System Report for CCL24-Eval Task 7: Prompting GPT-4 for Chinese Essay Fluency Evaluation

Dan Zhang, Thuong Hoang, Ye Zhu
Deakin University
{dan.zhang thuong.hoang, ye.zhu}@deakin.edu.au

Abstract

This report presents the methodology and results of utilizing GPT-4 for CCL24-Eval Task 7 of Chinese Essay Fluency Evaluation (CEFE). The task is divided into three tracks: Identification of Error Sentence Types, Rewriting Error Sentences, and Essay Fluency Rating. We employed a few-shot prompt engineering to guide GPT-4 in performing this task. Our approach integrated fine-grained error analysis with advanced NLP techniques to provide detailed, actionable feedback for students and teachers. Despite some successes, particularly in generating semantically similar and syntactically relevant corrections, our analysis revealed significant challenges, especially in multiple-label classification and the accurate identification of error types. The report discusses these findings and suggests areas for further improvement.

1 Introduction

The rapid development of education and the widespread use of the Internet have significantly increased the volume of essay evaluations. This growth poses significant challenges in terms of the cost and efficiency of human evaluation. To overcome these challenges, researchers and institutions have increasingly integrated Artificial Intelligence (AI) and Machine Learning (ML) algorithms into educational settings. Various NLP tasks have emerged, including automated item generation (Zou et al., 2022), grammar error detection and correction (Lu et al., 2022), reading and text complexity assessment, writing analysis and feedback (Jia et al., 2022), and automated writing evaluation.

For essay evaluation systems, particularly those targeting primary and middle school students' writing, the goal is to provide objective, accurate, and timely evaluations by analyzing various aspects of essays, including language use, content, and structural coherence. One of the critical aspects of essay evaluation is the fluency of expression, which reflects not only the smoothness and normative use of language but also the overall writing proficiency and ability to convey ideas clearly.

Essay fluency is essential for assessing and improving writing quality. It encompasses several linguistic features, such as sentence length, lexical complexity, and sentence structure (Yang et al., 2019). Current approaches to essay fluency evaluation typically involve scoring based on these linguistic features, treating it as a grammar correction task to identify and correct spelling and grammatical errors, or regarding it as a faulty sentence detection task to determine the presence of errors. However, these methods often consider essay fluency assessment as an isolated natural language processing (NLP) task, lacking systematic integration across multiple levels and perspectives. Moreover, existing studies tend to define grammatical error types in broad categories like redundancy, omission, misuse, and disorder, which are insufficiently detailed for providing precise feedback.

To advance the field of grammatical error recognition and correction in examination essays written by native Chinese-speaking primary and secondary school students, the China National Conference on Computational Linguistics (CCL-2024) has included Chinese Essay Fluency Evaluation (CEFE) as one of its

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 285-293, Taiyuan, China, July 25 - 28, 2024.
Volume3: Evaluations

shared tasks. This task systematically classifies text errors at various granularities, provides human-annotated data, and proposes three tracks that cover error detection and correction, sentence rewriting, and essay fluency rating.

To address this task, we employed few-shot prompting techniques and leveraged the advanced generation capabilities of GPT-4 (OpenAI, 2023) to make predictions across the three tracks. Our approach integrates fine-grained error analysis with advanced NLP techniques to deliver detailed, actionable feedback for students and teachers. Specifically, we used prompt engineering (Dai et al., 2023) to instruct GPT-4 to rewrite erroneous sentences with minimal modifications. Additionally, we guided GPT-4 to rate the overall fluency of essays. Ultimately, our goal is to improve the quality and fluency of student essays, providing them with insights necessary for continuous improvement in their writing skills.

2 Task Description

The CEFE shared task systematically defines fine-grained error types that affect essay fluency and offers corrective suggestions. This fine-grained approach helps students understand their writing issues more clearly and assists teachers in quickly gauging students' writing levels, facilitating more effective writing instruction. The evaluation task involves a comprehensive analysis of essay fluency from multiple linguistic angles: lexical, syntactic, and semantic, and provides modification suggestions. The task is divided into three tracks:

Track 1: Identification of Error Sentence Types This track focuses on recognizing both coarse-grained and fine-grained errors commonly found in student essays. It approaches the issue from two perspectives: character-level and component-level errors. There are four types of coarse-grained grammatical errors (Shen et al., 2023): Character-Level Error (CL), Incomplete Component Error (IC), Redundant Component Error (RC), and Incorrect Constituent Combination Error (ICC). Additionally, there are fourteen fine-grained error types, which provides a more detailed understanding of specific errors. Additionally, there are fourteen fine-grained error types, providing a more detailed understanding of specific errors. The detailed error types are listed in the table 1 below.

Coarse-grained Error	**Fine-grained Error**
Character-level	Missing Word Typo Missing Punctuation Punctuation Misuse
Incomplete Component	Subject unknown Predicate Incompleteness Object Incompleteness Other Incompleteness
Redundant Component	Subject Redundancy Function Word Redundancy Other Redundancy
Incorrect Constituent Combination	Improper Word Order Verb-object Mismatch Other Mismatch

Table 1: Coarse-grained and Fine-grained Error Types

Track 2: Rewriting Error Sentences This track involves rewriting incorrect sentences in primary and secondary school compositions. The challenge is to provide minimal modifications to the erroneous sentences while preserving the original semantics. The revisions should make as few changes as possible, focusing on correcting errors without altering the intended meaning.

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 285-293, Taiyuan, China, July 25 - 28, 2024.
Volume3: Evaluations

Track 3: Essay Fluency Rating This track involves assessing the overall fluency of essays. For this evaluation, essays are rated based on three levels of fluency: Excellent, Average, and Poor. These tracks collectively aim to enhance the understanding and evaluation of essay fluency, providing actionable insights for both students and educators.

3 Methodology

Track 1: Identification of Error Sentence Types To effectively identify error types, we designed a series of prompts tailored to leverage the capabilities of GPT-4. We also combined few-shot learning (Brown et al., 2020) in the prompt engineering (Chung et al., 2024). Our prompt design process involved the following steps:

- Initial Exploration: We began by generating a diverse set of example prompts to assess GPT-4's initial performance in identifying error types. These examples included sentences with known errors annotated with both coarse-grained and fine-grained error types.
- Prompt Refinement: Based on initial results, we refined our prompts to enhance clarity and specificity. Each prompt was designed to clearly define the task, provide context, and include labeled examples.
- Few-Shot Learning: We utilized few-shot learning by incorporating several annotated examples within the prompts. This approach helped GPT-4 understand the task requirements and improved its ability to generalize from the examples provided.
- Iterative Testing and Optimization: We iteratively tested and optimized our prompts using feedback from each iteration to fine-tune the examples and instructions.

This process ensured that the prompts effectively guided GPT-4 in accurately identifying error types.

The final few-shot learning-based prompt we used for track 1 is in Appendix Figure1:

Track 2: Rewriting Error Sentences To achieve effective sentence rewriting, we designed a series of prompts tailored to guide GPT-4 in producing accurate and semantically consistent corrections. The design process (Chung et al., 2024) included the following steps:

- Contextual Examples: We incorporated contextually relevant examples within the prompts to demonstrate how erroneous sentences should be corrected. These examples showcased minimal yet precise modifications, emphasizing the importance of maintaining original semantics. By illustrating corrections in context, we helped GPT-4 understand the nuances of effective rewriting.
- Clear Instructions: Each prompt provided clear and concise instructions specifying the need to make minimal changes while retaining the original meaning of the sentences. This guidance helped GPT-4 focus on correcting errors without altering the intended message, ensuring that the corrections were both accurate and meaningful.
- Few-Shot Learning: We utilized few-shot learning by including multiple annotated examples within each prompt. This approach allowed GPT-4 to learn the task requirements and apply them to new unseen sentences effectively.
- Iterative Refinement: Through iterative testing and refinement, we continuously improved our prompts to enhance GPT-4's performance. Feedback from each iteration was used to adjust the examples and instructions, ensuring the model generated accurate and semantically consistent corrections. This iterative approach allowed us to fine-tune the prompts for optimal results.

By following these steps, we ensured that GPT-4 was effectively guided in rewriting erroneous sentences, producing corrections that were both accurate and semantically consistent.

The final few shot learning prompt we used for track 2 is in Appendix Figure 2:

Track 3: Essay Fluency Rating To effectively rate essay fluency, we designed a series of prompts that provided clear instructions and examples. Our prompt design process included the following steps:

- Definition of Fluency Levels: We provided detailed descriptions of the three fluency levels (Excellent, Average, and Poor) within the prompts. This helped GPT-4 understand the criteria for each category.
- Annotated Examples: We included multiple annotated examples of essays rated at different fluency levels. These examples served as references, illustrating the characteristics of each fluency level.
- Few-Shot Learning: By incorporating several examples within each prompt, we utilized few-shot learning to help GPT-4 generalize the rating criteria across different essays.
- Iterative Refinement: We iteratively tested and refined our prompts to enhance clarity and effectiveness. Feedback from each iteration was used to adjust the examples and instructions, ensuring optimal performance.

The final few shot learning prompt we used for track 3 is in Appendix Figure 3:

4 Experiment Results and Analysis

Track 1 Result In Track 1, GPT-4 (OpenAI, 2023) encountered significant challenges in error identification, leading to incomplete and empty predictions:

The model failed to recognize certain error types, resulting in missing annotations. This indicates that GPT-4 struggled with multiple label classification tasks, particularly in accurately identifying all relevant error types within a sentence.

As for the empty predictions, there were instances where the model outputted empty results for error types, even when multiple errors were present. This further highlights GPT-4's difficulty in handling tasks requiring multiple labels.

The performance issues observed indicate that GPT-4 is not well-suited for multiple label classification tasks, especially in the context of identifying fine-grained grammatical errors.

We summarize the key challenges and insights for track 1: Complexity of Multiple Labels:

- Multiple label classification requires the model to identify several error types within a single sentence, which appears to be a challenging task for GPT-4.
- Prompt Sensitivity: The model's performance was highly sensitive to the phrasing and structure of the prompts. Despite iterative refinement, the prompts may not have been effective enough to guide the model accurately.
- Error Types Granularity: The fine-grained nature of error types required a deep understanding of grammatical rules and context, which GPT-4 struggled to achieve consistently.

These findings suggest that further refinement and alternative approaches are needed to improve GPT-4's performance in multiple label classification tasks.

Track 2 Result Our evaluation process for sentence rewriting involved comparing the corrected sentences generated by GPT-4 against reference annotations.

To evaluate the performance of our model, we utilized several metrics: Exact Match (EM), BERT Perplexity (Bert PPL), Levenshtein distance, BLEU-4, BERTScore (Zhang et al., 2019), Precision, Recall, and F0.5. An aggregated score, AvgScore, was calculated using a specific formula.

Exact Match (EM) measures the percentage of sentences that match the reference sentence exactly. BERT Perplexity (Bert PPL) (Devlin et al., 2018) indicates the fluency of the generated text based on the perplexity score from a BERT model. Levenshtein Distance measures the number of single-character

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 285-293, Taiyuan, China, July 25 - 28, 2024.
Volume3: Evaluations

edits required to change the generated sentence into the reference sentence. BLEU-4 (Papineni et al., 2002) evaluates the n-gram precision up to 4-grams, indicating the similarity between the generated and reference sentences. BERTScore (Zhang et al., 2019) uses BERT embeddings to measure the semantic similarity between the generated and reference sentences. Precision is the ratio of correctly predicted positive observations to the total predicted positives. Recall is the ratio of correctly predicted positive observations to all observations in the actual class. F0.5 is a weighted harmonic mean of precision and recall that gives more weight to precision.

The AvgScore was calculated using the following formula:

$$\text{AvgScore} = \frac{\text{EM} + \text{BLEU-4} + \text{F0.5} + \text{BERTScore}}{4} - \text{Levenshtein} - \text{Bert PPL}$$

The results are shown in table 2.

Model	EM	Bert PPL	Levenshtein	BLEU-4	BERT Score	Precision	Recall	F0.5	Score
Baseline model	9.67	3.26	1.46	88.72	97.28	37.33	20.53	32.08	52.22
Best Rank model	11.5	2.91	2.7	88.00	97.37	38.75	25.72	35.19	52.41
GPT-4	3.78	12.29	12.66	70.75	91.89	0.41	1.06	0.46	16.77

Table 2: Comparison of performance on Track 2.

The low precision score indicates that a significant portion of the model's corrections are not correct. A recall score greater than 1 suggests potential issues with the calculation or interpretation, as recall should typically be between 0 and 1. The F-0.5 score, which prioritizes precision, is also low, reinforcing the conclusion that the model's corrections are often inaccurate.

Performance analysis

- The performance analysis in Track 2 reveals strengths in generating semantically similar and syntactically relevant corrections, as indicated by high BLEU-4 and BERTScore values.
- However, the model struggles with exact matches and precision, suggesting that further refinement is needed to improve the accuracy of corrections.
- The overall AvgScore of 16.77 highlights a moderate performance, indicating room for improvement in future iterations.

Track 3 Result Our evaluation process for the essay fluency rating task involved comparing the fluency ratings generated by GPT-4 against reference annotations. The evaluation employs several metrics: Accuracy (ACC), Precision, Recall, Macro F1, and Quadratic Weighted Kappa (QWK). An aggregated score, AvgScore, is calculated using a weighted combination of these metrics. The AvgScore is calculated using the following formula:

$$\text{AvgScore} = 0.5 * \text{F1} + 0.2 * \text{QWK} + 0.3 * \text{ACC}$$

QWK is first normalized to the [0,1] range before calculating the weighted score.

The results are shown in the below table 3.

We achieved an accuracy of 48.59%, indicating that nearly half of the essay fluency ratings were correctly classified. This reflects a moderate level of correctness in the model's predictions.

The precision score of 55.79% suggests that over half of the predicted fluency ratings were correct. This high precision indicates that the model is good at identifying correct positive instances, meaning the model's predictions are reliable when it does identify a positive case.

Model	ACC	Precision	Recall	F1	QWK	AvgScore
Baseline model	45.52	41.42	39.15	38.57	0.1543	44.48
Best Rank model	51.66	49.89	46.54	47.42	0.1818	51.03
GPT-4	48.59	55.79	40.09	39.25	0.1202	45.40

Table 3: Comparison of Performance on Track 3.

The recall score of 40.09% shows that the model identified about 40% of the actual positive instances. While this is lower than the precision score, it indicates that there is room for improvement in capturing all relevant positive cases.

The F1 score of 39.25 reflects a balanced consideration of precision and recall. This score indicates that the model maintains a reasonable trade-off between precision and recall, providing a holistic measure of its classification performance.

Quadratic Weighted Kappa (QWK):

- The QWK score of 0.1202, normalized to 0.5601, indicates a moderate agreement with human ratings.
- This suggests that the model's predictions align reasonably well with human judgments, though there is still room for improvement.

AvgScore:

- The AvgScore of 45.40 is a composite measure reflecting the overall performance across all metrics.
- This score shows that GPT-4 model performs fairly well but also highlights areas where further refinement could lead to better results.

Performance analysis

- The performance analysis in Track 3 demonstrates a relatively strong model for assessing essay fluency, with high precision and a moderate level of accuracy and agreement with human ratings.
- However, the lower recall score indicates that the model could benefit from strategies aimed at capturing more positive instances.

5 Conclusion

The results of our evaluation indicate that while GPT-4 shows potential in tasks such as rewriting sentences and rating essay fluency, it faces significant challenges in accurately identifying error types, particularly in multiple-label classification tasks. Our approach demonstrated strengths in generating semantically similar corrections and achieving moderate agreement with human ratings. However, these findings highlight the need for further refinement and alternative strategies to improve the model's accuracy and overall performance in these tasks.

References

Xinshu Shen, Hongyi Wu, Xiaopeng Bai, Yuanbin Wu, Aimin Zhou, Shaoguang Mao, Tao Ge, and Yan Xia. 2023. *Overview of CCL23-Eval Task 8: Chinese Essay Fluency Evaluation (CEFE) Task*. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 3: Evaluations)*, pages 282–292.

Bowei Zou, Pengfei Li, Liangming Pan, and Aiti Aw. 2022. *Automatic true/false question generation for educational purpose*. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 61–70.

Yiting Lu, Stefano Bannò, and Mark Gales. 2022. *On assessing and developing spoken 'grammatical error correction' systems*. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 51–60.

Qinjin Jia, Yupeng Cao, and Edward Gehringer. 2022. *Starting from "zero": An incremental zero-shot learning approach for assessing peer feedback comments*. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 46–50.

Yiqin Yang, Li Xia, and Qianchuan Zhao. 2019. *An automated grader for Chinese essay combining shallow and deep semantic attributes*. *IEEE Access*, volume 7, pages 176306–176316. IEEE.

OpenAI, R. 2023. *GPT-4 Technical Report. arXiv 2303.08774*. *View in Article*, volume 2, number 5.

Wei Dai, Jionghao Lin, Hua Jin, Tongguang Li, Yi-Shan Tsai, Dragan Gašević, and Guanliang Chen. 2023. *Can large language models provide feedback to students? A case study on ChatGPT*. In *Proceedings of the 2023 IEEE International Conference on Advanced Learning Technologies (ICALT)*, pages 323–325. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. *BERT: Pre-training of deep bidirectional transformers for language understanding*. *arXiv preprint arXiv:1810.04805*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. *BERTScore: Evaluating text generation with BERT*. *arXiv preprint arXiv:1904.09675*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. *BLEU: a method for automatic evaluation of machine translation*. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and others. 2020. *Language models are few-shot learners*. *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and others. 2024. *Scaling instruction-finetuned language models*. *Journal of Machine Learning Research*, volume 25, number 70, pages 1–53.

A Appendix

中小学作文病句类型识别是一个多标签分类问题。每条句子可能包含一个或多个错误类型。
本任务旨在分析每条句子，并精确地识别出每个病句对应的粗粒度和细粒度错误类型。

病句类型定义：
病句错误类型包括词法、句法、语义错误，具体分为四个粗粒度错误类型和十四个细粒度错误类型如下：

字符级错误：包含 缺字漏字，错别字错误，缺少标点，错用标点。
成分残缺型错误：包含 主语不明，谓语残缺，宾语残缺，其他成分残缺。
成分赘余型错误：主语多余，虚词多余，其他成分多余。
成分搭配不当型错误：语序不当，动宾搭配不当，其他搭配不当。

请参考以下示例对输入的中小学生作文句子进行分析，识别并标注出现的每一种病句类型。

需要同时注意句子可能存在的多种错误，并给出相应的粗粒度和细粒度错误类型。只输出预测结果
输出格式应包括两个列表："CourseGrainedErrorType" 和 "FineGrainedErrorType"。
示例：
Example 1
Example 2

作文句子="{sentences}"

Figure 1: The final few-shot learning-based prompt used for track 1.

中小学作文病句改写任务的目标是对中小学生作文中出现的错误句子进行最小化修改，
同时确保语义保持不变。这是一个文本生成任务，输入为错误的句子，输出为修改后的句子。

操作指南：
请仔细分析输入的错误句子，并进行适当的修改以纠正错误，确保修改后的句子语义保持一致。
输出应仅包含修改后的句子（revisedSent），不包含其他信息。

示例：
Example 1
Example 2

作文句子="{sentences}"

请确保所有修改尽可能地简洁，并严格遵循语法规则，以提供清晰、准确的改写。

Figure 2: The final few-shot learning prompt used for track 2.

任务描述：
你是一名特级中小学语文老师。中小学作文流畅性评级任务是一个多分类任务，目的是对一篇作文的流畅性进行评级。
流畅性是指文本的语言表达是否通顺、自然，以及文章结构是否合理。

流畅性等级：

优秀：作文语言表达极为流畅，逻辑清晰，段落过渡自然，无明显语法错误或不自然表达。

一般：作文整体表达尚可，存在一些语法错误或表达不自然，逻辑和结构表现平均，影响阅读体验。

不及格：作文在语言表达上非常不流畅，逻辑不清晰或结构混乱，存在多处语法错误或不自然的表达，明显影响理解。

操作指南：
输入：一篇中小学生作文。
输出：根据作文的流畅性，给出相应的评级（优秀、一般、不及格）。只输出`essay_score_level`

示例输入和输出：
`Example 1`
`Example 2`

参照以上示例，对以下作文进行分类，只输出对`essay_socre_level`的预测{优秀，一般，不及格}：

作文句子="`{essay}`"

Figure 3: The final few-shot learning-based prompt used for track 3.