Exploring Large Language Models for Automated Essay Grading in Finance Domain

1st Garima Malik
Data Science Lab
Toronto Metropolitan University
Toronto, Canada
garima.malik@torontomu.ca

2nd Mucahit Cevik
Data Science Lab
Toronto Metropolitan University
Toronto, Canada
mcevik@torontomu.ca

3rd Sojin Lee
Co-founder & CEO
Blees Technologies Inc. (Blees AI)
& Olive AI Limited (Olive AI)
Toronto, Canada
ms.sojinlee@gmail.com

***Abstract*—This study explores the application of large language models (LLMs) for the automated grading of essays in the finance domain. The focus is on generating grades for six Assessment Indicators (AIs) related to finance and accounting for each essay. Our research highlights the potential of LLMs and showcases custom prompt engineering's effectiveness in a domain-specific Automated Essay Scoring (AES) task. We propose two distinct prompting techniques: unified and discrete. The unified technique generates grades for all AIs using a single comprehensive prompt, while the discrete technique employs separate prompts for each AI. To enhance the effectiveness of these models, we apply In-Context learning through One-shot and Few-shot methods. Through extensive experimentation, we show that LLMs outperform fine-tuned BERT-like baselines, demonstrating consistency and generalizability in their results. However, challenges remain with output post-processing and the cost of processing input tokens.**

. ***Index Terms*—Large Language Models (LLMs), Automated Essay Grading, In-Context Learning, Generative AI, Natural Language Processing (NLP)**

I. Introduction

Automated Essay Scoring (AES) and Automated Essay Grading are often used interchangeably. Both involve grading written assignments using computer algorithms [1]. Automated grading tasks include a wide variety of assessments, including standardized exams, long-answer questions, and high school essays. The grading criteria can vary significantly: some assessments focus on linguistic accuracy, while others prioritize the inclusion of key concepts [2]. In this research, we address an automated grading problem aimed at evaluating a student's understanding of core concepts in finance and accounting by checking if these ideas are effectively discussed in their essay. The procedure for the automated grading task we are considering is illustrated in Figure 1, each student essay is combined with prompt engineering and processed through large language models to generate the overall grades. Each student is evaluated on six core concepts, referred to as Assessment Indicators (AIs). The language models assign a grade of either 'Y' (Correct) or 'N' (Incorrect) based on whether the AIs are discussed in the essay. The overall grade is determined by the student's performance across these six core concepts.

Most AES research focuses on evaluating the overall quality of the essay, providing a holistic, single score [3, 4]. These studies primarily assess concepts related to grammar, coherence, and organization of the essays [2]. In this study, we present a domain-specific automated grading problem where student essays are evaluated based on their understanding of financial reporting, inventory, performance management, Return on Investment (RoI), financial recommendations, and future improvement strategies through case studies in accounting and finance. Helmeczi et al. [5] introduced a similar AES problem in the finance domain; however, they approached it as a classification problem by treating individual sentences as inputs to the classification models, rather than using the entire student essay.

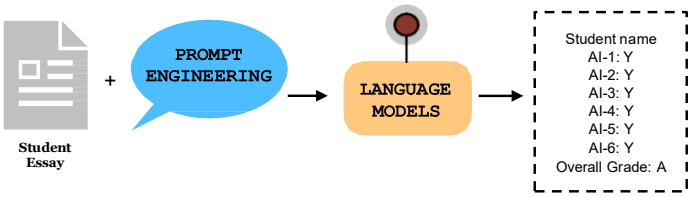


Fig. 1: Overview of automated essay grading task using language models considered in our study

Traditionally, automated grading tasks are considered regression-based [6] or classification-based [5] tasks which can be processed through machine learning and transformer-based architectures [7]. With the latest advancements in Large Language Models (LLMs) in Natural Language Processing (NLP), we can enhance grading tasks from basic language-based essay evaluation to domain-specific automated grading. This study leverages LLMs impressive capabilities, robust logical reasoning, and ability to comprehend human instructions to test complex concepts in varied domains[8].

To advance research in AES within the finance and accounting domain, we employ open-source LLMs and prompt engineering to grade student essays automatically. LLMs offer immediate and consistent feedback, simplifying the grading process with precise and informative prompts [7]. Additionally, we apply In-Context learning to LLMs, enhancing performance compared to Zero-shot settings where models are provided only with prompts and student essays, without sample grades [2]. To design effective prompts for each Assessment

Indicator, we propose two different prompt structures: one where each AI grade is predicted individually and another where all AI grades are predicted simultaneously. The main contributions of our work are summarized in terms of research questions below:

RQ1: How do state-of-the-art language models perform for each Assessment Indicator (AI) in the grading task compared to existing methods?

We conduct extensive experiments with four state-of-the-art language models: GPT-3.5 [9], GPT-4o [10], Mistral8x7B [11], and Gemini-1.5-pro [12] for the automated grading task. Utilizing the proprietary FinCase-AES dataset, which evaluates basic financial reporting concepts, we demonstrate the effectiveness of these language models in predicting grades across six different assessment indicators for each student essay. In addition, we implemented fine-tuned BERT-based baselines to benchmark the performance of the language models against existing methods. Our results highlight the superior performance of GPT-4o, which consistently outperformed other models, and the significant potential of language models in enhancing automated essay scoring systems in the finance domain.

RQ2: Which prompting strategy, UPM or DPM, is more effective for the automated grading task?

We define and evaluate two prompting strategies, the Unified Prompting Method (UPM) and the Discrete Prompting Method (DPM), for grading student essays. For each AI, specific prompts are formulated, and grades are generated according to a specified output format. Our experiments demonstrate that both prompting methods provided consistent performance across each AI. However, we observe that the UPM approach, which grades the student essay for all AIs in a single pass, offers efficiency and cost advantages over the DPM approach, which processes each AI separately.

RQ3: How do One-shot and Few-shot learning improve performance compared to zero-shot on the FinCase-AES dataset?

We investigate the impact of One-shot and Few-shot learning techniques on improving the performance of zero-shot learning for both unified and discrete prompting methods across all AIs. We aim to enhance the model's ability to generalize from limited labeled data.

The rest of the paper is organized as follows. Section II discusses the work related to automated grading task and language models. A detailed description of our dataset, prompt structures, and methods are provided in Section III. Section IV presents the experimental results, the evaluation of prompting methods, and In-Context learning. Finally, we conclude in Section VI.

II. Literature Review

The ability of students to showcase their intellectual development in a specific field through writing is a crucial aspect of the academic process, as emphasized by Hyland [13]. However, manual grading of essays is time-consuming and often lacks consistency both within and across raters. AES systems aim to address these issues by reducing the graders' workload and enhancing grading consistency [2]. Traditionally, AES tasks were approached as standard machine learning problems, where textual features from essays were extracted and used to train models with scores as labels [14]. With advancements in deep learning, AES problems have been tackled using methods like Convolutional Neural Networks (CNNs) [15], Long Short Term Memory Networks (LSTMs) [16], and pre-trained language models [17]. The Bidirectional Encoder Representations from Transformers (BERT) model has proven to be state-of-the-art for AES tasks, offering precise score predictions.

Recent studies have explored the capabilities of open-source LLMs for automated grading tasks. Table I lists the relevant studies which utilize various language models with different prompt engineering techniques for AES tasks. Most studies utilize the ASAP dataset[1], which comprises 12,978 essays scored on eight different levels and evaluated using Cohen's Quadratic Weighted Kappa (QWK) as the primary metric. Beyond ASAP, research has also included multilingual essay datasets in languages such as Chinese [7], Japanese [19], and Turkish [6]. Typically, these essays are written on specific topics and graded on a scale from 0 to n, with n representing the highest score, and the scoring rubrics clearly defining the criteria for each level. In this study, we utilize a unique essay grading dataset where, instead of assigning a single score to the entire essay, we evaluate finance case study answers based on six core concepts. Each concept is graded individually using language models. Table I highlights our study and illustrates the differences in methodology, prompt engineering, and evaluation metrics.

III. Methodology

This section provides a detailed description of the dataset, the structure of prompts, and the language models used in our study with In-Context learning, concluding with experimental details.

A. Dataset

For the automated grading task, we utilize the FinCase-AES (Finance Case Studies - Automated Essay Scoring) dataset. This proprietary dataset includes case-study-based questions related to accounting and general finance, assessing students'

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

TABLE I: Supporting literature for AES task utilizing LLMs.

Study	Dataset	Method	Models	Metrics
Firoozi et al. [6]	Turkish AES	Fine-tuning	BERT, GPT	QWK
Helmeczi et al. [5]	Proprietary (Finance)	Classification using PET and SetFit	BERT, DeBERTa	F1-score, Accuracy
Lee et al. [18]	ASAP & TOEFL	Multi-trait specialization	GPT-3.5, Llama2, Mistral	QWK
Stahl et al. [2]	ASAP	Persona prompts	Mistral, Llama2	QWK
Takeuchi and Okgetheng [19]	Japanese Essay Data (300)	Fine-tuning using Lora	Open-calm	Accuracy
Mansour et al. [20]	ASAP	Four distinct prompts with and without rubrics	GPT, Llama2	QWK
Lee et al. [21]	1,000 Science essays	In context learning with CoT	GPT-3.5, GPT-4	Accuracy
Song et al. [22]	2,870 Chinese Essay	Standard and Role playing prompts	LR, SVM, ChatGLM	QWK
Xiao et al. [7]	ASAP, 13,372 Chinese Essay	Fine-tuning LLMs	BERT, GPT-4, GPT3.5, Llama3	QWK
Our study	**FinCase-AES (Proprietary)**	**Unified and Discrete prompting**	**GPT-3.5, GPT-4o, Mistral, Gemini**	**Balanced Accuracy, Macro F1-score**

understanding of financial reporting, revenue recognition, inventory valuation, and financial risk assessment. Each input in the dataset consists of multiple paragraphs addressing the case-study questions in English. The dataset comprises a total of 92 student essays. General statistics about these essays are presented in Table II.

TABLE II: FinCase-AES dataset characteristics

	Essay Length		Sentence Count	
Dataset	Avg.	Std.	Avg.	Std.
FinCase-AES	245.6	78.6	12.7	5.0

To grade these essays, we define six main assessment indicators (AIs). Each AI evaluates a specific concept in financial reporting. Reviewers can assign one of three grades to each AI in a student essay: 'Y' for Correct, 'P' for Partially Correct, and 'N' for Incorrect. Table III shows the distribution of grades for each AI across the 92 essays. Notably, AI-1 and AI-5 exhibit significant class imbalance and lack the 'N' grade. This information should provide a clear understanding of the task while respecting the proprietary nature of the dataset.

After assigning grades to each AI, we apply straightforward rules to determine the overall ordinal grades, which are defined as follows: 'A' being the highest grade and 'D' being the lowest grade. Since we are working with proprietary datasets,

TABLE III: Grades distribution across each AI. (-) indicates absence of grade.

Grades	AI-1	AI-2	AI-3	AI-4	AI-5	AI-6
Y	91	80	72	52	87	43
P	-	9	13	29	-	30
N	1	3	1	11	5	19

our industrial experts manually graded six AIs for 92 essays. These manually assigned grades provide a benchmark to evaluate the performance of language models in generating grades automatically.

- A: Competent with Distinction
- B: Competent
- C: Reconsideration Required
- D: Not Competent

To better understand the correlation between essay length and overall grade, we convert the grades to numerical values ranging from 1 to 4 and plot these grades against essay length in Figure 2. Our analysis shows a weak correlation, with a value of 0.06, indicating no significant relationship between essay length and overall grade. This suggests that students who provide correct answers to the AIs receive good grades regardless of the length of their essays.

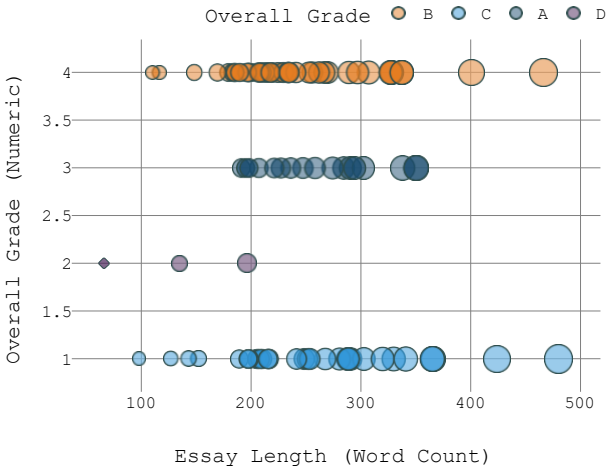


Fig. 2: Correlation of overall grade of essay with length of the essay.

B. Structure of Prompts

To apply the language models to student essays, we first curate the prompts for each AI that needs to be graded. Table V shows the prompt structure of each AI. Specifically, AI-2, AI-3, AI-4, and AI-6 are further divided into two components each. After predicting the component grades as 'Y' or 'N', we apply rule-based functions to determine the final grade as 'Y', 'P', or 'N'. Each AI provides instructions or rubrics to assess the student's essays. Furthermore, we propose two different prompt structures for the automated grading task defined below. Table V presents the prompt structures in the Unified Prompting Method (UPM) and the Discrete Prompting Method (DPM).

- **Unified Prompting Method**: In the UPM approach, the student essay is graded for each AI in one go, producing the result in a specific output format, as provided in Table V.
- **Discrete Prompting Method**: the DPM approach involves passing the prompts for each AI one by one, along with the student essays, to grade each AI separately. There is no output format provided in DPM, as language models are expected to generate the grade only without any explanation.

C. In context Learning

This section explains the application of In-Context learning for grading task with FinCase-AES dataset. We explore how providing one or multiple exemplary essays, together with their grades helps with essay-scoring task. Below, we define the types of In-Context learning utilized in our study.

- **Zero-shot**: In Zero-shot settings, we apply the student essays directly using the defined prompting techniques described in Section III-B. Zero-shot learning involves making predictions on new, unseen classes without any specific examples from those classes during training.

TABLE IV: Prompt definition for each AI.

AI	Prompt
AI-1	**Mark Y**: If student discuss the topic of Performance management **Mark N**: If students didn't discuss the topic
AI-2-C1	**Mark Y**: If the student identifies key metrics associated with the allocation of responsibilities and benefits. **Mark N**: If the student omits discussion on the key metrics.
AI-2-C2	**Mark Y**: If the student states that the key metric is "not achieved" and provides examples or explanations, such as "not achieved – the manager's responsibility remains as the machine still requires maintenance." **Mark N**: If the student only mentions whether the criteria are achieved without further explanations or examples.
AI-3-C1	**Mark Y**: If the student discusses performance metrics related to pricing. **Mark N**: If the student fails to address the performance management criteria related to pricing.
AI-3-C2	**Mark Y**: If the student states that retention metrics are "satisfactory" and provides examples or explanations, such as "satisfactory – new training improved retention by 5.6% " **Mark N**: If the student states that the criteria are met or not but does not provide any further explanations or examples demonstrating a deeper understanding.
AI-4-C1	**Mark Y**: If the student states that Return on Investment (ROI) is "satisfactory"; and provides examples with reasonable ROI calculations. **Mark N**: If the student fails to discuss ROI or does not provide examples or calculations
AI-4-C2	**Mark Y**: If the student explains that the comparison between ROI and RI is "satisfactory" and provides examples or explanations, such as "Department A's ROI exceeds RI due to higher returns."; **Mark N**: If the student states whether the criteria are met or not without further explanations or examples.
AI-5	**Mark Y**: If the student provides a recommendation on whether department's manager should receive bonus. The correctness of the decision is not the focus, but rather that a recommendation is made. **Mark N**: If no recommendation is provided.
AI-6-C1	**Mark Y**: If the student accurately discusses the financial impact of "Asset Depreciation." The specific details of the depreciation calculation are not necessary as long as the student addresses the impact on asset value, depreciation expense, or net book value. **Mark N**: If no impact is provided.
AI-6-C2	**Mark Y**: If the student provides a future improvement strategy for department's performance. Provide next steps. **Mark N**: If no future improvement strategy is provided.

Here, the language model leverages auxiliary information

TABLE V: Prompting structure applied in a unified and discrete manner.

Unified Prompting	Discrete Prompting

CONTEXT:
`You are tasked with grading student essays on the topic of revenue recognition. The essays should be evaluated based on the following rubric and only show grades as 'Y' or 'N'. No explanations required.`
PROMPT:
AI−1: Mark Y or N:
...
AI−6: Mark Y or N:
OUTPUT FORMAT:
`For each student response, provide the following in a list [(AI name, Grade)]`
STUDENT ESSAY:
[Insert student essay here]
	CONTEXT:
	`You are tasked with grading student essays on the topic of revenue recognition following with student essays and rubrics. Only show the grade no explanations.`
	PROMPT:
	AI−1: Mark Y or N:
	STUDENT ESSAY:
	[Insert student essay here]
	...
	PROMPT:
	AI−6: Mark Y or N:
	STUDENT ESSAY:
	[Insert student essay here]

and generalizes from the known prompts to predict grades for the essays. This approach eliminates the need for labeled examples.

- **One-shot**: For One-shot learning, we enhance the model's ability to predict grades by providing it with one sample text from the student essays for each AI. This example is randomly selected and accompanied by the actual grade marked by a human expert. The model uses this single example to learn and make accurate predictions for grades [2, 7].
- **Few-shot**: In Few-shot, we further improve the model's predictive capabilities by providing three examples for each AI in our experiments. These examples are randomly selected and represent varied instances of the same grade category, specifically focusing on the 'Y' grade. By offering a small, diverse set of labeled examples, the model gains a broader understanding of the grading context and criteria [2, 7]. This approach allows the model to generalize better by seeing multiple representations of the 'Y' grade, thereby enhancing its ability to apply the grading rubric accurately to new student essays.

D. Essay Grading Baselines

Similar to the approach used in the works of Yang et al. [23], Han et al. [24], and Xiao et al. [7], we fine-tune a simple BERT [25] pre-trained model checkpoint using the FinCase-AES dataset for each AI. Below, we define our baseline to compare the efficacy of the language models.

- BERT-Original: For this baseline, we fine-tune the Bert-base-uncased model checkpoint for each AI. This model is trained using actual student essay tokens without any pre-processing. Due to significant class imbalance for AI-1 and AI-5, the BERT model is fine-tuned for 2 classes, while for the rest of the AIs, fine-tuned for 3 classes of grades.
- BERT-Summarized: In this model, we first generate summaries of the essays using a pre-trained BART[2] model checkpoint. These summaries are then used as input for fine-tuning the BERT-base-uncased checkpoint.

E. Language Models

For this study, we focused on state-of-the-art LLMs for the automated essay grading task using the FinCase-AES dataset. These models have larger context windows, enabling them to process and understand longer texts more effectively, which is crucial for grading lengthy essays. LLMs generate human-like text, making them ideal for producing natural and coherent language output, enhancing the quality of automated grading by closely mimicking human evaluators without fine-tuning.

However, using these models for larger input sizes incurs costs, as they are hosted by major industries on their servers. While T5 model variants could be alternatives, they have smaller context windows of 512 tokens and limited generalizing capabilities without fine-tuning, making them less suitable for our task. Below, we briefly describe the LLMs employed in our experiments.

- **GPT-3.5**: Developed by OpenAI, GPT-3.5 [9] is a large-scale language model with 175 billion parameters. It supports extensive text generation tasks and has a token limit of 4,096 tokens per request, enabling the processing of long-form text efficiently.

[2]https://huggingface.co/docs/transformers/en/model_doc/bart

- **GPT-4o**: GPT-4o [10] is OpenAI's newest flagship model, offering GPT-4-level intelligence but with enhanced speed and multimodal capabilities, including text, voice, and vision inputs and outputs. It has a context window of 128,000 tokens, making it suitable for more complex tasks. GPT-4o is also more cost-efficient and faster compared to its predecessors, improving performance across various applications.
- **Gemini-1.5-pro**: This model, although less commonly referenced than GPT models, is known for its specialized capabilities in domain-specific tasks. It supports a token limit of 128,000 tokens, allowing it to manage detailed and specific text generation and comprehension tasks effectively [12].
- **Mistral8x7B**: A highly specialized model with 56 billion parameters (8 layers of 7 billion each), designed for optimized performance in structured data and text generation. It has a token limit of 32,000 tokens, making it well-suited for intricate and lengthy text analysis and generation tasks [11].

F. Experimental Setup

To assess the performance of language models on the FinCase-AES dataset, we first extract the grades from the language model outputs. Given the unpredictable nature of these models in terms of token prediction, we carefully post-process the dataset to match the grades extracted from the LLMs with the actual grades assigned by experts.

For evaluation, we consider two metrics to account for the imbalanced grade distribution across all AIs. First, we use balanced accuracy, which is essentially the average recall of each grade within each AI. Second, we report the macro F1-score, which is the harmonic mean of precision and recall for the grades within each AI. Since we have six AIs, we also report the average performance of all models across all AIs using the average of balanced accuracy and macro F1-score. Table VI details the hyperparameters used for fine-tuning the baselines and employing LLMs for the automated grading task. For LLMs, we experimented with different temperature values ranging from 0.0 to 0.6, with 0.2 and 0.0 proving to be the most efficient. All experiments were repeated for three random trials due to the limited number of instances in our dataset.

TABLE VI: Hyperparameter settings

Model	Size	Hyperparameter values
BERT-Original	110M	Epoch = 5, Batch_size = 4
BERT-Summarized	110M	Epoch = 5, Batch_size = 4
GPT-3.5	1800B	Temp = 0.0
GPT-4o	175B	Temp = 0.2
Gemini-1.5-pro	1.5B	Temp = 0.2
Mistral8x7B	56B	Temp = 0.2

IV. Results

This section presents the numerical study to compare the effectiveness of language models for the FinCase-AES dataset.

RQ1: How do state-of-the-art language models perform for each Assessment Indicator (AI) in the grading task compared to existing methods?

Table VII shows the performance of various language models compared to the baseline. All language models surpass the Bert-Original baseline, with GPT-4o and Gemini-1.5-pro also surpassing the Bert-Summarized baseline. The average macro F1-score indicates that Bert-Summarized performs particularly well for AI-4 (84.6%) and AI-6 (74.4%), where other language models struggle. This may be because the summarized input helps BERT models learn the grades more effectively for AI-6. GPT-4o shows strong performance for AI-1 and AI-5, likely due to the simplicity of the prompts, making it easier for language models to predict these grades. For fine-tuning baselines, the grade distribution requires a substantial number of samples for each grade in each AI to learn the patterns effectively. GPT-3.5 and Mistral8x7B models struggle to beat the baseline performance. GPT-3.5, being a relatively simpler model with a smaller size and shorter context window compared to the newly released GPT-4o, accounts for its mediocre performance with the FinCase-AES dataset.

RQ2: Which prompting strategy, UPM or DPM, is more effective for the automated grading task?

Table VII also compares the proposed prompting strategies for the FinCase-AES dataset. For the UPM strategy, GPT-4o and Gemini-1.5-pro stand out with average macro F1-scores of 71.1% and 70.8%, respectively. For the DPM strategy, GPT-4o proves to be the best model with a macro F1-score of 72.8%. Figure 3 shows the average balanced accuracy for the language models. In terms of balanced accuracy, Gemini-1.5-pro is the best model with the UPM strategy, while GPT-4o excels with the DPM approach. We also observe that the performance of language models with UPM is more consistent, as indicated by smaller standard deviation values, compared to the DPM approach.

RQ3: How do One-shot and Few-shot learning improve performance compared to zero-shot on the FinCase-AES dataset?

Table VIII presents the In-Context Learning results in terms of average macro F1-score and average balanced accuracy. One-shot learning shows minor improvements in performance, likely because a single sample instance may confuse the model, and more samples are needed for better context. Due to token limit constraints, we experimented only with values of n equal to 1 and 3. Few-shot results demonstrate a significant improvement over Zero-shot settings, with an increase of 2-3% in macro F1-score for GPT-4o and Gemini-1.5-pro using both UPM and DPM prompting strategies.

TABLE VII: Comparison of prompting strategies with language models in zero-shot settings versus Fine-Tuning BERT models. Performance metrics are presented as "mean ± standard deviation". The best performance is highlighted in bold and marked with (*) if it surpasses the fine-tuning baseline.

Model	AI-1	AI-2	AI-3	AI-4	AI-5	AI-6	Avg (F1)
Bert-Original	0.486 ±0.000	0.380 ±0.144	0.440 ±0.124	0.328 ±0.093	0.657 ±0.297	0.643 ±0.232	0.489 ±0.097
Bert-Summarized	0.486 ±0.000	0.487 ±0.176	0.568 ±0.318	0.846 ±0.087	0.486 ±0.000	0.744 ±0.128	0.603 ±0.057
UPM							
GPT-3.5	1.000 ±0.000	0.349 ±0.080	0.537 ±0.271	0.355 ±0.116	0.833 ±0.289	0.294 ±0.129	0.561 ±0.051
GPT-4o	1.000 ±0.000	0.571 ±0.209	0.468 ±0.041	0.738 ±0.018	1.000 ±0.000	0.489 ±0.170	**0.711*** ±0.009
Gemini-1.5-pro	1.000 ±0.000	0.613 ±0.220	0.608 ±0.117	0.801 ±0.080	0.768 ±0.261	0.459 ±0.134	0.708* ±0.080
Mistral8x7B	1.000 ±0.000	0.375 ±0.020	0.476 ±0.217	0.354 ±0.042	0.824 ±0.305	0.316 ±0.118	0.558 ±0.087
DPM							
GPT-3.5	0.658 ±0.297	0.495 ±0.151	0.401 ±0.211	0.574 ±0.099	0.653 ±0.301	0.250 ±0.101	0.505 ±0.091
GPT-4o	1.000 ±0.000	0.589 ±0.257	0.625 ±0.138	0.803 ±0.040	0.829 ±0.297	0.521 ±0.031	**0.728*** ±0.118
Gemini-1.5-pro	1.000 ±0.000	0.253 ±0.095	0.246 ±0.198	0.371 ±0.052	0.819 ±0.314	0.340 ±0.033	0.505 ±0.087
Mistral8x7B	1.000 ±0.000	0.506 ±0.182	0.360 ±0.102	0.547 ±0.105	0.829 ±0.297	0.177 ±0.080	0.570 ±0.078

TABLE VIII: Evaluation Results of In-Context learning (One-shot and Few-shot) with Language Models. Performance metrics are represented as "mean ± standard deviation". Performance improvements compared to zero-shot settings are highlighted in bold.

	One-shot (n = 1)			
	UPM		**DPM**	
Model	Avg. (F1)	Avg. (Acc)	Avg. (F1)	Avg. (Acc)
GPT-3.5	0.569 ±0.048	0.613 ±0.057	0.555 ±0.048	0.619 ±0.071
GPT-4o	0.679 ±0.006	0.724 ±0.015	**0.714** ±0.041	0.747 ±0.050
Gemini-1.5-pro	0.691 ±0.051	0.788 ±0.008	0.562 ±0.040	0.649 ±0.016
Mistral8x7B	0.580 ±0.029	0.660 ±0.009	0.618 ±0.066	0.647 ±0.062
	Few-shot (n = 3)			
	UPM		**DPM**	
GPT-3.5	0.542 ±0.028	0.571 ±0.027	0.618 ±0.071	0.653 ±0.073
GPT-4o	**0.749** ±0.048	0.779 ±0.052	**0.759** ±0.037	0.768 ±0.026
Gemini-1.5-pro	**0.736** ±0.034	0.795 ±0.052	0.568 ±0.056	0.678 ±0.066
Mistral8x7B	0.638 ±0.019	0.596 ±0.018	0.577 ±0.244	0.604 ±0.254

Figure 4 shows the performance changes in macro F1-score metrics for each AI in the FinCase-AES dataset using the UPM approach. In Figure 4a, we observe no improvements for AI-1 and minor improvements for AI-3. Additionally, the performance of language models decreases for AI-6 and AI-2. Most models show improvement for AI-5, with Mistral 8x7B benefiting significantly from One-shot learning. In Figure 4b, there is no improvement for AI-1 as it was already performing well in Zero-shot settings. However, Few-shot learning shows better improvements compared to One-shot learning, with only minor performance drops observed.

Similarly, Figure 5 shows the changes in macro F1-scores for each AI with the application of In-Context learning using the DPM approach. In Figure 5a, we observe that Gemini-1.5-pro shows improvements for AI-2 and AI-4, suggesting that the complexity of these prompts is better understood by the model with sample instances. AI-5 improvements are seen across all language models, although there is a performance drop for GPT-3.5. In Figure 5b, maximum improvements are observed for each AI across all language models. While no clear pattern of performance improvement is visible, Few-shot learning consistently outperforms One-shot learning in terms of performance gains.

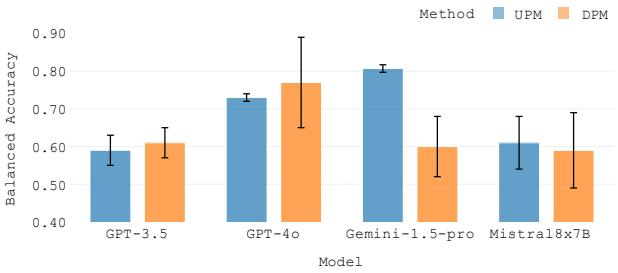


Fig. 3: Comparison of prompting strategies for language models in terms of average balanced accuracy across AIs over three random trials

V. Discussion and Study Limitations

Our results highlight the efficacy of language models over existing baselines for the finance domain essay grading task. We find that discrete prompting works better for the GPT-4o model, while unified prompting works best for the Gemini-1.5-pro model. However, there are some limitations to our study: Firstly, the dataset we used is limited in terms of student essays. Although preliminary experiments show that the language models perform well, we believe this performance will translate to a larger set of essays as well.

Secondly, there are challenges with the post-processing of outputs from language models. The GPT-4o model, being very advanced and relatively new, follows prompt instructions well and generates precise results, producing only grades in the desired format. However, other models generate extra explanations or different grade formats, such as numbers or different letters instead of 'Y' or 'N'. We found it challenging to tune our prompts accordingly. While we attempted to create generic prompts for all models, they worked well with GPT-4o but required post-processing for the other models.

Thirdly, there is the cost of processing input tokens and output inference. These language models are larger in size and cannot be easily uploaded and experimented with on-premises. Consequently, we rely on hosted servers, and the cost of accessing these servers through APIs is tied to the input and output token costs. On average, we spent $10.58 with GPT-4o for 92 essays, $2.33 with GPT-3.5, $2.71 with Mistral, and $12.23 with Gemini. While this cost is manageable for 92 essays, it would increase drastically with a larger dataset.

VI. Conclusion

In this paper, we explore the application of language models for the automated grading of student essays in the finance domain. Our primary focus is on evaluating the consistency and effectiveness of these models using different prompting techniques. The proprietary dataset used in this study tests students' financial knowledge through case-study-based questions, with each essay graded on six different assessment indicators. We conducted experiments comparing various language models against standard Automated Essay Scoring (AES) models using the FinCase-AES dataset. We propose and evaluate two distinct prompt engineering approaches: the unified prompting method and the discrete prompting method.

Our results indicate that the newly released GPT-4o model, with its advanced capabilities, outperforms a fine-tuned BERT model using both unified and discrete prompting methods. The Gemini-1.5-pro model emerged as the second-best model in our analysis. We found that language models perform well with simpler assessment indicators (AIs), such as AI-1 and AI-5, but struggle with more complex ones like AI-2 and AI-6.

In-Context learning techniques demonstrated overall improvements in performance across all AIs, with Few-shot learning proving more effective and consistent than One-shot learning for the FinCase-AES dataset. Our findings suggest that with precise and well-tuned prompts, language models can deliver consistent performance for AES tasks. However, there are notable limitations, including the larger model size, data security concerns, and the higher cost associated with input processing and output inference.

For future work, we plan to experiment with a larger number of student responses to further validate our findings. Additionally, we aim to explore similar datasets to strengthen the robustness of our experiments. Fine-tuning the language models specifically for the AES task also appears to be a promising direction for improving performance and warrants further investigation.

References

[1] E. B. Page, "The imminence of... grading essays by computer," *The Phi Delta Kappan*, vol. 47, no. 5, pp. 238–243, 1966.

[2] M. Stahl, L. Biermann, A. Nehring, and H. Wachsmuth, "Exploring llm prompting strategies for joint essay scoring and feedback generation," *arXiv preprint arXiv:2404.15845*, 2024.

[3] D. Ramesh and S. K. Sanampudi, "An automated essay scoring systems: a systematic literature review," *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2495–2527, 2022.

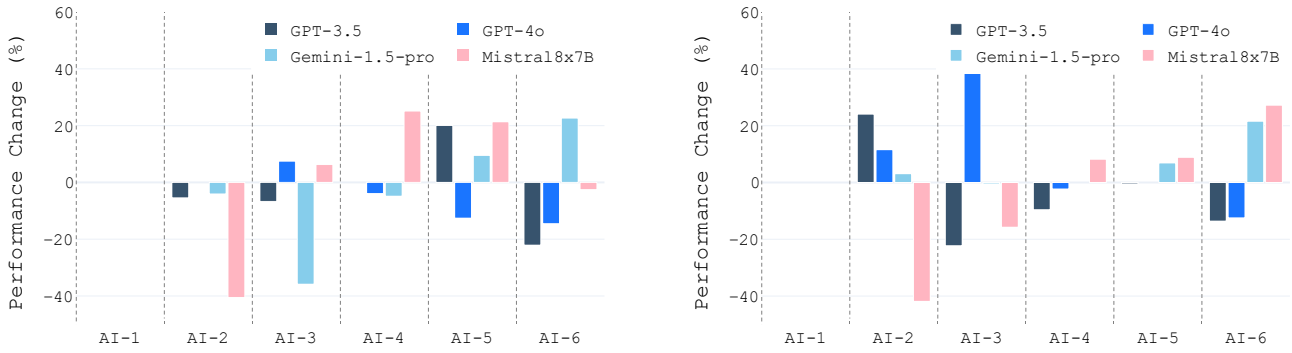
[4] Z. Ke and V. Ng, "Automated essay scoring: A survey of the state of the art." in *IJCAI*, vol. 19, 2019, pp. 6300–6308.

[5] R. K. Helmeczi, S. Yildirim, M. Cevik, and S. Lee, "Few shot learning approaches to essay scoring." in *Canadian AI*, 2023.

[6] T. Firoozi, O. Bulut, and M. G˙IERL, "Language models in automated essay scoring: Insights for the turkish language," *International Journal of Assessment Tools in Education*, vol. 10, no. Special Issue, pp. 149–163, 2023.

[7] C. Xiao, W. Ma, Q. Song, S. X. Xu, K. Zhang, Y. Wang, and Q. Fu, "Human-ai collaborative essay scoring: A dual-process framework with llms," *arXiv preprint arXiv:2401.06431*, 2024. [Online]. Available: https://arxiv.org/abs/2401.06431

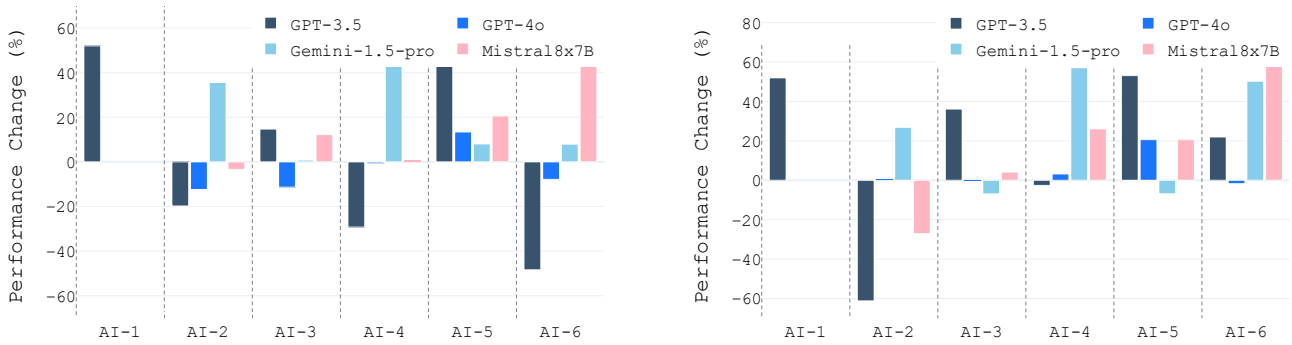
[8] M. Kostic, H. F. Witschel, K. Hinkelmann, and M. Spahic-Bogdanovic, "Llms in automated essay eval-



(a) One-shot

(b) Few-shot

Fig. 4: Average Macro-F1 performance change in % for each AI in One-shot and Zero-shot experiments with UPM approach.



(a) One-shot performance improvement

(b) Few-shot performance improvement

Fig. 5: Average Macro-F1 performance change in % for each AI in One-shot and Zero-shot experiments with DPM approach.

uation: A case study," in *Proceedings of the AAAI Symposium Series*, vol. 3, no. 1, 2024, pp. 143–147.

[9] J. Ye, X. Chen, N. Xu, C. Zu, Z. Shao, S. Liu, Y. Cui, Z. Zhou, C. Gong, Y. Shen *et al.*, "A comprehensive capability analysis of gpt-3 and gpt-3.5 series models," *arXiv preprint arXiv:2303.10420*, 2023.

[10] O. Platform, "gpt-4o-2024-05-13," 2024. [Online]. Available: https://platform.openai.com/docs/models/gpt-4o

[11] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7b," 2023.

[12] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivie`re, M. S. Kale, J. Love *et al.*, "Gemma: Open models based on gemini research and technology," *arXiv preprint arXiv:2403.08295*, 2024.

[13] K. Hyland, "Writing in the university: Education, knowledge and reputation," *Language teaching*, vol. 46, no. 1, pp. 53–70, 2013.

[14] Y. Salim, V. Stevanus, E. Barlian, A. C. Sari, and D. Suhartono, "Automated english digital essay grader using machine learning," in *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*. IEEE, 2019, pp. 1–6.

[15] F. Dong and Y. Zhang, "Automatic features for essay scoring–an empirical study," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 1072–1077.

[16] K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in *Proceedings of the 2016 conference on empirical methods in natural language*

processing, 2016, pp. 1882–1891.

[17] J. Lun, J. Zhu, Y. Tang, and M. Yang, "Multiple data augmentation strategies for improving performance on automatic short answer scoring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 09, 2020, pp. 13 389–13 396.

[18] S. Lee, Y. Cai, D. Meng, Z. Wang, and Y. Wu, "Prompting large language models for zero-shot essay scoring via multi-trait specialization," *arXiv preprint arXiv:2404.04941*, 2024.

[19] K. Takeuchi and B. Okgetheng, "Estimating japanese essay grading scores with large language models," *Language Resources and Evaluation*, vol. 58, no. 2, pp. 345–367, 2024.

[20] W. Mansour, S. Albatarni, S. Eltanbouly, and T. Elsayed, "Can large language models automatically score proficiency of written essays?" *arXiv preprint arXiv:2403.06149*, 2024.

[21] G.-G. Lee, E. Latif, X. Wu, N. Liu, and X. Zhai, "Applying large language models and chain-of-thought for automatic scoring," *Computers and Education: Artificial Intelligence*, p. 100213, 2024.

[22] Y. Song, Q. Zhu, H. Wang, and Q. Zheng, "Automated essay scoring and revising based on open-source large language models," *IEEE Transactions on Learning Technologies*, 2024.

[23] R. Yang, J. Cao, Z. Wen, Y. Wu, and X. He, "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*, 2020, pp. 1560–1569.

[24] J. Han, H. Yoo, J. Myung, M. Kim, H. Lim, Y. Kim, T. Y. Lee, H. Hong, J. Kim, S.-Y. Ahn *et al.*, "Fabric: Automated scoring and feedback generation for essays," *arXiv preprint arXiv:2310.05191*, 2023.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.