Can Large Language Models Automatically Score Proficiency of Written Essays?

Watheq Mansour[1*], **Salam Albatarni**[2], **Sohaila Eltanbouly**[2], **Tamer Elsayed**[2]
[1]The University of Queensland, Brisbane, Australia
[2]Computer Science and Engineering Department, Qatar University, Doha, Qatar
w.mansour@uq.edu.au, {salam.albatarani, sohaila.eltanbouly, telsayed}@qu.edu.qa

Abstract

Although several methods were proposed to address the problem of automated essay scoring (AES) in the last 50 years, there is still much to desire in terms of effectiveness. Large Language Models (LLMs) are transformer-based models that demonstrate extraordinary capabilities on various tasks. In this paper, we test the ability of LLMs, given their powerful linguistic knowledge, to analyze and effectively score written essays. We experimented with two popular LLMs, namely ChatGPT and Llama. We aim to check if these models can do this task and, if so, how their performance is positioned among the state-of-the-art (SOTA) models across two levels, holistically and per individual writing trait. We utilized prompt-engineering tactics in designing four different prompts to bring their maximum potential to this task. Our experiments conducted on the ASAP dataset revealed several interesting observations. First, choosing the right prompt depends highly on the model and nature of the task. Second, the two LLMs exhibited comparable average performance in AES, with a slight advantage for ChatGPT. Finally, despite the performance gap between the two LLMs and SOTA models in terms of predictions, they provide feedback to enhance the quality of the essays, which can potentially help both teachers and students.

Keywords: ChatGPT, Llama, Automated Essay Scoring, Natural Language Processing

1. Introduction

Automated Essay Scoring (AES), one of the earliest research problems in natural language processing (Page, 1966), aims to develop automated tools for assessing the proficiency of written essays. It can save a huge amount of time spent by teachers assessing essays manually while providing free feedback to students. In addition to being more consistent over time, AES systems can deliver more impartial assessments than humans.

Over the past 50 years, a wide variety of approaches were proposed to address AES, including learning from hand-crafted features (Phandi et al., 2015; Chen and He, 2013), and neural approaches including pre-trained language models (Dong et al., 2017; Wu et al., 2022). Most of them focus on predicting an overall holistic score that closely matches human judgment. Other studies focus on providing feedback by estimating quality scores on multiple traits of the essay, e.g., coherence (Higgins et al., 2004), relevance (Louis and Higgins, 2010), and organization (Persing et al., 2010) among others.

Recently, Large Language Models (LLMs) were introduced as cutting-edge transformer-based models demonstrating impressive capabilities on various tasks (Meoni et al., 2023; Kocmi and Federmann, 2023). In this paper, we check if the language proficiency retained by LLMs (represented by ChatGPT[1] and Llama 2[2] in our study) is useful in scoring essays. We test the capabilities of LLMs in this task across two levels, holistically and per individual writing trait. We also compare its performance to current state-of-the-art (SOTA) models. Furthermore, we analyze its performance and check its consistency in scoring the same essays.

To this end, our aim in this work is to answer the following research questions:

- **RQ1:** What is the effect of prompt engineering on the performance of LLMs for AES?
- **RQ2:** Is the performance of LLMs for AES consistent across prompts?
- **RQ3:** How far is the performance of ChatGPT and Llama from SOTA, holistically and per trait?

We carried out the experiments on the full Automated Student Assessment Prize (ASAP) dataset[3] comprising 8 tasks and 12978 essays. We designed four different prompts in an incremental fashion, where each adds more information to the prior one, and collected its responses for all essays. We also recorded the predicted scores and feedback for each essay and released them to the research community for further analysis.

Our developed prompts led to notable enhancements in the performance of LLMs, reaching a peak

---

*The work was done while at Qatar University.

[1] `https://chat.openai.com/chat`

[2] `https://ai.meta.com/llama/`

[3] `https://www.kaggle.com/c/asap-aes`

LREC-COLING 2024, pages 2777–2786
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

Quadratic Weighted Kappa (Cohen, 1968) score of 0.606 and 0.562 by ChatGPT and Llama 2, respectively. Although those scores lag behind SOTA models with respect to scoring prediction, LLMs have an advantage in terms of generated feedback. For all the experimented essays, they provided feedback to enhance their quality.

Our contribution in this work is four-fold:

- We evaluate the predictive performance of two LLMs, namely ChatGPT and Llama 2, for AES at the holistic and trait-based levels and compare it against the SOTA models.
- We show the impact of prompt engineering on ChatGPT's and Llama's performance by designing four different prompts with incremental task elaboration.
- We study the consistency of ChatGPT and Llama 2 in AES.
- We make all responses of the LLMs (including their feedback) publicly available to enable future analysis and research.[4]

The rest of the paper is organized as follows. Section 2 discusses the work related to AES and LLMs. The detailed description of our experiment and prompt designs are provided in Section 3. Section 4 presents the experimental results, the analysis of prompt engineering, and performance consistency. Finally, we conclude in Section 6.

2. Related Work

Previous attempts to address the AES problem relied either on crafting features to represent essays, syntactically (Yannakoudakis et al., 2011) or lexicallay (Chen and He, 2013; Phandi et al., 2015), or adopting neural networks (Xie et al., 2022). As a matter of fact, SOTA performance was achieved using a set of Multi-layer perceptrons that simultaneously conduct regression and ranking optimization (Xie et al., 2022).

Following the advancement of neural networks, LLMs were utilized in various applications. Consequently, a handful of researchers explored finetuning LLMs, e.g., BERT (Devlin et al., 2018), for the AES task, which yielded a remarkable performance (Wang et al., 2022; Yang et al., 2020).

ChatGPT and Llama[5] were built using reinforcement learning from human feedback to capture contextual meanings in a much deeper way, leading to a better language generation. Those models can be utilized for multiple tasks, e.g., code generation and text paraphrasing. Although work utilizing LLMs for AES is limited, the field is quickly evolving. Existing work solely focus on proprietary models. For example, Mizumoto and Eguchi (2023) utilized OpenAI's text-davinci-003 model for text completion to automatically score the essays of the TOEFL11 dataset and employed the generated scores as features for a Bayesian regression model. Naismith et al. (2023) tested GPT-4 (Achiam et al., 2023) on assessing the discourse coherence of essays, yielding promising results. Similarly, Yancey et al. (2023) focused on GPT-3.5 and GPT-4 with Common European Framework of Reference for Languages (CEFR) short essays dataset. Both models fell short behind a simple XGBoost baseline.

Although prior research highlighted the underperformance of LLMs, some gaps persist. Firstly, existing studies focus solely on OpenAI's paid models. Secondly, none of them have tested these models on the ASAP dataset, an essential benchmark in AES for comparing with SOTA models.

In this paper, we aim to assess the capability of using LLMs in AES over two levels, holistically and per trait. Moreover, different prompts are crafted to study the effect of prompt engineering on the performance of LLMs.

3. Methodology of Study

This section presents the outline of the adopted models and dataset, the design of the experimented prompts, and the experimental design.

3.1. Dataset and Models

We used the ASAP dataset, which is broadly used to evaluate AES systems (Ke and Ng, 2019). ASAP was proposed in a competition on Kaggle in 2012. It contains 8 different tasks written in English by students ranging from grades 7 to 10. There are three types of tasks: persuasive, source-dependent response, and narrative. Table 1 describes the properties of each task.

We have chosen two LLM models for our experiments, namely, ChatGPT and Llama. ChatGPT has demonstrated remarkable capabilities on various tasks and gained significant attention across diverse domains. In our experiments, we utilized the ChatGPT API, namely gpt-3.5-turbo-0301 (a Snapshot from March 1st, 2023).[6] This model is optimized for dialogue, and its performance is on par with Instruct Davinci. Nevertheless, ChatGPT API is not open source, which imposes scalability limitations due to its associated cost.[7] Alternatively, Llama is an open-source LLM developed by

---

[4] `https://github.com/Watheq9/AES-with-LLMs`

[5] Hereafter we will refer to Llama 2 as Llama.

[6] `https://platform.openai.com/docs/models`

[7] The total cost of the experiments with ChatGPT in this work amounted to approximately $120.

Task	Essay Type	Avg. length	Score Range	# Essays
1	Persuasive	350	2-12	1783
2		350	1-6	1800
3	Source-Dependent	150	0-3	1726
4		150	0-3	1772
5		150	0-4	1805
6		150	0-4	1800
7	Narrative	250	0-30	1569
8		650	0-60	723

Table 1: Properties of the different tasks in the ASAP dataset.

Meta. It outperforms other open-source language models (e.g., Falcon and MPT) on different benchmarks (Touvron et al., 2023). In this work, we used Llama-2-13b-chat-hf.[8] For both models, the temperature is set to 0 to guarantee reproducible results.

Note that the essay might contain some anonymization tokens that are placeholders added to anonymize sensitive information and ensure privacy. They replace personal names, locations, numbers, times, dates, organizations, and other specific information that could identify individuals or institutions. The anonymization tokens are: @PERSON#, @LOCATION#, @NUM#, @TIME, @MONTH, @DATE, and @CAPS#, where # refers to a number.

Figure 1: The preprocessing instructions.

3.2. Prompt Design

Four prompts are designed to assess the performance of ChatGPT and Llama. The prompts are structured in an incremental fashion, with each subsequent prompt building upon the preceding one by adding extra task elaboration. The key inputs in all the prompts are the task (i.e., the prompt the essay was written in response to) and the essay. Then, the scoring rubric[9] and one-shot example are added to the subsequent prompts.

Ensuring the clarity of the prompt given to the LLMs is vital. Therefore, we follow some prompt-engineering tactics advised by OpenAI[10] to enhance the prompt comprehension. First, proper delimiters are used to distinguish the inputs (essay, rubrics, and one-shot examples) from the instruction text, preventing prompt injection issues. Second, the prompt is designed as a series of sequential instructions, pushing the models to complete the steps in order. Third, we instruct the models to preprocess the input text (as shown in Figure 1), as certain placeholders are present within the ASAP essays. Finally, we ask for a JSON output structure specifying the desired outputs. Overall, we designed four prompt variants that meet these guidelines as follows:

- **Prompt A** (shown in Figures 2 and 3) is a simple prompt asking to evaluate the input essay *without defining the rubrics*. It anchors our expectations on the performance of ChatGPT and Llama without any extra information.

Your job is to evaluate the provided essay on a scale from 1 to 6.
{preprocessing_instruction}
Here is a summary of the required steps:
1. Evaluate the essay (after suitable replacement of the anonymization tokens wherever needed) on a scale from 1 to 6.
2. Format your response in JSON as follows:
{
"Total_score": your evaluation score,
"Commentary": your feedback on the essay
}
Here is the prompt delimited by <>: <task_prompt>
And here is the essay, which you need to evaluate, delimited by triple backticks: “‘input_essay“‘

Figure 2: Prompt A, a simple prompt asking ChatGPT to score an essay according to the score range.

You will be given a prompt and an essay that was written in response to that prompt. Your job is to score the provided essay on a scale from 1 to 6.
{preprocessing_instruction}
Here is the prompt delimited by <>: <task_prompt>
And here is the essay, which you need to score, delimited by triple backticks: “‘input_essay“‘
Format your response in JSON as follows:
{
"total_score": your evaluation score,
}
The response should include only the JSON format.

Figure 3: Prompt A, a simple prompt asking Llama to score an essay according to score range.

- **Prompt B** *adds the rubric guidelines* to prompt

---

[8] https://huggingface.co/meta-llama/Llama-2-13b-chat-hf

[9] Scoring guidelines that specify each score criteria

[10] https://platform.openai.com/docs/guides/gpt-best-practices/strategy-write-clear-instructions

A, allowing the models to understand the task more. We utilized the rubrics attached to each task in ASAP, which include definitions for each score point in the scale. The rubric assists in providing an objective framework to assess the quality and fulfillment of specific criteria.

- **Prompt C** expands on Prompt B by *adopting a one-shot example strategy*. This prompt will demonstrate whether ChatGPT and Llama are able to learn from the provided example and enhance its performance.
- **Prompt D** (shown in Figures 4, 5, and 6) *formats prompt C as a chat between a system, an assistant, and a user*, dividing the input and output into suitable roles.[11] Thus, the expected input and output hopefully become much clearer to the LLM. Figure 7 presents an example of ChatGPT's response to an essay from Task 7 using this prompt.

```
{"role": "system",
"content": “You are a helpful pattern-following assistant that evaluates essays written by students in response to a given prompt. Your job is to evaluate the provided essay per the following Rubric Guidelines delimited by <>: <{rubric_guidelines}>
{preprocessing_instruction}
Here is a summary of the required steps
1. Evaluate the essay (after suitable replacement of the anonymization tokens wherever needed) per the described rubric guidelines.
2. Format your response in JSON as follows:
{
"Total_score": your evaluation score,
"Commentary": your feedback on the essay
}
Here is the prompt delimited by <>: <task_prompt> ”’
},
{ "role": "system",
"name": "example_user",
"content": example_input},
{"role": "system",
"name": "example_assistant",
"content": example_output},
{"role": "user",
"content": actual_input,}
```

Figure 4: Prompt D for ChatGPT, formats prompt C as a chat between a system, an assistant, and a user.

The prompts for ChatGPT and Llama contain the same instructions but in different formats. We tried multiple arrangements of the instructions inside the prompt to see the best instruction order for each LLM. After several trials with the order of the instructions inside the prompts, we noticed two main differences. First, ChatGPT benefits from adding a concise list of required steps, whereas Llama did not exhibit similar enhancements from these steps. Second, ChatGPT provided the required JSON format, which was one of the required steps, regardless of the position of this requirement inside the prompt. On the other hand, with Llama, the required format is only generated when explicitly specify it at the very end of the prompt. Another difference in the prompts between ChatGPT and Llama is that in Llama's prompts, we did not ask for feedback. We did this to keep the responses concise because Llama often offers extra feedback even when it is not asked to provide one. Figures 2 and 3 show the structure of prompt A for ChatGPT and Llama. Prompts B and C have similar formats with the addition of the rubric and the one-shot example.

For Prompt D, ChatGPT has clear and well-defined roles for formatting the prompt as a chat. Figure 4 shows prompt D for ChatGPT. Similarly, Llama has the capability to provide system instruction but with a different format. Llama has an advantage that it can store the history of the messages. So we utilized this by adding the system instructions and the one-shot example as history, and the prompt contained only the input essay. However, we noticed Llama tends to forget the constraint of requiring the response to be in JSON format, so we repeated this instruction with each input. Figures 5 and 6 show the history and prompt D for Llama.

```
[ ’system’, ’You are a helpful pattern-following assistant that evaluates essays written by students in response to a given prompt. Your job is to score the provided essay per the following Rubric Guidelines delimited by <>: <{rubric_guidelines}>
{preprocessing_instruction}
Here is the prompt delimited by <>: <task_prompt>
Format your response in JSON as follows:
{
"total_score": your evaluation score,
}
The response should include only the JSON format.
],
[ ’[INST]’, ’example_input’ ],
[ ’[/INST]’, ’{example_output}’ ]
```

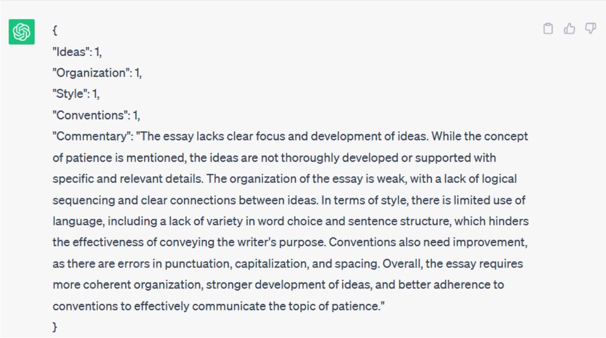
Figure 5: The history message provided to Llama for Prompt D. In Llama [INST] indicates the user input, and [/INST] indicates Llama's response.

---

[11] Such roles are defined by the LLM.

```
{input_essay}
The response should include only the JSON format.
```

Figure 6: Prompt D for Llama. It contains the input essay only. All other instructions are included in the conversation history.



```
{
"Ideas": 1,
"Organization": 1,
"Style": 1,
"Conventions": 1,
"Commentary": "The essay lacks clear focus and development of ideas. While the concept
of patience is mentioned, the ideas are not thoroughly developed or supported with
specific and relevant details. The organization of the essay is weak, with a lack of logical
sequencing and clear connections between ideas. In terms of style, there is limited use of
language, including a lack of variety in word choice and sentence structure, which hinders
the effectiveness of conveying the writer's purpose. Conventions also need improvement,
as there are errors in punctuation, capitalization, and spacing. Overall, the essay requires
more coherent organization, stronger development of ideas, and better adherence to
conventions to effectively communicate the topic of patience."
}
```

Figure 7: A sample of ChatGPT response for essay with id 17834 from task 7 using prompt D.

3.3. Experimental Design

Considering ChatGPT's paid API, testing 4 prompts on all tasks is quite costly. Given that ASAP tasks are divided into three categories, we performed the experiments in two phases for both ChatGPT and Llama. In the **first phase**, we experiment with one task from each category on all of our prompts to choose the best prompt for each category. In the **second phase**, we apply that prompt to the remaining tasks in that category. From each category, we opt for the task with the shortest overall length (rubric guidelines, essay resource, and task question). So, we selected task 1 (persuasive), task 5 (source-dependent), and task 7 (narrative) for the first phase.

4. Experimental Results and Discussion

In this Section, we discuss the performance of ChatGPT and Llama for AES. Moreover, we provide a thorough examination of the differences between ChatGPT and Llama when it comes to judging essay proficiency and their consistency. Recall that in this work, we aim to answer the following research questions:

- **RQ1:** What is the effect of prompt engineering on the performance of LLMs for AES?
- **RQ2:** Is the performance of LLMs for AES consistent across prompts?
- **RQ3:** How far is the performance of ChatGPT and Llama from SOTA, holistically and per trait?

To evaluate the AES performance of ChatGPT and Llama, we use Quadratic Weighted Kappa (QWK) (Cohen, 1968), which is a widely used measure for AES. It is worth noting that, for ASAP tasks 7 and 8, we computed the holistic scores out of the predicted trait scores per the formula mentioned in the corresponding task guidelines.

Our **baselines** are SOTA on both the holistic and trait levels. As such, we contrast our LLM models with Xie et al. (2022) and Jiang et al. (2023), which achieved SOTA performance for in-prompt AES and cross-prompt AES respectively, from the holistic perspective. We also compare our LLM models against Mathias and Bhattacharyya (2020) over the traits level.

4.1. Effect of Prompt Engineering (RQ1)

Table 2 presents the results of the first phase of our experiments, which reveal several observations.

Effect on ChatGPT For ChatGPT, first, prompts B, C, and D exhibit better performance than prompt A (except for the source-dependant task, where prompt A outperforms prompts C and D). This is expected as they all provide more context that is apparently needed for ChatGPT to comprehend the task better. Second, the best prompt is different across different task categories. More specifically, prompts C, D, and B were the best for the persuasive, narrative, and source-dependent categories, respectively. This is a very interesting outcome that highlights the importance of prompt engineering for AES in general and AES different task categories in particular. Moreover, a closer look at the task definition of the source-dependent category explains why prompt B was the best; in such category, the source text is provided, and students were mandated to include evidence from it, making it a somewhat "self-contained" task and prompt B was just enough for that with no need for showing an example (as in prompts C and D). That also explains why the source-dependent category, represented by task 5, seems to be much easier (over all prompts) than the other categories. In contrast, the persuasive and narrative categories benefited from the one-shot setup of prompts C and D; however, both still exhibit low absolute performance.

Effect on Llama On the other hand, Llama exhibits a totally different performance than ChatGPT. As for the persuasive task category, prompt A achieved the best performance. Recall that in the argumentative/persuasive task, essays typically present evidence and facts to support their stated argument. Since Llama's pre-training data incorporates a more extensive corpus of factual information than others (Touvron et al., 2023), it is possible

LLM	Task	Prompt			
		A	B	C	D
ChatGPT	Task 1 (Persuasive)	0.057	0.080	**0.120**	0.113
	Task 5 (Source-Dep.)	0.552	**0.576**	0.476	0.455
	Task 7 (Narrative)	0.060	0.027	0.055	**0.123**
Llama	Task 1 (Persuasive)	**0.331**	0.092	0.154	0.170
	Task 5 (Source-Dep.)	0.047	0.346	**0.418**	0.180
	Task 7 (Narrative)	0.268	**0.359**	0.037	0.224

Table 2: Performance of LLMs measured in QWK over different prompt designs per task category. The best performance in each row (model and task category) is boldfaced.

that it grasps the task better with the basic instructions in prompt A, while the added context in the other prompts confuses the model and makes predictions more challenging. As for the remaining tasks, adding the rubrics in prompt B improved the context clarity and the scoring criteria over prompt A. Moreover, it seems that adding a one-shot example appeared to add noise to the prompt for the narrative task, although it was useful in the source-dependent task.

Overall When observing both the performance of ChatGPT and Llama, we can notice a huge discrepancy. For example, Llama's performance varies notably between different prompts for the same task, whereas this behavior is not observed with ChatGPT.

4.2. Consistency (RQ2)

To show this discrepancy in more detail, we assessed the consistency of the scores generated by ChatGPT and Llama across the different prompts. As the prompts build upon one another, it is anticipated that the performance of the LLM will have a reasonable agreement between the various prompts because *they all share the same core instructions*. To perform this analysis, we calculate the QWK score between each pair of prompts for tasks 1, 5, and 7 to assess the level of agreement among each LLM response across the different prompts. Table 3 shows the results of this experiment for the 6 pairs of prompts. For ChatGPT, the adjacent pairs always have a higher QWK; for example, the A-B pairs have higher scores than A-C and A-D for the three tasks. This behavior is expected since the adjacent prompts have higher context similarity. The average QWK for ChatGPT for tasks 1, 5, and 7 are 0.62, 0.58, and 0.64. On the other hand, Llama showed a significant disagreement between the responses of the different prompts, with no observable increasing or decreasing trend. The average QWK is 0.4, 0.2, and 0.17 for tasks 1, 5, and 7, respectively.

From this analysis, Llama demonstrated inconsistent performance in AES, which suggests that it is highly sensitive to prompt selection. ChatGPT showed higher agreement across the different prompts and tasks, indicating that ChatGPT exhibits consistent performance in AES. It is worth noting that in the ASAP dataset, the average agreement score between the two human raters is 0.75 [12], which indicates the complexity of the AES task. These results show the critical role of prompt engineering in enhancing or diminishing the performance of LLMs on the AES task. Moreover, choosing the right prompt is highly dependent on *both* the LLM and the nature of the task.

This also highlights that, despite the astonishing ability of LLMs to generate coherent and good-quality text, they struggle to distinguish between good and bad essays. This issue persists even when providing more context about the scores and a sample of a scored essay.

4.3. LLMs vs. SOTA (RQ3)

We next turn to measure the performance of the two LLMs in predicting the holistic score and trait scores of the essays over the entire set of ASAP tasks. With that, we aim to (1) comprehensively contrast the performance of the two LLMs against each other, and (2) position their performance within the best AES models in the literature.

Based on the first stage, for each LLM, we selected the prompt that achieved the highest QWK for each task category. Specifically, for ChatGPT, we selected prompts C, D, and B, for the persuasive, narrative, and source-dependent categories, respectively. Likewise, for Llama, we opted for prompts A, B, and C, for the persuasive, narrative, and source-dependent categories, respectively.

Table 4 compares the performance of ChatGPT and Llama per ASAP task with SOTA models (as reported in their respective studies) on the holistic level. First, we notice that the average performance of the two LLMs across tasks is quite comparable, with a slight advantage for ChatGPT. The results also show that ChatGPT does a reasonably good

---

[12] For rater1_domain1 and rater2_domain1

Task	LLM	Prompt Pairs						Average
		A - B	A - C	A - D	B - C	B - D	C - D	
1	ChatGPT	0.772	0.526	0.449	0.672	0.585	0.741	0.62
	Llama	0.286	0.595	0.615	0.235	0.293	0.537	0.43
5	ChatGPT	0.836	0.709	0.458	0.734	0.405	0.312	0.58
	Llama	0.059	0.046	0.222	0.524	0.181	0.162	0.20
7	ChatGPT	0.804	0.748	0.477	0.756	0.475	0.607	0.64
	Llama	0.457	0.148	0.152	0.092	0.125	0.021	0.17

Table 3: QWK performance of each prompt pair for ChatGPT and Llama

Task	Xie et al. (2022)	Jiang et al. (2023)	ChatGPT	Llama
1	**0.856**	0.762	0.120	0.331
2	**0.750**	0.686	0.193	0.562
3	**0.756**	0.637	0.198	-0.059
4	**0.851**	0.673	0.416	0.201
5	**0.847**	0.778	0.576	0.418
6	**0.858**	0.664	0.606	0.363
7	**0.838**	0.742	0.123	0.359
8	**0.779**	0.677	0.276	0.201
Average	**0.817**	0.702	0.313	0.297

Table 4: QWK performance of ChatGPT and Llama vs. SOTA holistic models. Best per row is boldfaced.

job in source-dependent tasks 4-6, where the performance is significantly better than in the other (source-independent) tasks, while Llama showed a better performance only in Task 2 (which is persuasive). Second, it is quite apparent that the exhibited performance of both LLMs is nowhere near the SOTA models, with average QWK of 0.817, 0.702, 0.313, and 0.297 for Xie et al. (2022), Jiang et al. (2023), ChatGPT, and Llama, respectively. While those SOTA models are task-specific and trained on the ASAP dataset directly, higher performance was expected from the LLMs, given the vast amount of data they were trained on and their ability to generate good-quality text. It is clear that although LLMs were able to understand the task and give a score, it was not able to capture the input essay as a whole and score it accordingly.

On the traits level, Table 5 draws a comparison with SOTA models per trait for tasks 7 and 8, the only tasks that have traits in ASAP. The results deliver the same message of the holistic level; ChatGPT and Llama are very far behind.

4.4. Analysing the Provided Feedback

Despite the underwhelming performance of both LLMs in scoring essays, that does not mean that LLMs cannot help in sharpening the essay quality by providing feedback for the writer. Here, we show a sample of such feedback and discuss the difference between ChatGPT and Llama. Table 6 shows the feedback of both LLMs on two sample essays from Task 3 (source-dependent), one is of low quality and the other is of high quality.

Interestingly, for ChatGPT, the feedback provided for the low-score essay was short, and there was no elaboration or meaningful comments on how to improve the essay. On the contrary, Llama elaborated better on why that essay had a low score. For the high-score essay, both models were able to explain where the essay could be improved; however, again, Llama was able to better explain where the essay could be improved.

While this is a very small sample (only two essays), it shows promising potential for LLMs to provide meaningful feedback on the writing quality of the given essays, which indeed opens a potential research direction for future work on using LLMs in the AES task. We plan to evaluate a larger set of feedback responses in future experiments to further validate and extend these findings.

5. Limitations

Our study has indeed several limitations.

- First, in our experiments, we were keen on making the length of the prompt as short as possible to fit within the maximum-allowed limit of tokens (which is 4,096 tokens) and reduce the corresponding cost.
- Second, we *randomly* chose one example for the one-shot prompts and did not experiment with different examples. It is also possible that adding few-shot examples might achieve better results.

Task	Model	Content	Organization	Voice	Word Choice	SF	Conventions	Style
7	BEA 2020	0.771	0.676	-	-	-	0.621	0.659
	ChatGPT	0.045	0.068	-	-	-	0.097	0.079
	Llama	0.091	0.023	-	-	-	0.327	0.154
8	BEA 2020	0.586	0.632	0.544	0.559	0.586	0.558	-
	ChatGPT	0.185	0.256	0.155	0.15	0.203	0.311	-
	Llama	0.276	0.271	0.278	0.268	0.121	0.089	-

Table 5: Comparison of ChatGPT's and Llama's performance against Mathias and Bhattacharyya (2020) (BEA 2020) per trait scores across tasks 7 & 8. SF refers to Sentence Fluency.

Essay	ChatGPT	Llama
5985 (Task 3, **Low-Score**)	The response is completely irrelevant and incorrect. It does not address the prompt and does not demonstrate any understanding of the text.	The response shows evidence of a minimal understanding of the text. The essay does not address the prompt or the features of the setting that affect the cyclist. Instead, the essay focuses on the writer's personal feelings and experiences, which are not related to the prompt. The essay also contains some inaccuracies, such as the statement that the cyclist was "cut hurt" and that the features of the setting did not affect the cyclist. Overall, the response lacks a clear understanding of the text and the prompt.
7000 (Task 3, **High-Score**)	The response demonstrates a partial understanding of the text. The essay discusses how the features of the setting affect the cyclist, specifically how the old directions and abandoned towns with no good water could have killed him. However, the response could have provided more examples from the essay to support the conclusion. Additionally, the response could have extended the understanding beyond the literal by discussing how the cyclist's experience reflects the fluidity of the world and the importance of being prepared for nature's challenges.	The response demonstrates a partial understanding of the text, although may not develop all parts equally. The essay provides some examples from the text to support the conclusion, but the analysis is not comprehensive and does not fully connect the support to a conclusion or assertion made about the text. The response correctly identifies the features of the setting that affect the cyclist, such as the desert environment, the lack of water, and the abandoned towns. However, the analysis could be more detailed and nuanced. For example, the response could discuss how the setting affects the cyclist's mood and confidence, or how the abandoned towns and lack of water create a sense of desolation and hopelessness. Additionally, the response could provide more specific examples from the text to support the analysis. Overall, the response demonstrates some understanding of the text, but could be more comprehensive and nuanced in its analysis.

Table 6: Feedback generated by ChatGPT and Llama for a sample essay of actual low and high scores.

- A further limitation is that OpenAI does not keep track of the historical context of previous requests. So, we had to send the same evaluation instructions in each API request, which incurred additional costs.

- Finally, we experimented with one closed- and another open-source LLMs. Experimenting with more LLMs might remove some of the above limitations and bring additional insights.

6. Conclusion

In this study, we examined the ability of two LLMs, namely ChatGPT and Llama, to assess written essays holistically and per trait, analyzed their performance, and studied their consistency. We designed 4 prompts that showed different performance over different task categories. Our analysis shows that the performance of the LLMs is highly dependent on the prompt and the task type. Moreover, Llama, in particular, is very sensitive to small changes in the prompt, while ChatGPT is more robust and consistent. Although both LLMs fell short behind other SOTA models, it is inevitable to say that LLMs are high-quality text-generating machines. However, it is important to highlight their limitations when it comes to *evaluating* text. Our study suggests a possibility of utilizing the vast linguistic knowledge those LLMs have to provide feedback for improving essays. However, according to our study, LLMs are not yet reliable for predicting the score a student would get on an essay.

In future work, we plan to analyze the generated feedback and its usefulness, besides examining the performance of other LLMs, such as Bard, on the task. Expanding our study to other datasets is another future direction. Fine-tuning the LLMs for the AES task might also improve the performance; therefore, that direction is also worth exploring.

7. Acknowledgements

The work of Salam Albatarni was supported by GSRA grant# GSRA10-L-2-0521-23037 from the Qatar National Research Fund (a member of Qatar Foundation). The work by the rest of the authors was made possible by NPRP grant# NPRP14S-0402-210127 from the Qatar National Research Fund. The statements made herein are solely the responsibility of the authors.

8. Bibliographical References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Hongbo Chen and Ben He. 2013. Automated essay scoring by maximizing human-machine agreement. In *Conference on Empirical Methods in Natural Language Processing*.

Jacob Cohen. 1968. Weighted kappa: nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

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*.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada. Association for Computational Linguistics.

Derrick Higgins, Jill Burstein, Daniel Marcu, and Claudia Gentile. 2004. Evaluating multiple aspects of coherence in student essays. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 185–192.

Zhiwei Jiang, Tianyi Gao, Yafeng Yin, Meng Liu, Hua Yu, Zifeng Cheng, and Qing Gu. 2023. Improving domain generalization for prompt-aware essay scoring via disentangled representation learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12456–12470, Toronto, Canada. Association for Computational Linguistics.

Zixuan Ke and Vincent Ng. 2019. Automated essay scoring: A survey of the state of the art. In *IJCAI*, volume 19, pages 6300–6308.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*.

Annie Louis and Derrick Higgins. 2010. Off-topic essay detection using short prompt texts. In *proceedings of the NAACL HLT 2010 fifth workshop on innovative use of NLP for building educational applications*, pages 92–95.

Sandeep Mathias and Pushpak Bhattacharyya. 2020. Can neural networks automatically score essay traits? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 85–91, Seattle, WA, USA → Online. Association for Computational Linguistics.

Simon Meoni, Eric De la Clergerie, and Theo Ryffel. 2023. Large language models as instructors: A study on multilingual clinical entity extraction. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 178–190, Toronto, Canada. Association for Computational Linguistics.

Atsushi Mizumoto and Masaki Eguchi. 2023. Exploring the potential of using an ai language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2(2):100050.

Ben Naismith, Phoebe Mulcaire, and Jill Burstein. 2023. Automated evaluation of written discourse coherence using GPT-4. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 394–403, Toronto, Canada. Association for Computational Linguistics.

Ellis B Page. 1966. The imminence of... grading essays by computer. *The Phi Delta Kappan*, 47(5):238–243.

Isaac Persing, Alan Davis, and Vincent Ng. 2010. Modeling organization in student essays. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 229–239.

Peter Phandi, Kian Ming A. Chai, and Hwee Tou Ng. 2015. Flexible domain adaptation for automated essay scoring using correlated linear regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 431–439, Lisbon, Portugal. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yongjie Wang, Chuan Wang, Ruobing Li, and Hui Lin. 2022. On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation. *arXiv preprint arXiv:2205.03835*.

Yongchao Wu, Aron Henriksson, Jalal Nouri, Martin Duneld, and Xiu Li. 2022. Beyond benchmarks: Spotting key topical sentences while improving automated essay scoring performance with topic-aware bert. *Electronics*, 12(1):150.

Jiayi Xie, Kaiwei Cai, Li Kong, Junsheng Zhou, and Weiguang Qu. 2022. Automated essay scoring via pairwise contrastive regression. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2724–2733.

Kevin P. Yancey, Geoffrey Laflair, Anthony Verardi, and Jill Burstein. 2023. Rating short L2 essays on the CEFR scale with GPT-4. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 576–584, Toronto, Canada. Association for Computational Linguistics.

Ruosong Yang, Jiannong Cao, Zhiyuan Wen, Youzheng Wu, and Xiaodong He. 2020. Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1560–1569.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.