A Comparative Study of Automated Grading on Programming Assignments Based on LLMs

Wu-Ja, Lin
Computer Science and Information Engineering
National Formosa University
Hu-Wei township, Yunlin county, Taiwan.
wjlin@nfu.edu.tw

Hui-Lin, Weng
Computer Science and Information Engineering
National Formosa University
Hu-Wei township, Yunlin county, Taiwan.
lin921105@gmail.com

Yu-Guang, Zhang
Computer Science and Information Engineering
National Formosa University
Hu-Wei township, Yunlin county, Taiwan.
41143129@nfu.edu.tw

***Abstract*—Programming is a required course for undergraduate students in many fields, and instructors face the challenge of assessing a large amount of assignments each week, making it difficult to provide timely feedback to students. To address this issue, numerous studies have proposed to grade programming assignments in an automatic way. This paper applies three widely used large language models, Gemini, Mistral, and ChatGPT, to grade programming assignments and evaluates their performances. The results not only show the strengths/weakness of applying LLM, compared to traditional approaches, in automatic grading, but also presents the differences among the LLMs.**

***Index Terms*—automatic grading, large language models (LLM), programming assignments, Gemini, Mistral, ChatGPT, timly feedback.**

I. Introduction

With the rise of remote education and massive open online courses, traditional methods of grading programming assignments face many challenges. The manual grading and feedback process in the past is insufficient to handle courses with a large number of students enrolled. To reduce teacher workload and maintain consistency in grading assignments, various automated tools have been developed [1]- [5]. Since last year, large language models (LLMs) have been shown their potentials in automating grading and feedback generation. This study aims to explore the application of LLMs (Gemini, Mistral, and ChatGPT) in programming assignment grading and compare their grading consistency with human. By setting grading standards in line with those of teachers, this research analyzes these LLMs in terms of grading, code improvement, feedback generation time, pricing, and more, highlighting the advantages and disadvantages of each model in automated grading.

II. Methodology

Large Language Models (LLMs): Gemini(gemini-2.0-flashexp), Mistral(mistral-large-latest), ChatGPT-1(gpt-3.5-turbo), and ChatGPT-2(GPT-4o-mini) are used to automatically grade Java Programming assignment and provide revision recommendation feedback in this paper. These LLMs are instructed to grade assignments based on the following criterias: correctness, proper input prompt, error handling, logic clarity, program readability, and execution efficiency. The grades generated by LLMs are compared with those given by human and the cost of LLM assessments are also included in this study. In addition to quantitive evaluation (e.g., grade generated and cost), a qualitative evaluation of applying LLMs on assessment is also provided in this paper.

III. Evaluation and Results

The results shown in Figure 1 and Table I are assessments of Java programming assignments of 48 students. Figure 1 shows the number of students in various score intervals, within which scores are generated by LLMs and human. The results indicate that Gemini's scoring range is narrower and more stringent, whereas Mistral, ChatGPT-1, and ChatGPT-2 exhibit wider scoring ranges. However, Mistral tends to assign fewer high scores, while ChatGPT-1 and ChatGPT-2 generally provide relatively higher scores. Table I presents the mean and standard deviation of scores given by LLMs and human. Figure 2 illustrates the distribution of the (average PR, PR standard deviation) of each student. It shows that the PR values for the same student given by various models are quite different. The total number of input tokens, output tokens generated by various LLMs, and the cost associated with various LLMs are listed in Table II and Figure 3. The Germini generates more output tokens and the cost is low when compared with other LLMs. A qualitative evaluation of applying LLMs on assessments is given in Table III for readers' reference. The evaluations in Table III is made based on the revision recommendation feedbacks generated by these LLMs. In summary, Gemini applies stricter grading criteria, emphasizing comments and naming conventions of the programs. Mistral is more lenient in error checking, while the ChatGPT series

maintains a more balanced approach. The feedback generation time required by various LLMs are listed in Table IV. It shows that the feedback generation time and output token count do not exhibit a linear relationship (Figure 3, Table II, and Table IV). Gemini generates feedback quickly with a high token count, demonstrating high reasoning efficiency. Mistral produces fewer tokens but at a slower speed, possibly due to deeper reasoning processes. ChatGPT-2 exhibits high variability, indicating strong adaptability, whereas ChatGPT-1 is more stable but occasionally experiences delays. The pricing varies based on the number of input and output tokens as well as the pricing strategies. Among them, Mistral has the highest cost, while ChatGPT2 has the lowest. Germini provides a good balance on the number of output tokens and the pricing cost. For other details, please refer to Table II. Overall, Gemini is well-suited for quickly generating comprehensive feedback, Mistral excels at concise and in-depth analysis, and the ChatGPT series offers flexibility. For programming assignments assessment, users could choose the LLM which best fits their requirements.

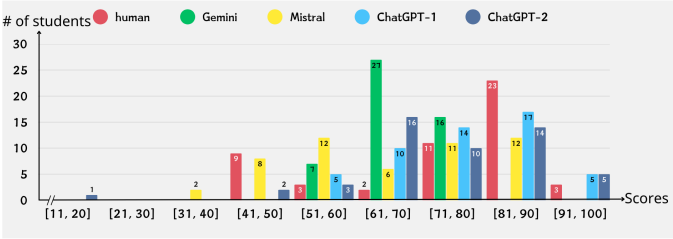

Fig. 1: Output score histogram of various LLMs.

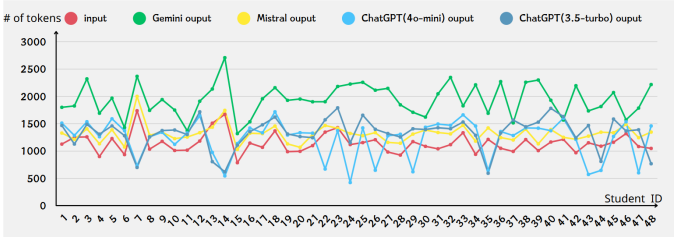

Fig. 2: Output token counts of various LLMs for the same programming assignment.

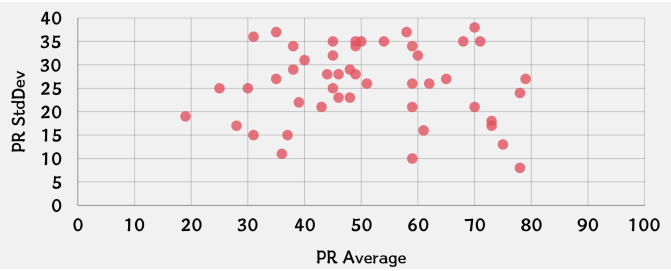

Fig. 3: Scatter Plot of the Mean and Standard Deviation of Scores from Four Models for 48 Students.

TABLE I: average and standard deviation of the scores given by LLMs and human.

	human	Gemini	Mistral	ChatGPT-1	ChatGPT-2
avg. score	75.5	67.0	66.9	78.5	75.4
std. dev.	16.0	6.3	15.3	11.0	14.3

TABLE II: Estimated cost of various LLMs

LLM	Total number of input tokens	Total number of output tokens	Total Cost
Gemini	55107	93145	Approx. $0.071
Mistral	55107	63323	Approx. $0.490
ChatGPT-1	55107	63880	Approx. $0.210
ChatGPT-2	5107	58823	Approx. $0.044

TABLE III: Qualitative evaluation of various LLMs.

	Gemini	Mistral	ChatGPT-1	ChatGPT-2
Pros	Strict grading, consistent	Flexible, error tolerance	Fast, simple tasks	Handles complexity well
Cons	Overlooks creativity	Reduced accuracy	Struggles with complexity	Details may be missed
Strengths	Standardized require-ments	Innovation-friendly	Quick execution	Effective suggestions
Weaknesses	Mechanical feedback	Slow response	Low accuracy	Unstable, inconsistent feedback

TABLE IV: Feedback generation time required by LLMs.

LLM	Min Time	Max Time	Avg. Time	Total Time
Gemini	11.4s	27.9s	18.6s	15m59s
Mistral	37.7s	1m21s	47.8s	39m52s
ChatGPT-1	9.2s	4m28s	29.4s	29m26s
ChatGPT-2	12.8s	52.2s	26.5s	22m22s

REFERENCES

[1] Hahn, M. G., Baldiris Navarro, S. M., & de-La-Fuente-Valentin, L., "LUD: An Automatic Scoring and Feedback System for Programming Assignments," in *2022 International Conference on Advanced Learning Technologies (ICALT)*, Jul. 2022, doi: 10.1109/ICALT55010.2022.00118.

[2] Gaona, E. F., Camacho, C. E. P., Castro, W. M., Castro, J. C. M., Rodríguez, A. D. S., & Avila-Garcia, M. S., "Automatic grading of programming assignments in Moodle," in *2021 9th International Conference in Software Engineering Research and Innovation (CONISOFT)*, Oct. 2021, pp. 161-167, IEEE.

[3] Delgado-Pérez, P., & Medina-Bulo, I., "Customizable and scalable automated assessment of C/C++ programming assignments," *Computer Applications in Engineering Education*, vol. 28, no. 6, pp. 1449-1466, Dec. 2020.

[4] Clegg, B., Villa-Uriol, M. C., McMinn, P., & Fraser, G., "Gradeer: An Open-Source Modular Hybrid Grader," in 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET), May 2021, pp. 60-65, IEEE.

[5] Solecki, I., Porto, J., Alves, N. D. C., Gresse von Wangenheim, C., Hauck, J., & Borgatto, A. F., "Automated assessment of the visual design of Android apps developed with the app inventor," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, Feb. 2020, pp. 51-57.