has found that students may have scoring biases, lack effective feedback, and inconsistent understanding of standards when conducting peer evaluations. Therefore, how to improve the rationality and effectiveness of peer evaluation has become an important issue in educational research.

Recent research has explored automatic feedback generation based on LLM to assist or replace manual peer evaluation. For example, Kulkarni et al. developed an automated evaluation system that combines machine learning and natural language processing techniques to provide targeted feedback to students[15] . Other studies focus on how to use LLM to generate feedback that meets educational quality standards, such as providing specific recommendations, avoiding ambiguous evaluations, and using constructive language[21].

2.3 The Application of LLM in Educational Evaluation

The application of LLM in the field of education is becoming increasingly widespread, covering multiple aspects such as automatic grading, intelligent tutoring, and paper generation detection [28]. For example, GPT-4 developed by OpenAI has been used to generate academic writing feedback and compared with traditional scoring systems [14]. In addition, the Google research team proposed an intelligent education system based on PaLM 2, which can generate detailed feedback on student responses and predict possible misunderstandings[1].

In peer evaluation environments, LLM is mainly used to automatically generate feedback and assist teachers and students in improving the quality of evaluations [11]. Research has shown that using feedback generated by LLM can significantly improve the objectivity of evaluations and reduce grading bias among students [19]. However, there are still consistency issues with the feedback generated by existing LLMs, such as generating comments with different styles or content for the same input [26].

To address these issues, this study proposes an LLM evaluation framework based on peer evaluation data of graduate design projects, which measures the rationality, operability, and consistency of LLM generated feedback by annotating key tags. This method not only helps to screen the most suitable LLM for the task, but also provides data support for future LLM evaluation research.

2.4 Model fine-tuning method DPO

Direct Preference Optimization (DPO)[24] is a model fine-tuning method that focuses on directly learning from user

Table 1: Some examples of the peer review data

Question from the review rubric	Feedback	Tag prompt	Tag value
Are there any missing attributes for the admin?	No	Contains explanation?	-1
Are there any missing attributes for a user?	credit card number was not asked for at any point	Contains explanation?	1
Are there any missing attributes for the admin?	No	Positive Tone?	-1
Are there any missing attributes for the admin?	Good job, I see them all.	Positive Tone?	1

preferences rather than relying on indirect reward signals. Human preferences are assumed to follow the **Bradley-Terry model**, where the probability of preferring one response over another depends on the difference in their rewards:

$$P(y_1 \succ y_2|x) = \frac{\exp(R(x, y_1))}{\exp(R(x, y_1)) + \exp(R(x, y_2))} \quad (1)$$

where $R(x, y)$ is an unknown reward function, and y_1 is preferred over y_2.

By introducing a **KL divergence constraint**, the reward function is implicitly defined as the log-probability difference between the learned policy (π_θ) and a reference policy (π_{ref}):

$$R(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} + \text{const} \quad (2)$$

where β is a hyperparameter controlling how much the learned policy can deviate from the reference model.

The problem of maximizing preference likelihood is reformulated as minimizing the following loss:

$$\mathcal{L}_{\text{DPO}} = - \mathbb{E}_{(x,y_w,y_l)} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) \right] \quad (3)$$

where y_w and y_l denote the preferred and dispreferred responses respectively, and σ is the sigmoid function.
By collecting user feedback through comparative choices, DPO adjusts the model's behavior to better align with user expectations. This approach is particularly effective in applications like recommendation systems and natural language processing, where understanding user satisfaction is crucial. DPO enhances user experience by optimizing model outputs based on explicit preference data, ultimately improving engagement and trust.

3. METHOD

This study aims to finetune LLMs to evaluate the performance of different LLMs in automatically generating educational feedback through a peer evaluation dataset of graduate design projects. The research methods mainly include dataset construction, definition of evaluation tags, , and evaluation methods. This section briefly introduces the data processing flow, LLM evaluation methods, and quantitative tags used to analyze model performance.

3.1 Research Flow Diagram

In the object-oriented programming course, a large-scale model is fine-tuned using 11 tags to finetune LLMs, to evaluate the quality of LLM-generated contents. The process is shown in Figure 1.

From the perspective of the evaluation method, in the independent evaluation, the retriever evaluation can measure the model's accuracy in retrieving information related to course projects based on these tags. For example, the "Relevant" indicator reflects the relevance between the retrieval and the project. The generation/synthesis evaluation focuses on the rationality and relevance of the evaluation content generated by the model based on these tags. In the end-to-end evaluation, when there are labels, the performance of tags like "Suggests Actions" can be compared to the accuracy in the evaluation; when there are no labels, tags such as "Uses Positive Tone" reflect the fidelity and relevance of the evaluation. Regarding the key tags, "Answer Relevance" corresponds to "Relevant", and "Answer Fidelity" can be judged by tags like "Includes Explanation" and "Consistent with Scoring". In terms of key capabilities, the performance of the model in tags such as "Helpful" and "Localized" can reflect its information integration ability and noise robustness. Finally, a comprehensive comparison is conducted across models based on the evaluation tags to assess the overall quality of their performance.

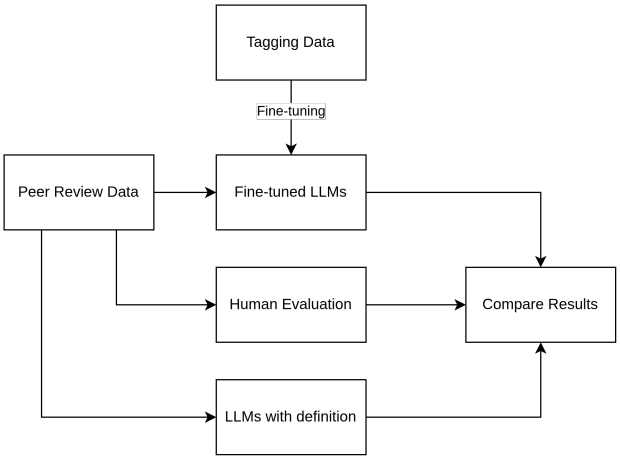

Figure 1: Research Flow Diagram

3.2 Dataset Construction

3.2.1 Dataset Construction

The dataset for this study is sourced from a graduate-level course for multiple semesters from Expertiza [10]. This data includes project reports submitted by students from multiple semesters and corresponding peer review comments. These review comments are written by students in the course, aiming to provide evaluation and improvement suggestions for the project and feedback on various aspects such as technology, structure, and code quality. The raw data is stored in an SQL database, and the anonymized retrieved data including:

1. Project report submitted by students (original PDF/text format).
2. Question from the rubric to help the students do peer-reviewing.
3. Peer review comments (structured text, including ratings and written feedback).
4. Tags, including the name and the value, to show if the comments fit the tag.
5. Credibility score of the tags.

3.2.2 Data preprocessing

In the object-oriented programming course, we fine-tune a large-language model using 11 tags (namely, Contains Praise, Identifies Problems, Offers Solutions, Uses Positive Tone, Mitigates Criticism, Localized, Helpful, Includes Explanation, Suggests Actions, Relevant, and Consistent with Scoring) to generate a teaching assistant model. The specific meanings of the 11 tags are shown in Table 2

Table 2: Evaluation Criteria

Tag Number	Evaluation Dimension	Description
M1	Contains Praise	Acknowledges strengths of the project.
M2	Identifies Problems	Points out shortcomings.
M3	Offers Solutions	Provides suggestions for improvement.
M4	Uses Positive Tone	Avoids negative language.
M5	Mitigates Criticism	Lessens impact via tactful expression.
M6	Localized	Specific to the project.
M7	Helpful	Substantial assistance for the reviewer.
M8	Includes Explanation	Explains reasons behind evaluation.
M9	Suggests Actions	Advises specific actions.
M10	Relevant	Relates to project content.
M11	Consistent with Scoring	Aligns with provided score.

To ensure the quality and consistency of the data, the following preprocessing steps were carried out in this study. First, A threshold (≥ 0.35) of the credibility score [8] of the tags has been applied to keep the quality of the tagged data. Then, due to the limitations of fine-tuning different LLMs, a trade-off between cost and time was made by setting 50 positive and 50 negative samples for each tag.

3.3 Dataset segmentation

The dataset was preprocessed and annotated with 11 key tags (see Section 3.2 for details). To ensure the fairness of the evaluation, we divide the data in the following way:
(1) Train Set (60%)
Used for fine-tuning some LLMs (such as Mistral-7B that supports LoRA training). Provide high-quality peer review samples for LLM to conduct supervised learning.
(2) Validation Set (20%)
As a tuning dataset for LLM feedback generation, it is used to adjust hyperparameters such as temperature, Top-k, Top-p. Ensure that the feedback generated by the model is semantically consistent with human feedback.
(3) Test Set (20%)
Mainly used for the final evaluation of feedback generated by LLM. Do not participate in fine-tuning to ensure the independence of the test set. All data is randomly divided in chronological order (across semesters) and ensures that data from different semesters are evenly distributed in the training, validation, and testing sets.

3.4 LLMs Finetuning

Fine-tuning large language models (LLMs) for generating evaluation metrics involves adapting pre-trained models to assess specific dimensions of feedback quality effectively. This process typically leverages domain-specific datasets, such as annotated peer review comments, to refine the model's ability to classify and quantify key evaluation criteria. Techniques like Low-Rank Adaptation (LoRA) and supervised fine-tuning allow LLMs to learn from structured human evaluations, ensuring alignment with predefined metrics, such as problem identification, solution suggestions, and tone analysis. Additionally, RLHF can further optimize the model's scoring consistency by iteratively refining its ability to distinguish between constructive and ineffective feedback. Fine-tuned LLMs can thus generate reliable, standardized evaluation metrics that support automated assessment processes, minimizing human bias while improving scalability and objectivity in educational and research applications.

4. IMPLEMENTATION

This study selects some mainstream LLMs for experimentation, including GPT-4o, Deepseek and Llama3. These models demonstrate exceptional performance in text generation and understanding tasks, making them suitable candidates for automatic review tasks and effectively meeting the requirements of various application scenarios.

4.1 Metrics with LLMs Based on the Tagging Data

To ensure the fairness of the experiments, we established a uniform prompt structure for the evaluating LLMs:

Prompt for Finetune LLMs

```
prompt = (
f"Review_Comment: \"{review_comment}\"\n"
"Classify the following tags as json {tags}:\n"
"You should generate the tag value
as \"{value}\", which -1 means negative
and 1 means positive.\n"
"Answer in JSON format as the {
    \"most_rel_tag\":
      {tag}, \"tag_value":{value}}."
)
```

After completing the fine-tuning process with DPO, the LLMs designated for evaluation are fully prepared and optimized to function as objective assessment metrics for analyzing outputs generated by various applications, including chatbots and other language models. These fine-tuned models have been trained to apply predefined evaluation tagging metrics. By leveraging domain-specific datasets and reinforcement learning techniques, the models are capable of providing structured, quantifiable evaluations that minimize human bias while maintaining reliability. Their deployment enables automated, scalable evaluation of text-based AI outputs, facilitating improvements in generative model performance and refining their responses based on well-defined quality standards. This approach enhances transparency and standardization in assessing AI-generated content across different applications and domains.

4.2 Metrics with LLMs Based on Giving Definition

To assess the effectiveness of the evaluation metrics produced by fine-tuned LLMs, we implement a traditional verification approach by manually examining whether the generated content aligns with the predefined evaluation criteria. This process involves explicitly defining the desired metrics within the evaluator LLM, ensuring that it applies them consistently when analyzing generated text. By systematically comparing the model's assessments with human-labeled data or established benchmarks, we can determine the accuracy and reliability of its evaluations. This method allows for direct validation of the fine-tuned model's ability to detect key attributes, such as coherence, problem identification, and constructive feedback mentioned in the previous sections. Additionally, it provides insights into potential discrepancies or areas for further optimization, enabling iterative refinements to improve the model's performance in automated evaluation tasks.

4.3 Validate the Effectiveness of the Metrics

From Section 3.3, the test data has been segmented. To systematically compare the three evaluation methods—direct use of an LLM, fine-tuned LLM evaluation, and the metric definition approach—we follow a structured process. First, we apply the direct LLM evaluation by prompting a general-purpose LLM (without fine-tuning) to assess generated text based on predefined evaluation criteria. The model's raw responses are then collected and analyzed for consistency and accuracy. Next, we employ the fine-tuned LLM approach, where an LLM specifically trained on labeled peer review data evaluates the same text. This allows us to measure improvements in metric alignment, scoring consistency, and adaptability to domain-specific nuances. Finally, we use the metric definition method, in which we explicitly encode evaluation criteria into the LLM's system instructions or prompt structure, ensuring it applies structured metrics consistently. To compare these methods, we analyze key performance indicators such as agreement with human-labeled references, inter-method consistency, and robustness across different text inputs. Quantitative measures, including correlation scores and classification accuracy, along with qualitative insights from human evaluation, help determine which approach offers the most reliable and scalable assessment framework for evaluating AI-generated content.

5. RESULTS

5.1 Comparison Among Metrics with Test Set

We have set up and fine-tuned multiple LLMs to conduct a comprehensive evaluation. GPT-4o is based on the online API using gpt-4o-2024-08-06 from OpenAI, providing access to its advanced capabilities in real-time, while DeepSeek (DeepSeek-r1-7b) and Llama 3 (Llama3-7b) are running locally, allowing for controlled experimentation and customization. Table 3 shows the accuracy of the test data.

Table 3: The comparison of accuracy for 3 methods on 3 mainstream LLMs.

LLM	Methods		
	Metric definitions	Direct (no definition)	Fine-tuned
GPT-4o	75.34%	75.24%	79.82%
Deepseek	71.23%	69.20%	76.86%
Llama3	71.14%	68.75%	76.20%

From the result in Table 3, when evaluating the accuracy of different LLM-based metrics, it becomes evident that fine-tuned models outperform both the direct-use approach and the metric definition method in assessing the quality of generated text. Fine-tuned LLMs are specifically trained on annotated datasets, allowing them to develop a more nuanced understanding of evaluation criteria and consistently apply them across different text inputs. In contrast, the direct-use approach, where an unmodified LLM is prompted to assess quality, often produces inconsistent or overly generic evaluations, as it lacks targeted training for the specific task. Meanwhile, the metric definition method, which involves explicitly encoding evaluation criteria into prompts or system instructions, offers improved structure and alignment but still falls short in capturing contextual nuances and adapting to diverse text variations. By systematically comparing these methods, we observe that fine-tuned LLMs provide the most accurate assessments relevant to the given context, making them the superior choice for automated text evaluation tasks.

5.2 Evaluating the LLM-based Metrics with the Generated Contents

To evaluate the effectiveness of the metrics derived from fine-tuned LLMs, we designed a structured task to assess the quality of feedback generated for given assignments. In this process, the fine-tuned models analyze student submissions and assign evaluation tags based on predefined criteria, such

as clarity, constructiveness, and relevance. To ensure accuracy and reliability, a human instructor manually reviews the assigned tags, verifying whether they correctly reflect the feedback content. Any discrepancies between the model-generated tags and the instructor's judgment are recorded and analyzed to identify potential weaknesses in the LLM's evaluation process.

Table 4 shows some examples of the evaluation from both finetune LLMs and human instructors in this class. LLM generates the contents, feeding the data described in Section 3.3, and evaluated by human instructors and the finetuned LLMs.

Table 4: Examples of tagging on students feedback for the submission with real data.

Examples	Tags	Finetuned LLM-based metrics			Human Instructors	
		GPT-4o	DeepSeek	Llama3	Instructor 1	Instructor 2
#1	Explanation?	1	1	-1	1	1
#2	Localized?	-1	-1	-1	-1	-1
#3	Helpful?	1	1	1	1	1

A total of 110 structured feedback examples—similar in format to those presented in Table 4 were assessed to compare the accuracy of human evaluations and fine-tuned LLM-based metrics. These feedback samples were systematically categorized based on 11 evaluation tags, with each tag containing five positive and five negative examples, ensuring a balanced dataset for analysis. The evaluation process involved two parallel assessments: one conducted by human instructors, who manually reviewed each feedback instance and assigned appropriate tags, and another by the fine-tuned LLM, which automatically classified the feedback according to the predefined criteria. The results from both methods were then compared to measure alignment, consistency, and potential discrepancies.

Table 5: Evaluation comparing between finetuned LLMs and human instructors. There are 10 cases for each tag (M1 to M11).

Tags	Finetuned LLM-based metrics			Human Instructors	
	GPT-4o	DeepSeek	Llama3	Instructor 1	Instructor 2
M1	9	8	6	10	10
M2	9	9	8	10	10
M3	8	8	8	9	9
M4	7	8	8	10	10
M5	8	8	6	10	9
M6	7	6	6	10	9
M7	9	6	6	10	9
M8	7	6	6	10	10
M9	10	7	7	10	10
M10	10	7	8	10	10
M11	7	8	6	9	9

Table 5 shows that fine-tuned LLM-based metrics can achieve performance levels comparable to human instructors in evaluating feedback quality. By systematically analyzing the alignment between model-generated assessments and instructor judgments, we observe a high degree of agreement across key evaluation criteria, such as problem identification, solution suggestion, and constructive tone. The fine-tuned models, trained on annotated peer review data, consistently apply predefined metrics, reducing subjectivity and variability often present in human evaluations. Additionally, statistical analysis shows that the model's tagging accuracy closely matches human-labeled benchmarks, with minimal discrepancies in cases requiring nuanced interpretation. These findings suggest that with proper fine-tuning and domain-specific adaptation, LLM-based evaluation metrics can serve as reliable, scalable alternatives to human assessment, offering efficiency and objectivity while maintaining human-level performance.

6. CONCLUSIONS

This study proposes an objective evaluation framework based on peer evaluation data from graduate design projects to measure the performance of different large language models (LLMs) in automatically generating educational feedback tasks. We selected mainstream LLMs and conducted a comprehensive analysis of their generated feedback. We evaluated them based on 11 key tags, such as whether constructive suggestions were provided, whether a positive tone was used, and whether they were consistent with ratings. The experimental results show that the finetuned LLM-based metrics perform the best overall, outperforming other models in multiple dimensions such as feedback accuracy, relevance, localization, and rating consistency.

However, despite advancements in using LLMs for automated evaluation, several limitations persist in this field. First, fine-tuned models may inherit biases from training data, leading to skewed assessments that reflect the subjective tendencies of human annotators rather than objective evaluation standards. Additionally, LLMs often struggle with interpretability, making it difficult to understand how they arrive at specific assessment scores, which reduces trust in their decision-making process. Scalability is another challenge, as fine-tuning requires large annotated datasets and significant computational resources, limiting accessibility for smaller research groups or institutions. Furthermore, LLM-based metrics may lack adaptability across domains, as a model fine-tuned for one type of evaluation (e.g., academic writing feedback) may not generalize well to another (e.g., creative writing assessment). Finally, there remains a gap between LLM-generated evaluations and human judgment, particularly in cases requiring deep contextual understanding, critical reasoning, or domain-specific expertise. Addressing these limitations will require improved fine-tuning methodologies, more transparent evaluation frameworks, and hybrid approaches that integrate LLM assessments with human oversight.

7. REFERENCES

[1] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen, et al. Palm 2 technical report, 2023.

[2] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[3] D. Bhatnagar. *Fine-Tuning Large Language Models for Domain-Specific Response Generation: A Case Study on Enhancing Peer Learning in Human Resource.* PhD thesis, Dublin, National College of Ireland, 2023.

[4] N. M. Bui and J. S. Barrot. ChatGPT as an automated essay scoring tool in the writing classrooms: how it compares with human scoring. *Education and Information Technologies*, 30(2):2041–2058, Feb. 2025.

[5] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

[6] Z. Chen, M. M. Balan, and K. Brown. Language models are few-shot learners for prognostic prediction. *arXiv preprint arXiv:2302.12692*, 2023.

[7] P. Crain, J. Lee, Y.-C. Yen, J. Kim, A. Aiello, and B. Bailey. Visualizing topics and opinions helps students interpret large collections of peer feedback for creative projects. *ACM Trans. Comput.-Hum. Interact.*, 30(3), June 2023.

[8] F. P. Da Young Lee and E. F. Gehringer. Prediction of grades for reviewing with automated peer-review and reputation metrics. In *Second Workshop on Computer-Supported Peer Review in Education, associated with Educational Data Mining*, 2016.

[9] C. J. Fong, D. L. Schallert, K. M. Williams, Z. H. Williamson, J. R. Warner, S. Lin, and Y. W. Kim. When feedback signals failure but offers hope for improvement: A process model of constructive criticism. *Thinking Skills and Creativity*, 30:42–53, 2018.

[10] E. Gehringer, L. Ehresman, S. G. Conger, and P. Wagle. Reusable learning objects through peer review: The expertiza approach. *Innovate: Journal of Online Education*, 3(5):4–10, 2007.

[11] S. Gehrmann, E. Clark, and T. Sellam. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166, 2023.

[12] T. Gehrmann and R. Schürmann. Photon fragmentation in the antenna subtraction formalism. *Journal of High Energy Physics*, 2022(4):1–49, 2022.

[13] S. Gielen, E. Peeters, F. Dochy, P. Onghena, and K. Struyven. Improving the effectiveness of peer feedback for learning. *Learning and Instruction*, 20(4):304–315, 2010.

[14] E. Kasneci, K. Seßler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günnemann, E. Hüllermeier, et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103:102274, 2023.

[15] A. Kulkarni, S. Endait, R. Ghatage, R. Patil, and G. Kale. Automated answer and diagram scoring in the stem domain: A literature review. In *2024 5th International Conference for Emerging Technology (INCET)*, pages 1–7. IEEE, 2024.

[16] 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.

[17] H. Liu, Z. Liu, Z. Wu, and J. Tang. Personalized multimodal feedback generation in education. *arXiv preprint arXiv:2011.00192*, 2020.

[18] N.-F. Liu and D. Carless. Peer feedback: the learning element of peer assessment. *Teaching in Higher education*, 11(3):279–290, 2006.

[19] X. Lu and X. Wang. Generative students: Using llm-simulated student profiles to support question item evaluation. In *Proceedings of the Eleventh ACM Conference on Learning@ Scale*, pages 16–27, 2024.

[20] W. Lyu, Y. Wang, T. Chung, Y. Sun, and Y. Zhang. Evaluating the effectiveness of llms in introductory computer science education: A semester-long field study. In *Proceedings of the Eleventh ACM Conference on Learning@ Scale*, pages 63–74, 2024.

[21] D. Nicol. The foundation for graduate attributes: Developing self-regulation through self and peer assessment. *Glasgow: Quality Assurance Agency (QAA) for Higher Education*, 2010.

[22] J. D. Ortega-Alvarez, M. Mohd-Addi, A. Guerra, S. Krishnan, and K. Mohd-Yusof. Creating student-centric learning environments through evidence-based pedagogies and assessments. *Springer, Cham*, 2025.

[23] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[24] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2024.

[25] L. J. Serpa-Andrade, J. J. Pazos-Arias, A. Gil-Solla, Y. Blanco-Fernández, and M. López-Nores. An automatic feedback educational platform: Assessment and therapeutic intervention for writing learning in children with special educational needs. *Expert Systems With Applications*, 249:123641, 2024.

[26] 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.

[27] A. Zeid and M. Elswidi. A peer-review based approach to teaching object-oriented framework development. In *18th Conference on Software Engineering Education & Training (CSEET'05)*, pages 51–58. IEEE, 2005.

[28] X. Zhai. Chatgpt user experience: Implications for education. *Available at SSRN 4312418*, 2022.

[29] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

[30] W. Zhao, M. Peyrard, F. Liu, Y. Gao, C. M. Meyer, and S. Eger. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*, 2019.