

FinNLP-AgentScen 2024

**Joint Workshop of the 8th Financial Technology
and Natural Language Processing (FinNLP) and
the 1st Agent AI for Scenario Planning
(AgentScen) in conjunction with IJCAI 2023**

Proceedings of the Workshop

August 3, 2024

Jeju, South Korea

Preface

Welcome to FinNLP, a forum dedicated to fostering international collaboration and knowledge-sharing in the application of Natural Language Processing (NLP) within the dynamic domain of FinTech. As we convene for this workshop, our goal is to explore the intersection of FinTech and NLP, identifying challenging problems, shaping future research directions, and expanding the horizons of this interdisciplinary field. This year's 8th FinNLP is held in conjunction with the 1st Agent AI for Scenario Planning (AgentScen) workshop.

Agent AI is emerging as a crucial research direction following significant advancements in multimodal large language models. This evolution opens new avenues for using AI in business analysis while also increasing the complexities involved in scenario planning. Scenario planning is particularly valuable in fields characterized by high uncertainty. Traditional forecasting methods often fall short in areas such as long-term strategic planning, geopolitics, and emerging industries. Although scenario planning is not a new concept, its integration with modern technologies like NLP presents exciting new opportunities. Recent research highlights how generative NLP technologies can enhance, automate, and diversify scenario planning. NLP's ability to process vast textual datasets, detect emerging trends, and generate detailed narratives makes it a powerful tool for scenario planners.

The modern era, marked by rapid technological changes, geopolitical instability, and evolving socio-economic landscapes, epitomizes the VUCA paradigm—Volatile, Uncertain, Complex, and Ambiguous. In this environment, traditional predictive methods, which rely on linear extrapolations of current trends, are inadequate. Unlike predictive models that aim for precision based on historical and current data, scenario planning ventures into a different realm. It doesn't just forecast an extrapolated future; it constructs multiple narratives, each illuminating a potential future. Scenario planning acknowledges the multifaceted, uncertain nature of the future, considering various driving forces and uncertainties to craft stories that present alternative futures, including those that may seem improbable. This approach is about preparing for a spectrum of possibilities rather than predicting a single, precise outcome.

Aligned with this perspective, we are launching a new workshop to discuss the potential of agent AI in scenario planning. Given its strategic importance in business, we believe this

is an excellent opportunity to integrate with FinNLP, which focuses on all financial-related topics. This collaboration allows participants in FinNLP to brainstorm and contribute to advancements in agent AI for scenario planning.

This workshop would not have been possible without the contributions of numerous individuals, and we extend our heartfelt gratitude to each of them. Special thanks to Dr. Jimin Huang and his dedicated team for successfully orchestrating the shared task named Financial Challenges in Large Language Models (FinLLM). We are also deeply grateful to all the program committee members who invested substantial time and expertise in providing insightful feedback on submissions and guiding the selection process for FinNLP-AgentScen-2024.

Lastly, we extend our sincere thanks to the project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO), for their invaluable financial support. Their partnership has been instrumental in enabling us to realize the goals of FinNLP-AgentScen and advance research in this domain.

We hope that FinNLP-AgentScen will continue to serve as a catalyst for groundbreaking research and meaningful connections, propelling the fields of FinTech and NLP towards new heights of innovation and excellence.

Chung-Chi Chen, Tatsuya Ishigaki, Hiroya Takamura, Akihiko Murai, Suzuko Nishino,
Hen-Hsen Huang, Hsin-His Chen
FinNLP-AgentScen 2024 Organizers
August 2024

Organizations

Organizing Committee

Chung-Chi Chen, AIST, Japan

Tatsuya Ishigaki, AIST, Japan

Hiroya Takamura, AIST, Japan

Akihiko Murai, AIST, Japan

Suzuko Nishino, AIST, Japan

Hen-Hsen Huang, Institute of Information Science, Academia Sinica, Taiwan

Hsin-Hsi Chen, Department of Computer Science and Information Engineering, National Taiwan University, Taiwan

Shared Task Organizers

Qianqian Xie, The FinAI, Singapore

Xiao-yang Liu, Open Finance, Columbia University, US

Yangyang Yu, Stevens Institute of Technology, US

Dong Li, Wuhan University, China

Benyou Wang - Chinese University of Hong Kong, Shenzhen, China

Alejandro Lopez-Lira, University of Florida, US

Yanzhao Lai - Southwest Jiaotong University, China

Min Peng, Wuhan University, China

Sophia Ananiadou, University of Manchester, UK; Archimedes RC, Greece

Hao Wang, Sichuan University, China

Jimin Huang, The FinAI, Singapore

Zhengyu Chen, Wuhan University, China

Ruoyu Xiang, New York University, US

VijayaSai Somasundaram, University of Florida, US

Kailai Yang, University of Manchester, UK

Chenhan Yuan, University of Manchester, UK

Zheheng Luo, University of Manchester, UK

Zhiwei Liu, University of Manchester, UK

Yueru He, Columbia University, US

Yuechen Jiang, Stevens Institute of Technology, US

Haohang Li, Stevens Institute of Technology, US

Duanyu Feng, Sichuan University, China

Program Committee

Paulo Alves, Católica Porto Business School
Yupeng Cao, Stevens Institute of Technology
Damir Cavar, Indiana University Bloomington
Yung-Chun Chang, Taipei Medical University
Nelson Correa, Andinum, Inc.
SarmisthaDas, Indian Institute of Technology, Patna
Jinhua Du, Investments AI, AIG
Pablo Duboue, Textualization Software Ltd.
Ismail El Maarouf, Imprevicible
Duanyu Feng, Sichuan University
Lubingzhi Guo, University of Glasgow
Sarang Gupta, Columbia University
Yueru He, Columbia University
Kiyoshi, Izumi, The University of Tokyo
Jinhang Jiang, McKesson Corporation
Ashvini Kumar Jindal, LinkedIn
Juyeon Kang, 3DS Outscale
Ken Kawamura, Revelata
Meisin Lee, Monash University
Haohang Li, Stevens Institute of Technology
Chit-Kwan Lin, Revelata, Inc.
Zhiwei Liu, University of Manchester
Xiao-Yang Liu, Columbia University
Alejandro Lopez-Lira, University of Florida
Zheheng Luo, University of Manchester
Hilal Pataci, University of Texas at San Antonio
Elvys Linhares Pontes, Trading Central Labs
Pawan Rajpoot, Rakuten India
Hiroki Sakaji, Hokkaido University
Selja Seppälä, University College Cork
Varad Srivastava, Barclays
Ke Tian, Rakuten Group, Inc
Kanji Uchino, FUJITSU RESEARCH OF AMERICA, INC.
Chuan-Ju Wang, Academia Sinica
Shih-Hung Wu, Chaoyang University of Technology

Kailai Yang, University of Manchester
Yangyang Yu, Stevens Institute of Technology
Chenhan Yuan, The University of Manchester

Table of Contents

Using Pre-trained Language Model for Accurate ESG Prediction	
Lei Xia, Mingming Yang and Qi Liu -----	1
Can GPT models be Financial Analysts? An Evaluation of ChatGPT and GPT-4 on mock CFA Exams	
Ethan Callanan, Amarachi Mbakwe, Antony Papadimitriou, Yulong Pei, Mathieu Sibue, Xiaodan Zhu, Zhiqiang Ma, Xiaomo Liu and Sameena Shah -----	23
Examining the Effect of News Context on Algorithmic Trading	
Surupendu Gangopadhyay and Prasenjit Majumder -----	33
GPT-Signal: Generative AI for Semi-automated Feature Engineering in the Alpha Research Process	
Yining Wang, Jinman Zhao and Yuri Lawryshyn -----	42
RACCOON: Real-world Advanced financial analysis through COmprehensive Natural language dataset	
Seonghyun Kim, Kanghee Lee, Minsu Jeong and Junghan Yoon -----	54
TopoLedgerBERT: Topological Learning of Ledger Description Embeddings using Siamese BERT-Networks	
Sander Noels, Sébastien Viaene and Tijl De Bie -----	63
Probing Numerical Concepts in Financial Text with BERT Models	
Shanyue Guo, Le Qiu and Emmanuele Chersoni -----	73
GENWISE: Thematic Discovery from Textual Data	
Minnie Kabra, Abhinav Nagpal, Aayush Sacheti, Mohit Kumar and Salil Joshi -----	79
FINALE : Finance Domain Instruction-Tuning Dataset with High-Quality Rationales via Chain-of-Thought Prompting	
Sangmin Lee, Suzie Oh, Saeran Park, Guijin Son and Pilsung Kang -----	89
Capturing Analysts' Questioning Strategies in Earnings Calls via a Question Cornering Score (QCS)	
Giulia D'Agostino, Andrea Rocci and Chris Reed -----	107
FinNLP-AgentScen-2024 Shared Task: Financial Challenges in Large Language Models - FinLLMs	
Qianqian Xie, Jimin Huang, Dong Li, Zhengyu Chen, Ruoyu Xiang, Mengxi Xiao, Yangyang Yu, Vijayasai Somasundaram, Kailai Yang, Chenhan Yuan, Zheheng Luo, Zhiwei Liu, Yueru He, Yuechen Jiang, Haohang Li, Duanyu Feng, Xiao-Yang Liu, Benyou Wang, Hao Wang, Yanzhao Lai, Jordan Suchow, Alejandro Lopez-Lira, Min Peng and Sophia Ananiadou -----	119
University of Glasgow at the FinLLM Challenge Task: Adapting Llama for Financial News Abstractive Summarization	
Lubingzhi Guo, Javier Sanz-Cruzado and Richard McCreadie -----	127
Wealth Guide: A Sophisticated Language Model Solution for Financial Trading Decisions	
Sarmistha Das, R E Zera Marveen Lyngkhai, Sriparna Saha and Alka Maurya -----	133
L3iTC at the FinLLM Challenge Task: Quantization for Financial Text Classification & Summarization	
Elvys Linhares Pontes, Carlos-Emiliano González-Gallardo, Mohamed Benjannet,	

Caryn Qu and Antoine Doucet -----	141
Revelata at the FinLLM Challenge Task: Improving Financial Text Summarization by Restricted Prompt Engineering and Fine-tuning	
Ken Kawamura, Zeqian Li, Chit-Kwan Lin and Bradley McDanel -----	146
'Finance Wizard' at the FinLLM Challenge Task: Financial Text Summarization	
Meisin Lee and Lay-Ki Soon -----	153
Upaya at the FinLLM Challenge Task 1 and 2: DistFin: Distillation based Fine-Tuning for Financial Tasks	
Ashvini Kumar Jindal, Pawan Kumar Rajpoot and Ankur Parikh -----	159
BAI-Arg LLM at the FinLLM Challenge Task: Earn While You Argue - Financial Argument Identification	
Varad Srivastava -----	165
CatMemo@IJCAI 2024 FinLLM Challenge: Fine-Tuning Large Language Models using Data Fusion in Financial Applications	
Yupeng Cao, Zhiyuan Yao, Zhi Chen and Zhiyang Deng -----	174

Using Pre-trained Language Model for Accurate ESG Prediction

Lei Xia^{1,2*}, Mingming Yang², Qi Liu^{1†},

¹The University of Hong Kong, ²Tencent AI Lab,

brianleixia@connect.hku.hk, shanemmyang@tencent.com, liuqi@cs.hku.hk

Abstract

Environmental, Social, and Governance (ESG) has been crucial in investment decision-making in recent years, with an increase in ESG-centric research emerging. Concurrently, Natural Language Processing (NLP) has emerged in analyzing ESG-related texts. However, there is a lack of models and datasets specifically tailored for ESG categorization. This study presents a novel approach leveraging Pretrained Language Models (PLMs) and Large Language Models (LLMs) to tackle ESG text classification tasks. We introduce a pipeline for creating specialized datasets for ESG analysis by using keyword search and LLMs APIs to label data. Through continued pre-training PLMs such as BERT, DistilRoBERTa, and RoBERTa on our datasets, our approach significantly surpasses traditional baseline performances. We also introduce ESGLlama and FinLlama, domain-specific models derived from Llama2, with FinLlama demonstrating exceptional efficacy in financial benchmarks and ESG text comprehensions¹. Final evaluations reveal that our models achieve significant advancements in ESG classification, outperforming established baselines.

1 Introduction

Environmental, Social, and Governance (ESG) considerations represent the cornerstone of contemporary sustainable or responsible investment strategies. Over the past decade, ESG has become the preeminent framework for socially responsible investments and decision-making within the financial sector. However, a significant challenge remains relying on voluminous annual sustainability reports for informed decision-making. The comprehensive nature of these reports necessitates substantial effort for thorough analysis, highlighting the critical

demand for automated solutions. In this context, Natural Language Processing (NLP) emerges as an indispensable tool, enabling navigating through extensive sustainability narratives and extracting pivotal ESG insights precisely.

Recent advancements in NLP have streamlined the identification and interpretation of ESG information, enabling more nuanced analysis. This research background sets the stage for exploring the integration of NLP in enhancing the efficiency and depth of ESG analysis. Additionally, existing research has applied pre-trained language models (PLMs) in ESG-related NLP tasks such as climate change-related text detection and controversy detection (Nugent et al., 2020; Huang et al., 2023; Schimanski et al., 2023; Webersinke et al., 2021). However, a significant gap exists in the processing and collecting textual ESG data. This results in a scarcity of publicly accessible, high-quality ESG textual datasets, especially for established text categorization tasks within the ESG domain. Recent developments in large language models (LLMs) are more powerful than small PLMs and have demonstrated their potential in performing various NLP tasks like language understanding and generation. But, no such research focuses on using LLMs to solve ESG-related tasks.

In this study, we tackle the significant gap in the availability of ESG-related datasets and apply PLMs and LLMs to challenging ESG classification tasks. We also use keyword search and LLMs APIs to annotate datasets for both 4-class and 9-class ESG classification. Further, we enriched our dataset collection with conversational history data, which proved crucial for Supervised Fine-Tuning (SFT) processes. SFT, a pivotal concept in our approach, refers to refining pre-trained models by training them on labeled datasets specific to the target task, thus enabling the models to learn task-specific patterns and adapt to the problem domain. This comprehensive fine-tuning involved

*Work was done when Lei Xia was interning at Tencent AI Lab.

†Qi Liu is the corresponding author.

¹Our code and data can be found at <https://github.com/brianleixia/LLM4ESGPrediction>

both PLMs and LLMs, significantly boosting their performance in ESG-related tasks. Moreover, we developed two fine-tuned LLMs, ESGLLama and FinLlama, based on the Llama2, which demonstrated substantial improvements over baseline models. FinLlama also excelled in financial benchmarks. In summary, our key contributions are the following:

- We propose a pipeline by utilizing keyword search and LLMs APIs to annotate data and construct three datasets for ESG analysis: pre-training corpus, classification dataset, and ESG SFT dataset.
- We introduce three domain-specific PLMs: ESG-BERT, ESG-DistilRoBERTa, and ESG-RoBERTa. These models notably surpass their base models and baseline.
- We conduct two fine-tuned Llama2 models: ESGLLama and FinLlama. FinLlama exhibits remarkable improvements in financial benchmarks.
- We compare PLMs and LLMs across various experimental settings, comprehensively analyzing their performance.

2 Datasets Construction Pipeline

In response to the notable scarcity of datasets tailored for ESG domain analysis, we propose a pipeline, as illustrated in Figure 1, which encompasses data preprocessing, labeling procedures, and model training to enhance ESG data analysis capabilities systematically. Initially, data is sourced from various open sources and cleansed according to predefined rules. During the preprocessing phase, data is preliminarily categorized using keyword searches. Subsequent labeling employs APIs from LLMs to ensure high classification accuracy. Human evaluations are conducted to validate the labeled data, which then facilitates the construction of specialized datasets for further model pre-training and fine-tuning.

Specifically, we have constructed three types of datasets to enhance the accuracy of ESG prediction tasks: (1) Pre-training Dataset. This expansive corpus of ESG-related texts is designed to bolster the initial training of domain-specific models, thereby improving their ability to interpret ESG contexts accurately. (2) Classification Datasets. These datasets

are segmented into four-class and nine-class categories for ESG texts, playing a pivotal role in the fine-tuning process to enhance model precision in ESG categorization. (3) SFT Dataset. Tailored for the Supervised Fine-Tuning (SFT) of Large Language Models (LLMs), this dataset incorporates conversational data generated by LLMs during the labeling procedure to boost the models' proficiency in ESG classification tasks.

2.1 Data Collection and Processing

For data collection, we searched and collected datasets mainly from two resources: huggingface² and kaggle³. Refer to more details of our collected data in Appendix B. After data collection, we extract textual content pertinent to ESG analysis. We standardized the datasets to a **sentence-level** format in the initial data processing phase, facilitating uniform analysis across diverse data sources. Following the standardization, a data-cleaning procedure was implemented. This involved the removal of URLs and special characters from the text, ensuring that the datasets were devoid of extraneous information that could potentially skew the analysis. The processed data amounted to approximately 18 million sentences.

2.2 Data Labeling

Keyword Search The keyword search initiates data identification across ESG subdomains, segregating text relevant to Env, Soc, Gov and Non-ESG content. This meticulous process enabled us to partition the corpus into distinct segments, each corresponding to a specific aspect of ESG. While this method predominantly isolated relevant ESG-related text, it is essential to acknowledge that it might not entirely preclude the presence of Non-ESG data within these preliminary datasets. We argue that Non-ESG data within the pre-training phase could inadvertently enhance the model's robustness by exposing it to a broader spectrum of textual content. Details of keywords are in Appendix C.

After filtering the texts by keyword searching, we got the preliminary results in Table 8. To validate the effectiveness of our classification approach, these visualizations effectively confirm the appropriateness of the categorized data, with predominant terms such as "GHG emission" and "climate change" in the Environmental domain, "human

²<https://huggingface.co/>

³<https://www.kaggle.com/>

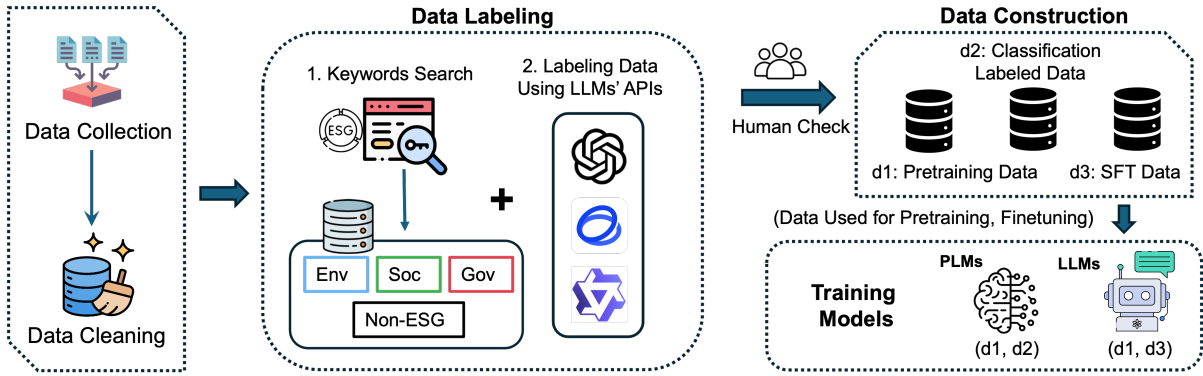


Figure 1: The work pipeline encompasses data collection, preprocessing, and labeling, followed by model training. Data is initially collected from open sources and cleansed. Using keyword searches and enhancing label accuracy through LLM’s APIs, with further validation by human evaluation. The resultant dataset is used for pre-training and fine-tuning classification tasks.

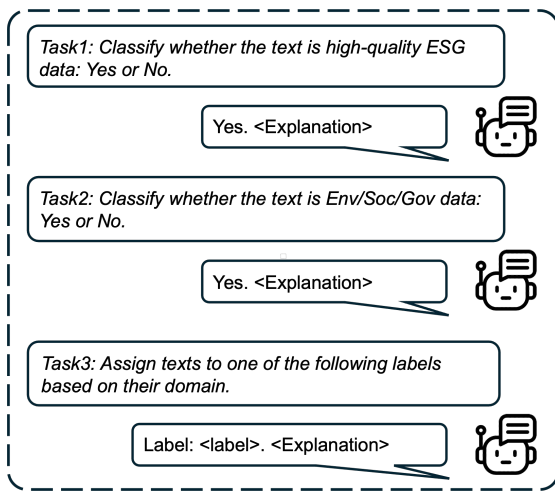


Figure 2: Representation of task decomposition and the task descriptions alongside responses from LLM

rights" and "customer" in Social, and "director" and "financial statement" in Governance, reflecting the accurate representation of domain-specific high-frequency words. Our next objective was to refine the accuracy of our labeled data further. To achieve this, we planned to leverage LLMs for an additional layer of filtering and validation. Details of visualizations are in Appendix D.

Labeling Data Using LLMs Before labeling the data, we recognized a complexity gradient in categorization tasks, where tasks with fewer categories are inherently simpler than those with more. Studies such as Bang et al. (2023) (Bang et al., 2023) suggest that LLMs may underperform in specific, challenging downstream tasks, including multi-class classification tasks. To address this, we devised a structured approach to simplify the ESG classification challenge, as depicted in Figure 2. In this stage, the overall task is divided into three simpler tasks, where Task1 and Task2

comprise the four-class task (Env, Soc, Gov, Non-ESG), and an additional Task3 is required to construct the nine-class task. Specifically, the nine-class classification involves three environmental categories (Climate Change, Natural Capital, Pollution and Waste), three social categories (Human Capital, Product Liability, Community Relations), two governance categories (Corporate Governance, Business Ethics and Values), and one Non-ESG category. The final three categories of the nine-class task are unified into a single ternary (3-class) task, applying the same categorization principles as the four-class task but with an added layer of specificity. Significantly, this ternary categorization is based on data already classified under the four-class schema, further refining the process.

For each sub-task, we employed APIs from three different LLMs: Qwen (*qwen-max*), GLM (*glm-4*), and GPT-3.5 (*gpt-3.5-turbo-instruct*). This multi-model strategy was underpinned by several rationales: Firstly, LLMs are prone to ‘hallucination’, often generating less reliable outputs due to their randomness. Utilizing multiple models helps mitigate significant data bias and enhances the diversity of the labeled data. Secondly, the decision to leverage several LLMs’ APIs was economically driven, aiming to reduce costs associated with extensive data filtering and labeling tasks. Lastly, employing multiple models concurrently significantly enhances the efficiency of the data labeling process. Details regarding the prompt design and an example of LLM response are in Appendix F.

2.3 Data Construction and Analysis

Pre-training Dataset In constructing the pre-training dataset, we initially aggregated datasets categorized as Env, Soc, and Gov based on key-

word searches. Recognizing the challenges associated with processing excessively long texts, we implemented a filtration step to exclude these from the dataset. Then, we executed a 90-10 split to segregate the data into training and evaluation subsets. The evaluation set is crucial in monitoring the training loss and establishing an early stop during the pre-training phase.

Classification Dataset The development of the labeled classification dataset involved multiple meticulous steps. Initially, we processed the outputs from the llms used for each classification task and subjected these to a rigorous human review to verify the LLM-generated classifications. This review process was crucial as it helped refine the data for the four-class and nine-class categorizations, specifically excluding Non-ESG data due to its inherent complexities and the limitations of LLM outputs, which may not always guarantee the absolute accuracy of the responses. Consequently, the Non-ESG dataset was compiled in a two-fold approach: approximately 8,500 samples were selected from the LLM responses, and an additional 5,500 samples were isolated following a keyword search, cumulatively amounting to around 14,000 Non-ESG samples. A notable issue identified was the class imbalance within the nine-class dataset. To rectify this, we implemented a normalization strategy by capping the maximum number of instances per class at 3,000, leading to a more balanced distribution. Furthermore, we applied stratified sampling for both datasets to ensure equitable class representation. Details of dataset distribution are in Appendix E.

Supervised Fine-Tuning Dataset SFT is a critical refinement process in NLP, enhancing a large language model’s adaptability to specific tasks. This alignment improves the model’s precision and adaptability for specific tasks. In line with best practices like those demonstrated by the Alpaca model (Taori et al., 2023), its instruction dataset has three fields: instruction, input, and output. We constructed a SFT Dataset similarly for ESG classification tasks with the following instructions:

1. Identification of ESG-related text: "If the following text is ESG related data."
2. Four-Class classification: "Classify the following text into one of the four ESG categories: {categories}."

3. Nine-category Class: "Classify the following text into one of the nine ESG categories: {categories}."

The dataset preparation involved reformatting existing four-class and nine-class datasets to align with these instructions, generating 95,412 data points. We also employed stratified sampling to select about 28,000 data points, ensuring diverse and balanced coverage across the instructions.

3 Methodology

3.1 Pre-trained Based Method

Baseline Our baseline employs FinBERT (Huang et al., 2023), a model adapted from BERT for the financial sector. FinBERT has been extended to address ESG-related classifications.

Datasets The dataset used for pre-training, detailed in Section 2.3, comprises 5,257,347 training sentences and 584,150 validation sentences, obtained via keyword search. While keyword searches are prone to including non-ESG phrases, resulting in false positives, this is beneficial for pre-training. It allows the model to learn the broader context of sustainability topics by exposing it to relevant and irrelevant samples.

Training Models As detailed in Section 2.3, we utilized this dataset to pre-train models including BERT (Devlin et al., 2019), DistilRoBERTa (Sanh et al., 2019), and RoBERTa (Liu et al., 2019). Instead of starting from scratch, we engaged in Continual Pre-Training (CPT), a strategy that allows a model to assimilate new data while preserving previously acquired knowledge. This approach is advantageous for adapting models to evolving data streams or new, unseen data. By continuing to pre-train on an established model’s checkpoint, we infused domain-specific ESG knowledge into the models. Consequently, we selected the model with the smallest validation loss as our final pre-training models: ESG-BERT, ESG-DistilRoBERTa, and ESG-RoBERTa. Details regarding pre-training process are in Appendix G.

3.2 LLM Based Method

Baseline Llama2 (Touvron et al., 2023), an open-source large language model. We choose Llama2 (Llama2-7b-chat-hf) as a baseline for the ESG classification task.

Datasets Our LLM-based methods utilize two main types of datasets: the pre-training corpus and Supervised Fine-Tuning (SFT) datasets. The pre-training corpus has been substantially expanded to include not only the ESG-related texts discussed in Section 2.3 but also a significant volume of financial texts, primarily sourced from financial reports, totaling 5,282,943 sentences. For SFT, we employed two distinct datasets. The first SFT dataset, as introduced in Section 2.3, consists of conversational data generated during the labeling of ESG data using LLMs. The second SFT dataset is more extensive, integrating the conversational data and additional financial instruction tuning data as outlined in FinGPT (Wang et al., 2023) and the ESG_Chat dataset⁴. The ESG_Chat dataset comprises dialogues between humans and LLMs, focusing on strategies to enhance ESG scores. Then, we adopted a targeted sampling strategy, producing a refined subset of 86,425 sentences.

Fine-tuning Models To enhance the LLM’s understanding of ESG-related themes, we enriched the model with ESG-related knowledge, resulting in the creation of two specialized models: *ES-GLlama* and *FinLlama*. ESGLlama underwent fine-tuning through Supervised Fine-Tuning (SFT) using conversational data tailored for ESG classification tasks, notably improving its accuracy within ESG contexts (as discussed in datasets, the first SFT dataset). Meanwhile, FinLlama was developed to tackle a broader spectrum of financial tasks, integrating extensive financial texts and targeted instruction-tuning data, ranging from sentiment analysis to financial Question Answering (QA). For fine-tuning FinLlama, we employed a two-stage training approach. Initially, the Llama2 model underwent Continual Pre-Training (CPT) using a combined corpus of ESG-centric texts and additional financial documents, including financial news and annual reports. Subsequently, in the second stage, we conducted supervised fine-tuning on the model pre-trained in the initial phase using the second SFT dataset (as discussed in datasets).

4 Experiments

4.1 Test on Public Dataset

To evaluate the generalizability of our trained models for ESG-related tasks, we conducted tests using

⁴https://huggingface.co/datasets/zadhart/ESG_Chat

publicly available datasets: environmental_2k⁵, social_2k⁶ and governance_2k⁷ which are derived from annual reports spanning 2017-2021. Each dataset is expertly annotated for binary classification, where '0' indicates "No" and '1' denotes "Yes" outcomes. We fine-tuned our models ESG-BERT, ESG-RoBERTa, and ESG-DistilRoBERTa on these datasets with a partitioning scheme of 64% training, 16% validation, and 20% testing.

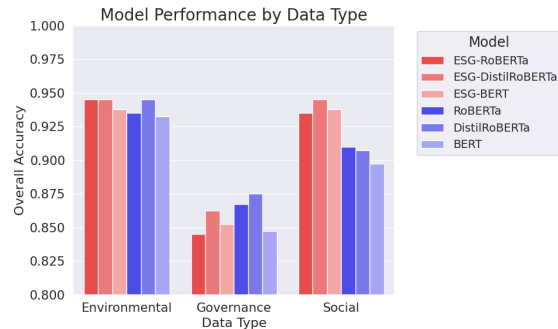


Figure 3: Overall perf. of models on public datasets

To evaluate the effectiveness of our models, the results are shown in Table 1. As we can see, the pre-trained models demonstrate superior performance compared to the baselines across the publicly accessible dataset. Notably, all pre-trained models consistently outperform their corresponding baseline models within the Social domain shown in Figure 3. Among them, ESG-DistilRoBERTa stands out with the highest precision (0.9415), recall (0.9449), and F1 score (0.9431), indicating robust performance. In the Environmental domain, ESG-RoBERTa shows remarkable precision (0.9436) and an equivalent F1 score, underscoring its effectiveness.

However, the Governance domain exhibits a contrasting scenario, with mixed results despite pre-training enhancements. The baseline DistilRoBERTa model outperforms the pre-trained versions in this domain, achieving the highest metrics with a precision of 0.8404, recall of 0.8444, and F1 score of 0.8424. This discrepancy suggests that while pre-training generally enhances model capabilities, its impact is less pronounced in the Governance domain. The observed variance may stem from misalignments between the pre-training content and the specifics of the publicly available

⁵https://huggingface.co/datasets/ESGBERT/environmental_2k

⁶https://huggingface.co/datasets/ESGBERT/social_2k

⁷https://huggingface.co/datasets/ESGBERT/governance_2k

Model	Env			Soc			Gov		
	P	R	F1	P	R	F1	P	R	F1
BERT	0.9207	0.9285	0.9244	0.8960	0.8899	0.8927	0.8048	0.8168	0.8104
<i>ESG-BERT</i>	0.9300	0.9284	0.9292	0.9354	0.9345	0.935	0.8141	0.8085	0.8112
DistilRoBERTa	0.9340	0.9436	0.9385	0.9035	0.9044	0.9039	0.8404	0.8444	0.8424
<i>ESG-DistilRoBERTa</i>	0.9364	0.9397	0.9380	0.9415	0.9449	0.9431	0.8252	0.8271	0.8261
RoBERTa	0.9279	0.9246	0.9262	0.9041	0.9135	0.9076	0.8292	0.8421	0.8352
<i>ESG-RoBERTa</i>	0.9340	0.9436	0.9385	0.9311	0.9345	0.9327	0.8048	0.7976	0.8011

Table 1: Performance metrics across environmental, social, and governance domains on public datasets. **Bold** shows the best results among baseline and corresponding pre-trained model, and underlined indicates the best results in each column.

governance data, suggesting a need to refine the fine-tuning parameters better to tailor the models to this domain’s nuances.

4.2 Test on Classification Datasets

Evaluate PLMs We fine-tuned our pre-trained models on ESG classification tasks (four-class and nine-class) using our constructed classification data. The training parameters were standardized at a batch size of 32 across 50 epochs while learning rates were adjusted based on model and task specifics. For the four-class classification, the learning rates were set at $3e-6$ for the BERT model and $1.25e-6$ for both DistilRoBERTa and RoBERTa. For the nine-class task, BERT was fine-tuned at $3e-6$, DistilRoBERTa at $1.75e-6$, and RoBERTa at $1.15e-6$. These rates were meticulously selected to optimize each model’s performance on its respective task. An *early stopping* mechanism was implemented during fine-tuning to curb overfitting and enhance computational efficiency. The models chosen for further utilization demonstrated the best performance on the validation set across the 50 epochs, specifically those achieving the lowest validation loss.

Table 2: Four-Class Evaluation Results of PLMs

Model	P	R	F1	Acc
FinBERT	0.7357	0.7150	0.7165	0.7222
BERT	0.8668	0.8658	0.8641	0.8667
dtRoBERTa	0.8672	0.8687	0.8662	0.8684
RoBERTa	0.8610	0.8596	0.8582	0.8602
ESG-BERT	0.9074	0.9077	0.9071	0.9083
ESG-dtRoBERTa	0.9027	0.9040	0.9014	0.9034
ESG-RoBERTa	0.9086	0.9100	0.9086	0.9102

To assess the effectiveness of our pretrained models, we conducted tests on two sets: a four-class and a nine-class classification task, with results detailed in Table 2 and Table 3, respectively. The

Table 3: Nine-Class Evaluation Results of PLMs

Model	P	R	F1	Acc
FinBERT	0.7160	0.7154	0.7081	0.7273
BERT	0.8393	0.8357	0.8361	0.8419
dtRoBERTa	0.8240	0.8153	0.8179	0.8239
RoBERTa	0.8187	0.8196	0.8174	0.8275
ESG-BERT	0.8606	0.8637	0.8617	0.8693
ESG-dtRoBERTa	0.8575	0.8552	0.8556	0.8616
ESG-RoBERTa	0.8611	0.8591	0.8592	0.8662

evaluations included baseline models, our specifically pre-trained models, and their base models. For the four-class task, ESG-RoBERTa excelled, achieving the highest metrics with a precision of 0.9086, a recall of 0.9100, an F1 score of 0.9086, and an accuracy of 0.9102, significantly surpassing the baseline finbert-esg model, which only reached an accuracy of 0.7222. This demonstrates a clear superiority over the baseline, with even the base models outperforming finbert-esg when fine-tuned. In the nine-class task, ESG-BERT led with the highest recall of 0.8637 and an F1 score of 0.8617, while ESG-RoBERTa achieved the top accuracy of 0.8662. These results highlight the advantages of our ESG-specific pretraining and fine-tuning strategy, markedly improving upon the performance of the baseline finbert-esg-9-categories model.

Evaluate LLMs We will evaluate the performance of the baseline and our fine-tuned models across six different *experimental settings*: Zero-Shot, One-Shot, In-Context Learning (ICL), Zero-Shot with Chain of Thought (CoT) (Kojima et al., 2023), One-Shot with CoT, and ICL with CoT. The dataset used for SFT in ESG text classification was constructed from ESG SFT data, as detailed in Section 2.3. More details about the ESG classification SFT dataset can be found in Appendix I. To process the results from our models, particularly the baseline, we utilized a *regular expression* matching technique to extract predicted labels from model

outputs. Details regarding classification prompts design are in Appendix J.

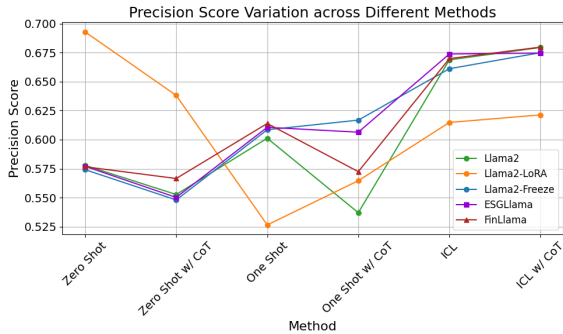


Figure 4: Four-Class Precision of LLMs

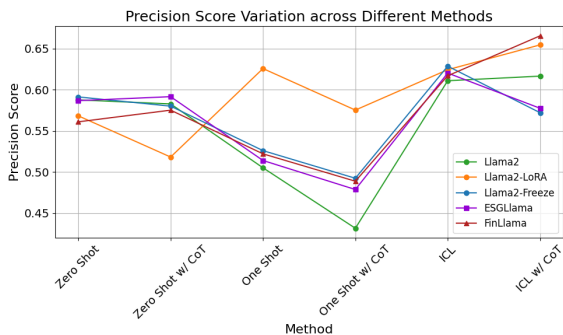


Figure 5: Nine-Class Precisions of LLMs

For four-class classification, Figure 4 shows that both models consistently outperform the baseline across most experimental settings. Notably, even the baseline model improves significantly when subjected to SFT with our ESG classification dataset. Interestingly, the Freeze fine-tuning method generally surpasses the LoRA approach, except in zero-shot scenarios where LoRA excels, possibly indicating its tendency to overfit slightly. The integration of CoT prompts typically reduces performance in zero-shot and one-shot settings, except for ICL tasks. This reduction may stem from CoT’s incompatibility with classification tasks, which require straightforward decision-making rather than stepwise logic processing. However, incorporating demonstration examples in ICL tasks enhances the model’s grasp of classification logic, significantly improving outcomes in ICL-CoT settings by providing richer context and sample diversity. Furthermore, FinLlama achieves superior precision over ESGLlama with the addition of CoT.

In the nine-class classification, the increase in category complexity and diversity presents more significant challenges, as indicated by lower overall performance metrics than in the four-class scenario. Performance visualization in Figure 5 shows that

both ESGLlama and FinLlama substantially outperform the baseline across most configurations, affirming the enhanced capability of our fine-tuned models in handling ESG-related texts. FinLlama excels in ICL, mainly when provided with ample examples, showcasing its deep understanding of the financial domain. Conversely, the performance notably drops in one-shot learning scenarios, where providing a single instance per class introduces significant bias and variability, impairing the model’s accuracy. However, increasing the number of examples markedly improves performance, underscoring the benefits of more extensive training datasets. The comparison between LoRA and Freeze methods reveals that LoRA outperforms Freeze in one-shot settings, suggesting that LoRA’s parameter adjustments are better suited for absorbing limited class-specific information efficiently. Additional analyses are in Appendix K.

4.3 Test on Financial Benchmark

To assess the FinLlama model’s performance in financial NLP tasks, we evaluate it on FinGPT benchmark (Wang et al., 2023). Our evaluation concentrated on two critical tasks: financial text sentiment analysis and headline classification, utilizing the fingpt-headline dataset⁸. Results, presented in Table 4, clearly show that FinLlama significantly outperforms the baseline Llama2 model across these tasks. This superior performance across financial sentiment analysis and headline classification tasks validates the effectiveness of our constructed pre-training and Supervised Fine-Tuning (SFT) datasets.

Table 4: Perf. of models on Financial Benchmarks

Dataset	Llama2		FinLlama	
	Acc	F1	Acc	F1
FPB	0.4703	0.4140	0.7855	0.7838
FiQA	0.7964	0.7744	0.7782	0.8096
TFNS	0.3811	0.3037	0.8405	0.8408
NWGI	0.5656	0.4833	0.6501	0.6445
Headline	0.4314	0.6182	0.8783	0.6975

5 Results Analysis

Performance of Pre-trained Models Our analysis highlighted that classification task complexity increases with the number of categories. This was evident from the lower convergence rates in

⁸<https://huggingface.co/datasets/FinGPT/fingpt-headline>

the nine-class task compared to the four-class task. ESG-RoBERTa excelled in the four-class task due to its larger parameter set, which enhances its text understanding capabilities. In contrast, ESG-BERT performed better in the nine-class task, suggesting that its pretraining objectives and architecture might offer superior generalization across more diverse categories. Performance evaluations on a publicly available dataset confirmed the effectiveness of our pre-trained models, as shown in Table 1. Particularly in the Social domain, reflecting the quality of our pre-trained models. The extensive testing on a public dataset validated our pretraining dataset’s quality and demonstrated our models’ improved comprehension of ESG-related content, enhancing classification accuracy.

Performance of Large Models Both ESGLlama and FinLlama consistently outperform the baseline across most testing scenarios, with notable improvements in the baseline model following SFT with our ESG classification dataset. This enhancement highlights the dataset’s quality and the effectiveness of SFT. A distinct observation is Freeze is generally better than LoRA because the trend of line changes in its results is consistent with those of other experimental setups, and more examples can improve its results. The integration of CoT typically reduces performance in zero-shot and one-shot scenarios. Still, it improves outcomes in ICL tasks due to additional context and examples provided. Transitioning to a nine-class framework increases task complexity, generally lowering performance metrics. In ICL tasks, FinLlama shows superior proficiency, particularly when additional samples are included, reflecting its adeptness at navigating complex classification landscapes. Conversely, performance drops in one-shot scenarios, underscoring the challenges of minimal data learning. Moreover, testing FinLlama on financial benchmark further validates its superiority in financial NLP, highlighting FinLlama’s effectiveness in financial discourse analysis.

Case Study In this analysis, we compare the performance of ESG-RoBERTa and FinLlama in deciphering complex ESG-related texts. An example from Table 5 shows ESG-RoBERTa misclassifying a case as "Business Ethics and Values," likely due to its reliance on keywords such as "reputation" and "financial costs" associated with ethical issues. Conversely, FinLlama employs ICL with CoT reasoning to accurately categorize the exact

text as "Product Liability." It provides a detailed rationale, dissecting the text to link it to product liability issues and showcasing its ability to perform stepwise, human-like reasoning for accurate and interpretable results. FinLlama’s response highlights its adeptness at integrating extensive contextual information and performing better than PLMs in this scenario. This case study underscores fine-tuned LLMs’ advanced comprehension and reasoning capabilities in detailed, context-rich environments. Despite some limitations, it demonstrates considerable scope to further enhance LLMs for more effective ESG text categorization.

Text: *A warranty or product liability claim whether or not meritorious could harm our reputation result in substantial financial costs or divert managements attention which could have an adverse effect on our business operating results and financial condition.*
True Label: *Product Liability*

ESG-RoBERTa
Answer: *Business Ethics and Values* ✗

FinLlama (ICL-CoT):
Response: Label: Product Liability\n\nExplanation: The text mentions the potential for a warranty or product liability claim to harm the company’s reputation, result in financial costs, and divert management’s attention, which could have an adverse effect on the company’s business operations and financial condition. This falls under the category of Product Liability, as it relates to the potential legal liability of a company for any defects or hazards in its products.
Answer: *Product Liability* ✓

Table 5: An example of results of PLM and LLM.

6 Conclusion and Future Work

We proposed a pipeline to address the lack of ESG-related datasets, utilizing keyword searches and LLM APIs to annotate and construct three types of data for ESG text classification tasks. This approach has significantly enhanced the performance of pre-trained models on ESG classification tasks. We also introduced domain-specific LLMs, ESGLlama and FinLlama, which were fine-tuned on our datasets, marking a major advancement in applying LLMs to ESG-related challenges. Notably, FinLlama has surpassed existing financial benchmarks. Comparative analysis reveals that while PLMs generally perform better, LLMs offer greater interpretability and adeptly handle complex contexts by integrating contextual information. Moving forward, we will further evaluate our developed datasets, and leverage the superior classification accuracy of PLMs to enhance and refine LLMs’ performance in ESG analysis.

Limitations

The limitations of the current work include: (1) The present model is not equipped to handle long text data (e.g., document-level data) as our data are normalized to the sentence level. Future work will focus on model training and inference with long text data. (2) The current dataset presents a cross-domain issue, where a text may pertain to both environmental and governance categories. In the future, we will refine our dataset to enhance its classification clarity and granularity, ensuring texts are either distinctly classified into specific categories or appropriately labeled as belonging to multiple categories.

References

- Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. 2019. [Publicly Available Clinical BERT Embeddings](#). *arXiv preprint*. ArXiv:1904.03323 [cs].
- Alex Andonian, Stella Biderman, Sid Black, Preetham Gali, Leo Gao, Eric Hallahan, Josh Levy-Kramer, Connor Leahy, Lucas Nestler, Kip Parker, Michael Pieler, Shivanshu Purohit, Tri Songz, Wang Phil, and Samuel Weinbach. 2021. [GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch](#).
- Dogu Araci. 2019. [FinBERT: Financial Sentiment Analysis with Pre-trained Language Models](#). *arXiv preprint*. ArXiv:1908.10063 [cs].
- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multi-task, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.
- Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. [SciBERT: A Pretrained Language Model for Scientific Text](#). *arXiv preprint*. ArXiv:1903.10676 [cs].
- Julia Anna Bingler, Mathias Kraus, Markus Leippold, and Nicolas Webersinke. 2022. [Cheap talk and cherry-picking: What ClimateBert has to say on corporate climate risk disclosures](#). *Finance Research Letters*, 47:102776.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](#). *arXiv preprint*. ArXiv:1810.04805 [cs].
- Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2021. Glm: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360*.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. [Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks](#). *arXiv preprint*. ArXiv:2004.10964 [cs].
- Allen H. Huang, Hui Wang, and Yi Yang. 2023. [FinBERT: A Large Language Model for Extracting Information from Financial Text*](#). *Contemporary Accounting Research*, 40(2):806–841. [_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1911-3846.12832](https://onlinelibrary.wiley.com/doi/pdf/10.1111/1911-3846.12832).
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. [Large language models are zero-shot reasoners](#). *Preprint*, arXiv:2205.11916.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.
- Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. [BioBERT: a pre-trained biomedical language representation model for biomedical text mining](#). *Bioinformatics*, 36(4):1234–1240.
- Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. [Large Language Models in Finance: A Survey](#). *arXiv preprint*. ArXiv:2311.10723 [cs, q-fin].
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [RoBERTa: A Robustly Optimized BERT Pretraining Approach](#). *arXiv preprint*. ArXiv:1907.11692 [cs].
- Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2020. [FinBERT: A Pre-trained Financial Language Representation Model for Financial Text Mining](#). volume 5, pages 4513–4519. ISSN: 1045-0823.

- Dakuan Lu, Hengkui Wu, Jiaqing Liang, Yipei Xu, Qianyu He, Yipeng Geng, Mengkun Han, Yingsi Xin, and Yanghua Xiao. 2023. [BBT-Fin: Comprehensive Construction of Chinese Financial Domain Pre-trained Language Model, Corpus and Benchmark](#). *arXiv preprint*. ArXiv:2302.09432 [cs].
- Lu Lu, Jinghang Gu, and Chu-Ren Huang. 2022. Inclusion in csr reports: The lens from a data-driven machine learning model. In *Proceedings of the First Computing Social Responsibility Workshop within the 13th Language Resources and Evaluation Conference*, pages 46–51.
- Alexandra Luccioni, Emily Baylor, and Nicolas Duchene. 2020. Analyzing sustainability reports using natural language processing. *arXiv preprint arXiv:2011.08073*.
- Tim Nugent, Nicole Stelea, and Jochen L. Leidner. 2020. [Detecting ESG topics using domain-specific language models and data augmentation approaches](#). *arXiv preprint*. ArXiv:2010.08319 [cs].
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep Contextualized Word Representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Matthew Purver, Matej Martinc, Riste Ichev, Igor Lončarski, Katarina Sitar Šuštar, Aljoša Valentinčič, and Senja Pollak. 2022. Tracking changes in esg representation: Initial investigations in uk annual reports. In *Proceedings of the First Computing Social Responsibility Workshop within the 13th Language Resources and Evaluation Conference*, pages 9–14.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
- Zacharias Sautner, Laurence Van Lent, Grigory Vilkov, and Ruishen Zhang. 2023. Firm-level climate change exposure. *The Journal of Finance*, 78(3):1449–1498.
- Tobias Schimanski, Andrin Reding, Nico Reding, Julia Bingler, Mathias Kraus, and Markus Leippold. 2023. [Bridging the Gap in ESG Measurement: Using NLP to Quantify Environmental, Social, and Governance Communication](#).
- Dominik Stambach, Nicolas Webersinke, Julia Anna Bingler, Mathias Kraus, and Markus Leippold. 2022. A dataset for detecting real-world environmental claims. *Center for Law & Economics Working Paper Series*, 2022(07).
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models*. <https://crfm.stanford.edu/2023/03/13/alpaca.html>, 3(6):7.
- 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*.
- Francesco S. Varini, Jordan Boyd-Graber, Massimiliano Ciaramita, and Markus Leippold. 2021. [ClimaText: A Dataset for Climate Change Topic Detection](#). *arXiv preprint*. ArXiv:2012.00483 [cs].
- Neng Wang, Hongyang Yang, and Christina Dan Wang. 2023. [FinGPT: Instruction Tuning Benchmark for Open-Source Large Language Models in Financial Datasets](#). *arXiv preprint*. ArXiv:2310.04793 [cs, q-fin].
- Nicolas Webersinke, Mathias Kraus, Julia Anna Bingler, and Markus Leippold. 2021. Climatebert: A pretrained language model for climate-related text. *arXiv preprint arXiv:2110.12010*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhajan Kam-badur, David Rosenberg, and Gideon Mann. 2023. [BloombergGPT: A Large Language Model for Finance](#). ArXiv:2303.17564 [cs, q-fin].
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. [PIXIU: A Large Language Model, Instruction Data and Evaluation Benchmark for Finance](#). *arXiv preprint*. ArXiv:2306.05443 [cs].
- Yi Yang, Mark Christopher Siy UY, and Allen Huang. 2020. [FinBERT: A Pretrained Language Model for Financial Communications](#). *arXiv preprint*. ArXiv:2006.08097 [cs].
- Yuwei Yin, Yazheng Yang, Jian Yang, and Qi Liu. 2023. [FinPT: Financial Risk Prediction with Profile Tuning on Pretrained Foundation Models](#). *arXiv preprint*. ArXiv:2308.00065 [cs, q-fin].

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Xuanyu Zhang, Qing Yang, and Dongliang Xu. 2023. XuanYuan 2.0: A Large Chinese Financial Chat Model with Hundreds of Billions Parameters. *arXiv preprint*. ArXiv:2305.12002 [cs].

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyang Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

A Related Work

ESG Related NLP The exploration of textual data in ESG reports has seen a marked increase in interest, covering various research topics. Recent studies have expanded beyond traditional analyses by adopting machine learning models to address societal issues such as stereotypes and inclusivity (Lu et al., 2022). Furthermore, diachronic distributional techniques have been utilized to trace the evolution of ESG terminology, revealing shifts in discourse (Purver et al., 2022). Traditional research often employs keyword-based analysis methods (Sautner et al., 2023), which lack contextual sensitivity (Varini et al., 2021). Recent shifts toward context-aware machine learning models have improved performance in diverse tasks such as climate content classification (Webersinke et al., 2021), topic detection (Varini et al., 2021), Q&A systems (Luccioni et al., 2020), and claim detection and verification (Stammach et al., 2022). Deploying fine-tuned BERT models, especially those trained on extensive business and financial news corpora like the Reuters News Archive, has effectively identified ESG controversies (Nugent et al., 2020).

Pre-trained Language Models The advent of robust Pre-trained Language Models (PLMs) such as BERT (Devlin et al., 2019), ELMo (Peters et al., 2018), RoBERTa (Liu et al., 2019) has significantly boosted NLP task performance across diverse domains. While domain-specific pre-training further augments their performance in specialized fields (Gururangan et al., 2020), with dedicated models like BioBERT (Lee et al., 2020) for biomedicine, ClinicalBERT (Alsentzer et al., 2019) for clinical care, and SciBERT (Beltagy et al., 2019) for scientific texts demonstrating targeted advancements. Additionally, ClimateBERT (Bingler et al., 2022) specifically addresses climate risk assessment. The landscape of Large Language Models (LLMs) encompasses models like T5 (Raffel et al., 2020), and the OpenAI GPT series, beginning with GPT-3 (Brown et al., 2020), renowned for setting benchmarks in generative tasks. Other notable GPT-style models include PaLM (Chowdhery et al., 2023), and GPT-NeoX (Andonian et al., 2021), alongside GLM (Du et al., 2021). Despite many LLMs being proprietary, open-source models like OPT (Zhang et al., 2022) and LLaMA (Touvron et al., 2023) foster extensive research and practical applications. Despite these advances, the application of PLMs in the nuanced ESG domain remains nascent. Our

work seeks to bridge this gap, leveraging PLMs to enhance ESG analysis and categorization.

Financial Language Models The application of language models in finance is rapidly expanding, as these models are increasingly used for specialized functions such as risk assessment and information extraction (Li et al., 2023). For instance, BloombergGPT (Wu et al., 2023) was initially trained with a mix of general and finance-specific datasets using BLOOM176B, while Xuan Yuan 2.0 (Zhang et al., 2023) and Fin-T5 (Lu et al., 2023) focus on the Chinese financial market, leveraging specialized pre-training. Fine-tuning for financial models predominantly targets sentiment analysis, news categorization, question-answering, summarization, and entity recognition. Noteworthy adaptations include FinBERT (Liu et al., 2020; Yang et al., 2020; Araci, 2019; Huang et al., 2023). Emerging models like PIXIU (Xie et al., 2023), and FinGPT (Yin et al., 2023) exemplify the advanced application of LLaMA architectures tailored for financial tasks. Unlike previous work, we fine-tuned LLMs to address ESG classification in finance and conduct a comprehensive analysis.

B Details of Collected Data

Below are the descriptions of the datasets we collected:

- *ESG-Prospectus-Clarity-Category*⁹: This dataset comprising 1,155 entries categorized into four ESG language classes: Specific, Ambiguous, Generic, and Risk. These entries were systematically extracted from the "Principal Investment Strategy" sections of sustainable (ESG) fund prospectuses through a specialized data extraction pipeline.
- *Esg-sentiment*¹⁰: Featuring text across nine emotion classes within the ESG spectrum (<Environmental, Social, Governance> * <Negative, Neutral, Positive>), each emotion assigns binary labels (0/1).
- *ESGBERT base-data*¹¹: This dataset extracted 13,846,000 sentences from annual reports (13,079,890 sentences), responsibility reports (695,631 sentences), sustainable reports

⁹<https://huggingface.co/Abhijeet3922>

¹⁰<https://huggingface.co/datasets/TrajanovRisto/esg-sentiment>

¹¹https://huggingface.co/datasets/ESGBERT/base_data

(259,163 sentences) and articles (143,289 sentences).

- *Environmental_claims*¹²: This dataset focuses on the binary classification of environmental claims made by publicly listed companies, containing 2,647 entries. It is designed to detect real-world environmental assertions.
- *DAX ESG Media Dataset*¹³: Comprising approximately 11k recent English language ESG documents (text is document level) related to German DAX companies, this dataset includes both company issued reports and third party data, alongside an auxiliary file detailing the Sustainable Development Goals (SDGs).
- *CLIMATE-FEVER*¹⁴: This dataset consists of 1,535 real-world climate change claims. Each claim is supported by five Wikipedia-sourced evidence sentences annotated to either support, refute, resulting in a total of 7,675 claim-evidence pairs.

Our data extraction involved the retrieval of the 'text' field across datasets, except the *DAX ESG Media Dataset*, from which the 'content' field was extracted, and the *CLIMATE-FEVER*, where both the 'claim' and the 'evidence' fields within the 'evidences' array were extracted. The summary of datasets is shown in Table 6.

C ESG Keywords

All keywords we used shown in Table 7 refer to (Schimanski et al., 2023).

D Word Clouds of keyword search

After filtering the texts by keywords searching. The texts are categorized into Environmental (Env), Social (Soc), Governance (Gov), and Non-ESG groups. The word clouds generated from these texts shown in Figure 6 offer a visual representation of the predominant themes within each category. In the Environmental domain, the word cloud prominently features terms such as "GHG emission" and "climate change," highlighting the focus on environmental impact. Socially oriented texts

¹²https://huggingface.co/datasets/climatebert/environmental_claims

¹³<https://www.kaggle.com/datasets/equintel/dax-esg-media-dataset>

¹⁴<https://www.sustainablefinance.uzh.ch/en/research/climate-fever.html>

Table 6: Summary of Collected ESG-Related Datasets

Dataset Name	Content Format	Size
ESG-Prospectus-Clarity-Category	<Text, Label>	2310 rows (546 kB)
Esg-sentiment	<Text, Environmental Negative, . . . , Social Positive>	679 rows (80.1 kB)
ESGBERT base-data	<Text>	13,846,000 rows (2.33 GB)
Environmental_claims	<text, label>	2647 rows (272 kB)
DAX ESG Media	<company, content, datatype, data, domain, esg_topics, internal, symbol, title>	11455 rows (130.11 MB)
CLIMATE-FEVER	<claim_id, claim, claim_label, evidences>	1,535 rows (3 MB)

Table 7: ESG Keywords Across Domains

Domain	Keywords
Environmental	adaptation, agricultural, air quality, biodiversity, biomass, climate, CO2, conservation, consumption, diversity, ecosystem, emissions, energy, environmental, flood, forest, fossil fuel, GHG, global warming, green, greenhouse, land use, methane, mitigation, nature, ozone, pollution, renewable, soil, solar, sustainability, water, recycling, clean energy, natural
Social	age, culture, race, accessibility, accident, accountability, awareness, charity, community, consumer protection, cyber security, data privacy, discrimination, diversity, education, employee benefit, empowerment, equality, ethics, fairness, gender, health, inclusion, mental well-being, parity, privacy, quality of life, religion, safety, social impact, volunteerism, welfare, wellbeing, workforce
Governance	audit, authority, bribery, compliance, corporate governance, corruption, crisis management, due diligence, ethics, framework, integrity, legal, lobby, oversight, policy, regulation, reporting, risk management, stakeholder engagement, transparency, whistleblower, board diversity, executive pay, shareholder rights, sustainable governance, corporate transparency, anti-corruption, business ethics

are characterized by frequent mentions of "human rights," "product," and "customer," reflecting the emphasis on societal concerns and stakeholder welfare. In the Governance category, words like "director," "financial statement," "management," and "shareholder" dominate, aligning with expectations for governance-related discourse. These visual insights from the word clouds roughly correspond with our anticipated high-frequency words for each ESG classification, underscoring the effectiveness of our keyword-based filtering approach. we got the preliminary results shown in Table 8.

Table 8: Summary of Processed Data

Domain	Num. of Sent.	Avg. Num. of Words		
		Q1	Mean	Q3
Env	2,143,453	19	30.43	36
Soc	2,796,077	20	31.46	37
Gov	1,851,303	20	31.75	38
Non-ESG	11,392,832	-	-	-
Total	18,183,665	-	-	-

E Data Distribution

Four-class and nine-class categorization criteria defined by Huang et al. (Huang et al., 2023)



Figure 6: ESG Domain Word Clouds After Keywords Search

Pre-training Dataset. We performed a 90-10 train-eval split to create the training and evaluation datasets, as shown in Table 9.

Table 9: Pre-training Dataset Statistics

Dataset	Num. of Sentences
Train	5,257,347
Valid	584,150
Total	5,841,497

For the four-class dataset. We used a 70:15:15 splitting ratio to construct the train-dev-test sets. The training set consisted of 37,155 instances, with 10,144 'Soc', 9,799 'Non-ESG', 9,192 'Env', and 8,020 'Gov'. The validation and test set each contained 7,962 instances, with 2,174 'Soc', 2,100 'Non-ESG', 1,969 'Env', and 1,719 'Gov' for validation, and 2,174 'Soc', 2,100 'Non-ESG', 1,970 'Env', and 1,718 'Gov' for testing. Results are shown in Figure 7.

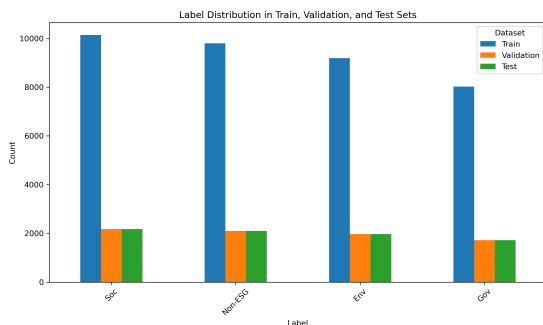


Figure 7: Four-class Label Distribution in Train, Val, Test Sets

For the nine-class dataset. We applied an 81:09:10 splitting ratio. The training set had 17,419 instances, with each label ('Human Capital', 'Prod-

uct Liability', 'Pollution and Waste', 'Business Ethics and Values', 'Corporate Governance', 'Community Relations', 'Non-ESG', 'Climate Change', 'Natural Capital'). The validation set contained 2,151 instances. Similarly, the test set had 1,936 instances. These datasets were constructed using stratified sampling to ensure a balanced representation of each class in the train-dev-test splits. Lastly, we fine-tuned our pre-trained models on these two datasets to adapt them for the four-class and nine-class ESG text classification tasks. Results are shown in Figure 8.

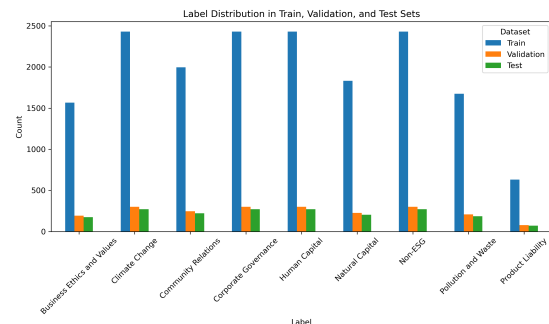


Figure 8: Nine-class Label Distribution in Train, Val, Test Sets

F LLM Labeling Prompts Design

We primarily utilize a combination of **few-shot learning** and Chain of Thought (CoT) in prompts design. Few-shot learning enables the model to learn from a limited quantity of text to align the acquired knowledge with our specific purpose. CoT (Wei et al., 2022) is a reasoning strategy that involves breaking down a problem into sub-problems and connecting them in a specific logical order based on a chain structure. The purpose of using a few shots is to familiarize the model with the ESG classification strategy using a small sample. Using

CoT is intended to enhance the model’s reasoning process.

For task 1: *Classify whether the text is high-quality ESG data: Yes or No.* The {Criteria} will be replaced by certain criteria, which are generated by GPT-4, and {Data} will be replaced by certain text we want to be classified.

System Prompt: "You are a helpful assistant in data managing, and good at using high-quality data criteria"
User Prompt: "To identify high-quality ESG data, we should consider the following criteria: {Criteria}
The following sentence is the data needed to define: {Data}
Answer 'Yes' or 'No' first, then give an explanation.
Let’s think step by step."

For task 2: *Classify whether the text is Env/Soc/-Gov data: Yes or No.*

System Prompt: "You are an expert in ESG data classification, especially {Type} ESG data classification."
User Prompt: "To identify {Type} ESG data, we should consider the following criteria: {Criteria}
Answer 'Yes' or 'No' first, then give an explanation.
Demonstrations:
{Demonstrations}
Text: {Data}
Answer: Let’s think step by step."

For task 3: *Assign texts to one of the following labels based on their domain.* The {Classes} are

corresponding to {Type}.

System Prompt: "You are an expert in classifying {Type} ESG data into finer categories such as {Classes}."
User Prompt: "For {Type} ESG data classification, consider: {Criteria}
First assign a class label based on these categories, or 'other' if uncertain. Then give an explanation.
Demonstrations:
{Demonstrations}
Text: {Data}
Answer: Let’s think step by step."

F.1 Annotation Examples

To illustrate the LLM’s responses to each task, we use text *"The possible threat of natural disasters affecting the ability to trade."* as an example, the responses are shown in Table 10.

G CPT PLMs Details

The continue pre-training parameters for our models were set as follows: a batch size of 16 for both training and evaluation, a total of 25 epochs, a seed value of 42 for reproducibility, and a maximum sequence length of 512. The models were trained with a gradient accumulation step of 2, weight decay of 0.01, and 16-bit precision (fp16) to optimize memory usage and accelerate the training process. The model checkpoints were saved and evaluated at the end of each epoch, as specified by the evaluation_strategy and save_strategy parameters set to 'epoch'. We conducted experiments on BERT, DistilRoBERTa, and RoBERTa at three different stages: epoch 5, epoch 15, and epoch 25. The results indicated that the models performed best at epoch 25. We utilized torch.distributed.launch for single-machine multi-GPU training. The pretraining of BERT, DistilRoBERTa, and RoBERTa took approximately 27 hours, 23.5 hours, and 33 hours, respectively. The training loss, validation loss, and validation accuracy during the training process are visualized in Figure 9. The smooth decrease in the loss curve and the gentle rise in the accuracy curve suggest that the models were converging and the training process was proceeding as expected. Consequently, we selected the model with the smallest

Table 10: Example of Data Through Three Tasks

Item	Content
Text	The possible threat of natural disasters affecting the ability to trade.
Task 1 Response	<p>Yes.</p> <ol style="list-style-type: none"> 1. Relevance: The sentence is directly related to environmental issues, specifically the potential impact of natural disasters on trade. 2. Accuracy: It is difficult to determine the accuracy of this statement without additional context or information about the specific location and type of natural disaster being referred to. However, in general, natural disasters can disrupt trade and supply chains, so the statement is plausible. 3. Source Credibility: Without knowing the source of this statement, it is difficult to assess its credibility. 4. Specificity: The statement provides a general idea of the potential threat posed by natural disasters to trade, but does not provide detailed insights or examples. 5. Objectivity: The statement appears to be objective and presents a balanced view of the potential impact of natural disasters on trade.
Task 2 Response	<p>Answer: Yes, the possible threat of natural disasters affecting the ability to trade is an environmental factor. Natural disasters can have a significant impact on the environment, including infrastructure, supply chains, and the ability to conduct business operations.</p>
Task 3 Response	<p>Label: Climate Change.</p> <p>Explanation: The text mentions the possible threat of natural disasters affecting the ability to trade. This indicates a concern related to climate change, as natural disasters such as hurricanes, floods, and wildfires are often associated with climate-related events. Climate change can disrupt trade by damaging infrastructure, disrupting supply chains, and impacting transportation systems. Therefore, the text falls under the category of Climate Change.</p>

validation loss at epoch 25 as our final pretraining models: ESG-BERT, ESG-DistilRoBERTa, and ESG-RoBERTa.

H FinLlama Training Details

Datasets. This fine-tuning was conducted on a specialized instruction-tuning dataset on financial domain delineated in FinGPT (Wang et al., 2023). Furthermore, we enhanced the dataset by incorporating the ESG_Chat dataset, which consists of dialogues between humans and Large Language Models (LLMs) focusing on methodologies to improve ESG scores. These conversations are structured to provide step-by-step guidance, with the LLM responses specifically tailored to offer structured, actionable advice. The characteristics of these datasets are detailed in Table 11.

Hyperparameters. Each stage was meticulously conducted throughout the training regimen over 3

epochs to ensure the models’ robust assimilation of the task-specific nuances. A consistent set of hyperparameters characterized the training to maintain uniformity across the models. Specifically, the batch size per device was set to 4, coupled with a gradient accumulation strategy involving four steps. This setup facilitated optimal resource utilization and stable training dynamics. The learning rate scheduler employed was of the cosine type, which aided in gradual learning rate adjustments, contributing to smoother convergence. For monitoring and model checkpointing, logging intervals were established at every 10 steps, and model states were preserved at every 100 steps, ensuring detailed progress tracking and the ability to revert to the most effective model state. The learning rate was judiciously chosen as 5×10^{-5} , balancing rapid adaptation and the preservation of pre-learned representations. The training progression

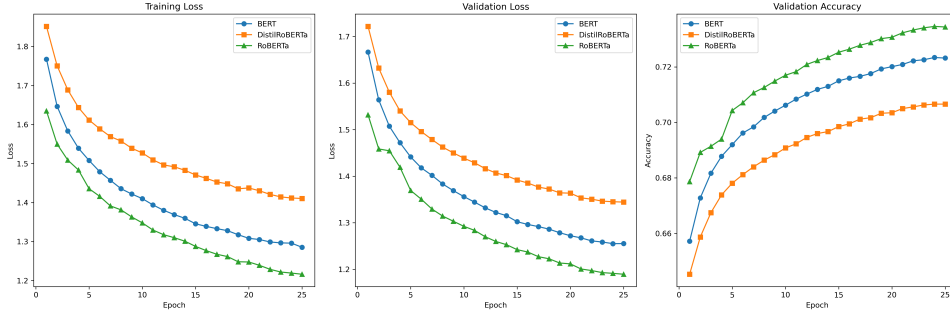


Figure 9: Continue Pre-training Log Loss and Accuracy across epochs

for both models was visually documented through loss curves, providing insightful glimpses into the models’ learning trajectories. Notably, a significant loss reduction was observed after the initial epoch for both models, indicative of their swift adaptation to the training objectives. For ESG llama, the training culminated with the loss stabilizing around 0.4, shown in Figure 10a, suggesting effective learning. Conversely, FinLlama exhibited a distinct two-phase training dynamic; the initial pretraining phase concluded with a loss of around 2.4, shown in Figure 10b, which, upon undergoing the subsequent Supervised Fine-Tuning (SFT) phase, settled at approximately 1.15 shown in Figure 10c. This delineation in training phases for FinLlama underscores the layered approach to model refinement, first broadening its financial domain comprehension, followed by targeted instruction-based fine-tuning to hone its capabilities for specific financial tasks. These models will be tested on our labeled ESG classification data. All experiments were conducted on NVIDIA V100 Tensor Core GPUs. Due to LLMs’ substantial parameter size and complex structure, fine-tuning and inference can be particularly time-intensive. We employed Parameter-Efficient Fine-Tuning (PEFT) techniques such as Low-Rank Adaptation (LoRA) and freeze during SFT phases to enhance efficiency. Additionally, we utilized LLaMA-Factory (Zheng et al., 2024) framework and vLLM (Kwon et al., 2023) to accelerate pre-training SFT and inference processes.

I ESG Classification SFT Dataset

Format: [{"instruction": "...", "input": "...", "output": "..."}]

Four-class Classification:

instruction: Classify the following text into one of the four ESG categories, choose an answer from {Env/Soc/Gov/Non-ESG}.

input: We maintain a health and safety management system aligned to ISO legal requirements in Australia and New Zealand.

output: Soc

Nine-class Classification:

instruction: Classify the following text into one of the nine ESG categories, choose an answer from {Climate Change/Natural Capital/Pollution and Waste/Human Capital/Product Liability/Community Relations/Corporate Governance/Business Ethics and Values/Non-ESG}.

input: Grievance mechanisms forms an important part of our stakeholder engagement process, and our human rights policy states that we will provide, or cooperate in providing, appropriate remediation if we have caused or contributed to adverse human rights impacts.

output: Human Capital

Table 12: An exam. of ESG classification SFT dataset.

The dataset we used for supervised fine-tuning is constructed from ESG SFT data in Section 2.3. The ESG classification SFT data was sampled and reconstructed from ESG SFT data by only selecting classification data and simplifying the result by retaining the text label without any additional ex-

¹<https://huggingface.co/datasets/FinGPT/fingpt-sentiment-train>

²<https://huggingface.co/datasets/FinGPT/fingpt-finred>

³<https://huggingface.co/datasets/FinGPT/fingpt-headline>

⁴<https://huggingface.co/datasets/FinGPT/fingpt-ner>

⁵https://huggingface.co/datasets/FinGPT/fingpt-fiqa_qa

⁶<https://huggingface.co/datasets/FinGPT/fingpt-fineval>

⁷<https://huggingface.co/datasets/zadhart/ESG-Chat>

Table 11: Instruction Financial Dataset Overview

Datasets	Train Rows	Test Rows	Description
finppt-sentiment-train ¹	76.8K	N/A	Sentiment Analysis Training Instructions
finppt-finred ²	27.6K	5.11K	Financial Relation Extraction Instructions
finppt-headline ³	82.2K	20.5K	Financial Headline Analysis Instructions
finppt-ner ⁴	511	98	Financial Named-Entity Recognition Instructions
finppt-fiqa_qa ⁵	17.1K	N/A	Financial Q&A Instructions
finppt-fineval ⁶	1.06K	265	Chinese Multiple-Choice Questions Instructions
ESG_Chat ⁷	914	N/A	Chat History about Improve ESG Score step-by-step

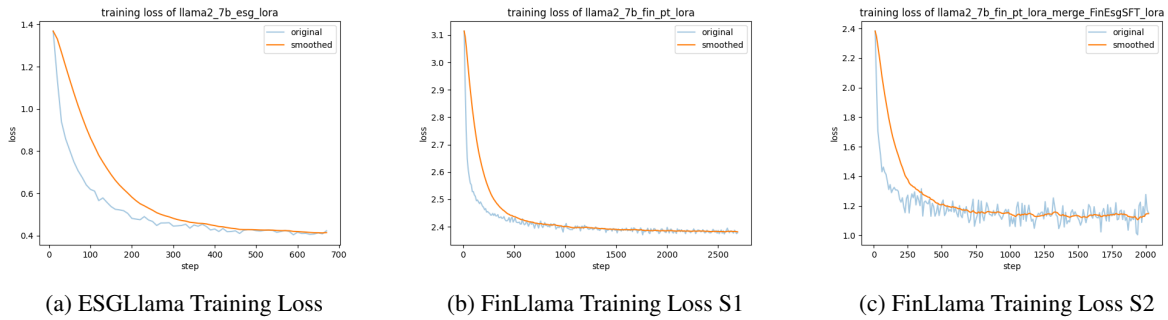


Figure 10: Training loss analysis during each stage of fine-tuning

planations. There are two main classification tasks contained in this dataset: four-class classification and nine-class classification. Finally, we obtained approximately 24k ESG Classification SFT Data. An example of the ESG classification SFT dataset regarding these two tasks is shown in Table 12. Using this dataset, we aim to enhance the baseline’s ESG text classification capability. This is because the baseline’s pre-training data may include financial text data partially related to ESG. We intend to modify the baseline for this task and evaluate its performance during the SFT training phase.

J Classification Prompts

System Prompt: "You are an expert in classifying ESG data. You will start your response with 'Label:'."

User Prompt: "Classify the following text into one of the four ESG categories, choose an answer from {Categories}"

Demonstrations:

{Demonstrations}

Text: {Text}

Label: So, the answer is"

For Four-Class classification task, we should specify the {Categories} by:

{Env/Soc/Gov/Non-ESG}

For Nine-Class classification task, we should

specify the {Categories} by:

```
{Climate Change/Natural Capital/  
Pollution and Waste/Human Capital/  
Product Liability/Community Relations/  
Corporate Governance/Non-ESG  
Business Ethics and Values}
```

To employ a chain-of-thought (CoT) setting, we need to slightly modify the system prompt and add let's think step by step at the end of the user prompt:

```
System Prompt: "You are an expert in  
classifying ESG data. You will  
response in this format:  
'Label:xxx. Explanation:xxx'.  
Your responses should be  
precise and concise."  
User Prompt: "...  
Label: Let's think step by step.  
So, the answer is"
```

K Additional LLM Classification Analysis

For Four-class classification. In evaluating our models, ESGLlama and FinLlama, within our experimental framework, we employed Precision, Recall, F1 Score, and Accuracy as our performance metrics. Initially, let us delve into the precision aspect, which serves to illustrate the models' exactness in classification tasks. Through the analysis of precision scores and the accompanying graphical representations shown in Figure 4, it becomes evident that both ESGLlama and FinLlama surpass the baseline model across most experimental configurations. Furthermore, even the baseline model, when subjected to Supervised Fine-Tuning (SFT) using our constructed ESG classification dataset, demonstrates enhanced performance compared to its original state. Interestingly, the Freeze fine-tuning approach generally outperforms the LoRA method, except in zero-shot settings. This observation could be attributed to the Freeze technique requiring a broader range of parameters for fine-tuning, thereby facilitating a deeper understanding of downstream tasks. In contrast, LoRA's superior performance in zero-shot scenarios might hint at a slight overfitting issue; external demonstration examples, not included in the training set, could potentially disrupt the model's inference processes.

The Freeze approach, in this context, better preserves the model's generalization capabilities and intrinsic reasoning faculties.

The incorporation of Chain of Thought (CoT) prompts leads to a performance decline in zero-shot and one-shot settings, except for the Iterated Chain of Learning (ICL) tasks. This decline could stem from the absence of stepwise reasoning chains in our training data, coupled with the inherent incompatibility of the CoT methodology with classification tasks—CoT primarily suits logic-based problem-solving. Nevertheless, the addition of demonstrations in ICL tasks enriches the model's learning of classification logic through increased sample exposure, culminating in the most favorable outcomes under ICL CoT configurations.

Further examination of performance metrics, as detailed in the corresponding table shown in Table 13, reveals that the LoRA method, applied directly to the baseline on our ESG classification dataset, achieves the highest precision (0.6928), recall (0.5557), F1 score (0.5488), and accuracy (0.5697) in zero-shot tasks. This outcome not only underscores the constructed dataset's validity but also establishes a benchmark for subsequent comparisons. Furthermore, the bold formatting in the table highlights the highest precision scores across six method settings for each model, underscoring the best-performing configurations. The underlined values denote the top performance metrics across all models and settings, establishing a benchmark for comparison. The star symbol (*) identifies the best baseline result for the LoRA and Freeze fine-tuning methods, serving as a reference point for assessing the fine-tuned models' enhancements. The directional arrows (↑↓) provide a visual cue for performance fluctuations in comparison to the baseline, elucidating the impact of our fine-tuning strategies on model precision. Against this backdrop, both ESGLlama and FinLlama exhibit a decline, albeit still outperforming the baseline, especially in ICL settings. Notably, FinLlama achieves superior precision over ESGLlama with the addition of CoT, underscoring the nuanced impact of our training methodologies on model performance. In summary, the table elucidates the nuanced interplay between fine-tuning methodologies, the inclusion of CoT prompts, and the iterative learning approach on model precision. The discernible improvement in precision with ESGLlama and FinLlama, particularly in ICL settings, reaffirms the efficacy of our fine-tuning strategies in embedding ESG-specific

knowledge into large language models.

For Nine-class classification, the analysis of performance metrics, particularly precision, elucidates a notable trend: as the complexity and diversity of classification categories increase, the task inherently becomes more challenging, as evidenced by the overall diminished performance compared to the four-class scenario. This trend underscores the escalated difficulty in distinguishing among a greater number of classes.

The precision score visualization (Figure 5) demonstrates that both ESGLlama and FinLlama significantly outperform the baseline model across most methodological settings. This superiority highlights our fine-tuned models' enhanced understanding and classification capability in the context of ESG-related texts. FinLlama demonstrates superior proficiency in iterative contrastive learning (ICL), particularly in scenarios with increased sample availability, indicating a profound comprehension of financial texts and their nuances. The analysis further reveals a pronounced decrement in performance for the one-shot learning setting across more granular classification tasks. Providing only one example per class introduces considerable bias and may confound the model's judgment due to the high variance associated with minimal data. Conversely, enriching the model with a broader set of examples significantly ameliorates performance, aligning with the expected benefits of expanded training data. This intricate classification landscape observes a notable divergence in the efficacy of the LoRA and Freeze fine-tuning methods. Interestingly, The LoRA approach exhibits superior performance in the one-shot setting compared to Freeze, suggesting that LoRA's parameter adaptation might be more conducive to effectively assimilating sparse class-specific information.

Delving deeper into the details presented in the accompanying Table 14, the most commendable performance is attributed to FinLlama under the ICL with Chain of Thought (CoT) augmentation, achieving a precision score of 0.6654. This result significantly surpasses the baseline precision of 0.6164 and even outstrips the baseline model fine-tuned with LoRA on the ESG classification data, which scored 0.6544. This evidence conclusively demonstrates the potent efficacy of FinLlama, particularly when augmented with CoT in complex classification scenarios.

Model	Methods	Overall			
		Precision	Recall	F1 Score	Accuracy
Llama2	Zero Shot	0.5778	0.5025	0.4815	0.5093
	w/ CoT	0.5527	0.4613	0.4252	0.4776
	One Shot	0.6012	0.5056	0.4706	0.5109
	w/ CoT	0.5370	0.3767	0.2680	0.3931
	ICL	0.6687	0.5408	0.5077	0.5446
	w/ CoT	0.6794	0.5193	0.4803	0.5229
LoRA	Zero Shot	<u>0.6928*</u>	<u>0.5557*</u>	<u>0.5488*</u>	<u>0.5697*</u>
	w/ CoT	0.6381	0.4973	0.5128	0.5053
	One Shot	0.5265	0.3896	0.2924	0.3976
	w/ CoT	0.5646	0.3291	0.2442	0.3360
	ICL	0.6148	0.5157	0.4821	0.5232
	w/ CoT	0.6213	0.3971	0.3247	0.4019
Freeze	Zero Shot	0.5741	0.5000	0.4787	0.5068
	w/ CoT	0.5480	0.4613	0.4276	0.4775
	One Shot	0.6085	0.5113	0.4761	0.5168
	w/ CoT	0.6168	0.3932	0.2873	0.4073
	ICL	0.6611	0.5382	0.5036	0.5422
	w/ CoT	0.6749	0.5181	0.4767	0.5216
ESGLlama	Zero Shot	0.5770	0.4997	0.4768	0.5054
	w/ CoT	0.5502	0.4594	0.4205	0.4753
	One Shot	0.6106	0.5373	0.5140	0.5389
	w/ CoT	0.6064	0.3984	0.3128	0.4147
	ICL	0.6738	0.5508 ↓	0.5203 ↓	0.5548 ↓
	w/ CoT	0.6746 ↓	0.4882	0.4323	0.4935
FinLlama	Zero Shot	0.5766	0.4961	0.4745	0.5024
	w/ CoT	0.5665	0.4669	0.4297	0.4828
	One Shot	0.6139	0.5375	0.5139	0.5394
	w/ CoT	0.5724	0.3856	0.3011	0.4017
	ICL	0.6698	0.5497 ↓	0.5174 ↓	0.5535 ↓
	w/ CoT	0.6797 ↓	0.4917	0.4365	0.4971

Table 13: Four-class evaluation results compare with baseline and our fine-tuned LLMs. **Bold** shows the best results in six method settings according to each model, and underline illustrates the best performance in each column. Star (*) is the best baseline result for two fine-tuning methods (LoRA and Freeze). Arrow (↑↓) signifies performance compared with Star (*).

Model	Methods	Overall			
		Precision	Recall	F1 Score	Accuracy
Llama2	Zero Shot	0.5875	0.4404	0.4454	0.4886
	w/ CoT	0.5826	0.4106	0.4171	0.4654
	One Shot	0.5049	0.4322	0.3877	0.4737
	w/ CoT	0.4314	0.3556	0.2895	0.3838
	ICL	0.6108	0.4029	0.4017	0.4411
	w/ CoT	0.6164	0.4624	0.4932	0.5057
LoRA	Zero Shot	0.5681	0.4901	0.4759	0.5294*
	w/ CoT	0.5180	0.4112	0.3895	0.4473
	One Shot	0.6256	0.5347*	0.4795*	0.5186
	w/ CoT	0.5751	0.3915	0.3450	0.3972
	ICL	0.6242	0.1946	0.1340	0.2257
	w/ CoT	0.6544*	0.1834	0.1465	0.2123
Freeze	Zero Shot	0.5911	0.4458	0.4488	0.4974
	w/ CoT	0.5799	0.4122	0.4161	0.4664
	One Shot	0.5258	0.4445	0.4148	0.4866
	w/ CoT	0.4922	0.4005	0.3353	0.4323
	ICL	0.6285	0.4189	0.4265	0.4649
	w/ CoT	0.5719	0.2432	0.2337	0.2862
ESGLlama	Zero Shot	0.5866	0.4271	0.4340 ↓	0.4778
	w/ CoT	0.5914	0.4190	0.4258	0.4726
	One Shot	0.5138	0.4446 ↓	0.4136	0.4855 ↓
	w/ CoT	0.4785	0.4031	0.3373	0.4318
	ICL	0.6201 ↓	0.4143	0.4235	0.4576
	w/ CoT	0.5773	0.2533	0.2470	0.2965
FinLlama	Zero Shot	0.5608	0.4293	0.4301 ↓	0.4830 ↓
	w/ CoT	0.5750	0.4123	0.4164	0.4664
	One Shot	0.5219	0.4376 ↓	0.4069	0.4757
	w/ CoT	0.4886	0.4062	0.3399	0.4349
	ICL	0.6168	0.4127	0.4163	0.4638
	w/ CoT	0.6654 ↑	0.2504	0.2478	0.2908

Table 14: Nine-class evaluation results compare with baseline and our fine-tuned LLMs. **Bold** shows the best results in six method settings according to each model, and underline illustrates the best performance in each column. Star (*) is the best baseline result for two fine-tuning methods (LoRA and Freeze). Arrow (↑↓) signifies performance compared with Star (*).

Can GPT Models be Financial Analysts?

An Evaluation of ChatGPT and GPT-4 on Mock CFA Exams

Ethan Callanan^{1,†}, Amarachi Mbakwe^{2,†,‡}, Antony Papadimitriou^{3,†}, Yulong Pei^{3,†}, Mathieu Sibue^{3,†}, Xiaodan Zhu¹, Zhiqiang Ma³, Xiaomo Liu³, and Sameena Shah³

¹Queen’s University

²Virginia Tech

³J.P. Morgan AI Research

¹{e.callanan,xiaodan.zhu}@queensu.ca, ²bmarachi@vt.edu, ³{first.last}@jpmchase.com

Abstract

Large language models (LLMs) have demonstrated remarkable performance on a wide range of natural language processing tasks, often matching or even outperforming state-of-the-art task-specific models. They have the potential to make a significant impact on financial professions and to have profound influence on the finance industry. In this study, we leverage mock exam questions of the Chartered Financial Analyst (CFA) program to conduct a comprehensive evaluation of ChatGPT and GPT-4 in financial analysis, considering zero-shot, chain-of-thought, and few-shot scenarios. We present an in-depth analysis of the models’ performance and limitations, and estimate whether they would have a chance at passing the CFA exams. Finally, we outline insights into potential strategies and improvements to enhance the applicability of LLMs in finance. In this perspective, we hope this work paves a way for future studies to continue enhancing LLMs for financial analysis.¹

1 Introduction

Tracking the progress of the most advanced large language models (LLMs) and their performance on major financial professional certifications has a profound impact on the financial industry. In general, language models and natural language processing (NLP) systems have played a pivotal role in enhancing various services, such as customer relations, financial question answering (Wang et al., 2022), document understanding (Kim et al., 2022), and report summarization (Abdaljalil and Bouamor, 2021). Despite these advancements, applying NLP in finance poses unique challenges, such as the distinct nature of financial tasks, linguistic structures,

and specialized terminology. As a result, the performance of general NLP models often falls short when applied to finance-related tasks — the specific challenges of financial reasoning problems warrant further investigation.

LLMs have the potential to make a significant impact on financial professions, and by extension on professional qualifications such as the Chartered Financial Analyst (CFA) Program.² With more than 190,000 charterholders across 160 markets worldwide, the CFA Program is arguably the most recognized certification in finance. Its exams are known for their meticulous yet practical assessment of financial expertise, making their resolution an ideal use case to gauge the capabilities of LLMs in handling complex financial analyses and reasoning. A human being often spends years to learn the required knowledge for the CFA examination.

Which of the following is most likely an assumption of technical analysis?

- A. Security markets are efficient*
- B. Market trends reflect irrational human behavior*
- C. Equity markets react quickly to inflection points in broad economy*

(a) Level I sample question

Paris Rousseau, a wealth manager at a US-based investment management firm, is meeting with a new client. The client has asked Rousseau to make recommendations regarding his portfolio’s exposure to liquid alternative investments [...]

[Table Evidence]

The AFFO per share for Autier REIT over the last 12-months is closest to:

- A. \$6.80;*
- B. \$7.16;*
- C. \$8.43.*

(b) Level II sample question

Figure 1: CFA example questions (source: CFA Institute); the question appears in bold, the multiple choices in blue and italic, and the vignette/case description in orange and italic.

In this paper, we rigorously assess the out-of-the-box capabilities of LLMs in real-world financial reasoning problems by conducting an evaluation

²<https://www.cfainstitute.org/en/programs/cfa/exam>

¹The code used in this paper is available upon request to any {first.last}@jpmchase.com among the authors.

[†]Equal contribution.

[‡]Work done while interning at J.P. Morgan AI Research.

on mock exam questions of the CFA Program. Our work focuses on two closed-source, non-domain specific LLMs, ChatGPT and GPT-4, using various popular prompting techniques. Although there are other LLMs available, the top models in the GPT series, e.g., GPT-4, do represent the state of the art on most benchmarked and in-house tasks, and are adequate to support the main conclusions of this study. In summary, our contributions are as follows:

- We conduct the first comprehensive evaluation of state-of-the-art LLMs on CFA mock exams, considering zero-shot, few-shot, and chain-of-thought prompting scenarios. We demonstrate that some of the models have a decent chance to pass the tests.
- We present an in-depth analysis of the models’ performance and limitations in solving these financial analysis and reasoning problems, including investigations at different topics and levels of the exams.
- We outline insights into potential strategies and improvements to enhance the applicability of LLMs in finance, suggesting new avenues for research and development.

2 Related Work

LLMs and Finance. As highlighted in (Brown et al., 2020; Wei et al., 2022), LLMs exhibit remarkable generalization across diverse topics. However, their application to finance, a domain demanding intricate reasoning with specific concepts, mathematical formulas, and visual aids, poses significant challenges. Approaches like continued pre-training (Araci, 2019; Wu et al., 2023), supervised fine-tuning (Mosbach et al., 2023; Yang et al., 2023), and retrieval augmented generation using external knowledge (Lewis et al., 2020) have been proposed to address these challenges. Notably, (Li et al., 2023) has extensively benchmarked the out-of-the-box capabilities of newer instruction-tuned LLMs in finance.

Evaluation of LLMs on Exams. Various studies have scrutinized LLMs in exams like the United States medical licensing exam (Kung et al., 2023), free-response clinical reasoning exams (Strong et al., 2023), college-level scientific exams (Wang et al., 2023), and the Bar exam (Katz et al., 2023). Notably, (Wang et al., 2023) found LLMs lacking in complex scientific reasoning, while (Bang et al., 2023) demonstrated ChatGPT’s outperformance in

NLP tasks. Our paper contributes by evaluating the financial reasoning abilities of ChatGPT and GPT-4 (Li et al., 2023) on the CFA exams. Refer to Appendix C for more detailed related work.

Topic	Level I			Level II		
	Calc.	#Tab	Len	Calc.	#Tab	Len
Ethics	0.7%	0.01	125	0.0%	0.00	1013
Quant. Meth.	70.5%	0.26	131	27.8%	0.00	1256
Economics	50.6%	0.25	121	66.7%	2.00	1115
Fin. Reporting	57.7%	0.35	151	53.6%	2.79	1383
Corp. Issuers	59.3%	0.28	120	44.4%	1.67	930
Equity Invest.	52.5%	0.19	112	45.8%	1.00	1053
Fixed Income	43.0%	0.06	87	50.0%	1.45	779
Derivatives	20.7%	0.00	65	75.0%	2.00	816
Alter. Invest.	36.4%	0.06	85	66.7%	2.00	840
Port. Manage.	38.3%	0.18	110	56.3%	2.13	1077
Overall	42.4%	0.17	116	45.5%	1.47	1058

Table 1: Question characteristics by topic; percentage of questions requiring calculation, average number of table evidence per question, and average prompt length (estimated using the tiktoken Python package).

3 The Chartered Financial Analyst Exam and Data

The CFA Program, a three-part exam, assesses investment tools, asset valuation, portfolio management, and wealth planning fundamentals. It is pursued by individuals in finance, accounting, economics, or business for roles in investment, risk, and asset management upon successful completion.

The CFA Institute does not release official past exams, so we use mock CFA exams written by CFA Charterholders and based on past CFA Institute assessments to conform to current testing formula and level of difficulty.

Other than availability, evaluating a model on the CFA Program poses another challenge in that the level III questions are open-ended written response questions, necessitating expensive human expert grading. As such, in this work, we focus on levels I and II and leaving Level III for future work.

Each exam level adheres to a specific format. Level I has 180 multiple choice questions (MCQs) on ten finance topics (Table 1). Level II includes 22 vignette-based item sets with 88 MCQs. Level III combines vignette-supported essay questions and MCQs. Example MCQs from the CFA Institute are illustrated in Figure 1.

We collected a total of five Level I exams and two Level II exams. In our experiments, we ensure each topic is represented in a similar proportion to the original CFA section (Figures 2 and 3 in

Appendix A). Table 1 summarizes the statistics of the exam questions we collected.

4 Experiment Setup

Prompting Paradigm. Our study examines the following typical prompting methods:

- **Zero-shot (ZS) prompting.** Tracking the off-the-shelf performance of generically trained AI models such as ChatGPT and GPT-4 is critical for programs like the CFA. We accordingly test ZS prompting performance.
- **Few-shot (FS) prompting.** We test 2-shot (2S), 4-shot (4S), 6-shot (6S), and 10-shot (10S) settings. When selecting examples, we apply two different strategies: (i) randomly sampling from the entire set of questions within the exam level (2S, 4S and 6S), and (ii) sampling one question from each topic in the exam level (10S). The latter aims at enabling the models to discern distinct attributes of the topic.
- **Chain-of-thought (CoT) prompting.** We follow ZS CoT (Wei et al., 2022), which has the added benefit of allowing us to analyze the “problem-solving process” of the models and determine where and why an answer goes wrong.

Implementation Details. We conduct the experiments using the OpenAI ChatCompletion API (gpt-3.5-turbo-0613 and gpt-4-0613 versions, 32K context window for FS prompting), with the temperature parameter set to zero. The prompt templates and settings are in Appendix B. To confirm the models had not memorized the mock exams as part of their training data, we employ memorization tests as in (Kiciman et al., 2023).

Metric. We compare the predictions against the exams’ solution set. *Accuracy* served as our sole evaluation metric throughout this study.

5 Overview of the Experiment Results

LLMs struggle more on Level II than on Level I. No matter the prompting paradigm employed, both models encounter more difficulties correctly answering the Level II item-sets than the independent questions from Level I (Table 2). We suggest that three factors might have negatively affected the performance of LLMs in Level II.

Firstly, the case description of a Level II item-set increases the length of the input prompt and could dilute the useful information it contains. Indeed, prompts for Level II are on average $10\times$

longer than the Level I ones; confronting Tables 1 and 2 shows that topics associated with poor performance usually present longer contexts in both Level I and II. In addition, the detailed case descriptions from Level II depict realistic day-to-day situations that contrast with the more general questions from Level I — LLMs thus need to abstract from case-specific details so as to identify the underlying finance concepts. Secondly, each item from the grouped item-set in Level II tends to go more in-depth about a specific finance topic than those in Level I, thus leading to more specialized and intricate problems. Lastly, Level II features a slightly higher proportion of questions requiring calculations and a much higher proportion containing table evidence (Table 1). Given the known limitations of LLMs on numerical and table reasoning (Frieder et al., 2023; Chen et al., 2022), this could also result in the low accuracy on Level II.

GPT-4 outperforms ChatGPT in almost all experiments, but certain finance topics remain challenging for both. As shown in Table 2, GPT-4 consistently beats ChatGPT in all topics in Level I and most topics in Level II, irrespective of the prompting paradigm.

In Level I, both LLMs perform best in Derivatives, Alternative Investments, Corporate Issuers, Equity Investments, and Ethics. The explicit mention of common finance notions in the questions (e.g., options, arbitrage, etc.) could be a factor, notions which ChatGPT and GPT-4 might have encountered during pretraining or instruction-tuning and that may help facilitate resolution. For Derivatives and Ethics, the question complexity is reduced due to the low amount of calculations and table understanding required to answer correctly (Table 1). However, both models perform relatively poorly in Financial Reporting and Portfolio Management (especially in ZS and CoT), with ChatGPT also struggling a lot more on highly computational topics such as Quantitative Methods. Indeed, the problems within these topics are more case-based, applied, computational, and CFA-specific than the ones from the aforementioned topics. They also tend to include more table evidence and complex details (Table 1).

The results are more nuanced in Level II. ChatGPT struggles on Alternative Investments and Fixed Income compared to GPT-4, while ChatGPT outperforms GPT-4 in Portfolio Management and Economics. Interestingly enough, both models now

Exam	Level I						Level II					
	ChatGPT			GPT-4			ChatGPT			GPT-4		
	ZS	CoT	2S	ZS	CoT	10S	ZS	CoT	10S	ZS	CoT	4S
Ethics	59.2	59.2	64.6	80.3	78.9	82.4	31.3	37.5	21.9	43.8	56.3	62.5
Quantitative Methods	53.9	50.0	59.7	78.0	76.0	76.0	44.4	55.6	54.2	66.7	66.7	72.2
Economics	68.0	63.7	68.0	74.1	73.6	76.2	66.7	58.3	62.5	41.7	58.3	58.3
Financial Reporting	54.0	53.4	60.1	68.2	70.8	70.0	39.6	31.3	44.8	54.2	66.7	56.3
Corporate Issuers	71.4	69.8	74.2	74.4	74.6	75.3	55.6	50.0	50.0	77.8	77.8	83.3
Equity Investments	59.4	60.9	62.5	80.3	70.5	68.8	60.4	60.4	60.9	65.0	58.8	62.5
Fixed Income	55.6	60.2	63.6	74.9	60.2	73.6	38.9	27.8	34.4	60.0	62.2	55.6
Derivatives	61.1	68.5	73.0	90.5	93.8	96.0	50.0	58.3	47.9	66.7	58.3	58.3
Alternate Investments	60.7	60.7	62.9	75.9	77.1	72.1	33.3	33.3	58.3	66.7	50.0	83.3
Portfolio Management	58.3	48.3	61.7	63.7	71.7	79.6	47.2	66.7	59.7	36.1	55.6	61.1
Overall	58.8	58.0	63.0	73.2	74.0	74.6	46.6	47.2	47.6	57.4	61.4	61.9

Table 2: ChatGPT and GPT-4 accuracy across topics on Level I and II exams for ZS, CoT, and FS prompting. For FS, the table only retains the results from the k -shot that achieves highest overall performance.

Model	Setting	Level I	Level II
ChatGPT	ZS	58.8	46.6
	CoT	58.0	47.2
	2S	63.0	46.6
	4S	62.3	45.7
	6S	62.2	47.0
	10S	62.4	47.6
GPT-4	ZS	73.2	57.4
	CoT	74.0	61.4
	2S	73.9	60.2
	4S	73.8	61.9
	6S	74.5	60.2
	10S	74.6	60.2

Table 3: Overall performance (accuracy) of ChatGPT and GPT-4 on Level I and II in ZS, CoT, and FS settings.

demonstrate low answer accuracy in the Ethics item-sets of Level II. This could originate from the more in-depth, situational, and detailed character of the problems from Level II in comparison to Level I.

CoT prompting yields limited improvements over ZS. CoT does not help LLMs in the evaluation as much as we expected, although CoT performs better than ZS in almost all cases (Table 3). In Level I, CoT prompting hardly benefits GPT-4 and deteriorates the performance of ChatGPT instead. Particularly, both models are affected in Quantitative Methods due to the hallucinations in mathematical formula and calculations. In Level II, CoT prompting yields a decent 7% improvement over ZS for GPT-4, but a disappointing 1% for ChatGPT. CoT benefits both LLMs in Ethics and Portfolio Management, where its explicit step-by-step reasoning over long and intricate evidence is usually helpful. Section 6 further investigates the reasons explaining such observations. We note that it is not easy to identify topics where CoT consistently improves or worsens the models’ perfor-

mance across levels, e.g., GPT-4 sees an accuracy improvement of 23% in Level II Financial Reporting, while ChatGPT has a 21% decrease.

A few in-context exemplars help more than CoT.

Compared with ZS and CoT prompting, FS offers significant performance improvement for ChatGPT on the Level I exams (Table 3). 2S prompting yields the best performance across all topics and overall in Level I for ChatGPT. The dominance is not as significant in Level II, but FS prompting still manages to achieve the best overall score for both models. Interestingly, for Level II, ChatGPT gains the most from 10S prompting, which suggests a more holistic FS approach across multiple topics helps the model crack complex questions. The overall trend in the results is that FS prompting seems to offer better assistance to less complex models (ChatGPT) when tested on seemingly simpler exams (Level I)

We argue that the better performance from FS credits to the answers associated with the examples in FS, which also might help the model understand how to best use the table evidence or other information contained in a question. The advantage from FS vanishes as the question complexity increases in Level II, where a combination of FS and CoT might be a potential approach worth further exploration.

6 Detailed Analysis on CoT

Surprisingly, CoT only marginally improves the models’ performance on most tests and is even slightly detrimental to the performance of ChatGPT on Level I exams (Table 3). We dive deeper into this phenomenon, as ZS CoT is often reported to outperform ZS prompting (Kojima et al., 2023).

To better understand CoT errors, we examine all instances where non-CoT is correct while CoT

Type of Error	ChatGPT	GPT-4
Knowledge	55.2%	50.0%
Reasoning	8.6%	10.7%
Calculation	17.2%	28.6%
Inconsistency	19.0%	10.7%

Table 4: Error modes introduced when using CoT on Level I questions. These errors do not occur with non-CoT prompting.

is incorrect, and categorize the errors as one of: Knowledge, Reasoning, Calculation, or Inconsistency. Knowledge errors are those where the model lacks critical knowledge required to answer the question. This includes an incorrect understanding of some concept, not knowing the relationship between concepts, or using an incorrect formula to answer a question requiring calculations. Reasoning errors occur when the model has all the correct knowledge, but either over-reasons in its response, or hallucinates some additional requirements or information not present in the question. Calculation errors are errors pertaining to some incorrect calculation (using a correct formula), or failing to accurately compare or convert results. Errors of inconsistency are when the model’s thinking is entirely correct, yet it chooses the wrong answer.

6.1 Underperformance of CoT on Level I

ChatGPT. Table 4 underlines that knowledge-based errors are the most common error mode for ChatGPT, constituting over half of all errors VS. non-CoT. This implies that, with CoT reasoning, the gaps in the internal knowledge of LLMs are magnified. As the model begins to think through its answer, it states its incorrect assumptions, which it proceeds to rationalize in the context of the question, thereby skewing the rest of the answer towards a wrong choice. Without using CoT reasoning, the model is able to make an “educated guess” where any incorrect knowledge has less of an opportunity of skewing the guess towards an incorrect answer. With a 1/3 chance of guessing correctly (plus any contextual hints that may lie in the question), guessing is a more accurate strategy when ChatGPT lacks the knowledge to reason correctly.

This same principal similarity explains calculation and reasoning errors, where one or a few off-track token generations can throw off the rest of the answer, resulting in an incorrect conclusion.

The instances where the model is entirely correct but makes an incorrect conclusion or selects

Type of Error	ChatGPT	GPT-4
Knowledge	70%	80%
Reasoning	20%	20%
Out of Tokens	10%	0%

Table 5: Error modes introduced when using CoT on Level II questions. These errors do not occur with non-CoT prompting.

the wrong answer are more enigmatic. In about half of these cases, it seemingly fails to generate a stop token upon coming to the conclusion, leading it to restate the concluding sentence with another option selected. In other cases, there appears to be a disconnect between the thought process and the answer selection. As we are using OpenAI’s API to retrieve structured output, our leading suspicion is that in these cases the ordering outlined in the system prompt is missed or ignored, and the answer is generated first.

GPT-4. The instances where CoT introduces errors for GPT-4 is half the number of instances where CoT introduces errors for ChatGPT. In these instances, GPT-4 also displays knowledge errors as the most common error mode. However, unlike ChatGPT, almost none of these knowledge errors stem from using incorrect formulas. This, along with the fact that there are less knowledge errors in total, shows that GPT-4 has more complete internal knowledge of both financial information and especially financial formulas and calculation methods. Even when GPT-4 finds the correct formula for a question involving calculations, it still struggles to perform the required calculation correctly. ChatGPT also frequently makes these sorts of errors in conjunction with wrong formula usage, which underlines the well-known and more foundational shortcoming of LLMs’ mathematical abilities (Frieder et al., 2023).

GPT-4 also displays far fewer inconsistency errors than ChatGPT. It appears to have a much stronger ability to connect its reasoning to the answers and to make comparisons. The one error type that GPT-4 makes more frequently than ChatGPT when it fails is reasoning errors. It seems that, along with GPT-4’s greater ability to reason, it has a greater chance of “talking itself” into incorrect lines of reasoning.

6.2 CoT Benefits on Level II

Level II exam questions require more interpretation of the information provided than Level I questions,

as test-takers must determine what parts of the case are relevant to the question, and some information may be missing altogether. Using CoT helps the model reason over the information and filter what is relevant to the question from the case, as evidenced by the results in Table 3. However, knowledge errors still persist in Level II, and outnumber reasoning errors for both ChatGPT and GPT-4 (Table 5).

7 Can LLMs pass the CFA exams?

7.1 CFA Pass Scores

The most intriguing question in this study probably is “Can LLMs pass the CFA exams?”. Conclusively determining whether a given score would suffice to pass the CFA exams is difficult because the CFA Institute refrains from disclosing the minimum passing score (MPS) for its examinations. The MPS is uniquely established for each individual exam, guided by the standards established by the CFA Institute in 2011. The CFA Institute employs the “Angoff Standard Setting Method” to ascertain the pass rates for CFA exams, and involves a group of CFA Charter holders convening to assess the difficulty level of the questions.

Exam	ChatGPT			GPT-4		
	ZS	CoT	FS	ZS	CoT	FS
Level I	F	F	F	P	P	P
Level II	F	F	F	U	P	P

Table 6: ChatGPT and GPT-4 ability to pass Level I and Level II Exams. P stands for pass, F stands for fail, and U stands for undetermined.

Although the CFA Institute maintains an air of secrecy surrounding its pass/fail thresholds, drawing from feedback provided by CFA exam takers on Reddit suggests that, for Level I, in general scoring approximately 70% in a majority of sections appears to more often than not lead to a pass. Attaining scores above 70% in *all* topics is not a requirement for pass, but maintaining an average score of 70% across topics considerably enhances the likelihood of a positive outcome³.

The estimates from the Reddit community regarding the MPS for Level II and III indicate that the two advanced exams have consistently featured

³<https://www.efinancialcareers.com.au/news/finance/whats-the-minimum-score-you-can-get-on-cfa-level-i-and-still-pass>

lower passing thresholds. In June 2019, their approximation on the MPS for Level III was at a mere 57.4%, and 62.8% for Level II. The section passing scores are ambiguous for Level II, but we can attempt to apply the same logic as aforementioned Level I exam but make an assumption that the cutoff for each is 60% instead of 70%⁴.

7.2 Pass Criteria and Outcomes

Given the information above, our proposed pass criteria are as follows:

- Level I - achieving a score of at least 60% in each topic and an overall score of at least 70%
- Level II - achieving a score of at least 50% in each topic and an overall score of at least 60%

Table 6 shows which model implementations were able to pass the exams. The FS implementations in both levels leverage the number of shots indicated in Table 2. Most of the settings showed a clear outcome, except for ZS on GPT-4 in Level II, which was a borderline case. ZS on GPT-4 attains >60% in six topics and a score between 50% and 60% in one topic. The topic performance seems high but the overall score, 57.39%, falls slightly short of the passing score proposed earlier, which therefore turns to be an unclear case.

8 Conclusion and Discussion

We conduct a thorough evaluation of ChatGPT and GPT-4 on the CFA exams and find that ChatGPT is unable to pass while GPT-4 is able under some FS and CoT settings. We note that CoT prompting provides marginal improvement for the models, but also exposes them to reasoning errors. Meanwhile, FS yields the best performance in most cases.

With these observations in mind, we propose future systems that could display greater performance by utilizing various tools. The most prevalent error mode of CoT, knowledge errors, could be addressed through retrieval-augmented generation using an external knowledge base containing CFA-specific information, or through fine-tuning on textbook data. Calculation errors could be avoided by offloading calculations to a function or API such as Wolfram Alpha. The remaining error modes, reasoning and inconsistency, could be reduced by employing a critic model to review and second guess the thinking before submitting the answer,

⁴<https://www.efinancialcareers.com.au/news/finance/whats-the-minimum-score-you-can-get-on-cfa-level-i-and-still-pass/>

or combining FS and CoT together to give richer examples of expected behavior. We hope this work paves the way for future studies to continue enhancing LLMs for financial reasoning problems through rigorous evaluation.

Acknowledgments

This research was funded in part by the Faculty Research Awards of J.P. Morgan AI Research. The authors are solely responsible for the contents of the paper and the opinions expressed in this publication do not reflect those of the funding agencies.

Disclaimer

This paper was prepared for informational purposes by the Artificial Intelligence Research group of JP-Morgan Chase & Co and its affiliates (“JP Morgan”), and is not a product of the Research Department of JP Morgan. JP Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

References

- Samir Abdaljalil and Houda Bouamor. 2021. An exploration of automatic text summarization of financial reports. In *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, pages 1–7.
- Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.
- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. [A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity](#).
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. [Convfnqa: Exploring the chain of numerical reasoning in conversational finance question answering](#).
- Simon Frieder, Luca Pinchetti, Alexis Chevalier, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Christian Petersen, and Julius Berner. 2023. [Mathematical capabilities of chatgpt](#).
- Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. 2023. Gpt-4 passes the bar exam. *Available at SSRN 4389233*.
- Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2022. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.
- TH Kung, M Cheatham, A Medenilla, C Sillos, L De Leon, C Elepaño, et al. 2023. Performance of chatgpt on usmle: Potential for ai-assisted medical education using large language models. *plos digit health* 2 (2): e0000198.
- Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. [Causal reasoning and large language models: Opening a new frontier for causality](#).
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Xianzhi Li, Xiaodan Zhu, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2023. Are chatgpt and gpt-4 general-purpose solvers for financial text analytics? an examination on several typical tasks. *arXiv preprint arXiv:2305.05862*.
- Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*.
- Eric Strong, Alicia DiGiammarino, Yingjie Weng, Preetha Basaviah, Poonam Hosamani, Andre Kumar, Andrew Nevins, John Kugler, Jason Hom, and Jonathan Chen. 2023. Performance of chatgpt on free-response, clinical reasoning exams. *medRxiv*, pages 2023–03.

- Bin Wang, Jiangzhou Ju, Yunlin Mao, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2022. A numerical reasoning question answering system with fine-grained retriever and the ensemble of multiple generators for finqa. *arXiv preprint arXiv:2206.08506*.
- Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. 2023. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. *arXiv preprint arXiv:2307.10635*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhajan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.
- Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.

Appendix

A Topic Distribution in Each Level

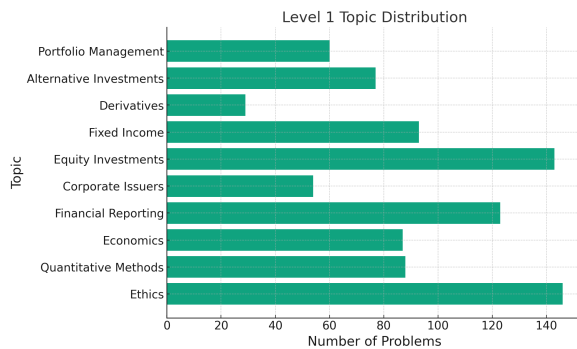


Figure 2: Level I exam topic distribution

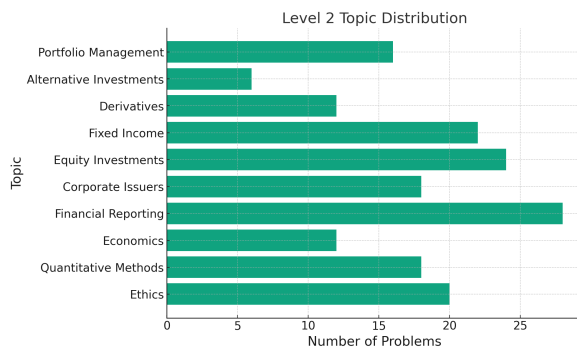


Figure 3: Level II exam topic distribution

B Prompt Templates Used in Our Work

B.1 Level I

Listing 1: ZS

SYSTEM: You are a CFA (chartered financial analyst) taking a test to evaluate your knowledge of finance. You will be given a question along with three possible answers (A, B, and C).

Indicate the correct answer (A, B, or C).

USER: Question:
{question}
A. {choice_a}
B. {choice_b}
C. {choice_c}

Listing 2: CoT

SYSTEM: You are a CFA (chartered financial analyst) taking a test to evaluate your knowledge of finance. You will be given a question along with three possible answers (A, B, and C).

Before answering, you should think through the question step-by-step. Explain your reasoning at each step towards answering the question. If calculation is required, do each step of the calculation as a step in your reasoning.

Indicate the correct answer (A, B, or C).

USER: Question:
{question}
A. {choice_a}
B. {choice_b}
C. {choice_c}

Listing 3: FS (2S example)

SYSTEM: You are a CFA (chartered financial analyst) taking a test to evaluate your knowledge of finance. You will be given a question along with three possible answers (A, B, and C).

Indicate the correct answer (A, B, or C).

USER: Question:
{question}
A. {choice_a}
B. {choice_b}
C. {choice_c}

ASSISTANT: {answer}

USER: Question:
{question}
A. {choice_a}
B. {choice_b}
C. {choice_c}

ASSISTANT: {answer}

USER: Question:
{question}
A. {choice_a}
B. {choice_b}
C. {choice_c}

B.2 Level II

For Level II, the case description of each item-set was inserted after the system prompt, before each question from the user.

C Related Work

LLMs and Finance. LLMs are trained on massive datasets that cover a broad range of topics and domains. Previous work has demonstrated the ability of LLMs to generalize surprisingly well to unseen downstream tasks, with little to no additional training data (Brown et al., 2020; Wei et al., 2022).

This raises an interesting question on the competitiveness of LLMs on specialized domains, such as finance. Indeed, the characteristics of most financial reasoning tasks — which rely on specific concepts and mathematical formula, frequently leverage diagrams and tables, often need multistep reasoning with calculations — make finance a challenging domain of application for LLMs. Several paths have been proposed to incorporate or emphasize domain-specific knowledge in LLMs: continued pre-training (Araci, 2019; Wu et al., 2023) and supervised fine-tuning on new data (Mosbach et al., 2023; Yang et al., 2023), retrieval augmented generation using a vector database of external knowledge (Lewis et al., 2020), etc. However, before considering such enhancements, only few papers have proceeded to extensively benchmark the out-of-the-box capabilities of newer instruction-tuned LLMs in finance (Li et al., 2023).

Evaluation of LLMs on Human Exams and other Benchmarks. Several previous studies have evaluated LLMs on various standard exams, such as United States medical licensing exam (Kung et al., 2023), free-response clinical reasoning exams (Strong et al., 2023), college-level scientific exams (Wang et al., 2023), and the Bar exam (Katz et al., 2023). The crucial contribution of these works is their analysis of the strengths and weaknesses of LLMs in realistic domain-specific settings, which guide subsequent research and practical use case resolutions.

For example, (Wang et al., 2023) evaluated ChatGPT and GPT-4 on a collection of Physics, Chemistry, and Math problems, and then concluded that current LLMs do not deliver satisfactory performance in complex scientific reasoning yet to be reliably leveraged in practice. In contrast, (Bang et al., 2023) found that ChatGPT outperformed fine-tuned task-specific models on four different NLP tasks, thus suggesting ChatGPT could be directly applied to solve industry use cases. Our paper aims at delving into the assessment of the inner financial reasoning abilities of ChatGPT and GPT-4.

Examining the Effect of News Context on Algorithmic Trading

Surupendu Gangopadhyay and Prasenjit Majumder

Dhirubhai Ambani Institute of Information and Communication Technology

Gandhinagar, India, 382007

{surupendu.g, p_majumder}@daiict.ac.in

Abstract

The application of reinforcement learning in algorithmic trading for spot trading, wherein the state is represented using price data, is a well-explored problem. However, these works perform spot trading in an index, which is not the market norm. Recent works have explored the use of reinforcement learning for performing high-frequency trading in the futures market. These works also combine news data and price data to represent the state. However, the news data is represented using news sentiment, which is not the optimal solution for representing the contextual information of news data. This paper proposes an RL framework that factors in the contextual information of the news data by using text embedding models and combines this with price data to perform futures trading. The paper investigates the impact of using different text embedding models on the trading strategy of the RL agent. Further, the paper also investigates whether using news context representation improves the trading decisions of the proposed model. The models are evaluated on NIFTY 50 index. The evaluation metrics show that using news contextual representation to represent the news data improves the trading performance of the RL agent.

1 Introduction

Reinforcement learning (RL) in algorithmic trading involves using trading agents that detect and leverage hidden signals in numerical and non-numerical data sources to open or close a position in the market. The goal of the agent is to learn an optimal trading strategy that maximizes the profit of the agent. In reinforcement learning, the agent interacts with the environment and modifies its policy based on the reward it receives from the environment. Additionally, the agent uses an exploration-exploitation strategy to learn the actions that eventually lead to the optimal policy. This property enables the agent to operate in unseen conditions as well. Thus,

in the case of using reinforcement learning for algorithmic trading, the agent can learn a dynamic trading strategy to operate in the highly temporal stock market environment.

The current literature on reinforcement learning for algorithmic trading widely uses price data, a numerical data source for designing the RL agent. The price data consists of historical OHLCV values and technical indicator values (Jeong and Kim, 2019; Wu et al., 2020; Théate and Ernst, 2021; Taghian et al., 2022; Yang et al., 2023). In these works, the trading agent primarily uses DQN (Mnih et al., 2015), which is an off-policy based RL algorithm. The trading agent performs spot trading and operates only once a day before the market closes. Thus the agent operates in a less noisy environment as the market trend already reflects the activity of the other stakeholders in the market. Also, these works directly perform spot trading instead of futures trading in an index, which is not the market norm.

Recent works have aimed to combine news data, which is a non-numeric data source, and price data to represent the state of the environment to perform trading in the share market (Gangopadhyay and Majumder, 2023a). The authors in this work represent the news data using the news sentiment. The agent operates in a noisy environment as the other stakeholders are also active in the market. The authors compare the performance of RL agents that use DQN and PPO (Schulman et al., 2017) which is an on-policy based RL algorithm. The authors show that in such conditions using PPO as the RL agent is well-suited as it uses the samples of the current trajectory to update the policy.

However, using news sentiment to represent the news data can give false positive and false negative signals from the news articles, which can impact the performance of the trading agent. Thus, using news sentiment to represent the contextual information of the news data is not an optimal solution,

and we need to explore the use of text embedding models to embed the context of the news data and use these news embeddings along with price data for representing the state. Therefore, based on this we investigate the following research questions:

- RQ1: How an RL agent can factor in the contextual representation of news data and combine it with price data to make trading decisions?
- RQ2: Can using contextual representation of news data along with price data improve the trading decisions of an RL agent?

In this paper, we propose an RL framework that leverages the contextual information of the news data and price data to perform futures trading at a high frequency. We use text embedding model to embed the news articles in the news data and use these embeddings along with price data to represent the state. Based on the findings of [Gangopadhyay and Majumder \(2023a\)](#) we use PPO as the RL agent. We use a feature extraction module to extract features from the state. In this work, we also compare the effect of using different text embedding models on the trading performance of the RL agent. We evaluate the effect of using different text embedding models by trading in the NIFTY 50 index. Our experiments show that factoring in the news context leads to an improvement in the trading performance of the model. The code repository of this work is available [here](#).

2 Related Work

The current literature on using reinforcement learning for algorithmic trading primarily focuses on spot trading in the stock market wherein the agent takes an action only once in a day before the market closes. Though some works have explored using reinforcement learning for algorithmic trading in the futures market. In the off-policy based approach the authors use DQN as the RL agent ([Jeong and Kim, 2019](#); [Wu et al., 2020](#); [Théate and Ernst, 2021](#); [Taghian et al., 2022](#); [Yang et al., 2023](#)). In the on-policy based approach the authors use policy gradient ([Lei et al., 2020](#)), PPO ([Hirchoua et al., 2021](#); [Gangopadhyay and Majumder, 2023a](#)), deterministic policy gradient ([Wu et al., 2020](#)) as the RL agent. Recent works have also explored RL agent that uses an ensemble of on-policy and off-policy based RL algorithms ([Yang et al., 2020](#); [AbdelKawy et al., 2021](#)).

In some works the authors use a discrete action space wherein the RL agent can perform three actions $[+1, -1, 0]$ ([Jeong and Kim, 2019](#); [Hirchoua et al., 2021](#); [AbdelKawy et al., 2021](#); [Lei et al., 2020](#)) indicating buy, sell and hold or two actions $[-1, +1]$ ([Taghian et al., 2022](#); [Yang et al., 2023](#); [Théate and Ernst, 2021](#)) indicating buy and sell. The use of discrete action space leads to the curse of dimensionality, which reduces the scalability of the RL agent to trade with a variable number of shares. Thus, in the case of discrete action space, the authors generally trade in a fixed number of shares. The papers that use on-policy based RL agents use a continuous action space ([AbdelKawy et al., 2021](#); [Gangopadhyay and Majumder, 2023a](#)) wherein the action lies between $[-1, +1]$, which overcomes the limitations of using a discrete action space.

The works that use only price data to represent the state use the OHLCV values, technical indicators, stock trends, account balance as the price data. Generally, the authors use the raw values of the price data of the previous day to represent the state. However, some works consider the price data as a time series and thus use a window of price data to represent the state. These works use encoder models such as GRU ([Wu et al., 2020](#); [Taghian et al., 2022](#)), attention-based GRU ([Lei et al., 2020](#)), transformers ([Yang et al., 2023](#)), CNN1D, CNN2D ([Taghian et al., 2022](#)) to extract the features from the sequence. [Taghian et al. \(2022\)](#) compare the performance of RL agents when using raw OHLC values and a window of OHLC values to represent the state and show that using a feature extraction module can improve the performance of an RL agent. [Gangopadhyay and Majumder \(2023a\)](#) perform HFT in the futures market using a combination of news data and price data to represent the state. The authors represent the news data using news sentiment and a feature extraction module that uses CNN to extract features over a window of prices and news sentiments.

In the current literature the reward function generally use the relative difference between the close prices or the difference in close prices as the reward function. The authors evaluate the trading models using evaluation metrics such as total profit, Sharpe ratio, Sortino ratio, maximum drawdown, and returns.

3 Proposed Approach

We propose an RL framework that trades in futures contracts in a minute-wise time series setting. At each time step t the agent uses the state s_t and reward r_t to determine the action a_t that will be executed at the next time step $t + 1$. The RL agent leverages the hidden signals in the news data and price data to determine the action. In this approach we consider all contracts as near month contracts, so the contracts will expire at the last Thursday of every month when the market closes. The agent can take a long or short position in the market and carry forward a position to the next day. When a contract expires all the open positions of the agent are automatically closed. We train the agent based on episodes, wherein each episode is the duration between the start and expiry of a contract. We use the environment used in [Gangopadhyay and Majumder \(2023a\)](#) to simulate the execution of buying and selling of futures contracts. We describe in detail the components of the RL framework in the subsections 3.1, 3.2, 3.3, and 3.4.

3.1 State

We use news data and price data to represent the state (s_t). We represent the price data at time step t by using the technical indicator values from time steps $t - w$ to t where w is the window size. The technical indicator values¹ consist of ADX, MACD, MOM, ATR, RSI, Slow %K, Williams %R, BBAND, EMA. At each time step i ($i \in [t - w, t]$), we construct a price vector denoted as $price_i$ which consists of these technical indicator values. Thus, the price data is a sequence of price vectors denoted as $[price_{t-w}, \dots, price_t]$.

We represent the news data at time step t by using k latest news article titles published between time steps $t - w'$ to t where w' is the window size. Thus, the news data is represented as a sequence of news article titles $[news_1, news_2, \dots, news_k]$. We use text embedding model to embed the context of a news article title and represent a news article title $news_j$ ($j \in [1, k]$) using a news embedding denoted as n_j . Thus, the news data is a sequence of news embeddings denoted as $[n_1, \dots, n_k]$. The state s_t is represented as combination of sequence of price vectors and news vectors.

As we are examining the effect of using different text embedding models on the performance

¹<https://www.incrediblecharts.com/indicators/technical-indicators-az.php>

of the trading agent, so in this work we compare the performance of the trading agent when using transformer encoder-based and transformer decoder-based text embedding models. In transformer encoder-based embedding models we use BERT (340M parameters) ([Devlin et al., 2018](#)) and FinBERT (340M parameters) ([Araci, 2019](#)). LLMs can provide a richer representation of the text due to a considerable increase in the number of the trainable parameters of the model and amount of data on which it is pretrained. All the major LLMs use the transformer decoder for text representation. So in transformer decoder-based embedding models we use Llama 2 (7B parameters) ([Touvron et al., 2023](#)) and Mistral (7B parameters) ([Jiang et al., 2023](#)).

As we are using a GPU resource poor setup for this work we do not finetune these text embedding models w.r.t. to our trading task rather we directly use these models in inference mode to get the embeddings. In BERT and FinBERT we take a sum of the token embeddings at the last layer to represent a news title. In Llama 2 and Mistral we use the embedding of last token in the sequence to represent a news title. Further, in this work we use 4-bit quantized AWQ ([Lin et al., 2023](#)) versions of Llama 2 and Mistral to enable inference of these models in the GPU resource poor setting.

3.2 Action

The action is the number of lots that agent intends to buy or sell. We use a continuous control setting in our proposed approach, so we define action space as $\mathcal{A} \in [-1, +1]$. The values of the action a_t lies within this action space. We define the maximum number of lots (max_lots) that the agent can buy or sell and use Equation 1 to get the actual number of lots that the agent wants to buy or sell. The use of a discrete action space will lead to the curse of dimensionality if we increase the number of lots that the agent can buy or sell, where as the continuous action space allows the agent to scale easily to trade in a high number of lots.

$$a_t = \lfloor max_lots \times a_t \rfloor \quad (1)$$

3.3 Agent

The agent uses PPO which is an on-policy based RL algorithm. As the state consists of a sequence of news vectors and price vectors, so the agent uses a feature extraction module to extract features from the state s_t and form a feature vector denoted as

f_t . PPO uses this feature vector f_t to determine the action a_t . The feature extraction module is shown in Figure 1.

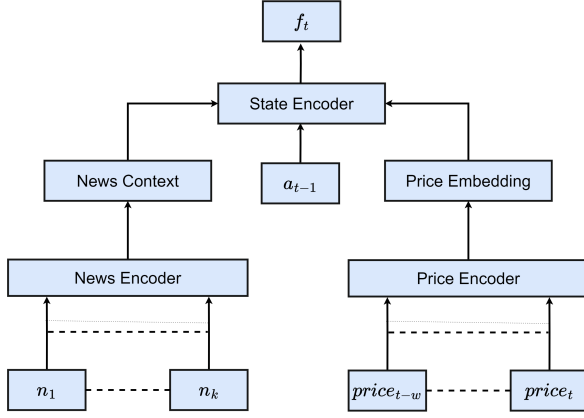


Figure 1: Architecture of feature extraction module

The feature extraction module consists of the following modules: news encoder, price encoder and state encoder. The news encoder module uses a CNN layer to capture the local contextual relationship between the sequence of news vectors to get the context vectors $[c'_1, \dots, c'_k]$ as output. It then uses Equation 2 to form a news sequence vector denoted as ns_v .

$$ns_v = \sum_k^{j=1} c'_j \quad (2)$$

It then passes ns_v through two fully connected neural layers to get a single value as output and applies a sigmoid function over the single value to get the news context value denoted as nc_v ($nc_v \in [0, 1]$) which represents the context encoded in the sequence of news vectors.

The price encoder module uses a CNN layer to capture the local contextual relationship between the sequence of price vectors to get the context vectors $[c_{t-w}, \dots, c_t]$ as output. It then uses Equation 3 to form a price sequence vector denoted as p_v . It then passes p_v through a fully connected neural layer to get the price embedding denoted as p_e which encodes the contextual relationship of the price vectors.

$$p_v = \sum_{i=t-w}^t c_i \quad (3)$$

The state encoder module concatenates nc_v , p_v and previous action take by the agent (a_{t-1}) and passes it through a fully connected neural layer to

form the feature vector f_t . We use CNN in the news encoder and price encoder, as it can effectively capture the context of a sequence (Gangopadhyay and Majumder, 2023b).

PPO consists of policy network and value network wherein the policy network determines the action and the value network gives the expected returns associated with the action. Both the networks consist of two fully connected neural layers which use the feature vector f_t to determine the action and value function. The policy network and value network share the parameters of the feature extraction module.

3.4 Reward Function

The reward function considers the short-term reward of an action w.r.t. change in close prices from t to $t + 1$ and long-term reward of an action w.r.t. change in the balance of the agent from t to $t + 1$ to calculate the reward denoted as r_t (Gangopadhyay and Majumder, 2023a). We use Equation 4 to calculate r_t where, c_t indicates the close price at time step t , $balance_t$ indicates the balance of the agent at time step t and a_t is the value that we get from Equation 1. The equation uses a λ value to balance the short-term and long-term rewards.

$$r_t = \lambda \times (a_t \times (c_{t+1} - c_t)) + (1 - \lambda) \times (balance_{t+1} - balance_t) \quad (4)$$

4 Dataset

The dataset consists of news data and price data. The news data comprises of archive news articles from the Economic Times ² from 2010-2021. We use a proprietary classifier to select only financial news articles from the news articles. The price data consists of minute-wise OHLC prices of NIFTY 50 index ³ from 2010-2021. We calculate the technical indicators values mentioned in the proposed approach from these prices and perform a z-normalization over the technical indicator values.

5 Evaluation Metrics

1. Total Profit: It is the profit earned at the end of the trading session. It is the difference between the balance earned at the end of the trading session and the balance at the start of the trading session.

²<https://economictimes.indiatimes.com/archive.cms?from=mdr>

³<https://www.kaggle.com/datasets/nishanthshalian/indian-stock-index-1minute-data-2008-2020>

2. Return (%): It is the percentage of the relative difference between the balances at the end of the trading session and start of the trading session.
3. Maximum Drawdown (MDD): MDD indicates the maximum loss a trading agent can incur in a trading session. It calculates this by measuring the relative difference the highest peak and lowest trough before the next peak is achieved. The duration between the two peaks indicates the time taken by the agent to recover from the loss.
4. Volatility: Volatility indicates the risk associated with an investment during a trading session. It is measured using the variance (σ) of the daily returns times the number of trading days (D) in a trading session. Volatility is calculated using Equation 5.

$$\text{Volatility} = \sigma\sqrt{D} \quad (5)$$

5. Sharpe Ratio: Sharpe Ratio is the ratio of expected returns ($E(R)$) in a trading session and the volatility in the trading session. Sharpe Ratio is calculated using Equation 6, wherein we assume the risk free rate to be zero.

$$\text{Sharpe Ratio} = \frac{E(R)}{\sigma} \quad (6)$$

6. Sortino Ratio: Sortino Ratio is the ratio of expected returns in a trading session and the standard deviation of the negative returns (σ_d) in a trading session. Sortino Ratio is calculated using Equation 7, wherein we assume the risk free rate to be zero.

$$\text{Sortino Ratio} = \frac{E(R)}{\sigma_d} \quad (7)$$

6 Experimental Setup

In this work we use Nvidia-RTX 20280Ti GPU consisting of 11GB VRAM. We use the data from 2010-2016 for training the trading models and we evaluate the models over the years 2017-2021. As per the market rules we set the size of a single lot for the years 2010-2017 to 25 and from 2018-2021 to 75. The maximum number of lots (max_lots) that agent can trade is 3. The starting balance for each test year is shown in Table 1, the balance is the product of the maximum number of lots that agent

can trade and the share price at the start of the trading session. We set the maximum sequence length of a news article title to 40 tokens. At each time step t we consider the news articles published in last 60 mins and thus set the value of w' to 60 and select the latest 10 news articles published within this window, thus we set k to 10. The graphs indicating the occurrence of sequence of news articles in the 1 hour window for training data and test data are shown in Figures 2 and 3. From the graphs we can observe that we are dealing with sparse data while dealing with news data. In this work, we consider the price data of last 5 minutes and thus set the value of w to 5. The λ value in the reward function is set to 0.85 (Gangopadhyay and Majumder, 2023a). The PPO algorithm uses Adam optimizer to perform optimization of the neural network layers. The parameters of policy network and value network of PPO and the parameters of the feature extraction module are given in Appendix A.1. The hyperparameters for training the different models is given Appendix A.2.

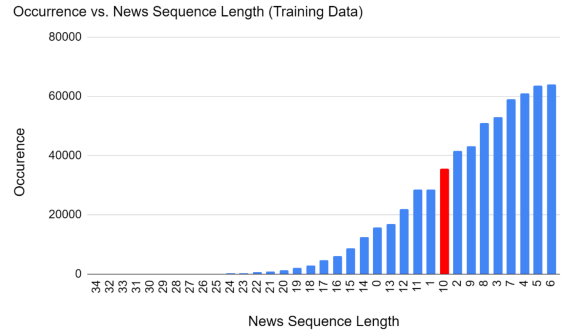


Figure 2: The frequency of occurrence of the length of the sequence of news articles present within 1 hr window (Training data)

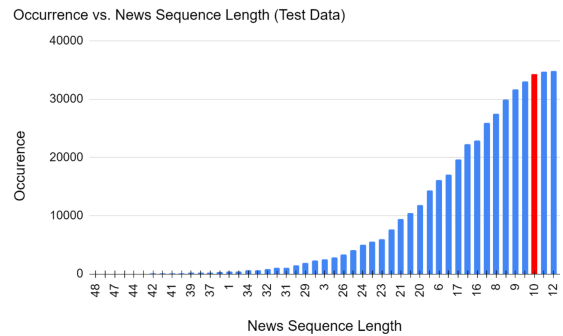


Figure 3: The frequency of occurrence of the length of the sequence of news articles present within 1 hr window (Test data)

Year	2017	2018	2019	2020	2021
Initial Balance	615757.5	2369632.5	2448382.5	2745483.75	3149122.5

Table 1: Initial balance at start of each test year

7 Results

In this section, we compare the trading performance of the proposed trading model when we use different text embedding models to embed the news data. As mentioned in the section 3.3, we compare the trading model w.r.t. BERT (768 dimension news embedding), FinBERT (768 dimension news embedding), Llama 2 (4096 dimension news embedding), and Mistral (4096 dimension news embedding). The year-wise total profit and average total profit is shown in Table 2. The year-wise return (%) and average return (%) is shown in Table 3. The year-wise MDD (%) and MDD duration (days) and average MDD (%) and average MDD duration (days) are shown in Table 4 and Table 5 respectively. The average volatility, Sharpe Ratio and Sortino Ratio are shown in Table 6, Table 7 and Table 8 respectively.

Years	Total Profit			
	BERT	FinBERT	Llama 2	Mistral
2017	229038.31	246737.37	152081.5	433339.06
2018	1083313.68	1154277.93	1235783.81	1314911.06
2019	1393744.87	1750031.24	1653581.06	936973.68
2020	3010310.81	3735349.31	3914199.37	3603385.68
2021	2008915.68	1240028.06	3296598.74	2266329
Avg. Profit	1545064.67	1625284.78	2050448.89	1710987.7

Table 2: Total Profit of the proposed trading model when using different text embedding models

Compared to the other text embedding models, we observe that using Llama 2 for news embedding gives the highest total profit and return (%), followed by Mistral, FinBERT, and BERT. Using FinBERT for news embedding gives the lowest MDD (%) and MDD duration of 26.277 % and 17.8 days, respectively, whereas using Llama 2 gives the highest MDD (%) and MDD duration of 27.813 % and 38.8 days respectively. We observe that the trading model faces similar loss percentages when using FinBERT and Llama 2, but the duration for which the trading model is at a loss is lower in FinBERT than in Llama 2. In terms of volatility, which indicates the risk of an investment,

Years	Return (%)			
	BERT	FinBERT	Llama 2	Mistral
2017	37.19	40.07	24.69	70.37
2018	45.71	48.71	52.15	55.49
2019	56.92	71.47	67.53	38.26
2020	109.64	136.05	142.56	131.24
2021	63.79	39.37	104.68	71.96
Avg. Return (%)	62.65	67.13	78.32	73.46

Table 3: Return (%) of the proposed trading model when using different text embedding models

Llama 2 has a volatility of 2.18, which is lower than BERT by 0.191, lower than Mistral by 0.181, and higher than FinBERT by 0.267. We also observe that using BERT gives the highest volatility and lowest return (%), indicating that it takes risky actions which yield lower returns. The Sharpe ratio and Sortino ratio of Llama 2 and Mistral are higher than BERT and FinBERT and thus indicate that using LLMs increase the volatility of the trading model but they also yield higher returns. The return (%) and MDD show that using FinBERT for embedding the news data improves the performance of the trading model than using BERT. Thus we infer that finetuning a pretrained language can improve the trading performance as it can better represent the context of a financial news article.

Years	MDD (%)			
	BERT	FinBERT	Llama 2	Mistral
2017	29.76	26.41	33.62	30.245
2018	27.01	25.77	27.6	26.09
2019	28.14	26.48	26.97	27.17
2020	26.19	24.89	25.88	25.81
2021	27.61	27.82	24.96	27
Avg. MDD (%)	27.74	26.27	27.81	27.26

Table 4: MDD (%) of the proposed trading model when using different text embedding models

Since, Llama 2 and Mistral are LLMs with higher model parameters and are pretrained on more text data than BERT and FinBERT. These factors lead to a more robust representation of the news data, thus improving the trading performance of the RL agent. Thus, using LLMs for text em-

Years	MDD Duration (Days)			
	BERT	FinBERT	Llama 2	Mistral
2017	30	14	147	50
2018	21	2	9	25
2019	34	9	24	29
2020	5	48	9	5
2021	16	16	5	39
Avg. MDD (Days)	21.2	17.8	38.8	29.6

Table 5: MDD duration (days) of the proposed trading model when using different text embedding models

bedding leads to higher returns accompanied with high investment risk, which corresponds with the efficient market hypothesis which states that to get higher returns, we need to take higher risk in investments (Fama, 1970). In the current literature Mistral (7B) has outperformed Llama 2 (13B) in various tasks (Jiang et al., 2023) whereas in this work, we find that the total profit and return (%) of Llama 2 are higher than that of Mistral by 3,39,461.199 and 4.858 %, respectively. However, the MDD (%) and MDD duration of Mistral is lower than Llama 2 by 0.547 % and 9.2 days. The performance of Llama 2 improves when using AWQ 4-bit quantization (Lin et al., 2023), but in this case, using AWQ 4-bit quantization may have affected the performance of Mistral.

However, the results show that the proposed approach can combine the contextual representation of the news data with price data to detect hidden signals in the data and exploit it to perform HFT in the futures market, thus answering our RQ1.

	Volatility			
	BERT	FinBERT	Llama 2	Mistral
Avg. Volatility	2.37	1.91	2.18	2.36

Table 6: Volatility of the proposed trading model when using different text embedding models

7.1 Comparison with existing approaches

Based on the return (%), we select the model that uses Llama 2 for news embedding as our best-performing model. We term this model as PPO_Llama_2. We compare this model with PPO_P, which uses only price data, i.e., a sequence

	Sharpe Ratio			
	BERT	FinBERT	Llama 2	Mistral
Avg. Sharpe Ratio	0.08	0.07	0.08	0.08

Table 7: Sharpe ratio of the proposed trading model when using different text embedding models

	Sortino Ratio			
	BERT	FinBERT	Llama 2	Mistral
Avg. Sortino Ratio	0.15	0.11	0.14	0.15

Table 8: Sortino ratio of the proposed trading model when using different text embedding models

of technical indicator values to represent the state, and PPO_FEM_PT, which uses price data and sentiment of news titles in the news data to represent the state (Gangopadhyay and Majumder, 2023a). We compare these models based on average return (%), average MDD (%), and average MDD duration. The comparison of the results of these models are shown in Table 9.

Trading Model	Return (%)	MDD (%)	MDD Duration (Days)
PPO_P	25.75	-26.81	47.6
PPO_FEM_PT	52.81 (+27.06)	-29.68 (-2.87)	41.6 (-6)
PPO_Llama_2	78.32 (+25.51)	-27.81 (+1.87)	38.8 (-2.8)

Table 9: Comparison of trading models based on Avg. Return (%), Avg. MDD (%) and Avg. MDD duration (days)

We observe that the return (%) of PPO_Llama_2 is higher than PPO_FEM_PT by 25.51 % and higher than PPO_P by 52.576 %. The duration for which PPO_Llama_2 faces a loss is less than PPO_FEM_PT by 2.8 days and less than PPO_P by 8.8 days. We observe that using the contextual representation of the news articles and a CNN layer to discover the relationship between the context of the sequence of news articles leads to a better representation of news data than using the aggregate of sentiments of news articles to represent the news data. We infer that news sentiment may not always convey the actual intent of a news article title and lead to generating false signals from the news data. Factoring in the news context reduces the false sig-

nals and greatly improves the performance of the RL agent and hence answers our RQ2.

8 Conclusion and Future Work

The results of the proposed trading model show that using news data context along with price data leads to an improvement over the model that uses news data sentiment along with price data or uses only price data. The results of FinBERT show that domain specific language models lead to an improvement in returns while also reducing the MDD. Increasing the text embedding model parameters improves the trading performance of the trading model, as we observed with the use of LLMs. Thus, in the future, we need to finetune the LLMs on financial texts and use this finetuned LLM to improve the trading performance of the RL agent. Future studies should also focus on using the article body instead of only the news title.

In this work, we deal with sparsity in availability of news as at time step t we used news data published in the last 60 mins preceding time step t . Increasing the number of hours led to non-convergence in training of the trading model. This sparsity affects the performance of the RL agent. The use of news data as the source of non-numeric data may also lead to some lag between when the information was available and when it was published, which may have affected the trading performance. In future work, we need to explore the use of multimodal news sources to bridge over the sparsity and time lag of news articles. Also, factoring in longer context length using the news article body instead of the title needs to be examined for better contextual representation. Future research should also focus on the adversarial training of the trading agents to guard against fake news, which can adversely affect the performance of the trading agent.

References

- Rasha AbdelKawy, Walid M Abdelmoez, and Amin Shoukry. 2021. A synchronous deep reinforcement learning model for automated multi-stock trading. *Progress in Artificial Intelligence*, 10(1):83–97.
- Dogu Araci. 2019. [Finbert: Financial sentiment analysis with pre-trained language models](#).
- 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*.
- Eugene F Fama. 1970. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2):383–417.
- Surupendu Gangopadhyay and Prasenjit Majumder. 2023a. How does news data impacts trading decisions? *Available at SSRN 4551629*.
- Surupendu Gangopadhyay and Prasenjit Majumder. 2023b. Text representation for direction prediction of share market. *Expert Systems with Applications*, 211:118472.
- Badr Hirchoua, Brahim Ouhbi, and Bouchra Frikh. 2021. Deep reinforcement learning based trading agents: Risk curiosity driven learning for financial rules-based policy. *Expert Systems with Applications*, 170:114553.
- Gyeun Jeong and Ha Young Kim. 2019. Improving financial trading decisions using deep q-learning: Predicting the number of shares, action strategies, and transfer learning. *Expert Systems with Applications*, 117:125–138.
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.
- Kai Lei, Bing Zhang, Yu Li, Min Yang, and Ying Shen. 2020. Time-driven feature-aware jointly deep reinforcement learning for financial signal representation and algorithmic trading. *Expert Systems with Applications*, 140:112872.
- Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. 2023. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978*.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Mehran Taghian, Ahmad Asadi, and Reza Safabakhsh. 2022. Learning financial asset-specific trading rules via deep reinforcement learning. *Expert Systems with Applications*, 195:116523.
- Thibaut Théate and Damien Ernst. 2021. An application of deep reinforcement learning to algorithmic trading. *Expert Systems with Applications*, 173:114632.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruiti

Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Xing Wu, Haolei Chen, Jianjia Wang, Luigi Troiano, Vincenzo Loia, and Hamido Fujita. 2020. Adaptive stock trading strategies with deep reinforcement learning methods. *Information Sciences*, 538:142–158.

Bing Yang, Ting Liang, Jian Xiong, and Chong Zhong. 2023. Deep reinforcement learning based on transformer and u-net framework for stock trading. *Knowledge-Based Systems*, 262:110211.

Hongyang Yang, Xiao-Yang Liu, Shan Zhong, and Anwar Walid. 2020. Deep reinforcement learning for automated stock trading: An ensemble strategy. In *Proceedings of the first ACM international conference on AI in finance*, pages 1–8.

A Appendix

A.1 Model Configuration

The configuration of the news encoder in the feature extraction module for different text embedding models is shown in Table 10. The configuration of the price encoder and state encoder in the feature extraction module for different text embedding models is shown in Table 11. The CNN layer uses a kernel size of 3 for all models. The parameters of policy network and value network of PPO is shown in Table 12.

Text Embedding Model	News Encoder		
	CNN	Layer 1	Layer 2
BERT	768×200	200×100	100×1
FinBERT			
Llama 2	4096×1000	1000×500	100×1
Mistral			

Table 10: Parameters of news encoder in the feature extraction module

Text Embedding Model	Price Encoder		State Encoder
	CNN	Layer 1	Layer 1
BERT	14×14	14×14	16×16
FinBERT			16×64
Llama 2			16×128
Mistral			16×16

Table 11: Parameters of price encoder and state encoder in the feature extraction module

Text Embedding Model	PPO	
	Policy Network	Value Network
BERT	16×16	
FinBERT	64×16	
Llama 2	64×64	
Mistral	64×64	

Table 12: Parameters of policy network and value network of PPO

A.2 Hyperparameters

The hyperparameters for training the models are shown in Table 13.

Text Embedding Model	BERT	FinBERT	Llama 2	Mistral
Learning Rate	0.0002	0.0002	0.00019	0.00019
Batch Size	128	128	128	128
Entropy Co-efficient	0.02	0.02	0.02	0.02
Epochs	6	6	7	6
Steps	2000	1500	1500	1500

Table 13: Hyperparameters for training the models

GPT-Signal: Generative AI for Semi-automated Feature Engineering in the Alpha Research Process

Yining Wang^{1,3}, Jinman Zhao², Yuri Lawryshyn³

¹Division of Engineering Science, ²Department of Computer Science,

³Centre for Management of Technology & Entrepreneurship(CMTE),

University of Toronto, Toronto, ON, Canada,

yining.wang@mail.utoronto.ca, jzhao@cs.toronto.edu, yuri.lawryshyn@utoronto.ca

Abstract

In the trading process, financial signals often imply the time to buy/sell assets to generate excess returns compared to a benchmark (e.g., an index). Alpha (Kakushadze, 2016) is the portion of an asset's return that is not explained by exposure to this benchmark, and the alpha research process is a popular technique aiming at developing strategies to generate alphas and gain excess returns. Feature Engineering, a significant pre-processing procedure in machine learning and data analysis that helps extract and create transformed features from raw data, plays an important role in algorithmic trading strategies and the alpha research process. With the recent development of Generative Artificial Intelligence(Gen AI) and Large Language Models (LLMs), we present a novel way of leveraging GPT-4 to generate new return-predictive formulaic alphas, making alpha mining a semi-automated process, and saving time and energy for investors and traders¹.

1 Introduction

In quantitative finance, we know many traditional financial signals such as the Price Earning (P/E) Ratio, Price/Book (P/B) ratio, Return on Equity (ROE), Return on Assets (ROA) etc. These signals all play an important role in helping people understand the financial situation of a company and get better ideas of the potential of that company in the stock market. The historical stock return data of different companies can be collected for stock market analysis and prediction (Li et al., 2023b). However, people are never enough of the existing traditional signals, and here comes the real magic of feature engineering in the alpha research process — finding new return-predictive signals.

Historically, feature engineering and formulaic alpha research processes have relied heavily on

human intuition and experience or complex algorithms (Zhang et al., 2020). Such processes for discovering new features could be overly subjective or time-consuming as they require sufficient domain-specific knowledge, a solid background in data engineering, and robust knowledge of various machine learning algorithms. However, the emergence of Generative Artificial Intelligence (Gen AI) gives us new insights and opportunities to reframe the feature extraction problem by automation.

As Gen AI has been rapidly developing in recent years, LLMs have become increasingly prevalent as a useful tool in real-life data science and deep learning applications among various fields. LLMs (Ouyang et al., 2022; Touvron et al., 2023; Jiang et al., 2024), based on deep neural networks with transformer architecture (Vaswani et al., 2017), are pre-trained on large-scale texts and fine-tuned by using reinforcement learning. The LLMs have strong performance on a variety of tasks such as content generation, question answering, arithmetic reasoning, computer programming and analysis, robust to data poisoning (Lyu et al., 2022), and are reckoned as a high-potential generative tool that can increase the efficiency in industry work and research.

The objective of this paper is to automate the process of generating new stock return-predictive financial signals using a Large Language Model (LLM), specifically GPT-4. The LLM will interpret information about a new financial dataset and create new, and significant signals. This system will utilize the LLM's advanced interpretative abilities to analyze financial texts and data, identify relevant patterns, and create valuable financial signals. Evaluation methods will be used to test the performance of the new signals in comparison to the existing signals; quantitative results will be presented.

¹Our code will be released at <https://github.com/Yiningww/GPT-signal>

In this work, we propose using LLM (GPT-4², specifically) to generate stock return-predictive new signals semi-automatically, which can help quantitative researchers and investors in the alpha mining process with much convenience and innovation. LLM creates new financial signals based on the user input information in the prompts, including the definition of several existing meaningful financial signals with sufficient coverage, historical signal data of multiple companies, and the respective historical returns at each time point. The process that GPT-4 employs for signal generation is not merely a one-off combination of the existing signals. It involves a series of refinements where the model learns which combinations yield the most informative signals, constantly improving the novelty and relevance of the signals it generates. The newly created signals will be evaluated by proposed evaluation methods. Based on the proposed framework, we conduct experiments on the S&P 500 companies in different sectors during different time frames, to compare with the baseline model and see the performance of new signals created by GPT-4. The main conclusions of our work can be summarized into the following points:

1. LLM(GPT-4) is able to analyze tabular structure data and generate new financial signals that meaningfully predict stock returns. These signals are developed based on the foundations of existing signals, historical data provided, and relevant information, with each new signal accompanied by its unique reasoning process detailed by the LLM.

2. The robustness of the generated signals is maintained when tested across different sectors of companies (i.e. Information Technology(IT), Health Care, Energy) within the S&P 500 index. Similar patterns of the new signals are observed in various selected sectors.

3. The model performance of newly created signals can outperform the models with baseline signals. Generally, the overall performance of these new signals tends to surpass that of the existing signals in all the selected sectors through 5 years (from year 2016 to year 2020).

4. GPT-4 can creatively combine the existing signals in non-linear and higher-order ways that go beyond simple linear combinations. This creative aspect of feature generation often results in signals that offer unique insights and are more than the sum of the existing parts. This data-driven approach to

signal construction is designed to discover novel patterns that are not immediately evident.

2 Related Work

LLM x Feature Engineering The utilization of Context-Aware Automated Feature Engineering (CAAFE) (Hollmann et al., 2024) mentioned in the work has a similar goal to this paper – implementing LLMs in automated machine learning(AutoML) (Hutter et al., 2019), generating new target-predictive features, and demonstrating the potential of LLMs for automating a broader range of data science tasks. CAAFE proposes to leverage the LLM and let the LLM generate codes that modify input datasets, creating target-predictive meaningful features that improve the performance of downstream prediction tasks in a repetitious workflow and with algorithmic feedback. The paper provides insights into our work, especially in prompting strategies for LLMs and evaluating methods of newly created features. LLMs, serving as tabular prediction models (Hegselmann et al., 2023), accept tiny tabular data sets as inputs, along with descriptive information (such as contextual information about the dataset, feature names with contextual information, data types, percentage of missing values, and 10 random rows from the dataset) about the dataset. While CAAFE focuses on various datasets, we focus on financial datasets with multiple companies’ historical financial signals and changes in historical returns.

LLM in Finance In the financial aspect, LLMs serve an important role in financial report generation, stock/market trends forecast, investor sentiment analysis, customized financial advice service etc., providing insights into market trends, performing risk management and evaluation, and even helping with trading decisions (Zhao et al., 2024). In addition, LLM’s capability of processing large-scale text data (Liu et al., 2023) makes it a prospective practice in the field of finance, enabling it to process natural language queries (Deng et al., 2023) and offer immediate advice and support.

In the prospect of LLMs and financial feature engineering, in particular alpha mining, paradigms such as Alpha-GPT (Wang et al., 2023a) are implemented for alpha mining, harnessing the power of human-AI interaction to increase the efficiency of alpha research. In Wang et al. (2023a)’s integration of GPT and alpha research, Alpha-GPT serves as a paradigm that enhances alpha genera-

²<https://openai.com/gpt-4>

tion through improved human-AI interaction. This system leverages a LLM to act as a mediator between quantitative researchers and the alpha search process. Alpha-GPT have three main advantages: First, it can interpret users’ trading ideas and translate them into appropriate expressions. Secondly, Alpha-GPT efficiently summarizes top-performing alphas in natural language, making them easier to understand. Finally, users can provide suggestions and modifications for the alpha search, which the model will automatically incorporate into future rounds of alpha mining. Alpha-GPT demonstrates that the output from the LLM can be a valuable reference for analyzing and revising prompt strategies, highlighting the importance of interaction with the LLM.

LLM Reasoning Having proved outstanding reasoning abilities, LLMs showcase proficient performance, especially in benchmarks such as arithmetic (Cobbe et al., 2021; Ling et al., 2017) and commonsense (Talmor et al., 2019). Many works have indicated the usefulness of prompting by implementing reasoning with LLMs like Few-shot learning (Brown et al., 2020), Emotional Prompt (Li et al., 2023a) and Chain-of-Thought (Wei et al., 2022; Kojima et al., 2022).

A recent trend highlights the use of LLMs for NLP tasks. For instance, Wan et al. (2023) uses in-context learning strategies on GPT-3 for Relation Extraction (RE). Wang et al. (2023b) and Xie et al. (2023) apply LLMs to the Named Entity Recognition (NER) task. Additionally, LLMs have been utilized for other tasks such as text summarization (Goyal et al., 2023) and sentiment analysis (Sun et al., 2023).

LLMs’ ability to understand table reasoning tasks and to analyze tabular data structure has also been confirmed in Chen (2022)’s work, showing that LLMs are capable and competitive at complex reasoning over table structures when combined with Chain-of-Thought (Wei et al., 2022; Kojima et al., 2022). LLMs can attain very strong performance with only a one-shot demonstration. In this work, we include tabular structured data in the prompt to LLM, based on the findings of the studies above, to utilize the LLM’s capability in complex reasoning.

3 Methodology

3.1 Prompt Design

The prompt mainly consists of two steps, building on Langchain’s Chase (2022) prompting template. Step 1 is to let GPT-4 generate the definition, the effect on predicting stock returns, and the preferred tendency of a set of existing signals we pick. After GPT-4 generates these definitions, we input this information for the second-step prompt, along with the overall instructions of the problem, several columns of data of some of the selected companies over a specific period, and the query (the actual question) we prompt to GPT-4. Zero-shot COT (Kojima et al., 2022) is used as a reasoning strategy, as the study shows that CoT can increase LLM’s accuracy even in a zero-shot learning strategy only by adding a simple prompt “Let’s think step by step”. A sample prompt is shown in Figure 1, including instructions for GPT-4 to reference; definitions, effect on predicting stock returns, and the preferred tendency of the 10 existing signals; sample data we randomly picked from our dataset; and the actual question (query).

3.2 Signal Evaluation

Spearman Rank Correlation Matrix Correlation Matrix is a method to measure the correlation between the variables and returns. The correlation coefficient ranges from -1 to 1. A value of 1 implies a strong positive relationship between two variables, -1 implies a strong negative relationship between the two variables, and a coefficient of 0 indicates that there is no linear relationship between the two variables. Traditional correlation matrices include Pearson-type correlations, which can be easily influenced by outliers and nonlinearities. Thus, we use the Spearman Rank Correlation Matrix as an alternative method, as it applies the Pearson correlation formula to the ranks of the data and can reduce distortions that influence the Pearson correlation to some extent. We calculate the correlation at each time point and take the average of the sum of the correlation coefficients.

$$Corr = \frac{1}{n} \sum (Corr_i)$$

where $Corr_i$ is the correlation coefficient of time i .

After obtaining the average correlation, heat maps are generated to display the correlation (calculated by the Pearson correlation formula and based

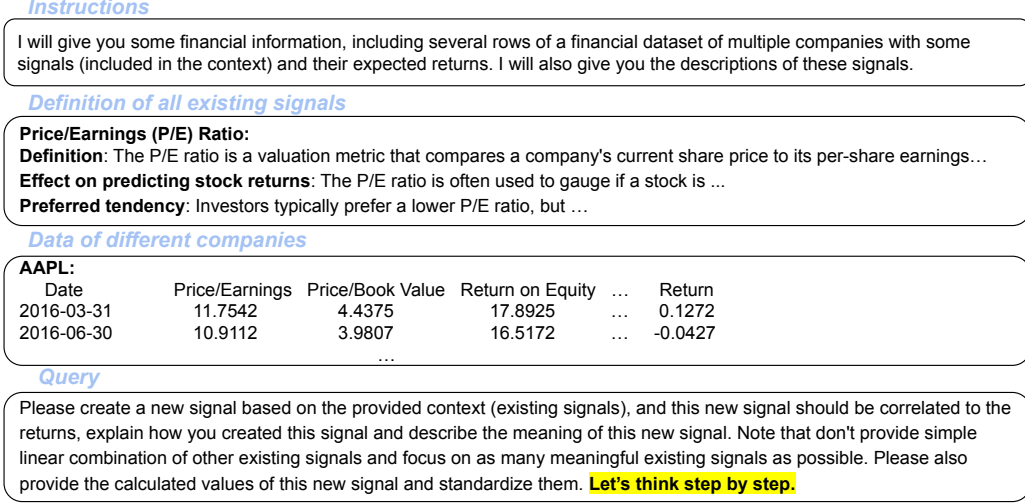


Figure 1: Prompt demonstration.

on the ranks of the data, instead of the actual data) between returns and each signal. While the coefficients can reveal the correlation between the signal and the return, they can vary with different periods and market situations. Hence, we also introduce another method to evaluate the signal, as shown in the next section.

Fama-MacBeth We adopt the Fama-MacBeth Two-Step Regression (Fama and MacBeth, 1973), a traditional method for evaluating how well signals describe returns. Data from n companies, including their historical signals and returns, are utilized for this evaluation. The Ordinary Least Squares (OLS), a commonly used approach, serves as the linear regression tool in our analysis process. Z-Score normalization is used on the signal values, as some of the signals have very large numerical values, while the values of percentage change in returns are very small.

Step 1: Each company's returns are regressed over time against the selected signals. The extent to which the returns are exposed to each signal is known as 'factor exposures' or 'beta coefficients'.

$$\begin{aligned}
 C_{1,t} &= \alpha_1 + \beta_{1,S1}S_{1,t} + \beta_{1,S2}S_{2,t} + \dots + \beta_{1,S_m}S_{m,t}, \\
 C_{2,t} &= \alpha_2 + \beta_{2,S1}S_{1,t} + \beta_{2,S2}S_{2,t} + \dots + \beta_{2,S_m}S_{m,t}, \\
 &\dots \\
 C_{n,t} &= \alpha_n + \beta_{n,S1}S_{1,t} + \beta_{n,S2}S_{2,t} + \dots + \beta_{n,S_m}S_{m,t}
 \end{aligned}$$

where $C_{i,t}$ is the expected return of company i at time t , α_i is the constant for company i , β_{i,S_j} is signal j 's beta coefficient at company i , and $S_{j,t}$ denotes signal j at time t for each company. t goes from 1 through T , indicating that each company's signals are regressed over time.

Step 2: We perform T Cross-sectional Regression at each time for all the companies: the cross-sectional stock returns are regressed against the factor exposures (beta coefficients) calculated in the first step, obtaining the risk premia coefficients for each signal.

$$\begin{aligned}
 C_{i,1} &= \gamma_{1,0} + \gamma_{1,1}\hat{\beta}_{i,S1} + \gamma_{1,2}\hat{\beta}_{i,S2} + \dots + \gamma_{1,m}\hat{\beta}_{i,S_m}, \\
 C_{i,2} &= \gamma_{2,0} + \gamma_{2,1}\hat{\beta}_{i,S1} + \gamma_{2,2}\hat{\beta}_{i,S2} + \dots + \gamma_{2,m}\hat{\beta}_{i,S_m}, \\
 &\dots \\
 C_{i,T} &= \gamma_{T,0} + \gamma_{T,1}\hat{\beta}_{i,S1} + \gamma_{T,2}\hat{\beta}_{i,S2} + \dots + \gamma_{T,m}\hat{\beta}_{i,S_m}
 \end{aligned}$$

where $C_{i,t}$ stands for the stock return of company i at time t , $\gamma_{t,0}$ is the constant term for each company C_i , $\gamma_{t,j}$ is the regression coefficient for factor j at time t , and $\hat{\beta}_{i,S_j}$ is the beta coefficient of company i for signal j . Note that i goes from 1 to n , as we include n companies in total.

4 Experimental Setup

Companies in different sectors, along with the historical signal data and percentage changes in returns, are included in the experiments. We opt for percentage changes in returns at various time points over actual return values because this offers a consistent standard across companies, accommodating the variance in return levels among different companies.

10 existing signals are as follows: Price/Earnings (P/E), Price/Book Value (P/B), Return on Assets (ROA), Return on Equity (ROE), Free Cash Flow per Share (FCF), Price/Cash Flow (P/CF), Enterprise Value/EBITDA (EBITDA), Gross Margin (GM), Net Margin (NM), Sales per

Creating a new nonlinear feature involves combining existing features in a nonlinear way that could potentially reveal more complex relationships with the stock returns. Given the provided context and the nature of the financial ratios, a possible nonlinear feature might capture a combination of profitability, valuation, and growth potential.

One such feature could be an "Investment Quality Score" that multiplies profitability ratios by inversely related valuation ratios, adjusted by growth evidence. For simplicity, let's utilize the following:

- **Profitability:** We'll use Return on Equity (ROE), a measure of financial efficiency and profitability relative to shareholders equity.

- **Valuation:** We will use the inverse of the Price/Earnings (P/E) Ratio and the inverse of the Price/Book Value (P/B) Ratio since lower values of these ratios are generally preferred and indicate potential undervaluation.

- **Growth:** Sales per Share could serve as a proxy for a company's growth potential. We'll transform this feature through a logarithm to moderate extreme values and emphasize relative changes in sales.

Our new feature, therefore, could be defined as:

Investment Quality Score (IQS) = (ROE * (1 / P/E) * (1 / P/B)) * log(Sales per Share)

This feature aims to score companies higher if they have high profitability, lower valuation ratios, and evidence of sustainable sales growth.

Now, let's calculate the values of this hypothetical feature for each company on each date. We'll only calculate the scores for the first two dates of the AAPL company as an example due to space constraints. Following these calculations, we'll need to standardize the newly created feature across all the data, which is often done by subtracting the mean and dividing by the standard deviation.

Figure 2: Sample output of GPT-4 after being asked to generate a new signal.

Share (SPS). The selection of existing signals, which are "popular" financial indicators commonly used for evaluating a company's financial health (Arkan et al., 2016; Charles Schwab, 2023), is primarily influenced by their coverage across the datasets, ensuring the chosen signals are broadly applicable and reflective of standard financial analysis practices.

At each cross-section, we obtain an Adjusted R-squared (R_{adj}^2) of the model. After GPT-4 generates new signals as the last section mentioned, we add each of the new signals to our existing signals and perform the two-step Fama-MacBeth regression. The performance of models with each new signal is compared with that of the baseline model (with only existing signals).

Dataset Based on the Global Industry Classification Standard (GICS)³ and looking at the S&P 500 index, we select companies in the Information Technology (IT) sector (43 companies), Health Care sector (31 companies), and Energy sector (19 companies), respectively. The full company lists are shown in the Appendix A. We download the companies' historical signal data from FactSet⁴ and historical returns from Yahoo Finance⁵, both of which are open-source financial websites. Data is processed to extract signal values, which are then merged with future one-month and three-month returns for analysis. This approach ensures a com-

prehensive dataset for evaluating financial performance.

5 Results

5.1 GPT-4 Output

With the prompts in the format shown in Section 3, GPT-4 is asked to generate several new signals by running the script multiple times, one new signal per run. Names and formulas are included in the outputs of GPT-4. Since we use a step-by-step prompting strategy, reasoning steps are also shown in the outcome, including the meaning, profitability, valuation, and growth of the new signal. Figure 2 shows part of a sample outcome of the new signal "Investment Quality Score (IQS)": GPT-4 provides its understanding of creating a new nonlinear signal, the reason why it creates such a new signal and the way of calculating the new signal. In addition, it calculates and standardizes values for the new signals based on the existing signal values we include in the prompt. The reasonings between other newly created signals are shown in B. The reasoning process demonstrates the potential of GPT-4 to produce outputs that are analytically sound and methodologically robust, rather than simply generating outputs arbitrarily.

6 new signals created by GPT-4 are listed below:

1. Profitable Valuation Score (PVS): $PVS = \frac{ROE}{P/E}$,
2. Risk-Adjusted Performance Score (RAPS): $RAPS = \frac{ROE}{P/E \cdot \beta}$, here β is 2 for calculation convenience.

³<https://www.msci.com/our-solutions/indexes/gics>

⁴<https://www.factset.com/>

⁵<https://finance.yahoo.com/>

Correlation Matrix

P/E	1.00	0.28	-0.30	-0.37	-0.30	0.45	0.50	0.21	-0.28	-0.36	-0.62	-0.65	-0.23	0.53	-0.50	-0.69	0.08
P/B	0.28	1.00	0.32	0.35	0.06	0.52	0.41	0.47	0.24	-0.24	0.14	0.04	-0.28	0.52	0.21	-0.03	0.11
ROA	-0.30	0.32	1.00	0.85	0.39	0.27	0.21	0.44	0.89	-0.00	0.75	0.72	-0.03	0.29	0.81	0.47	0.07
ROE	-0.37	0.35	0.85	1.00	0.47	0.14	0.10	0.33	0.72	0.14	0.90	0.86	0.10	0.40	0.87	0.49	0.05
FCF	-0.30	0.06	0.39	0.47	1.00	-0.08	-0.12	0.17	0.34	0.61	0.49	0.50	0.26	0.18	0.41	0.44	-0.01
P/CF	0.45	0.52	0.27	0.14	-0.08	1.00	0.77	0.49	0.23	-0.28	-0.06	-0.15	-0.46	0.45	0.10	-0.31	0.13
EBITDA	0.50	0.41	0.21	0.10	-0.12	0.77	1.00	0.39	0.20	-0.28	-0.15	-0.22	-0.41	0.42	0.01	-0.38	0.12
GM	0.21	0.47	0.44	0.33	0.17	0.49	0.39	1.00	0.56	-0.45	0.19	0.12	-0.20	0.47	0.42	-0.12	0.10
NM	-0.28	0.24	0.89	0.72	0.34	0.23	0.20	0.56	1.00	-0.12	0.66	0.65	0.04	0.21	0.79	0.43	0.04
SPS	-0.36	-0.24	-0.00	0.14	0.61	-0.28	-0.28	-0.45	-0.12	1.00	0.27	0.31	0.33	-0.16	0.05	0.44	-0.10
PVS	-0.62	0.14	0.75	0.90	0.49	-0.06	-0.15	0.19	0.66	0.27	1.00	0.96	0.26	0.16	0.88	0.70	0.00
RAPS	-0.65	0.04	0.72	0.86	0.50	-0.15	-0.22	0.12	0.65	0.31	0.96	1.00	0.36	0.10	0.82	0.76	-0.03
EVC	-0.23	-0.28	-0.03	0.10	0.26	-0.46	-0.41	-0.20	0.04	0.33	0.26	0.36	1.00	-0.03	0.07	0.50	-0.14
VEC	0.53	0.52	0.29	0.40	0.18	0.45	0.42	0.47	0.21	-0.16	0.16	0.10	-0.03	1.00	0.21	-0.23	0.09
PLF	-0.50	0.21	0.81	0.87	0.41	0.10	0.01	0.42	0.79	0.05	0.88	0.82	0.07	0.21	1.00	0.50	0.05
IQS	-0.69	-0.03	0.47	0.49	0.44	-0.31	-0.38	-0.12	0.43	0.44	0.70	0.76	0.50	-0.23	0.50	1.00	-0.06
Return	0.08	0.11	0.07	0.05	-0.01	0.13	0.12	0.10	0.04	-0.10	0.00	-0.03	-0.14	0.09	0.05	-0.06	1.00
	P/E	P/B	ROA	ROE	FCF	P/CF	EBITDA	GM	NM	SPS	PVS	RAPS	EVC	VEC	PLF	IQS	Return

Figure 3: Correlation of **all companies** with both existing and new signals. Note that the last six signals are newly created by LLM.

3. Efficiency Value Composite (EVC): $EVC = \frac{1.0}{ROA} \cdot \frac{1.0}{EBITDA} \cdot \frac{1.0}{PCF}$
4. Valuation Efficiency Composite Score (VEC): $VEC = \frac{(P/E+ROE+FCF)}{3.0}$
5. Profitability Leverage Factor (PLF): $PLF = \frac{ROE \cdot GM}{P/E}$
6. Investment Quality Score (IQS): $IQS = (ROE \cdot \frac{1}{P/E} \cdot \frac{1}{P/B} \cdot \log(SPS))$

For the evaluation period, we use ranges from years 2016 to 2020, with a frequency of 3 months, as the historical signals of the companies are reported quarterly. In addition, we use the signal values to predict the future quarterly returns (i.e. we use signals in March to predict returns in June). We demonstrate results for IT companies (with future quarterly returns), and other sectors' results are listed in the Appendix B.

5.2 Overall Results

Figure 3 shows the correlation matrix for all the companies in the 3 different sectors. New signal EVC still possesses the highest absolute value with returns and most of the other new signals. Note that although the values of the coefficients are small, they are already considered sufficiently large values in the case of predicting change in stock returns

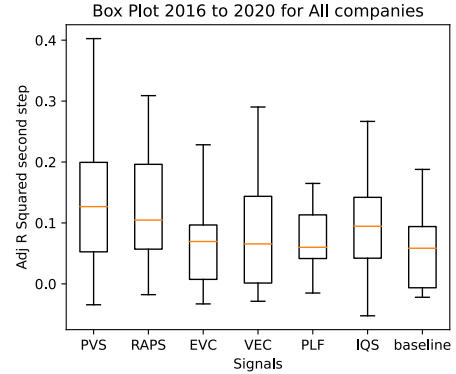
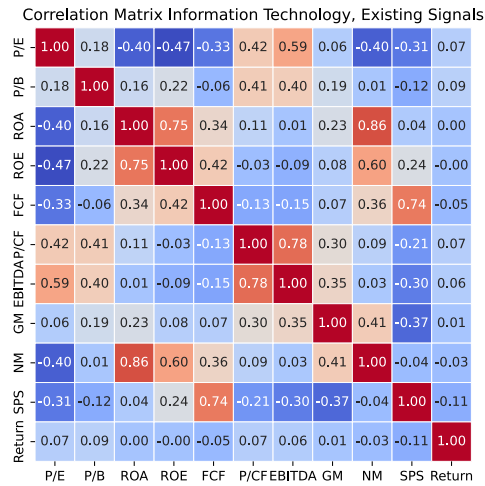


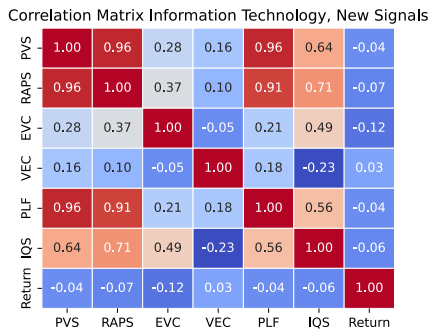
Figure 4: R_{adj}^2 values for Fama-MacBeth step 2 with companies in all 3 sectors. The last boxplot is the baseline without any new signals.

(Kawee Numpacharoen, 2012). These observations show that the new signals do have considerable correlations to the returns.

Figure 4 shows the box plot of R_{adj}^2 values for Fama-MacBeth step 2, evaluated on companies in all the 3 sectors. The box plot offers a comparative visual representation, showing the variability of the R_{adj}^2 values, which serve to gauge the explanatory capacity of our regression models enhanced by the introduction of novel signals. The new signals demonstrate a range of improvements in comparison to the baseline model, as denoted by the median and the interquartile ranges. The final box plot on



(a) Correlation of IT companies with existing signals.



(b) Correlation of IT companies with new signals.

Figure 5: Correlation for existing/new signals with returns.

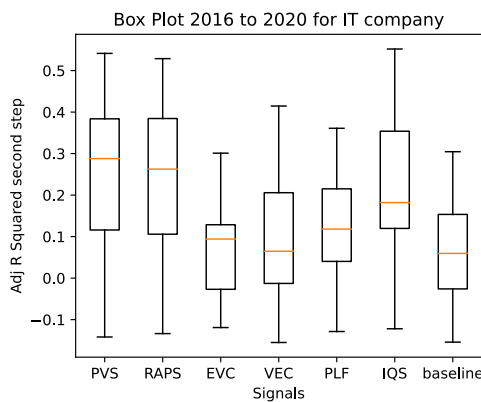


Figure 6: R_{adj}^2 values for Fama-MacBeth step 2 with companies in IT sector. The last boxplot is the baseline without any new signals.

the right illustrates the baseline model without the integration of new signals, establishing a benchmark that accentuates the predictive accuracy gains afforded by the existing features.

5.3 Correlation Heat Maps Break-down

Heat Maps of the correlation coefficients between existing and new signals and their future 3-month returns in IT companies are shown in figure 5a and figure 5b. The correlation coefficients between signals and returns are shown in the last column and the last row. Figure 5a is the correlation matrix in the IT sector between the existing signals and historical returns, and the last column is the correlation coefficients between the signals and returns. We can see that the absolute value of the coefficients ranges from 0 to 0.11. Figure 5b is the correlation matrix between the new signals and the returns, and we can see that the absolute value of the coefficients ranges from 0.03 to 0.12, which has an overall better performance than the existing signals.

The new signal EVC has the highest absolute correlation, surpassing the performance of all the existing signals. Besides, other new signals generated by GPT also have proper performance, all of which have absolute correlation coefficients larger than at least two of the existing signals.

Apart from the IT sector, we also evaluate the new signals on companies' data in the Health Care and Energy sectors. Corresponding heat maps are plotted in the same format, as shown in Appendix B. Similar patterns can be observed in different sectors, as many of the correlations of the new signals have a higher absolute value than the existing ones.

5.4 Fama-MacBeth Regression Break-down

The R_{adj}^2 values for the Fama-MacBeth step 2 regression models, each with a new signal added to the original set (the 10 existing signals, have been calculated and presented at Figure 6, and the median values of the R_{adj}^2 values are marked by the orange lines. The signal names in the graph represent the models with 10 existing signals plus each of the 6 new signals generated by GPT-4, respectively, noting the "baseline" represents the model with only the 10 existing signals. (The last boxplot is the baseline without any new signals). It is observed that the inclusion of these new signals results in improved performance for 5 out of 6 models with new signals, compared to the baseline model's performance. Box plots for companies in Health

Care and Energy sector are shown in Appendix B, with similar patterns observed.

6 Conclusions

In this work, we leverage an LLM (GPT-4) to generate 6 novel financial signals that enhance the performance of existing stock return-prediction models, addressing the limitations of traditional feature engineering techniques in financial analytics and the alpha research process. We demonstrate that GPT-4 is capable of analyzing existing signals' performance in historical data and extracting useful context information in the feature engineering process. The work results in the creation of innovative signals that capture patterns and interactions.

The new signals generated by GPT-4 demonstrate various advantages. First of all, GPT-4 adapts to changes in market conditions more thoroughly and dynamically than traditional models, permitting it to continually refine and optimize the process of signal generation based on data and human-AI interaction. Secondly, the LLM is able to process and analyze a large amount of data, and identify sophisticated patterns and relationships that are not obvious through traditional and standard statistical methods. Last but not least, the use of GPT-4 largely speeds up the feature engineering process, reducing the time required to develop complicated algorithms and explore new financial signals in the market.

References

- Thomas Arkan et al. 2016. The importance of financial ratios in predicting stock price trends: A case study in emerging markets. *Finanse, Rynki Finansowe, Ubezpieczenia*, (79):13–26.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. *Language models are few-shot learners*. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Charles Schwab. 2023. Five key financial ratios for stock analysis. <https://www.schwab.com/learn/story/five-key-financial-ratios-stock-analysis>. Accessed: 2023-04-25.
- Harrison Chase. 2022. *Langchain*. Available at: <https://github.com/langchain-ai/langchain>.
- Wenhu Chen. 2022. Large language models are few (1)-shot table reasoners. *arXiv preprint arXiv:2210.06710*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. *Training verifiers to solve math word problems*.
- X. Deng, V. Bashlovkina, F. Han, S. Baumgartner, and M. Bendersky. 2023. What do llms know about financial markets? a case study on reddit market sentiment analysis. In *Companion Proceedings of the ACM Web Conference 2023*, pages 107–110.
- Eugene F Fama and James D MacBeth. 1973. Risk, return, and equilibrium: Empirical tests. *Journal of political economy*, 81(3):607–636.
- Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2023. *News summarization and evaluation in the era of gpt-3*.
- Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *Conference Proceedings Name*, pages Start Page–End Page. Publisher Name, if available. Replace "Conference Proceedings Name" and page numbers with actual details.
- Noah Hollmann, Samuel Müller, and Frank Hutter. 2024. *Llms for semi-automated data science: Introducing for context-aware automated feature engineering*. Available at: <https://arxiv.org/abs/2305.03403> (Accessed: 28 February 2024).
- F. Hutter, L. Kotthoff, and J. Vanschoren. 2019. *Automated Machine Learning: Methods, Systems, Challenges*. Springer. Available for free at <http://automl.org/book>.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L elio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Th eophile Gervet, Thibaut Lavril, Thomas Wang, Timoth e Lacroix, and William El Sayed. 2024. *Mixtral of experts*.
- Zura Kakushadze. 2016. *101 formulaic alphas*.
- Amporn Atsawarungrangkit Kawee Numpacharoen. 2012. Generating correlation matrices based on the boundaries of their coefficients. *PLoS one*, 7:e48902.

- Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. [Large language models are zero-shot reasoners](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.
- Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. 2023a. [Large language models understand and can be enhanced by emotional stimuli](#).
- Zhenglin Li, Hanyi Yu, Jinxin Xu, Jihang Liu, and Yuhong Mo. 2023b. [Stock market analysis and prediction using lstm: A case study on technology stocks](#). *Innovations in Applied Engineering and Technology*, 2(1):1–6.
- Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. [Program induction by rationale generation: Learning to solve and explain algebraic word problems](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.
- X.Y. Liu, G. Wang, H. Yang, and D. Zha. 2023. [Fingpt: Democratizing internet-scale data for financial large language models](#). *Name of Journal or Conference if known*. Available online.
- Weimin Lyu, Songzhu Zheng, Tengfei Ma, and Chao Chen. 2022. [A study of the attention abnormality in trojaned bert](#)s. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4727–4741.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. [Training language models to follow instructions with human feedback](#). *Advances in Neural Information Processing Systems*, 35:27730–27744.
- Xiaofei Sun, Xiaoya Li, Shengyu Zhang, Shuhe Wang, Fei Wu, Jiwei Li, Tianwei Zhang, and Guoyin Wang. 2023. [Sentiment analysis through llm negotiations](#).
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#).
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). *Advances in neural information processing systems*, 30.
- Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. [GPT-RE: In-context learning for relation extraction using large language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore. Association for Computational Linguistics.
- S. Wang et al. 2023a. [Alpha-gpt: Human-ai interactive alpha mining for quantitative investment](#). Available at: <https://arxiv.org/abs/2308.00016v1> (Accessed: 28 February 2024).
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023b. [Gpt-ner: Named entity recognition via large language models](#).
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). *Advances in Neural Information Processing Systems*, 35:24824–24837.
- Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. [Empirical study of zero-shot NER with ChatGPT](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7935–7956, Singapore. Association for Computational Linguistics.
- Tianping Zhang, Yuanqi Li, Yifei Jin, and Jian Li. 2020. [Autoalpha: An efficient hierarchical evolutionary algorithm for mining alpha factors in quantitative investment](#). *arXiv preprint arXiv:2002.08245*.
- H. Zhao et al. 2024. [Revolutionizing finance with llms: An overview of applications and insights](#). Available at: <https://arxiv.org/abs/2401.11641> (Accessed: 28 February 2024).

A Company List

See Table 1.

B Other Results

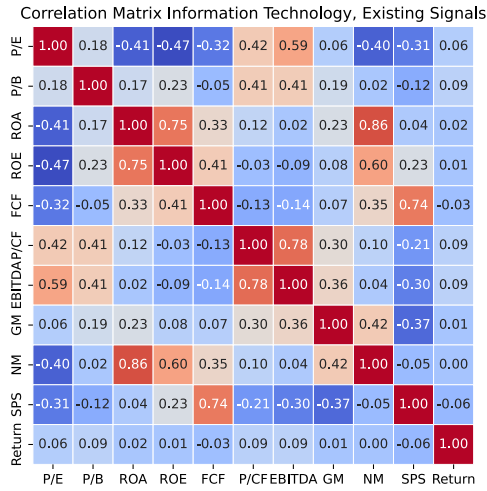
B.1 IT Companies with Future One-Month Returns

B.1.1 Correlation

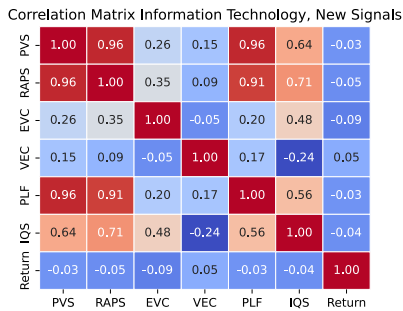
See Figure 7.

B.1.2 Fama-MacBeth

See Figure 8.



(a) Correlation of IT companies and future 1-month returns with existing signals.



(b) Correlation of IT companies and future 1-month returns with new signals.

Figure 7: Correlation of existing/new signals with returns.

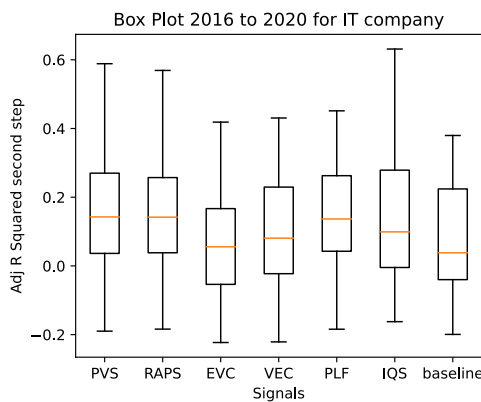


Figure 8: R_{adj}^2 values of IT companies and future 1-month returns for Fama-MacBeth step 2. The last box-plot is the baseline without any new signals.

Sector	Companies
Information Technology	"AAPL", "AKAM", "AMD" "ANET", "ANSS", "APH" "CDNS", "CDW", "CTSH" "ENPH", "EPAM", "FFIV" "FSLR", "FTNT", "GEN" "GLW", "IBM", "INTC" "IT", "JNPR", "KLAC" "LRCX", "MCHP", "MPWR" "MSFT", "MSI", "NOW" "NXPI", "ON", "PTC" "QCOM", "ROP", "STX" "SWKS", "TDY", "TEL" "TER", "TRMB", "TXN" "TYL", "VRSN", "WDC", "ZBRA"
Health Care	"ABBV", "ABT", "ALGN" "AMGN", "BAX", "BDX" "BIO", "BMY", "BSX" "CAH", "COR", "CRL" "CTLT", "CVS", "DGX" "DHR", "DXCM", "EW" "GILD", "HSIC", "TMO" "UHS", "VRTX", "VTRS" "IDXX", "ILMN", "INCY" "WST", "ZTS", "ISRG", "JNJ"
Energy	"APA", "COP", "CTRA" "EOG", "FANG", "HAL" "HES", "KMI", "MPC" "MRO", "OKE", "OXY" "PSX", "PXD", "SLB" "TRGP", "VLO", "WMB", "XOM"

Table 1: Company list of different sectors

B.2 Health Care Companies with Future One-Month Returns

B.2.1 Correlation

See Figure 9.

B.2.2 Fama-MacBeth

See Figure 10.

B.3 Health Care Companies with Future Three-Month Returns

B.3.1 Correlation

See Figure 11.

B.3.2 Fama-MacBeth

See Figure 12.

B.4 Energy Companies with Future One-Month Returns

B.4.1 Correlation

See Figure 13.

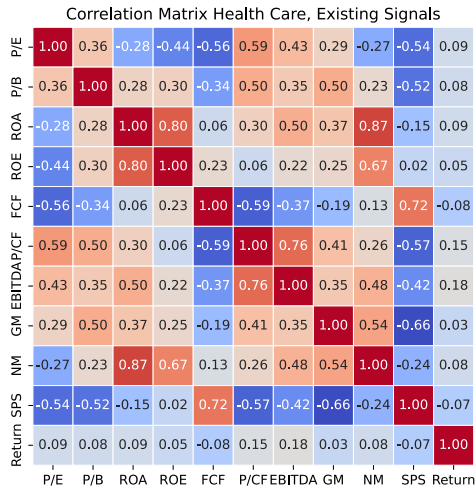
B.4.2 Fama-MacBeth

See Figure 14.

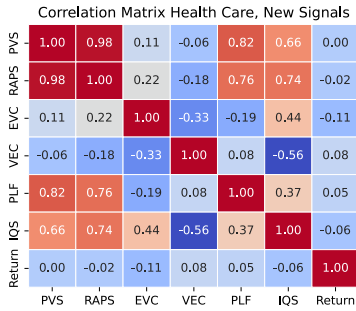
B.5 Energy Companies with Future Three-Month Returns

B.5.1 Correlation

See Figure 15.



(a) Correlation of Health Care companies and future 1-month returns with existing signals.



(b) Correlation of Health Care companies and future 1-month returns with new signals.

Figure 9: Correlation of existing/new signals with returns.

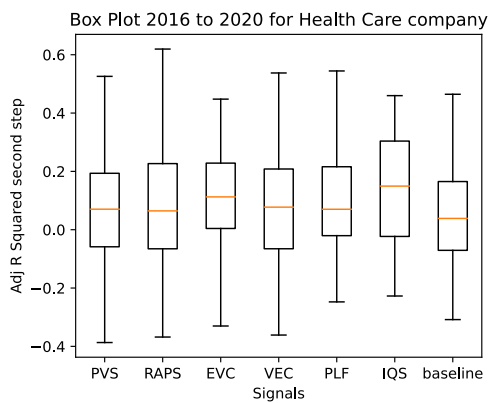
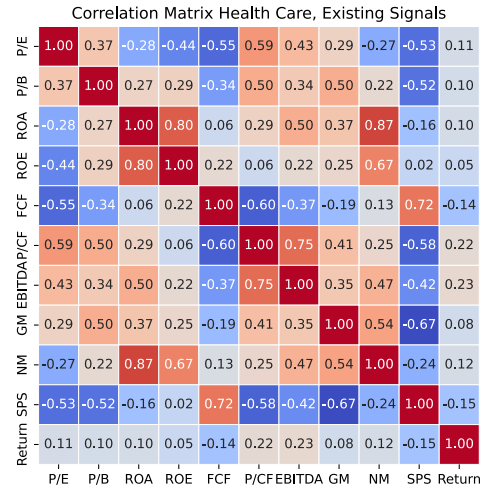


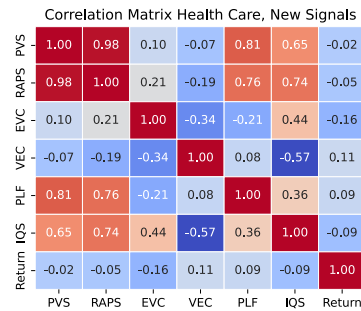
Figure 10: R_{adj}^2 values of Health Care companies and future 1-month returns for Fama-MacBeth step 2. The last boxplot is the baseline without any new signals.

B.5.2 Fama-MacBeth

See Figure 16.



(a) Correlation of Health Care companies and future 3-month returns with existing signals.



(b) Correlation of Health Care companies and future 3-month returns with new signals.

Figure 11: Correlation of existing/new signals with returns.

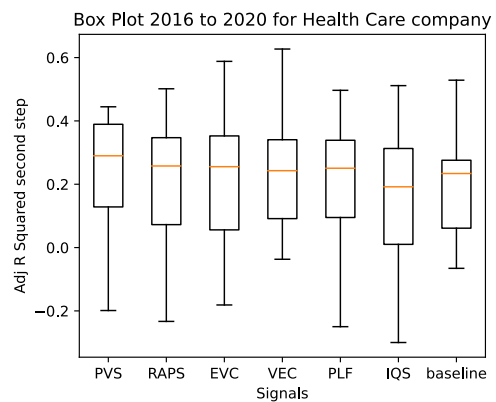
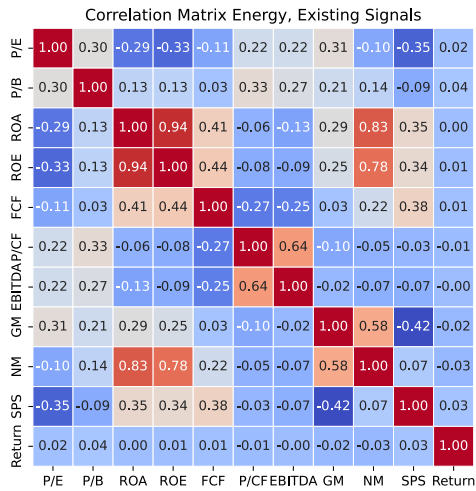
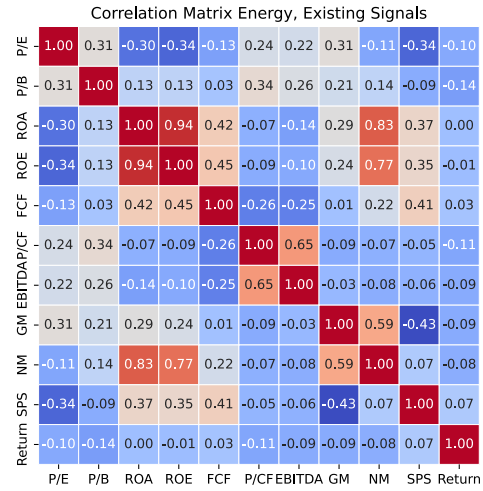


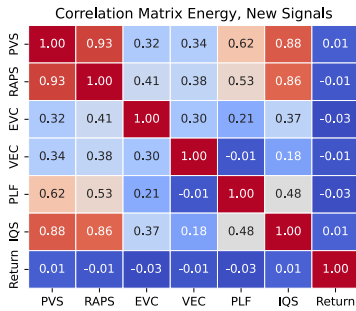
Figure 12: R_{adj}^2 values of Health Care companies and future 3-month returns for Fama-MacBeth step 2. The last boxplot is the baseline without any new signals.



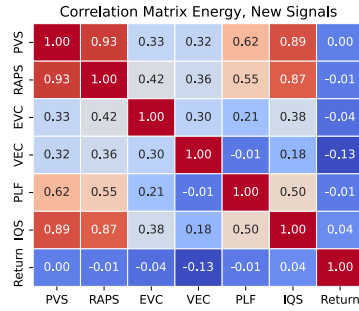
(a) Correlation of Energy companies and future 1-month returns with existing signals.



(a) Correlation of Energy companies and future 3-month returns with existing signals.



(b) Correlation of Energy companies and future 1-month returns with new signals.



(b) Correlation of Energy companies and future 3-month returns with new signals.

Figure 13: Correlation of existing/new signals with returns.

Figure 15: Correlation of existing/new signals with return.

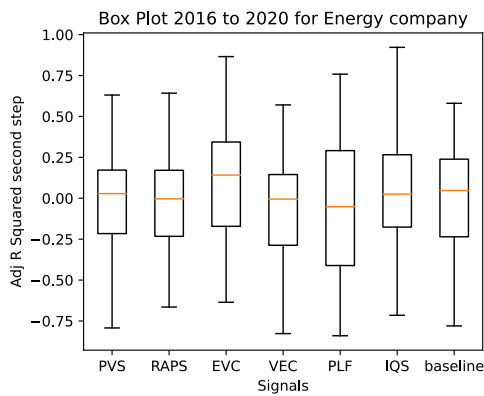


Figure 14: R_{adj}^2 values of Energy companies and future 1-month returns for Fama-MacBeth step 2. The last boxplot is the baseline without any new signals.

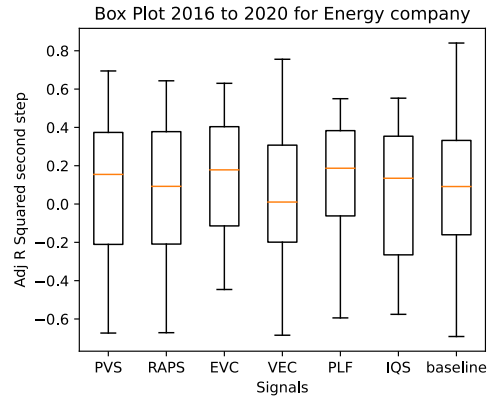


Figure 16: R_{adj}^2 values of Energy companies and future 3-month returns for Fama-MacBeth step 2. The last boxplot is the baseline without any new signals.



RACCOON: Real-world Advanced financial analysis through Comprehensive Natural language dataset

Seonghyun Kim¹, Kanghee Lee², Minsu Jeong³, Junghan Yoon³

¹Korea University, ²Konkuk University, ³Sungkyunkwan University,
¹qksksk657@korea.ac.kr, ²khlee91@konkuk.ac.kr, ³pisces03@skku.edu, ⁴yjh4037@g.skku.edu

Abstract

Our research introduces Raccoon¹, a benchmark dataset aimed at evaluating the cognitive capabilities of large language models (LLMs) in the complex domain of financial analysis. Traditional NLP benchmarks primarily focus on assessing the correctness of model outputs without examining the underlying cognitive processes. In contrast, Raccoon shows the simulation of human-like reasoning by integrating planning and reasoning tasks that mimic complicated human thought processes. Our study analyzes the extent to which LLMs understand the implicit meanings behind questions within the financial domain, and how these meanings are interpreted from various perspectives. To identify the differences, we compared the planning and reasoning processes of LLMs with those of human analysts. Our findings suggest that LLMs adopt more detailed approaches to problem-solving, which can sometimes limit their ability to effectively reach conclusions through reasoning. This comprehensive evaluation not only enhances our understanding of the cognitive limitations of current LLM architectures but also informs future development directions aimed at bridging the gap between artificial and human cognitive abilities in financial analysis.

1 Introduction

The emergence of LLMs in the field of computational linguistics has made considerable progress in natural language processing (NLP) tasks (Brown, Mann et al. 2020; Rosoł, Gašior et al.

2023). These models have not only demonstrated capabilities at or near human expert levels in specialized domains such as legal (Cui, Li et al. 2023) and clinical (Kwon, Ong et al. 2024). Despite these advances, a major gap remains in our understanding of how LLMs simulate human-like thought processes and reach conclusions (Huang, Chen et al. 2023). This gap is highlighted by evaluation methodologies and datasets that focus primarily on the model's ability to identify 'correct' answers, rather than clarifying the underlying cognitive processes involved (Yang, Qi et al. 2018; Liang, Bommasani et al. 2022). In contrast, human problem-solving requires clear and logical progression: understanding the problem, preparing necessary knowledge, and systematically connecting this knowledge to derive solutions (Phogat, Harsha et al. 2023; Song, Xiong et al. 2023). This core process also necessitates what is called a step-by-step agent-based approach (Wang, Wei et al. 2022; Zhang, Zhang et al. 2022; Sun, Zheng et al. 2023).

Our study aims to introduce a new benchmark dataset within the financial domain—an area characterized by complex thought processes encompassing political, economic, historical, and sociological considerations. This dataset is not only a challenge for LLMs to demonstrate human-like reasoning but also serves as a robust benchmark for assessing cognitive abilities in one of the most sophisticated contexts of human thought.

To address the complexities inherent in evaluating the cognitive capabilities of LLMs in the financial domain, our study concentrates on two pivotal aspects: planning and reasoning. These cognitive functions are essential for mimicking the complicated human thought process, which is

* All authors contributed equally to this research.

¹ The raccoon dataset is available for download at https://github.com/MrBananaHuman/finance_raccoon

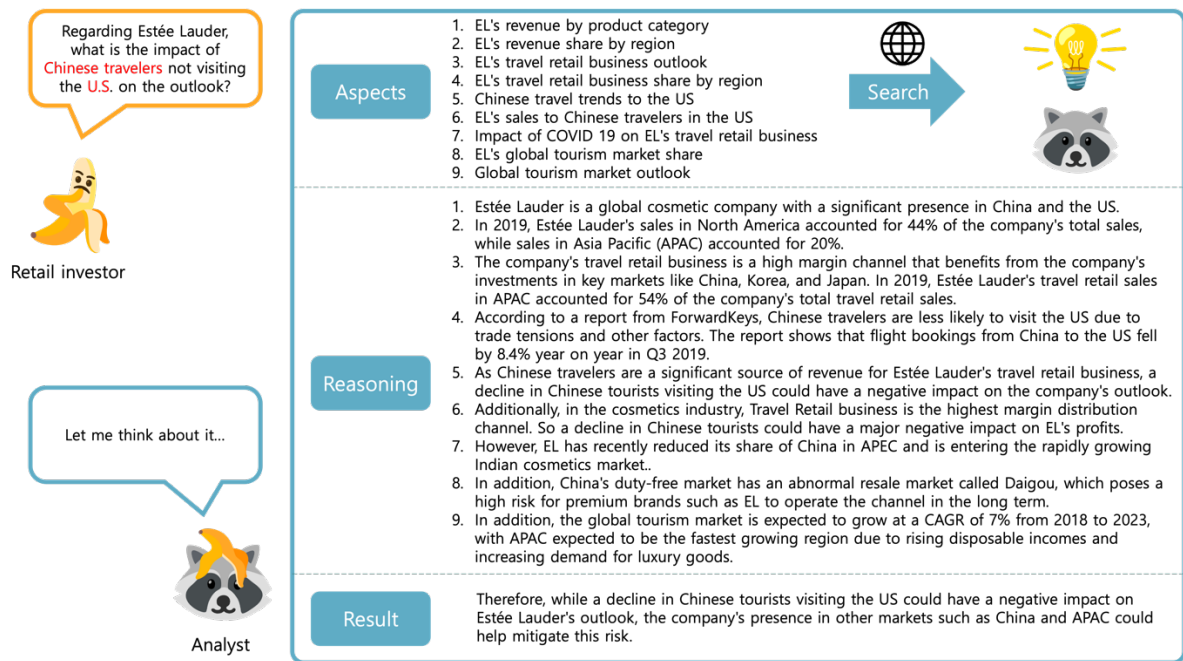


Figure 1: Overview of Raccoon dataset.

necessary for addressing complex and multifaceted issues.

Planning involves the ability to predict and clearly describe the steps necessary for problem-solving before beginning the process. This requires a strategic overview and sequence of actions aligned with achieving the intended outcomes. In the context of financial domain, this may include identifying relevant economic indicators, potential regulatory impacts, or market trends to consider.

On the other hand, reasoning is the process of connecting the dots between the information gathered during the planning stage and the specific question at hand. This includes deriving logical conclusions from a series of premises or known facts. This stage is crucial for navigating the complexities of financial data and interpreting it to make informed decisions or predictions.

Our study merges these two cognitive processes into a single framework for challenging LLMs with finance-related questions. (1) Aspect: Upon receiving a finance-related question, the model defines the aspects necessary to resolve the question. This involves sequentially outlining the key elements or considerations related to the query. (2) Reasoning: Once these aspects are planned and the necessary information is provided, the LLM generates a series of reasoning steps to construct a consistent and logical response.

We applied this two-step task to both human experts, LLMs and compared scenarios that did not include a planning stage. This comparative analysis not only highlights the importance of planning in complex problem-solving but also allows for evaluating the depth of understanding and cognitive similarities between humans and LLMs in handling complicated financial issues.

2 Related works

Inducing LLMs to generate reasoning steps can significantly assist them in identifying correct solutions in complex problems. Wei, Wang et al. 2022 implemented Chain of Thought (CoT) prompting in LLMs to trace reasoning pathways during problem-solving, improving both interpretability and accuracy (Chen, Ma et al. 2022; Mavi, Saparov et al. 2023; Lu, Peng et al. 2024). Despite these results, benchmarks often prioritize outcome correctness over reasoning process.

In financial analysis, the role of AI has traditionally been confined to predictive modeling. Jin, Tang et al. 2024 demonstrated the use of LLMs in forecasting stock market trends from historical data, yet their exploration into the reasoning processes of models was limited. This gap is starting to close with recent contributions like those from Chen, Chen et al. 2021; Son, Jung et al. 2023

who introduced datasets demanding semantic understanding and logical reasoning in economic contexts, though not fully replicating human-like cognitive processes.

Our research extends these efforts by offering a structured framework that assesses the capability of LLMs to perform step-by-step reasoning analogous with planning to that of a human financial analyst. This approach aligns with Chang, Wang et al. 2024, who discuss the operational paradigms of LLMs that emulate human cognitive processes, highlighting the need for frameworks that assess ethical fidelity alongside cognitive capabilities. Similarly, Momennejad, Hasanbeig et al. 2024 emphasize the necessity for integrating complex relational structures and functionalities like the human cognition to enhance the performance of LLMs in real-world tasks.

Furthermore, Li, Xu et al. 2024 underscore the growing role of LLMs in natural language generation (NLG) evaluation, focusing on their adaptability to produce coherent and contextually relevant assessments. Their work, employing techniques like Reinforcement Learning with Human Feedback (RLHF), strives to refine the generative capabilities of LLMs to enhance human-like reasoning processes, particularly in domain-specific tasks such as medical and financial text analysis.

Here, we address these issues by constructing a benchmark dataset for the financial domain that includes planning and reasoning tasks, with the goal of studying the differences in cognitive thinking between LLMs and humans.

3 Methods

3.1 Data Collection

Our study utilizes earnings call transcripts as the primary source of data. These transcripts were collected from Seeking Alpha, targeting companies listed in the S&P 500 index, spanning from the fourth quarter of 2019 to the second quarter of 2023. We extracted the necessary details such as ticker symbol, quarter, date, and participants using HTML tags. Additionally, supplementary information such as industry, sector, region, capitalization, and size were obtained via scraping the Nasdaq website. Each transcript was divided into sections, typically ‘Presentation’ and ‘Q&A’, using HTML tags. Within the Q&A sessions, statements made by each speaker were sequentially

recorded, ensuring the data preserved the flow of dialogue and interaction.

3.2 Data Preprocessing

We focused on the dialogues involving key corporate figures such as CFOs, Presidents, Chairmen, and CEOs. When an exchange pattern such as BOS-Analyst → President & CEO → JPMorgan-Analyst → Deutsche Bank-Analyst was identified, only the highlighted interactions were retained. All other non-sequential data were excluded to create concise, single-QA dialogues. Only questions pertaining to business conditions, forecasts, and economic outlooks were retained. Total 72 keywords were selected to filter questions related to future projections and market conditions, which included terms such as ‘Outlook’, ‘Projection’, ‘Market conditions’, ‘Economic climate’, and so on. This method ensured that only dialogues concerning strategic business outlooks and financial forecasts were processed for analysis.

3.3 Raccoon Dataset Construction

In this section, we detail the structure and procedures of our proposed benchmark dataset, as depicted in Figure 1. Our dataset is composed of several components, each tailored to reflect the decision-making process inherent in financial analysis.

The question transformation process involves converting the key content of each transcript into a concise query format. This aims to capture the essence of the transcript, ensuring that the questions generated encapsulate significant themes or central insights. For instance, if a transcript discusses a notable merger between two companies, the question derived from this discussion could be, “What are the potential financial impacts of the merger between Company A and Company B?”

We have identified ‘aspect’ as crucial elements in planning to these questions. Treated as search keywords, these aspects guide financial analysts in adequately addressing the queries. The aspects are organized sequentially to facilitate logical navigation through the search process. For instance, in responding to the merger question, aspects might include ‘Market share implications’, ‘Regulatory hurdles’, and ‘Synergy realization timelines.’

For each aspect, we associated virtual knowledge entries that provide the necessary information to address the aspect effectively. These entries are designed to emulate the type of data an

analyst might encounter when investigating a particular aspect in real-world scenarios. An example of such a knowledge entry for ‘synergistic savings’ might state, “Historically, similar mergers have reported an average of 15% synergy savings within the first two years.”

Furthermore, we have formalized a step-by-step reasoning process that links the question, aspect, and corresponding knowledge to generate coherent responses. This process mirrors the analytical thinking employed by financial analysts. The number of aspects and reasoning steps varies depending on the content of the original transcript.

Finally, we constructed a dataset comprising 50 such instances, each recorded with details such as the speaker, year, source data, and quarter. This structured approach not only enables the systematic simulation of financial analysis tasks but also serves as a robust framework for training machine learning models to emulate and generate human-like reasoning in the financial domain.

3.4 Categorization of Aspect

We categorized ‘aspect’ that is key part within financial transcripts, with a focus on real-world business scenarios. Each category is designed to highlight specific areas that are routinely evaluated by financial analysts. Below, we outline these seven principal categories, each accompanied by a revised example demonstrating diverse corporate perspectives.

Sales Portfolio Proportion: This category addresses the analysis of how sales are distributed among different products or services. It is crucial for understanding which segments are most lucrative or need strategic attention. For example, an aspect for Apple might be, “Proportion of total revenue derived from iPhone sales compared to other products.”

Customer List and Proportion: This focuses on identifying key customers and their sales contribution, which is vital for assessing risks associated with customer concentration. An aspect for Microsoft might be, “Percentage of total revenue contributed by enterprise clients in the cloud sector.”

Business Outlook: This category evaluates the prospects based on current conditions and planned strategies. An aspect for Tesla could be, “Expected growth in electric vehicle sales following the introduction of new model lines.”

Business Growth Strategy: We examine strategic initiatives aimed at business expansion. An example aspect for Amazon might be, “Strategies for market expansion in Asia through AWS services.”

Impact of Specific Events on Business: This category assesses the effect of external events on business operations. An aspect relevant to Nvidia might be, “Impact of global chip shortages on GPU production.”

Determine Economic Conditions Relevant to Your Business: This involves understanding macroeconomic factors that could impact a company. An example aspect for Goldman Sachs might be, “Effects of current interest rate trends on investment banking profitability.”

Sales and Operating Profit Guidance: This includes forecasts and expectations regarding sales and profitability, crucial for investor relations and strategic planning. An aspect for Coca-Cola might be, “Guidance on operating margins in light of fluctuating commodity prices.”

3.5 Response Generation

To compare response between human analyst and LLM, we constructed a dataset using responses generated by the Azure GPT API. This dataset is designed to investigate differences in reasoning processes when specific informational aspects are provided or not.

- Human: Financial analyst who have at least three years of experience at securities firm research centers.
- GPT-3.5: This model was prompted to generate reasoning without any prior provision of specific aspects or contextual knowledge; however, it was provided with 5-shot reasoning examples.
- GPT-3.5 with Raccoon: The model received both the aspects and the reasoning demonstrated in the human 5-shot scenarios, thereby aligning its generation process more closely with the analyst.

4 Evaluations

4.1 Human evaluation

In this section, we outline the qualitative evaluation methodology utilized to assess the effectiveness of our dataset, which includes both

qualitative and quantitative evaluation methods. The qualitative analysis focuses on the aspects and reasoning processes that the models generate in response to the transcripts.

The evaluation of aspects was approached from two distinct perspectives. First, the criterion of implicitness was used to determine if an aspect directly addresses the question's content or requires implicit, expert-level background knowledge pertinent to the discussed company and industry. An aspect is labeled as implicit if it draws upon knowledge not explicitly stated in the question but necessary to fully grasp the context. For example, a question regarding Estée Lauder that necessitates consideration of the US-China trade conflict would lead to an aspect deemed implicit, as it involves significant external economic factors impacting the business scenario.

Second, we assessed the relevance of each aspect. An aspect is considered relevant if it directly aids in answering the question or clarifying the topic discussed. Conversely, an aspect is marked as irrelevant if it does not align with the theme of the question. For instance, an aspect that discusses financial strategies would be relevant to a query about a company's future growth projections but would be irrelevant to a question focusing on the environmental impact of the company's operations. Following the evaluation of aspects, we also examined the reasoning generated by the models. The consistency metric checks if the reasoning steps maintain thematic and logical coherence throughout the response. A reasoning process is deemed consistent if each step logically follows the preceding one, without any abrupt deviations or shifts in logic. For example, reasoning that begins with a discussion on financial growth due to market expansion and then abruptly shifts to product quality without a logical link is considered inconsistent.

Lastly, the specificity of each reasoning step is evaluated based on its grounding in specific, verifiable data or detailed logical argumentation. Reasoning is classified as specific if it includes concrete data, references, or clearly defined logic. It is deemed nonspecific if it largely relies on vague statements or assumptions without significant support. For instance, a statement like, "The company will likely see a 10% increase in sales due to the new product launch, as indicated by early market tests," exemplifies specific reasoning. Conversely, a generalized statement such as, "The company will do better because it has good

products," lacks specificity due to its reliance on broad, unsubstantiated claims.

4.2 Token Overlap

In the process of our quantitative evaluation, we assess the token overlap ratio to determine the lexical similarity between the generated aspects and the corresponding question. We compare the tokens from the question with those of the aspects to investigate whether the generated aspects are directly extracted from the question content or whether they introduce novel yet related concepts. The initial step involves tokenizing the question sentences and their corresponding aspects using a GPT tokenizer. Let T denote the tokenizer function, Q represents the question sentence, and A_n signifies the n -th aspect derived from Q . The token overlap is computed as follows:

$$\text{Overlap}(Q, A_n) = \frac{|T(Q) \cap T(A_n)|}{|T(Q) \cup T(A_n)|}$$

The token overlap value ranges from 0 to 1, where 0 denotes no overlap and 1 indicates complete duplication. This methodology facilitates the quantitative evaluation of the shared lexical content between two sequences. Such a metric proves especially valuable in tasks necessitating the measurement of lexical similarity, including paraphrase detection and text entailment.

4.3 Perplexity

The second metric for quantitative assessment is the comparison of perplexity across pre-trained large-scale language models. We measure the perplexity of the aspect generated by the language model when given a question as input. Perplexity serves as a metric to evaluate the likelihood of the sentences produced by the language model. Given a sequence of output tokens $Y = \{y_0, \dots, y_m\}$, we calculate the perplexity as follows:

$$\text{PPL}(Y) = \sqrt[m]{\prod_{i=1}^m \frac{1}{p(y_i | y_0, \dots, y_{i-1})}}$$

Therefore, to verify the generative plausibility of the aspect corresponding to the question, we compare the aspects generated by humans and those produced by the GPT for the same query.

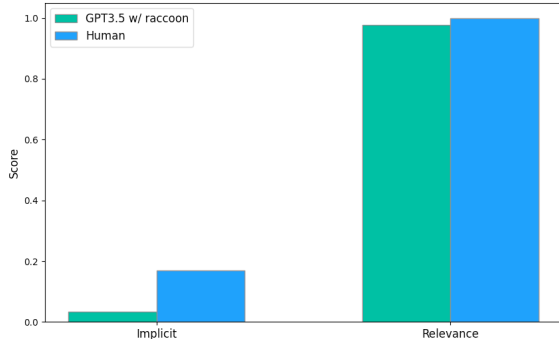


Figure 2: The performance of Human and GPT on aspect generation

5 Results

5.1 Aspect Analysis

To evaluate the performance of GPT in generating aspects, we conducted a human evaluation, as depicted in Figure 2. We found that GPT-generated responses contained a considerably lower proportion of implicit aspects compared to those generated by humans. The scores are normalized to a maximum of 1 point.

We classified the generated aspects into distinct categories and examined their distribution. According to the data presented in Figure 3, it is evident that human participants distributed their responses evenly across seven categories, with the distribution resembling an approximately uniform spread. In contrast, the aspects generated by GPT were disproportionately concentrated in certain categories, with more than half of the responses falling into specific ones. This analysis underscores a significant difference in the approach to problem-solving between humans and the GPT model. Humans tend to employ a diverse range of perspectives when addressing a question, which is reflected in the even distribution of response categories. On the other hand, GPT shows a tendency to focus narrowly on fewer categories, indicating a limitation in the model's ability to diversify its approach and consider multiple aspects of a problem. This pattern suggests that while GPT can effectively generate responses, its capacity to mimic the multifaceted approach typical of human reasoning is still constrained.

To substantiate this hypothesis, we examined the proportion of token overlap between the questions and the generated aspects. According to the data presented in Figure 4, the token overlap in aspects generated by humans was statistically significantly lower compared to those generated by GPT. This

suggests that human participants tend to generate more varied and conceptually distinct aspects that do not merely repeat the tokens present in the questions.

On the other hand, GPT demonstrated a higher tendency to reuse tokens from the questions in its generated aspects. This behavior indicates a more literal or direct interpretation and utilization of the input text, which may limit the model's ability to generate responses that introduce new or diverse perspectives independent of the explicit content of the questions. This pattern provides quantitative support for the earlier observation that GPT, while capable of generating relevant aspects, tends to do so in a less diverse and more question-bound manner compared to human responses.

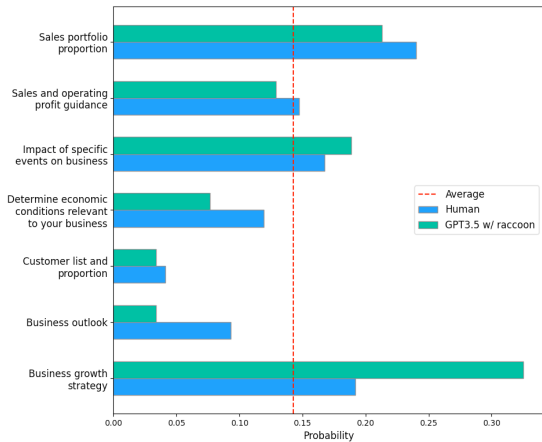


Figure 3: Categorical comparison of aspects

5.2 Reasoning Analysis

To compare reasoning steps, we discuss the methods used to compare reasoning steps by evaluating consistency, specificity, and answer validation across responses generated by human analysts and GPT models in Figure 5.

Consistency in reasoning was analyzed to determine how often GPT models repeated concepts from previous steps in subsequent reasoning processes, revealing a strong tendency towards redundancy. In contrast, human respondents frequently introduced logical leaps between steps, indicative of a more dynamic and less linear reasoning approach.

Specificity was assessed in scenarios where both GPT-3.5 with Raccoon and human participants were provided with specific aspects and knowledge. We observed that each reasoning step effectively referenced the necessary

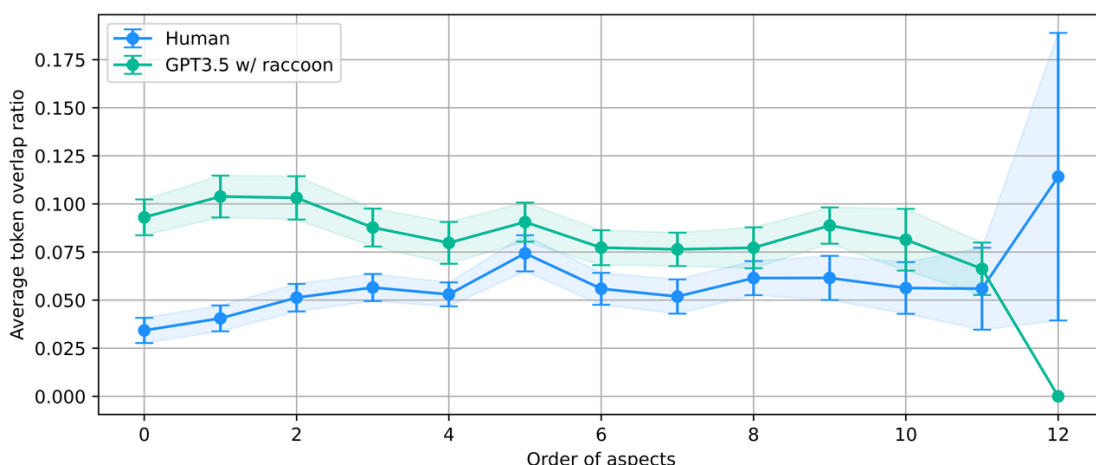


Figure 4: Averaged token overlap ratio for each number of aspects

information. This observation suggests that supplying well-defined aspects and relevant knowledge can reduce instances of ‘hallucination’ in GPT responses, where the model generates irrelevant or fictitious content.

Regarding answer validation, it was noted that GPT-3.5 often avoided definitive conclusions. Particularly with the GPT-3.5 Raccoon configuration, as the reasoning progressed, there was a noticeable tendency to generate conclusions that were not pertinent to the initial question posed. This pattern underscores a challenge in maintaining the relevance of the responses as the model attempts to integrate and reason with the provided knowledge and aspects.

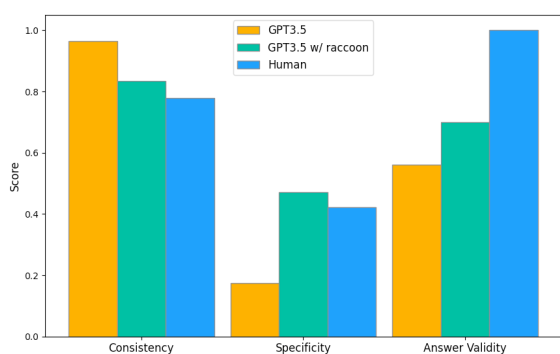


Figure 5: The performance of Human and GPT on reasoning generation

These evaluations underscore significant differences in reasoning quality between human analysts and language models, especially in maintaining consistency, utilizing relevant knowledge effectively, and producing valid

conclusions. The insights from this comparative analysis are crucial for understanding and enhancing the reasoning capabilities of AI models in complex analytical tasks.

5.3 Perplexity Analysis

In furthering our examination of the generative differences between GPT and human responses, we explored whether the observed patterns were specific to Azure's GPT-3.5 API by comparing the PPL of aspects generated by GPT (Figure 6). For this purpose, we used the following prompt template for the perplexity evaluation and ensured that the same prompt was used for all models:

You are a financial domain expert analyst. Please create search queries to answer questions related to the given ticker's company.
 Ticker: $\{ticker_id\}$
 Question: $\{question\}$
 Aspects: $\{aspect_list\}$

In the template, $\{ticker_id\}$ represents the target company's stock symbol, $\{question\}$ is the specific inquiry, and $\{aspect_list\}$ contains either aspects generated by GPT or those created by human analysts. We calculated the PPL for a total of 50 examples.

Across the board, it was observed that the aspects generated by humans yielded relatively higher PPL values in all public models (Llama-2, Llama-3, Mistral, Phi-2, Falcon) compared to those generated by GPT. Notably, statistical significance in PPL differences was found within the outputs of Llama-2, Llama-3, and Phi-2 models. These

findings suggest that decoder-based models indeed generate in a manner that is distinctly different from human creation. The significant disparities in PPL values underscore the dissimilarity in the naturalness or predictability of language between the two sources. While GPT-generated aspects tend to align more closely with the language patterns the models have been trained on, human-generated aspects seem to reflect a broader and possibly more unpredictable range of language use within the financial domain.

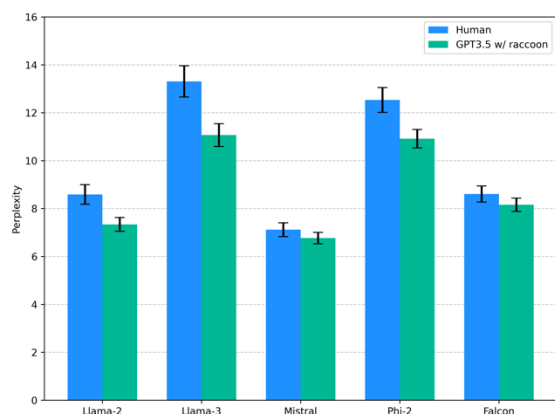


Figure 6: Comparative perplexity analysis of Human and GPT outputs across public LLMs.

6 Conclusion

In conclusion, our comprehensive investigation into the cognitive capabilities of LLMs within the financial domain has highlighted both the strengths and limitations inherent in these advanced AI systems. By examining the performance of LLMs in comparison to that of human experts in complex financial analysis tasks, our study has illuminated significant discrepancies in the depth and authenticity of the models' reasoning processes. While LLMs excel in producing relevant and logically coherent responses, they predominantly rely on explicit cues from input questions, which often limits their ability to generate diverse perspectives and understand implicit content.

Furthermore, despite enhancements through our 'Raccoon' configuration—which provides structured aspects and reasoning paths—challenges persist in ensuring consistency and relevance throughout the reasoning process. This configuration has indeed improved the performance of GPT models, but it also reveals that even advanced GPT models struggle to match the nuanced understanding and broader information

integration displayed by human analysts. This suggests that while LLMs can generate syntactically correct and contextually appropriate answers, they lack the human-like ability to seamlessly navigate and link multiple domains of knowledge, often resulting in a more constrained analytical scope.

Overall, the findings from this research underscore the critical need for continuous evolution in the design and development of LLMs, especially if they are to be effectively employed in complex, real-world tasks like financial analysis. By aligning model development with insights gained from rigorous comparative analyses with human cognitive processes, there is significant potential to enhance LLMs' capabilities. Such advancements could make these models not just supplementary tools but robust partners in augmenting human expertise, thereby ensuring their efficacy and reliability in practical applications across various domains.

References

- Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry and A. Askell. 2020. Language models are few-shot learners. *Advances in neural information processing systems* **33**: 1877-1901.
- Chang, Y., X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang and Y. Wang. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology* **15**(3): 1-45.
- Chen, W., X. Ma, X. Wang and W. W. Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.
- Chen, Z., W. Chen, C. Smiley, S. Shah, I. Borova, D. Langdon, R. Moussa, M. Beane, T.-H. Huang and B. Routledge. 2021. Finqa: A dataset of numerical reasoning over financial data. *arXiv preprint arXiv:2109.00122*.
- Cui, J., Z. Li, Y. Yan, B. Chen and L. Yuan. 2023. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv preprint arXiv:2306.16092*.
- Huang, J., X. Chen, S. Mishra, H. S. Zheng, A. W. Yu, X. Song and D. Zhou. 2023. Large

- language models cannot self-correct reasoning yet. arXiv preprint arXiv:2310.01798.
- Jin, M., H. Tang, C. Zhang, Q. Yu, C. Liu, S. Zhu, Y. Zhang and M. Du. 2024. Time Series Forecasting with LLMs: Understanding and Enhancing Model Capabilities. arXiv preprint arXiv:2402.10835.
- Kwon, T., K. T.-i. Ong, D. Kang, S. Moon, J. R. Lee, D. Hwang, B. Sohn, Y. Sim, D. Lee and J. Yeo. 2024. Large Language Models Are Clinical Reasoners: Reasoning-Aware Diagnosis Framework with Prompt-Generated Rationales. Proceedings of the AAAI Conference on Artificial Intelligence.
- Li, Z., X. Xu, T. Shen, C. Xu, J.-C. Gu and C. Tao. 2024. Leveraging large language models for nlg evaluation: A survey. arXiv preprint arXiv:2401.07103.
- Liang, P., R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu and A. Kumar. 2022. Holistic evaluation of language models. arXiv preprint arXiv:2211.09110.
- Lu, P., B. Peng, H. Cheng, M. Galley, K.-W. Chang, Y. N. Wu, S.-C. Zhu and J. Gao. 2024. Chameleon: Plug-and-play compositional reasoning with large language models. *Advances in Neural Information Processing Systems* **36**.
- Mavi, V., A. Saparov and C. Zhao. 2023. Retrieval-Augmented Chain-of-Thought in Semi-structured Domains. arXiv preprint arXiv:2310.14435.
- Momennejad, I., H. Hasanbeig, F. Vieira Fruijeri, H. Sharma, N. Jovic, H. Palangi, R. Ness and J. Larson. 2024. Evaluating cognitive maps and planning in large language models with CogEval. *Advances in Neural Information Processing Systems* **36**.
- Phogat, K. S., C. Harsha, S. Dasaratha, S. Ramakrishna and S. A. Puranam. 2023. Zero-Shot Question Answering over Financial Documents using Large Language Models. arXiv preprint arXiv:2311.14722.
- Rosoł, M., J. S. Gąsior, J. Łaba, K. Korzeniewski and M. Młyńczak. 2023. Evaluation of the performance of GPT-3.5 and GPT-4 on the Polish Medical Final Examination. *Scientific Reports* **13**(1): 20512.
- Son, G., H. Jung, M. Hahm, K. Na and S. Jin. 2023. Beyond classification: Financial reasoning in state-of-the-art language models. arXiv preprint arXiv:2305.01505.
- Song, Y., W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang and R. Yao. 2023. Restgpt: Connecting large language models with real-world restful apis. arXiv preprint arXiv:2306.06624.
- Sun, J., C. Zheng, E. Xie, Z. Liu, R. Chu, J. Qiu, J. Xu, M. Ding, H. Li and M. Geng. 2023. A survey of reasoning with foundation models. arXiv preprint arXiv:2312.11562.
- Wang, X., J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery and D. Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:2203.11171.
- Wei, J., X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le and D. Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* **35**: 24824-24837.
- Yang, Z., P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov and C. D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. arXiv preprint arXiv:1809.09600.
- Zhang, Z., A. Zhang, M. Li and A. Smola. 2022. Automatic chain of thought prompting in large language models. arXiv preprint arXiv:2210.03493.

TopoLedgerBERT: Topological Learning of Ledger Description Embeddings using Siamese BERT-Networks.

Sander Noels^{1,2}, Sébastien Viaene¹, Tijn De Bie²

¹Silverfin, Gaston Crommenlaan 12, 9050 Ghent, Belgium

²Department of Electronics and Information Systems, Ghent University, 9000 Ghent, Belgium

Correspondence: sander.noels@ugent.be

Abstract

This paper addresses a long-standing problem in the field of accounting: mapping company-specific ledger accounts to a standardized chart of accounts. We propose a novel solution, TopoLedgerBERT, a unique sentence embedding method devised specifically for ledger account mapping. This model integrates hierarchical information from the charts of accounts into the sentence embedding process, aiming to accurately capture both the semantic similarity and the hierarchical structure of the ledger accounts. In addition, we introduce a data augmentation strategy that enriches the training data and, as a result, increases the performance of our proposed model. Compared to benchmark methods, TopoLedgerBERT demonstrates superior performance in terms of accuracy and mean reciprocal rank.

1 Introduction

Ledger accounts are essentially the building blocks of accounting that provide detailed records of all financial activities within a company. They collectively form a hierarchical structure known as the chart of accounts (COA), which offers a comprehensive view of a company's financial activities. However, companies often customize their COA to accommodate their specific business needs and regulatory requirements, resulting in a lack of standardization and comparability (Jørgensen and Igel, 2021; Wang, 2023). This scenario has brought about various standardization initiatives aiming to enhance the quality of COAs. A standardized COA not only improves the accuracy of financial data, but also promotes comparability and improves the overall quality of the financial information reported (Rylee, 2017). Moreover, it paves the way for machine learning solutions that can generalize financial information between companies (Bergdorf, 2018).

The mapping of company-specific ledger accounts to a common COA remains a challenge. Existing academic and industry solutions typically create custom machine learning models for individual companies, or limit data to a predefined set of accounts (Bergdorf, 2018). Furthermore, the hierarchical information inherent in a COA, which could significantly enhance the performance of the mapping solution, has never been used.

We hypothesize that the hierarchical structure of the ledger accounts within the COA and their interrelationships can enhance the performance of the mapping solution. Considering the meaning and hierarchical nature of the ledger account descriptions, we believe that a model that understands these relationships should deliver superior ledger mapping results.

We propose TopoLedgerBERT, a novel sentence embedding method for ledger account mapping that overcomes the limitations of existing methods by exploiting the hierarchical nature of ledger accounts. To create the mapper, we take a ledger description from the custom COA, compute its embedding, and then calculate its distance to embeddings of standard ledger accounts. The standard ledger account having the minimum distance is considered the mapping for the custom ledger account. TopoLedgerBERT incorporates a graph distance concept to include hierarchical information from the COA during the sentence embedding process. By doing so, it captures not only the semantic similarity of ledger accounts but also their hierarchical structure. Additionally, this method is capable of making predictions for previously unseen companies and COAs.

Our main contributions are summarized as follows:

- We propose TopoLedgerBERT, a novel sentence embedding method specifically designed for the mapping of ledger accounts in accounting.
- Unlike existing solutions that require cus-

tomized machine learning models for individual companies, TopoLedgerBERT can adapt to new companies and COAs without any prior setup.

- We demonstrate that the TopoLedgerBERT model outperforms benchmark methods in terms of accuracy and mean reciprocal rank.
- Our empirical evaluation reveals that TopoLedgerBERT effectively captures hierarchical relationships between different ledger accounts.
- Finally, we demonstrate the importance of fine-tuning pre-trained embeddings and employing data augmentation strategies for improving the model's performance.

The remainder of this paper is as follows: Section 2 reviews the related work. Section 3 delves into the background and explains how a COA can be represented as a graph. Section 4 details our proposed TopoLedgerBERT model, followed by an empirical evaluation in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

2.1 Ledger Accounts and the Chart of Accounts

Ledger accounts serve as units that record and summarize a company's transactions. A ledger account typically includes an ID that possibly reflects the hierarchy and a short description that can sometimes be challenging to decipher. Each ledger account also contains a set of bookings for each financial year. A company generally comprises around 100 ledger accounts.

These accounts collectively form a hierarchy known as the "chart of accounts" (COA), a classification system for financial information. The COA segregates expenditures, revenues, equity, assets, and liabilities into distinct categories, providing a comprehensive breakdown of all financial transactions during a specific accounting period. However, a long-standing issue in this domain is the lack of standardization in the COA.

2.2 Factors Driving the Customization of COAs

Customizing COAs is driven by several factors. Companies often require specialized ledger accounts to support specific financial transactions relevant to their sector (Jorge et al., 2022). In addition to sector-specific needs, the complexities of tax law

and individual company requirements contribute to the customization of COAs. These complexities underline the subjective nature of bookkeeping, where companies vary in their practices, making standardization a challenging endeavor (Liu et al., 2021). Two examples can illustrate these complexities:

- **Tax Compliance:** US tax laws dictate varying rates at which different expenses are deductible. For example, expenses for business vehicles and office snacks are 100% deductible, while business meals are only 50% deductible. Consequently, separate accounts for these expenses become necessary for accurate tracking and compliance.
- **Business Needs:** Companies may have different needs for account specificity. For example, company A might be satisfied with a generalized account named "cars and trucks" for all its automobile-related expenses. However, company B might prefer more granular accounting and create separate accounts for "fuel", "vehicle maintenance", and "vehicle insurance".

Although customization can support specific business needs and regulatory compliance, it can also lead to inconsistencies and difficulties in comparing financial statements between companies (Dhole et al., 2015; Noels et al., 2022, 2023a). Therefore, while supporting the freedom for companies to customize their COAs, it is essential to develop methods for translating these custom COAs into a unified structure.

2.3 The Importance of Standardization

Standardization initiatives have been introduced, both on national and international scale, to improve various aspects of accounting (EUROSTAT, 2017; Jorge et al., 2022). For example, in 2008, the European Commission recommended the use of standard COAs for small enterprises in a report that highlights best accounting practices (Commission, 2008). Standardization initiatives are beneficial for various reasons. Primarily, they enhance the accuracy of financial data and, consequently, financial reporting. A unified COA allows for the uniform sorting and aggregating of financial data, facilitating consistent accounting practices and routines. This, in turn, helps to develop effective IT and control systems (Jorge et al., 2022). According to Rylee (2017), the standardization of COAs ensures a consistent and streamlined reporting of financial

data, thus preventing inaccuracies that might result from too complex COAs.

2.4 Standardization Initiatives in Related Work

The "multiple charts of accounts problem" is a longstanding issue in both academia and industry due to the unique customization of COAs by different companies. The challenge lies in the need for a system that can effectively generalize and merge financial information across various companies and sectors, even with minimal historical data (Bergdorf, 2018).

Past strategies have tried to circumvent this problem using company-specific classifiers or the use of uniform ledger accounts or COAs. Jørgensen and Igel (2021) propose a system capable of mapping transactions to the appropriate ledger accounts across different companies. However, this approach required one to first manually map specific account codes into a unified space. Munoz et al. (2022) limit themselves to a set of accounts used by different companies, employing hierarchical classifiers to predict the correct account code for specified invoice line items. Despite the innovative aspect of their approach, this method is limited to the account codes present in their training data, making it ineffective for accounts added after their training period. Noels et al. (2022, 2023a) define a COA with all possible financial accounts hierarchically structured within a financial statement, allowing them to represent every company. This allows them to compute the similarity between the financial statements of each company. However, due to the lack of uniformity, the use of this method for automated financial statement analysis remains limited, indicating the need for a method capable of automating the translation to a standardized COA.

All the methods discussed face the challenge known as the "cold-start problem", which occurs when a system has to make predictions for ledger accounts that were not present in the training dataset. This limitation has inspired efforts toward the automation of COA standardization. In response to this challenge, a viable solution lies in automating the translation of a company's unique financial structure into a unified accounting chart. Such a chart should be capable of accounting for all potential company transactions and could therefore be effectively implemented in machine learning systems. According to Jørgensen and Igel (2021), the most significant opportunity for improvement

lies in the automated transformation to a standardized COA, while still allowing company-specific COAs tailored to individual company needs or tax regulations.

2.5 Account Mapping in Academia and Industry

Companies like Oracle and Silverfin provide their own mapping solutions. Oracle's solution includes a COA mapping feature for data consolidation, but this rule-based system requires substantial maintenance due to the need for continuous adaptation. Silverfin uses machine learning in its AI mapping feature for better adaptability. The company approaches account mapping as a classification problem that depends on textual information as well as transaction details associated with a custom ledger account. Given this approach of considering the problem as a classification tied to a COA, it inevitably faces difficulties when encountering new COAs. Both aim to automate the process of mapping custom ledger accounts, highlighting their importance in the industry.

There is a growing academic interest in automated account mapping solutions, as demonstrated by a system proposed by Wang (2023). The system standardizes custom ledger accounts by automatically mapping them to standard ones using three pre-trained embedding methods (TF-IDF, Word2Vec, FinBERT). To the best of our knowledge, this is the first academic work that offers a solution to the laborious process of mapping custom ledger accounts to a standardized taxonomy.

The rise of large language models opens up new opportunities in financial data applications. The rationale is that these methods could be better adapted to the unique terminology used in financial data, potentially improving the accuracy of mapping custom ledger accounts (Noels et al., 2023b). According to Wu et al. (2023), the complexity of the financial domain, the unique terminology, scarce data, and privacy regulations have limited the success of large language models in providing valuable insights. Therefore, fine-tuning pre-trained embeddings, such as FinBERT, could potentially overcome these hurdles and offer more accurate representations of financial data, thus improving the mapping process (Araci, 2019).

Liu et al. (2021) introduce another promising approach, using siamese network-inspired neural networks to perform binary matching of transaction and account information. This method uses

pre-trained embedding models to place transactions and accounts in the same vector space. This approach could be applied to the account mapping problem, where custom ledger account descriptions are paired with their standardized ledger account descriptions. This approach offers a distinct advantage over classification-based methods, as it moves away from handling a fixed COA.

In conclusion, the siamese network-inspired approach appears to be a promising avenue for future research, as it is not restricted to a predefined number of accounts and can accommodate unseen ledger accounts and COAs. Additionally, the idea of fine-tuning language models on financial data presents a compelling opportunity. Given the unique terminology, models trained specifically for this mapping problem would be better equipped to handle the complexity and nuances of ledger descriptions. These insights inspire further exploration of the development of more robust, adaptable, and accurate systems to automate the account mapping process in financial accounting.

3 Background

This section provides an overview of the COA as a graph and the application of siamese BERT-networks in account mapping.

3.1 Chart of Accounts as a Graph

As stated by [Yang and Cogill \(2013\)](#), a vertex-labeled tree is a natural representation of the ledger accounts present within a COA.

As an example, we consider the *assets* section of a balance sheet. The *assets* section can be divided into *fixed* and *current assets*. Both of these ledger accounts can be subdivided into more detailed accounts, as shown in Figure 1. Note that the vertex-labeled representation of a COA is not limited to this specific example. A subset of ledger accounts and their reciprocal relationship are given for exemplary purposes.

This representation method clearly preserves the structural property of a COA. This means that a COA for every company can be represented by a vertex-labeled tree, where the vertex labels are the ledger account descriptions.

3.2 Siamese BERT-Networks

Siamese BERT-networks are a type of sentence embedding method that can embed a sentence, taking into account the semantic and contextual information of a sentence. In the context of account

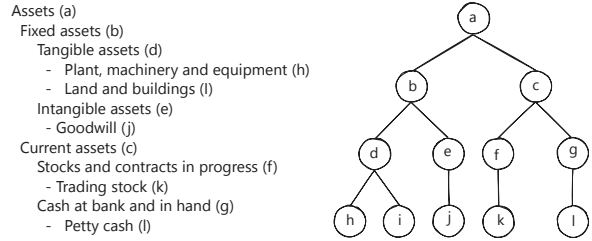


Figure 1: Left: Assets subsection of the balance sheet. Right: A vertex-labeled tree representation of the assets subsection of the balance sheet.

mapping, these networks use the bi-encoder transformer architecture of the Sentence-BERT network ([Reimers and Gurevych, 2019](#)) to learn a joint representation space of the ledger account descriptions and their standardized counterparts.

Training a sentence embedding method involves inputting pairs of sentences into the network. These pairs can be either positive pairs, which consist of semantically similar sentences, or negative pairs, which consist of semantically dissimilar sentences. The network then calculates the distance between the two inputs. It attempts to minimize the distance for similar inputs (e.g., identical account descriptions) and maximize the distance for dissimilar inputs (e.g., different account descriptions).

4 TopoLedgerBERT

4.1 Rationale Behind the Proposed Solution

The rationale behind the proposed solution is to address the limitations of current sentence embedding methods in the context of ledger account mapping. As demonstrated in previous studies, the hierarchical relationships between the ledger accounts play a significant role in understanding the financial position of a company ([Yang and Cogill, 2013](#); [Noels et al., 2022, 2023a](#)). Therefore, the embedding method we propose aims to capture not only the semantic similarity of the ledger accounts, but also their hierarchical composition within the COA. In doing so, we hypothesize that the proposed method can provide better embeddings for ledger accounts, leading to improved mapping accuracy and efficiency in the accounting domain.

Let us consider a specific example. The ledger account for *land and buildings* can be subdivided into *agricultural land* and *residential land*. In this context, the proposed method should consider *agricultural land* and *residential land* as more similar to each other than to other ledger accounts that are located farther apart on the graph. This is due to

their direct hierarchical relationship and position in the COA.

4.2 TopoLedgerBERT

As mentioned in Section 3.1, a COA can be symbolized by a vertex-labeled tree, represented as $T = (V, E, L, \phi)$. In this context, $V(T) = \{1, \dots, n\}$ stands for a set of n vertices, $E \subseteq V \times V$ indicates the set of edges, and L is a collection of unique labels corresponding to each vertex in V . These labels in L define the standardized ledger account descriptions within a target COA. Therefore, we introduce a function $\phi : V \rightarrow L$ that maps every vertex v in V to a unique label l in L .

We also define the shortest path distance $d(i, j)$ between the vertices i and j in T and construct the distance matrix $D(T)$ of T . Using $D(T)$, we construct a similarity matrix $S(T)$, where each value

$$s_{ij} = 1 - \frac{d_{ij}}{\max(D)} \quad (1)$$

represents the similarity between the vertices i and j . If the value of s_{ij} is high, it signifies close proximity within the tree T , while low values indicate a significant distance.

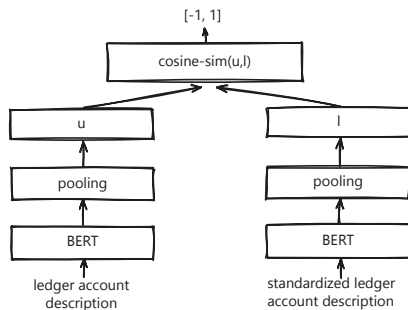


Figure 2: Diagram of the Sentence-BERT architecture for computing ledger account description similarity scores for the ledger account mapping problem.

We create positive training samples by pairing each original ledger account description with its corresponding ledger account from the COA, assuming that there is an appropriate standardized ledger account for each account description. We denote the set of original ledger account descriptions as U , and the corresponding standardized ledger account descriptions as L . Each sample in the positive training dataset D_+ is a triplet $(u, l_u, s_i) \in U \times L \times \{1\}$, where $s_i = 1$ indicates the highest similarity score.

To define the set D_+ , we introduce v_u which represents the vertex in V corresponding to the

original ledger account description u . The function $\phi : V \rightarrow L$ then maps each vertex v_u to its corresponding unique label l_u . Given these definitions, we can formally define D_+ as follows:

$$D_+ = \{(u, \phi(v_u), 1) | u \in U\}, \quad (2)$$

where u represents an original ledger account description, and $\phi(v_u)$ denotes its corresponding label in the COA.

For each tuple $(u, \phi(v_u), 1) \in D_+$, we generate K additional negative training samples. First, let us define the set $V_{\setminus v_u}^K \subseteq V \setminus \{v_u\}$ with $|V_{\setminus v_u}^K| = K$, a randomly selected subset of K vertices from V different from v_u . Subsequently, for each $v \in V_{\setminus v_u}^K$, we calculate the similarity score $s_{v_u v}$ using the similarity matrix $S(T)$. The set of negative training samples D_- is defined as:

$$D_- = \{(u, \phi(v), s_{v_u v}) | u \in U, v \in V_{\setminus v_u}^K\}, \quad (3)$$

with $|D_-| = K|D_+|$.

Finally, the augmented data set D_{aug} is the union of D_+ and D_- :

$$D_{aug} = D_+ \cup D_- \quad (4)$$

D_{aug} functions as the input for the TopoLedgerBERT model, as shown in Figure 2. The dataset augmentation method is visualized in Figure 3.

5 Empirical Evaluation

This section provides a thorough empirical evaluation of TopoLedgerBERT’s effectiveness in mapping ledger account descriptions to a standardized taxonomy, based on experiments using real-world financial data.

5.1 Data Description

The TopoLedgerBERT model uses a confidential dataset from Silverfin¹, an accountancy cloud service provider. This dataset comprises financial statement data from 5,608 UK companies from various industrial sectors and market sizes. Each company has mapped their original ledger account descriptions to the corresponding ledger accounts of one of six possible COAs, resulting in 521,265 unique mappings. The dataset is split into a 90%-10% training and testing set.

For each unique COA in the dataset, a vertex-labeled tree $T_{config} =$

¹<https://www.silverfin.com>

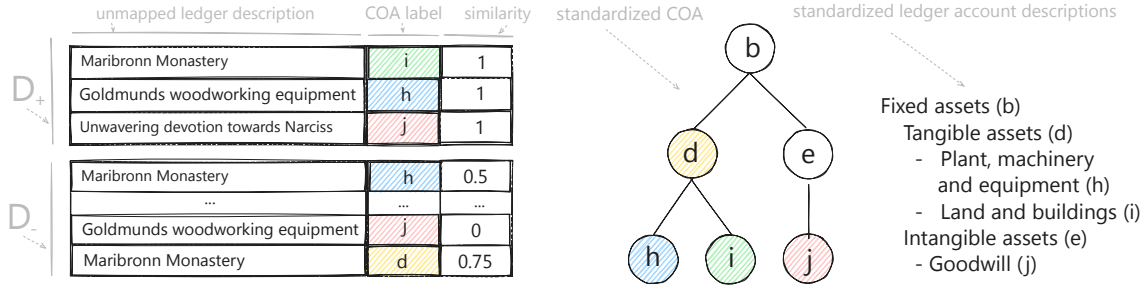


Figure 3: Example construction of an augmented dataset, D_{aug} , by TopoLedgerBERT for ledger account mapping.

$(V_{config}, E_{config}, L_{config}, \phi_{config})$ with $config \in \{1, 2, 3, 4, 5, 6\}$ was constructed. Each chart has $\{282467, 151524, 40183, 28573, 10311, 8207\}$ mappings, respectively.

5.2 Methods Evaluated

To assess the performance of our proposed TopoLedgerBERT model, we benchmark it against several established models. The following methods are considered:

- Standard SBERT: Utilizes the sentence-transformer model (*all-MiniLM-L6-v2*²) pre-trained on over 1B English sentence pairs (Reimers and Gurevych, 2019).
- Fine-tuned SBERT: An improved version of the Standard SBERT, incorporating multiple negatives ranking loss with in-batch negatives. The model’s training hyperparameters inspired by Decorte et al. (2023) are found in Appendix A.
- FinBERT: A BERT model variant specialized for financial texts (Araci, 2019). FinBERT is not utilized to generate sentence embeddings but rather to understand the financial context at the token level. A mean pooling strategy is used to derive sentence representations.
- TopoLedgerBERT: Our proposed model. It starts from the same sentence-transformer model (*all-MiniLM-L6-v2*) and employs a cosine similarity loss function that acknowledges hierarchical distances between different ledger accounts. Training hyperparameters can be found in the Appendix A.

5.3 Performance Metrics

In assessing the effectiveness of the proposed TopoLedgerBERT model, we employ four performance metrics: accuracy (Acc), mean reciprocal

²<https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>

rank (MRR), and two novel measures, the Mean Misprediction Distance (MMD) and the Mean Overall Distance (MOD).

The MMD quantifies the average hierarchical distance between the predicted and the actual label for mispredicted instances. A lower MMD implies that, although the prediction is incorrect, the predicted label is still relatively close to the true label in terms of the hierarchical structure of the COA. This metric provides a comprehensive overview of the model’s performance in preserving the hierarchical relationships in the COA.

Let $MD(v_i, v_j)$ be the distance of misprediction between the predicted vertex v_i and the true vertex v_j , which is defined as the length of the shortest path between v_i and v_j in the COA tree T . If the prediction is correct, i.e., $v_i = v_j$, then $MD(v_i, v_j) = 0$. Otherwise, $MD(v_i, v_j) > 0$.

The MMD is then defined as the average misprediction distance over all mispredicted instances in the test set, calculated as follows:

$$MMD = \frac{\sum_{v_i \neq v_j} MD(v_i, v_j)}{|\{(v_i, v_j) | v_i \neq v_j\}|} \quad (5)$$

where $\{(v_i, v_j) | v_i \neq v_j\}$ is the set of mispredicted instances.

Similarly, the Mean Overall Distance (MOD) can be defined, which also takes into account correctly predicted instances:

$$MOD = \frac{\sum_i MD(v_i, v_j)}{|\{(v_i, v_j)\}|}, \quad (6)$$

where $\{(v_i, v_j)\}$ is the set of all instances.

5.4 Experiment 1: Performance Evaluation of TopoLedgerBERT and Impact of Negative Sampling

In this experiment, we aim to investigate two main research questions:

1. How does the TopoLedgerBERT model perform in comparison to other established methods?
2. Does increasing the number of negative samples in the augmented dataset D_{aug} impact the performance of the TopoLedgerBERT model?

For the second question, we experiment with different values of K , where K represents the number of extra negative samples generated per positive sample in D_+ . The values we consider for K are $\{5, 10, 15, 20\}$. We denote the different TopoLedgerBERT models trained on augmented datasets with different negative sample sizes by TopoLedgerBERT@ K .

Metric	Acc	MRR
FinBERT	28.91	34.57
Standard SBERT	44.79	53.07
Fine-tuned SBERT	64.15	72.69
TopoLedgerBERT@5	65.30	72.62
TopoLedgerBERT@10	66.13	73.23
TopoLedgerBERT@15	66.80	73.73
TopoLedgerBERT@20	67.01	73.78

Table 1: Comparison of the performance of TopoLedgerBERT and benchmark models. Models are evaluated based on Acc and MRR. The best performing model for each metric is in **bold**.

The results of the empirical evaluation, summarized in Table 1, validate the superior performance of our proposed TopoLedgerBERT model over the benchmark methods in terms of accuracy and mean reciprocal rank.

The FinBERT model, despite being trained on financial data, does not perform as well as the other models. This is likely due to its token-level focus, which does not lend itself to generating meaningful sentence representations. On the other hand, the Standard SBERT and Fine-tuned SBERT models, which are sentence-transformer models, perform significantly better. This highlights the importance of sentence representation methods in the context of ledger account mapping.

When comparing the TopoLedgerBERT@ K models with different values of K , we observe a general trend of increasing performance with increasing K . The best performance is seen at $K = 20$, where the model achieves an accuracy of 67.01% and a MRR score of 73.78%. This suggests that augmenting the training data with more diverse negative samples containing information about the

ledger account taxonomy can effectively enhance the model performance. Learning from negative samples helps the model better distinguish between different ledger accounts and enhance its mapping abilities.

However, the impact of a further increase in K on the model’s performance is not straightforward. Excessive negative samples could potentially dilute the impact of the positive samples. Further investigations could be done by testing a wider range of K values.

5.5 Experiment 2: Hierarchical Relationship Understanding of TopoLedgerBERT

In this experiment, we evaluate the ability of the TopoLedgerBERT model to capture hierarchical relationships between different ledger accounts. We compare it with the Fine-tuned SBERT model to determine its effectiveness. The comparison is based on two metrics: MOD and MMD.

Metric	MOD	MMD
Fine-tuned SBERT	1.44	4.41
TopoLedgerBERT@20	1.02	3.90

Table 2: Comparison of MOD and MMD for Fine-tuned SBERT and TopoLedgerBERT@20. The best results are **bold**.

As shown in Table 2, the TopoLedgerBERT@20 model outperforms the Fine-tuned SBERT model in terms of both MOD and MMD. This suggests that the TopoLedgerBERT model is more effective at capturing hierarchical relationships in the COA and accurately mapping ledger accounts, thus leading to semantically better predictions.

The MOD value for the TopoLedgerBERT@20 model is significantly lower than the value for the Fine-tuned SBERT model. This indicates that the TopoLedgerBERT@20 model’s predictions are generally closer to the true labels in the COA, including both correct and incorrect predictions.

Regarding the MMD value, the TopoLedgerBERT@20 model achieves a lower value. This suggests a more effective use of hierarchical information by the TopoLedgerBERT@20 model, which leads to more accurate predictions, even for mispredicted samples.

Subsequently, we analyze the performance of TopoLedgerBERT@20 by comparing its misprediction distance distribution with that of Fine-tuned SBERT. This analysis provides additional insights

into the model’s ability to capture and leverage the hierarchical information from the COA.

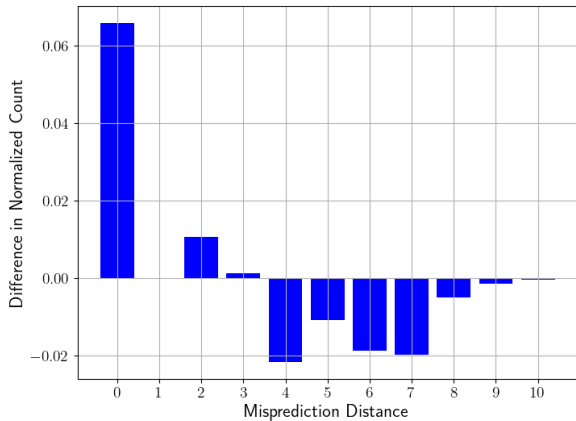


Figure 4: Misprediction Distance (MD) difference distribution between TopoLedgerBERT@20 model and Fine-tuned SBERT model.

The plot, as shown in Figure 4, shows the difference in the distribution of the MD between the TopoLedgerBERT@20 and the Fine-tuned SBERT models. A positive difference indicates that the TopoLedgerBERT@20 model has more predictions at that specific distance, while a negative difference suggests the opposite. Looking at the misprediction distance distribution difference plot, it is clear that the TopoLedgerBERT@20 model outperforms the Fine-tuned SBERT model in terms of both the count of correct predictions and the distribution of mispredictions.

TopoLedgerBERT@20 outperforms Fine-tuned SBERT by around 6% in correct predictions. While both models show a similar pattern in one-edge-away predictions, reflecting a preference for specificity, TopoLedgerBERT@20 has more two-edges-away mispredictions. This implies that TopoLedgerBERT@20 leans towards making close-by predictions.

TopoLedgerBERT@20 significantly performs better than Fine-tuned SBERT for predictions that are four edges away or more. This implies that the TopoLedgerBERT@20 model not only increases the count of correct predictions but also reduces the number of far-off predictions, hence preserving the hierarchical relationships in the COA better.

5.6 Discussion

The empirical evaluation shows that TopoLedgerBERT outperforms the benchmark models in terms of accuracy and mean reciprocal rank, validating its effectiveness in mapping custom ledger account

descriptions to a standardized taxonomy. The increased performance of the TopoLedgerBERT model is attributed to its innovative approach of incorporating hierarchical information from the COA into the training process and employing data augmentation strategies.

Additionally, the TopoLedgerBERT model performs better in capturing hierarchical relationships between different ledger accounts. The lower MOD and MMD scores indicate that the model can make more accurate predictions that are closer to the true labels in the COA.

The results confirm that the TopoLedgerBERT model shows a significant advance toward the ledger account mapping problem. Future research could explore further enhancements to the model, such as incorporating additional contextual information or exploring more complex data augmentation strategies.

6 Conclusion

In this paper, we address the significant problem of mapping custom ledger accounts to a standardized COA, a task that is crucial to improving the comparability of financial data between companies. We propose a unique method, TopoLedgerBERT, that leverages hierarchical information in the COA and fine-tuned sentence embeddings to create a mapping solution. Our empirical evaluation shows that TopoLedgerBERT not only improves the mapping accuracy, but also effectively captures the hierarchical relationships between different ledger accounts. Furthermore, the TopoLedgerBERT model shows its effectiveness in addressing the cold-start problem, generally seen when a system must predict for ledger accounts absent from the training dataset. Overcoming this issue not only solves the "multiple charts of accounts problem", as the model is not bound by a predefined set of ledger accounts or a COA, but it also paves the way for more streamlined and efficient automation of COA standardization. The empirical evaluation provides valuable insights into the proposed model’s potential benefits and areas for further improvement, ultimately contributing to improving the efficiency and precision of the accounting domain.

In the future, this work could be extended in several directions. Firstly, our proposed method can be applied to various sectors where target labels are hierarchically structured. Secondly, the incorporation of additional hierarchical informa-

tion or features may enhance the performance of the model. In addition, a detailed analysis could be performed to understand to what extent the effectiveness of our method depends on the quality and consistency of the COAs used. Third, a test of our proposed method across a larger, more diverse range of industries or countries may provide deeper insights into the model’s generalizability and robustness. Fourth, a more advanced strategy for selecting negative samples could supplement the current approach in generating more representative training samples. Lastly, incorporating evaluation metrics from the accounting domain can provide additional validation of the model’s utility in accounting applications.

Acknowledgments

This research received funding from the Flemish government, through Flanders Innovation & Entrepreneurship (VLAIO, project HBC.2020.2883) and from the Flemish government under the program “Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen”. We also acknowledge the invaluable feedback provided by Simon De Ridder, Karen Dedecker, and Pieter De Koninck of the AI Team at Silverfin.

References

- Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.
- Johan Bergdorf. 2018. Machine learning and rule induction in invoice processing: Comparing machine learning methods in their ability to assign account codes in the bookkeeping process.
- European Commission. 2008. Final report of the expert group: Accounting systems for small enterprises—recommendations and good practices. Luxembourg: European Commission.
- Jens-Joris Decorte, Severine Verlinden, Jeroen Van Haute, Johannes Deleu, Chris Develder, and Thomas Demeester. 2023. Extreme multi-label skill extraction training using large language models. *arXiv preprint arXiv:2307.10778*.
- Sandip Dhole, Gerald J Lobo, Sagarika Mishra, and Ananda M Pal. 2015. Effects of the sec’s xbrl mandate on financial reporting comparability. *International Journal of Accounting Information Systems*, 19:29–44.
- EUROSTAT. 2017. Epsas issue paper on the national approaches to harmonisation of chart of accounts. Working paper no. epsas wg 17/12, European Commission, Luxembourg. Paper by PwC on behalf of Eurostat.
- Susana Jorge, Giovanna Dabbicco, Caroline Aggestam-Pontoppidan, and Diana Vaz de Lima. 2022. New development: The development of standardized charts of accounts in public sector accounting. *Public Money & Management*, 42(7):530–533.
- Rasmus Kær Jørgensen and Christian Igel. 2021. Machine learning for financial transaction classification across companies using character-level word embeddings of text fields. *Intelligent Systems in Accounting, Finance and Management*, 28(3):159–172.
- Juan Liu, Lei Pei, Ying Sun, Heather Simpson, Jocelyn Lu, and Nhung Ho. 2021. Categorization of financial transactions in quickbooks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3299–3307.
- Justin Munoz, Mahdi Jalili, and Laleh Tafakori. 2022. Hierarchical classification for account code suggestion. *Knowledge-Based Systems*, 251:109302.
- Sander Noels, Simon De Ridder, Sébastien Viaene, and Tijl De Bie. 2023a. An efficient graph-based peer selection method for financial statements. *Intelligent Systems in Accounting, Finance and Management*, 30(3):120–136.
- Sander Noels, Adriaan Merlevede, Andrew Fechey, Maarten Vanhalst, Nick Meerlaen, Sébastien Viaene, and Tijl De Bie. 2023b. Automated financial analysis using gpt-4. In *Machine Learning and Knowledge Discovery in Databases: Applied Data Science and Demo Track*, pages 345–349, Cham. Springer Nature Switzerland.
- Sander Noels, Benjamin Vandermarliere, Ken Bastiaensen, and Tijl De Bie. 2022. An earth mover’s distance based graph distance metric for financial statements. In *2022 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFER)*, pages 1–8. IEEE.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.
- Carol D Rylee. 2017. So, you need a new chart of accounts. *Planning for Higher Education*, 45(3):60.
- Richard Wang. 2023. Standardizing xbrl financial reporting tags with natural language processing. *Available at SSRN 4613085*.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kam-badur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Steve Yang and Randy Cogill. 2013. Balance sheet outlier detection using a graph similarity algorithm. In *2013 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFER)*, pages 135–142. IEEE.

A Details of the Training Process

The strategies for learning sentence representations were executed using the widely used S-BERT³ implementation (Reimers and Gurevych, 2019). In the multiple negative ranking loss, the 'scale' hyperparameter was kept at the default value of 20. The training was always done for 1 epoch. Positive pairs were mixed randomly into groups of 64. The AdamW optimizer was used as the default optimizer, with a learning rate of 2e-5, and a 'WarmupLinear' learning rate schedule that includes a warm-up period covering 5% of the training data. Automatic mixed precision was employed to make the training process faster. We utilized mean pooling as the pooling technique.

³https://www.sbert.net/docs/package_reference/losses.html

Probing Numerical Concepts in Financial Text with BERT Models

Shanyue Guo, Le Qiu and Emmanuele Chersoni

Department of Chinese and Bilingual Studies

The Hong Kong Polytechnic University

Yuk Choi Road 11, Kowloon, Hong Kong, China

{shanyue.guo, emmanuele.chersoni}@polyu.edu.hk, lani.qiu@connect.polyu.hk

Abstract

Numbers are notoriously an essential component of financial texts, and their correct understanding is key to automatic system for efficiently extracting and processing information.

In our paper, we analyze the embeddings of different BERT-based models, by testing them on supervised and unsupervised probing tasks for financial numeral understanding and value ordering.

Our results show that LMs with different types of training have complementary strengths, thus suggesting that their embeddings should be combined for more stable performances across tasks and categories.

1 Introduction

The analysis of the linguistic and conceptual knowledge contained in the representations of Transformer architectures (Vaswani et al., 2017) has become of general interest since the introduction of pre-trained language models (LMs) (Radford et al., 2019; Devlin et al., 2019). A common paradigm for testing such knowledge is represented by *probing tasks*: a simple classification model takes as input a representation of a word/sentence from a pre-trained LMs (i.e. an *embedding*) and it is asked to solve a task involving human linguistic knowledge (e.g. subject-verb number agreement, coreference resolution etc.), and a good performance is considered as an indicator that the LM encodes the target knowledge (see Belinkov (2022) for an overview).

Understanding numbers is even more essential for the analysis of financial texts, where they may denote different types of concepts (e.g. amounts, percentages, time periods etc.), each one with its own scale of values. Ideally, a model should be able to estimate the correct magnitude of a numeral for a category, and carry out comparisons between category members. Probing provides important

insights about which models contain more information about a specific linguistic distinction, because it analyzes their performance in the most simple and controlled settings (Adi et al., 2017; Conneau et al., 2018; Chersoni et al., 2021), and therefore it can guide the choice on the base models for more sophisticated NLP pipelines and downstream tasks.

In our paper, we analyze the embeddings from three different BERTs with probing tasks for numeracy in the financial domain. Such tasks are meant to test numerical understanding, seen as the capacity of interpreting a numerical expression and assigning it to a specific conceptual category, and the capacity of ordering the values of each category on a scale. The BERT models were selected to assess the effect of different types of pretraining in handling numeracy: is a general numeracy-augmented pretraining sufficient to learn knowledge about numerals in the financial domain? Or the exposure to financial text is necessary for capturing the nuances of the meaning of numerals in this domain?

We show that the models perform similarly in a supervised probing task, where the LM embeddings are used to train a classifier. On the other hand, when tested with unsupervised tasks, more differences emerge: although the embeddings of MWP-BERT show more consistency in identifying numeral categories and ordering numeral values, there is not a single model doing consistently better in all categories and tasks. This might suggest the opportunity of combining different LM representations to achieve more stable performance in financial tasks.¹

2 Related Work

With the rising popularity of Natural Language Processing and text mining for finance (Loughran and McDonald, 2016), researchers quickly adapted pre-

¹Code and data available at: <https://anonymfile.com/KV10e/code-submission.zip>

Category	Count	Percentage	Sample Instance
Monetary	2646	37.34	\$CY don't let it close below 14.77
Temporal	2062	29.1	Alert sent to members at 9:59 AM
Percentage	1060	14.96	past Ecommerce sales up 50%
Quantity	843	11.9	\$MU Interesting that about 15k shares
Indicator	198	2.79	The close over the 200 dma on heavy volume
Option	158	2.23	\$ISRG bought weekly 387.50 puts
Product Number	120	1.69	a partnership announcement combined with sd- 101

Table 1: Descriptive statistics and sample instances of FinNum-1 (target numeral in **bold**).

trained LMs to the financial domain, mostly starting from general-domain architectures (e.g. BERT, Devlin et al. (2019)) and then carrying out additional training on a corpus from the financial domain (e.g. the FinBERT models, Araci (2019); Yang et al. (2020)). The domain adaptation process led to performance improvements, although most evaluations focused on sentiment analysis and related tasks, and improvements were not always consistent (Peng et al., 2021).

Given the interest of the NLP field in numerical understanding, several works focused on improving the mathematical reasoning capacities of LMs (Geva et al., 2020; Thawani et al., 2021; Chen et al., 2022; Petrak et al., 2023). However, despite recent progress, LMs seem to be still struggling with numerals, especially if rare/unseen in the training data (Wallace et al., 2019; Sharma et al., 2024).

In our study, we are interested in seeing whether i) BERT embeddings of numerals can be used to assign them to the right conceptual category, or superordinate class (Chen et al., 2018); ii) the information they contain can be used to infer their values by putting them on a category-specific scale.

3 Experimental Settings

3.1 Dataset

The experiments are conducted on the FinNum-1 dataset (Chen et al., 2018), which was introduced for a shared task in numerical understanding, and consists of numeral expressions in financial tweets categorized in 7 classes: "Monetary", "Percentage", "Option", "Indicator", "Temporal", "Quantity", and "Product Number". Given a numeral in context, a model has to assign it to the right conceptual class. Descriptive statistics and sample instances from FinNum-1 can be seen in Table 1.

The representation of each numeral in the dataset is extracted from the last layer of a LM, resulting in an embedding representation of size 768.

3.2 Models

We focused on BERT-based models to assess the impact on number representations of specific types of training on the same architecture. Other reasons are that such models are relatively lightweight and faster to run, and that they are *bidirectional*, therefore they represent a better choice than autoregressive models for extracting contextualized representations, which is what our tasks require. Recent literature proved that the fact that autoregressive LMs cannot see future tokens represents a drawback for the representation quality of their embeddings (Springer et al., 2024).

The first model that we use is the standard **BERT** model by Devlin et al. (2019). The second one is a domain-adapted version of BERT, **FinBERT** (Yang et al., 2020), which was initialized from a BERT Base checkpoint and then further pretrained on financial corpora. Finally, we include the **MWP-BERT** model by Liang et al. (2022), which incorporate several numeracy grounded pre-training objectives and has been proved to improve the quality of number representations in several mathematical reasoning tasks. We chose these specific models (all in their Base version) because they exemplify different types of additional training of the same architecture, allowing us to ask to what extent training on financial corpora and numeracy injection improve number representations in financial text.

3.3 Probing Tasks

Numeral understanding. The purpose of the supervised numeral understanding task is to investigate whether the embedding representation of numerical data by LMs exhibits discernible variations across distinct numerical categories. This experiment is conducted by assessing the capability of a simple linear classifier (logistic regression, in our case) to identify the correct category of numerals using the embedding dimensions as input features.

We divide the dataset instances in 7 subsets, one for each category, and in each category subset, we

add negative examples in a 1:1 ratio by randomly sampling instances from the other 6 categories. Subsequently, we employ our probing classifier and use stratified 10-fold cross-validation to obtain the average performance for each category subset. We assess performance in terms of the standard metrics of *accuracy*, *precision*, *recall*, and *F1-macro*.²

This first task is supervised, as a simple classifier is trained on a linguistic distinction (the semantic category of the number) on the basis of the LM representations (the token embeddings of the number). However, such methodology has been criticised as it involves an external classifier, and thus the relation between the performance and the knowledge already in the embeddings is not clear (Levy et al., 2023). To probe more directly the extent to which number properties exist in the embedding spaces, we adopt two additional unsupervised tasks.

Number outlier detection. Outlier detection (Camacho-Collados and Navigli, 2016) relies on the hypothesis that embeddings of the same classes form coherent clusters in the vector space. In this task, we introduce an *outlier* in clusters of numerals that belong to the same category, and then measure the reciprocal similarities between all the numerals in each cluster. If a LM has an accurate representation of a category, the outlier should be the one with the lowest similarity to the other cluster members.

We generated from FinNum-1 an evaluation dataset for outlier detection with 6966 clusters of 8 instances (7 instances of a class + 1 outlier). For space reasons, the details of the process are provided in Appendix A. The performance is evaluated based on detection *Accuracy* and *Outlier Position Percentage (OPP)*.

For each cluster, cosine similarity is computed between each number embeddings pair. Detection accuracy is determined by sorting the numbers according to the average similarities to other cluster members. If the embedding with the smallest average similarity is the one of the outlier, the model gets a hit, and accuracy is given by the number of hits divided by the number of clusters. OPP is computed instead with the following equation:

$$OPP = \frac{\sum_{S \in D} \frac{OP(S)}{|S|}}{|D|} \quad (1)$$

D represents the evaluation subset of a category. S represents a cluster in D . $OP(S)$ refers to the detected outlier position of this evaluation sample (the index in the sorted ranking, according to average cosine). $|S|$ refers to the number of embeddings in each cluster (8 in our case). Given that the cosine similarity is sorted in descending order, the OPP value can be seen as an indicator of how close the outlier is to the bottom of the ranking.

Value ordering. Finally, we want to check how accurate are the LMs in representing the *values* of the numerals within each category, in a relative ordering task. Recently, Grand et al. (2022) proposed a method based on *semantic projections* to interpret the content of word embeddings, by means of identifying vector subspaces that corresponding to human-interpretable semantic scales.

The method works as follows: i) identify words that can represent extreme values of a target on a scale, e.g. for SIZE words like *big*, *huge*, *gigantic* on one extreme, and *tiny*, *small*, *minuscule* on the other extreme; ii) average the corresponding embeddings at the two extremes to obtain a "prototype" of an extreme value for that scale (the concepts of 'very small' and 'very big'), and then connect the averaged vectors with a line; this line was used to represent the scale of human measurements of SIZE; iii) given a list of words/concepts to be ordered by their SIZE, project their embeddings onto the SIZE line and take the relative ordering of their values. Here, we adopt the same method to map number embeddings onto their values, and test if they can be ordered from the smallest to the biggest one in their respective category.

First, we identify the vectors corresponding to the maximum and minimum numbers within each category and we subtract them to obtain a scale vector of *value*. Then, we have to calculate the projection of each remaining number in the category onto *value*, defined via the classic scalar projection formula:

$$Proj(\overrightarrow{number}) = \frac{\overrightarrow{number} \cdot \overrightarrow{value}}{\|\overrightarrow{value}\|} \quad (2)$$

For each number in the dataset, we sort both the numbers and their corresponding representations based on the numerical value (the embeddings of

²Differently from Chen et al. (2018), here the task is simplified for the probing classifier: instead of a 7-way classification, we have 7 classification models that work in a one vs. all setting. Since models only have to make binary choices on whether instances belong to a class or not, we can expect similar or higher performance compared to Chen et al. (2018).

Model	Acc	Prec	Rec	F1
BERT	0.92	0.92	0.92	0.92
FinBERT	0.92	0.93	0.92	0.92
MWP-BERT	0.88	0.88	0.88	0.88

Table 2: Metrics for the numeral understanding task (averaged by class and rounded to the second decimal).

duplicated numbers are simply averaged). Subsequently, we compute a scale vector by subtracting the numerical representation of the smallest number from that of the largest number within the category. Finally, we calculate the projection of all the numbers in the category onto the *value* vector.

Performance is assessed using two different metrics: *pairwise ordering accuracy* and *Spearman correlation*. In the former, for a category of n instances, we generate n^2 evaluation pairs by pairing each instance with all the other ones in the same category and compare their numerical values, assigning a hit to a LM for every time it correctly picks the example with higher numerical value in a pair; in the latter, we measure the Spearman ρ between the order of the actual values of the numbers in the gold standard and the values obtained via projection of their embeddings.

4 Results

The scores for the numeral understanding task can be seen in Table 2. At a glance, it can be seen that all the models achieve a very high performance, with only MWP-BERT being slightly below 90% for all the evaluation metrics. All the model representations clearly contain relevant information for the identification of the right semantic class of a numeral expression, to the point that a performance around 90% can be obtained even with a linear classifier. More detailed, by-class figures can be found in Appendix B: unsurprisingly, Percentage is the easiest class for all models, probably because the presence in almost all contexts of the percentage sign provides a strong identification cue.³

Moving to the unsupervised tasks, we can observe in Table 3 that BERT Base is the best one for Accuracy in identifying the outlier, and MWP-

³Reviewer 3 requests us to report the results of the original FinNum-1 shared task for comparison. The highest F1-Macro that was reported in the FinNum-1 shared task was around 0.90, achieved by Fortia-1 with a convolutional neural network combining different types of word embeddings (word level, character level, ELMo etc.) (Azzi and Bouamor, 2019). However, given that they were operating in an actual multiclass classification setting while we adopted a one vs. all approach, we do not think the scores are directly comparable.

Category	BERT Base	FinBERT	MWP-BERT
Indicator	0.68/0.92	0.63/0.90	0.56/0.89
Monetary	0.38/0.75	0.39/0.77	0.40/0.78
Option	0.61/0.87	0.49/0.84	0.57/0.87
Percentage	0.63/0.88	0.51/0.85	0.67/0.92
Product			
Number	0.37/0.76	0.43/0.79	0.20/0.74
Quantity	0.34/0.71	0.31/0.72	0.26/0.69
Temporal	0.24/0.68	0.20/0.62	0.35/0.77
AVERAGE	0.46/0.79	0.42/0.78	0.43/0.81

Table 3: Outlier detection results for the metrics of Accuracy/OPP (best scores in **bold**).

BERT has an edge for the OPP metric. MWP-BERT does in general a better job in clustering a higher number of instances, as it gets the top scores for Monetary, Temporal and Percentage, the most frequent categories (they combine for more than 81% of the data points).

Category	BERT Base	FinBERT	MWP-BERT
Indicator	0.60/0.43	0.73/0.61	0.68/0.55
Monetary	0.40/0.60	0.46/0.73	0.41/0.28
Option	0.61/0.46	0.53/0.47	0.49/0.48
Percentage	0.37/0.34	0.42/0.21	0.46/0.58
Product Number	0.69/0.37	0.75/0.36	0.65/0.50
Quantity	0.71/0.38	0.63/0.38	0.59/0.51
Temporal	0.58/0.35	0.55/0.31	0.50/0.52
AVERAGE	0.57/0.42	0.58/0.44	0.54/0.49

Table 4: Pairwise accuracy/Spearman scores on the value ordering task (best scores in **bold**).

Table 4, showing the scores of Pairwise ordering accuracy, displays almost a tie across categories between the LMs: MWP-BERT is the best model for Percentage; FinBERT does better in the most finance-specific categories (Indicator and Monetary) and in Product Number; the Base model is best for the remaining ones. All models display moderate correlations with the actual number values, with MWP-BERT being significantly better than both BERT and FinBERT ($p < 0.01$ for a two-tailed test with the Fisher r-to-z transformation). This suggests that the numeracy-augmented model is the best choice for handling value ordering. However, it also has a sharp drop on Monetary, the most "finance-specific" and frequent category.

In general, a trend of unsupervised tasks is that representations of different models do well in different categories, which suggests that combining them might lead to a more stable performance. To quickly test this hypothesis, we tried to repeat all the above experiments by combining the FinBERT and the MWP-BERT embeddings, using the simple methods of averaging and concatenation. While

averaging led to inconsistent results, we observed a slight increase of performance for embeddings concatenation in numeral understanding, with the F1-score going up to 0.93, and outlier detection, with 0.46 of Accuracy and 0.82 of OPP. The results are more ambivalent for value ordering: pairwise ordering accuracy goes down to 0.49, but the combined vectors achieve the highest Spearman correlation with the gold standard value with 0.54. This also includes a much higher correlation with the Monetary category, going up to 0.43 from 0.28.

We think this is good preliminary evidence of possible improvements by combining the information in the two vector types, and it is likely that larger improvements could be achieved by adding trainable layers on the top of the original embeddings representations.

5 Conclusion

In this work, we used to simple probing task to analyze the knowledge of financial numerals encoded in different types of BERT-based LMs, particularly in relation with the categories of the numerals in financial text and with the capacity of ordering their values on the scale proper of each category.

While with a supervised probe the numeracy-augmented MWP-BERT does worse, in unsupervised tasks it the representation quality across categories looks generally better. The fact that some models experience highs and lows in some categories might be related to limited exposure in the pretraining phase, which suggest that, in financial tasks, it might be wise to combine different types of embeddings to stabilize the representation and simultaneously account for different aspects of numerical knowledge.

Acknowledgements

The authors acknowledge the support from the project “Analyzing the semantics of Transformers representations for financial natural language processing”(ZVYU), sponsored by the Faculty of Humanities of the Hong Kong Polytechnic University.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks. In *Proceedings of ICLR*.

Dogu Araci. 2019. FinBERT: Financial Sentiment

Analysis with Pre-trained Language Models. *arXiv preprint arXiv:1908.10063*.

- Abderrahim Ait Azzi and Houada Bouamor. 2019. Fortia1@ the NTCIR-14 FinNum Task: Enriched Sequence Labeling for Numeral Classification. In *Proceedings of the NTCIR Conference on Evaluation of Information Access Technologies*.
- Yonatan Belinkov. 2022. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, 48(1):207–219.
- José Camacho-Collados and Roberto Navigli. 2016. Find the Word that Does Not Belong: A Framework for an Intrinsic Evaluation of Word Vector Representations. In *Proceedings of the ACL Workshop on Evaluating Vector-Space Representations for NLP*.
- Chung-Chi Chen, Hen-Hsen Huang, Yow-Ting Shiue, and Hsin-Hsi Chen. 2018. Numeral Understanding in Financial Tweets for Fine-grained Crowd-based Forecasting. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*.
- Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. 2022. Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. *arXiv preprint arXiv:2211.12588*.
- Emmanuele Chersoni, Enrico Santus, Chu-Ren Huang, and Alessandro Lenzi. 2021. Decoding Word Embeddings with Brain-based Semantic Features. *Computational Linguistics*, 47(3):663–698.
- Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What You Can Cram into a Single Vector: Probing Sentence Embeddings for Linguistic Properties. In *Proceedings of ACL*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*.
- Mor Geva, Ankit Gupta, and Jonathan Berant. 2020. Injecting Numerical Reasoning Skills into Language Models. In *Proceedings of ACL*.
- Gabriel Grand, Idan Asher Blank, Francisco Pereira, and Evelina Fedorenko. 2022. Semantic Projection Recovers Rich Human Knowledge of Multiple Object Features from Word Embeddings. *Nature Human Behaviour*, 6(7):975–987.
- Tal Levy, Omer Goldman, and Reut Tsarfaty. 2023. Is Probing All You Need? Indicator Tasks as an Alternative to Probing Embedding Spaces. In *Findings of EMNLP*.
- Zhenwen Liang, Jipeng Zhang, Lei Wang, Wei Qin, Yunshi Lan, Jie Shao, and Xiangliang Zhang. 2022. MWP-BERT: Numeracy-augmented Pre-training for Math Word Problem Solving. In *Findings of NAACL*.

Tim Loughran and Bill McDonald. 2016. Textual Analysis in Accounting and Finance: A Survey. *Journal of Accounting Research*, 54(4):1187–1230.

Bo Peng, Emmanuele Chersoni, Yu-Yin Hsu, and Churen Huang. 2021. Is Domain Adaptation Worth your Investment? Comparing BERT and FinBERT on Financial Tasks. In *Proceedings of the EMNLP Workshop on Economics and Natural Language Processing*.

Dominic Petrak, Nafise Sadat Moosavi, and Iryna Gurevych. 2023. Arithmetic-Based Pretraining Improving Numeracy of Pretrained Language Models. In *Proceedings of *SEM*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models Are Unsupervised Multitask Learners. *OpenAI Blog*, 1(8):9.

Mandar Sharma, Rutuja Murlidhar Taware, Pravesh Koirala, Nikhil Muralidhar, and Naren Ramakrishnan. 2024. Laying Anchors: Semantically Priming Numerals in Language Modeling. In *Findings of NAACL*.

Jacob Mitchell Springer, Suhas Kotha, Daniel Fried, Graham Neubig, and Aditi Raghunathan. 2024. Repetition Improves Language Model Embeddings. *arXiv preprint arXiv:2402.15449*.

Avijit Thawani, Jay Pujara, and Filip Ilievski. 2021. Numeracy Enhances the Literacy of Language Models. In *Proceedings of EMNLP*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems*.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP Models Know Numbers? Probing Numeracy in Embeddings. In *Proceedings of EMNLP*.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. FinBERT: A Pretrained Language Model for Financial Communications. *arXiv preprint arXiv:2006.08097*.

A Outlier Detection: Dataset Construction

To construct an evaluation dataset for outlier detection, we created category clusters by following the steps in [Camacho-Collados and Navigli \(2016\)](#). We grouped 7 FinNum-1 instances from one class and then we randomly introduced one instance from another category as an outlier. To maximize the utilization of data, we employed a sliding window of size 7 through the list of the instances of each one of the 7 categories to create the clusters.

Following the segmentation of each category subset, we appended 6 data instances from different categories to each generated cluster (one outlier is sampled from each one of the other categories). Through this process, for every set of 7 data samples from each category, 6 distinct datasets containing outliers from different classes were generated. In total, the number of clusters is 6966.

B Full Scores for the Numeral Understanding Task

The specific scores for each system, broken down by class, can be seen in Tables 5, 6 and 7.

Category	BERT			
	Acc	Prec	Rec	F1
Indicator	0.91	0.92	0.91	0.91
Monetary	0.89	0.89	0.89	0.89
Option	0.92	0.92	0.92	0.92
Percentage	0.98	0.98	0.98	0.98
Product Number	0.91	0.91	0.91	0.91
Quantity	0.89	0.89	0.89	0.89
Temporal	0.92	0.92	0.92	0.92
AVERAGE	0.92	0.92	0.92	0.92

Table 5: Probing classifier results with BERT Base.

Category	FinBERT			
	Acc	Prec	Rec	F1
Indicator	0.93	0.93	0.93	0.93
Monetary	0.92	0.92	0.92	0.92
Option	0.87	0.88	0.87	0.87
Percentage	0.98	0.98	0.98	0.98
Product Number	0.93	0.93	0.93	0.93
Quantity	0.90	0.90	0.90	0.90
Temporal	0.94	0.94	0.94	0.94
AVERAGE	0.92	0.93	0.92	0.92

Table 6: Probing classifier results with FinBERT.

Category	MWP-BERT			
	Acc	Prec	Rec	F1
Indicator	0.91	0.91	0.91	0.91
Monetary	0.86	0.86	0.86	0.86
Option	0.89	0.89	0.89	0.88
Percentage	0.98	0.98	0.98	0.98
Product Number	0.81	0.82	0.81	0.81
Quantity	0.83	0.83	0.83	0.83
Temporal	0.89	0.89	0.89	0.89
AVERAGE	0.88	0.88	0.88	0.88

Table 7: Probing classifier results with MWP-BERT.

GENWISE: Thematic Discovery from Textual Data

Minnie Kabra* **Abhinav Nagpal*** **Aayush Sacheti*†**
American Express AI Labs American Express AI Labs Asato.ai
minnie.kabra@aexp.com abhinav.nagpal1@aexp.com sachetiaayush@gmail.com

Mohit Kumar
American Express AI Labs
mohit.kumar30@aexp.com

Salil Rajeev Joshi
American Express AI Labs
salilrajeev.joshi@aexp.com

Abstract

In this work, we introduce GENWISE - a generative AI-based framework designed to streamline extracting and organizing key information from textual data. Focusing on the prevalent issue in business where significant time is spent on manual data analysis, our framework employs cutting-edge generative AI, embedding, and clustering techniques towards a thematic discovery. We further deliver hierarchical thematic representations, enhancing the ease of understanding for users at different levels. Our methodology includes precise issue extraction through generative AI, utilization of the Retrieval-Augmented Generation framework for improved accuracy, and a 20% improvement in cluster coherency using the Enhanced Community Detection algorithm. This comprehensive pipeline is optimized explicitly for industrial settings, offering a significant leap in efficiency and thematic representation for complex data sets.

1 Introduction

Banks and other financial institutions have been accumulating unstructured data for decades, including customer complaints, emails, chats, and call transcripts. Despite sophisticated processes to organize this data, about 80% of the analysis¹ remains descriptive. The sheer volume and complexity of this unstructured data pose significant challenges when combined with structured data, which is heavily relied upon by financial institutions to gain a comprehensive understanding of their customers. Business teams spend hundreds of hours each month reading and summarizing this data based on customer interactions to extract actionable themes. Identifying key information and

grouping it semantically is both time-consuming and laborious. For example, sample records from a finance industry dataset mentioned in Table 1 contain several key segments, each as highlighted. This problem is compounded when these records are clustered based on these segments, represented at varying levels of detail.

In this paper, we address the challenge of thematic discovery from textual data using AI and ML techniques. We utilize cutting-edge generative AI, embedding techniques, and clustering methods to automatically identify key segments, transform them into a semantic format, and organize them hierarchically to ease the cognitive burden involved in the process.

The problem of thematic discovery has drawn academic attention in the past. Approaches ranging from simple rule-based extraction to statistical topic modelling and, more recently, neural techniques have been explored. However, we observed that these techniques had shortfalls in industrial settings. Bertopic's (Grootendorst, 2022) document representation is inadequate, or if it is adequate, like in TopicGPT (Pham et al., 2023), the algorithm is not suitable for industrial applications due to its high execution time in an online setting. Towards this end, we have curated an end-to-end system to discover themes from textual data in an unsupervised manner. The novelty of our system draws from the optimal use of underlying components and precise outcomes not feasible through existing systems.

Specifically, we make the following contributions:

- **Intuitive Representation:** The generated cluster representation presents the themes hierarchically at various levels of granularity, allowing senior leaders and business analysts to gain actionable themes.
- **Extracting important and distinct issues from**

*Equal contribution

†Work done as part of American Express AI Labs

¹<https://www.informationweek.com/machine-learning-ai/big-data-analytics-descriptive-vs-predictive-vs-prescriptive>

Table 1: Sample CPF B Complaints, key information is highlighted

Abridged CPF B Note text
(ID 7317133) I filed a dispute for incorrect information on my credit report. I received an email from the credit bureau stating that they are assuming the disputes are coming from a 3rd party. They in fact did not come from a 3rd party. I even called them to verify it was me and they still refused to process my disputes . I wasted money on mailing my disputes out...
(ID 7317093) I am writing to dispute the accuracy of the information on my credit reports provided by XXXX, Experian, and XXXX. After reviewing the reports, I have identified several inconsistencies that I believe require immediate attention and correction . I kindly request that you investigate and rectify the inaccuracies in accordance with the FCRA...

the text: We use generative AI models with precise prompts to identify the key segments present in the text. The textual segments are further aggregated in a semantic space using state-of-the-art embedding techniques.

- **RAG (Retrieval-Augmented Generation) framework:** We use RAG to ingest industrial domain knowledge to reduce hallucination from generative models and make the segment extraction more precise.
- **Community detection algorithm enhancements:** After testing several non-parametric clustering approaches, we selected a community detection algorithm and further improved its cluster coherency by 20% on average.

The remainder of the paper is organized as follows: Section 2 presents a brief history of existing theme discovery systems focusing on recent advances. Section 3 provides a high-level architecture for GENWISE and explains the role of RAG and other generative AI paradigms. Section 4 provides a comparison of our system’s performance on benchmarks as well as performance from state-of-the-art. Finally, we provide lessons learned in Section 5 that led to our ensemble architecture and note the conclusion in Section 6.

2 Related Works

Topic modelling is an information extraction technique that aims to extract a document’s intrinsic “themes/topics” from a collection of documents. There have been multiple methods proposed over the years for topic modelling, including statistical methods (Deerwester et al., 1990; Hofmann, 1999; Blei et al., 2003; Févotte and Idier, 2011), deep

learning-based methods using neural word embeddings (Moody, 2016; Dieng et al., 2020) and large language model (LLM) based approaches (Pham et al., 2023; Wang et al., 2023b) leveraging the zero-shot capabilities of the SOTA LLMs.

The earliest statistical approaches (Deerwester et al., 1990; Hofmann, 1999; Blei et al., 2003; Févotte and Idier, 2011) to model topics considered each document as a collection of words and modelled each document as a combination of underlying topics. LSI (Deerwester et al., 1990) decomposes a document term matrix using singular value decomposition (SVD) to identify the most prominent topics in each document. However, LSI has limited interpretability. Subsequently, Probabilistic latent semantic indexing (pLSI) (Hofmann, 1999) overcame this limitation by representing topics as multinomial random variables. Latent Dirichlet Allocation (LDA) (Blei et al., 2003), a hierarchical probabilistic model, generalized pLSI by incorporating Dirichlet conjugate priors for the word multinomial distributions over a topic and topic multinomial distributions over a document. However, these models have a limitation in that their document representation is inadequate as they do not consider the word context for creating document representation. They only consider the bag-of-words representations of the documents, ignoring the semantic relationships between words.

Of late, neural topic models (NTM) (Moody, 2016; Dieng et al., 2020) were proposed as computational power increased, and better text representation techniques such as (Mikolov et al., 2013) using neural models were discovered. One of the first proposed models was lda2vec (Moody, 2016) that employed word2vec (Mikolov et al., 2013) along with LDA. Lda2vec leverages the meaning encoded in a document to learn a better word representation by adding the document representation to the hub word representation to predict the context word representation. Here by hub, we refer to the key information or the cluster label. In the ETM model (Dieng et al., 2020), the topics contained in a document are represented in the same embedding space as the words. It produces a topic mixture from a logistic normal distribution and generates words for a topic by projecting the topic vector onto the vocabulary vector. ETM provides interpretable topics and achieves state-of-the-art results.

Recently, LLMs are increasingly used to analyze text automatically by prompting LLMs for tasks

Table 2: GENWISE compared to the closest works in the literature

Approach	GPTopic	TnT-LLM	TopicGPT	GENWISE
Input processing	Complete document	Queries an LLM to generate a summary of the document	Complete document	Use custom text cleansing modules including acronym expansion, spellchecker, etc., followed by querying an LLM to get the top-3 salient points in each document
Clustering	Uses HBDSCAN algorithm for clustering; Allows the user to specify a fixed number of topics; merges the topics using agglomerative clustering; Clustering occurs using either OpenAI / custom embeddings	Does not perform clustering explicitly	Does not perform clustering explicitly	Uses a hierarchical clustering approach generating themes at different granularities. Uses K-Means clustering in the 1 st level followed by two levels of an Enhanced Community Detection. Clustering occurs using embeddings obtained from an LLM.
Topic generation	Names and description of themes are generated by prompting an LLM with the top-k words related to the theme	Prompts LLM multiple times in sequence to list the topics present in the document. Follows a topic generation, followed by a topic update and then topic review prompts.	Queries an LLM to list the topics present in the document given some sample topics	Use the hub element generated from clustering to label the topic/theme
Topic assignment	The document is assigned the label of the cluster to which it belongs	Uses a light-weight logistic regression model trained on the labels assigned by an LLM	Prompts an LLM to classify a given document to one or more topics generated during the topic generation phase	The document is assigned the labels of its constituent salient points

such as summarization (Liu and Healey, 2023; Laban et al., 2023), clustering (Hoyle et al., 2023; Zhang et al., 2023; Viswanathan et al., 2023), and topic modelling (Grootendorst, 2022; Pham et al., 2023; Wang et al., 2023b; Reuter et al., 2024; Wan et al., 2024). TopicGPT (Pham et al., 2023) aims to generate and label topics in an automated fashion using LLMs. It generates new topics by passing sample documents and some sample topics to an LLM. This resulting set is refined to avoid duplication. Another work, TnT-LLM (Wan et al., 2024), creates a label taxonomy using an LLM following topic generation, update and review steps. A lightweight classifier is then trained on the generated label taxonomy for classification. Similar to our work, (Reuter et al., 2024) first performs clustering on the documents using the HBDSCAN (Campello et al., 2013) algorithm, followed by labelling the clusters formed by prompting an LLM with the top 500 words related to each cluster. These words can come from different documents clustered together in the same cluster. Table 2 describes the approach of each of the above works. GENWISE offers a significant advancement over these previous works providing an efficient end-to-end pipeline. Instead of using multiple prompts throughout its pipeline, GENWISE streamlines the process by prompting only once, enabling quicker and more effective theme generation. Moreover,

the hierarchical themes generated offers progressively finer themes providing a structured examination for users, starting from broader themes and moving to specific themes.

3 Solution Overview

Business analysts sift through large amounts of unstructured textual data to identify actionable themes. However, this data cannot be used in its raw form given that long texts, various ways of presenting the information, and domain information and jargon might be present as abbreviations. One needs to extract the key information to give it as an input to the clustering algorithm. We used generative large language models (LLM) to identify the key information of a raw text. In particular, we used open source LLMs such as Openchat² (Wang et al., 2023a) which is the best 7B parameter model at the time of writing this paper. As is widely known, a precise prompt is required for LLMs to extract the information suitably. We begin this section with details on prompt engineering for the financial text snippets. We subsequently explain the role of RAG, hierarchical representation, our enhancements to the clustering algorithm, and our high-level system architecture.

²<https://huggingface.co/openchat/openchat-3.5-1210>

3.1 Prompt engineering

We created the prompts using appropriate instruction placement, output format, multi-output responses, and negative instructions. Notably, the prompt engineering experiments were carried out with consideration for what different stakeholders expect the output to be.

1. **Instruction placement:** We noticed that providing the most important instructions at the beginning of the prompt was helpful to the LLM in carrying out the instruction. For example, *‘This is a textual note from the customer. note_text For this note, carry out the following tasks. task_list’*.
2. **Output format:** This helps to parse and use the LLMs output easily. For example, *‘For this note, provide the following information strictly in JSON format: output_format_example’*.
3. **Multi Output responses:** Asking for the multiple outputs in a single prompt to the LLM instead of using it multiple times to get output for a text at the different levels. For example, *‘The JSON object should list key ‘segments’ which summarize the text. For each segment, provide a ‘succinct description’ and a ‘concise label’*.
4. **Negative Instructions:** LLMs tend to hallucinate without precise instructions. Negative instructions help reduce hallucinations by bounding the tasks. For example, *‘Generate only the requested output, do not include any other language before or after the requested output. Do not repeat any information. Remove dates, amounts, and names.’*

3.2 RAG Framework

We address the challenge of LLMs misinterpreting acronyms by implementing the RAG framework, thus enhancing LLMs with internal knowledge for accurate acronym expansion. For example, "NPSL" may translate to "No present spending limit" for a financial company, while it expands to "National Premier Soccer League" as a general expansion. LLMs may hallucinate without accurately expanding the term *NPSL*.

We prevent this through a vector database (LlamaIndex³) that utilizes different indexing methods like VectorStore Index for semantic informa-

³<https://pypi.org/project/llama-index/>

tion and KeywordIndex for syntactic information, which is pivotal for acronym expansion.

We also used these indexes for ambiguous acronyms, i.e., which can be used as an acronym or word. "AM" is one such acronym. It can be either used as an acronym whose expansion is account manager (e.g., *am called to inform us*) or as a verb (e.g., *I am calling to ask*). We used these indexes to determine when such acronyms should be expanded. On an internal dataset, we found that when ambiguous acronyms are used, these indexes can identify them as acronyms 50% of the time. Moreover, when they are not used as an acronym, these indexes do not identify them as acronyms 80% of the time.

3.3 Levelwise clustering

Once the long descriptions corresponding to the key segments are extracted from the texts, they need to be clustered semantically. To present the semantic themes at various granularity levels, we create a hierarchy of clusters so that relationships between the clusters at different levels can be analyzed effectively.

Clustering techniques are applied at different levels to get granularity of themes, which can be crucial to understanding the data more deeply. Given that we cannot predetermine the number of themes in the data, we tried several non-parametric clustering algorithms such as Hierarchical Agglomerative clustering (HAC) (Hastie et al., 2009), mean-shift clustering (Derpanis, 2005), DBSCAN (Khan et al., 2014), etc., and observed best results with another algorithm fast community detection⁴. We further modified this solution and referred to it as "Enhanced Community Detection" (ECD). The overall hierarchical process works as follows:

- **Level 0 clusters (L_0):** Gets broad-level clusters with a primary objective of quickly partitioning large volumes of data. Extremely large clusters are not as informative as themes, and by suitably assuming a threshold for the maximum size of the cluster, we arrive at a broad estimate for the number of clusters. For this step, we use k-means clustering (Hartigan and Wong, 1979) as a fast, parametric clustering technique. This step is optional and depends on the volume of the data received.

⁴https://github.com/UKPLab/sentence-transformers/blob/master/examples/applications/clustering/fast_clustering.py

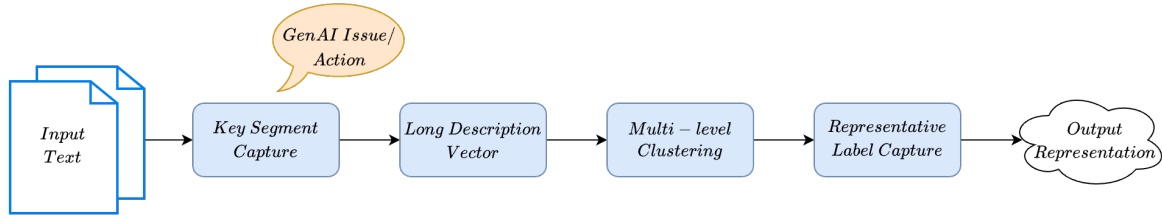


Figure 1: High-level system architecture for GENWISE

- **Level 1 clusters (L_1):** For all the k clusters obtained at the earlier step, ECD is applied to get level 1 clusters.
- **Level 2 clusters (L_2):** ECD is again applied with refined parameters to L_1 clusters to get further granularity (The parameters are refined using foundational algorithmic principles i.e., by reducing the cluster size threshold and increasing the similarity threshold). This meticulous process ensures that the clusters are more granular and specific, providing a deeper insight into the data.

Compared to HAC, the main advantages of community detection are its speed and ability to combine all the similar data points at once. On the other hand, HAC only combines two data points at a time, making it slower and prone to errors. Also, very similar points in HAC can result in totally different clusters depending on the initial configuration. This is not the case with ECD because many similar points are combined here in a single iteration rather than just 2 points. For instance, assume three similar data points: A, B, and C. The similarity of (A, B) is slightly greater than that of (A, C) and (B, C). In HAC, there can be a case where, after combining A and B, the similarity of (A, B) with C is insignificant. This is not the case with ECD, where all (A, B, C) will be combined simultaneously. Because of all these issues, ECD is much more suitable for obtaining coherent clusters.

3.4 Community detection algorithm Enhancements

The Enhanced Community Detection algorithm mentioned earlier clusters the data points based on cosine similarity between the points. In particular, it first identifies the community for the element, which includes all elements similar to that element. In many cases, communities can overlap, leading some elements to belong to multiple communities. However, we require each element to belong to a single community for the correct theme extraction. We must assign each element to only one

community to form non-overlapping communities. Therefore, given an empty set of non-overlapping communities, we add an element to a new community only if the element is not a part of any existing community.

We observed a code flaw in sorting elements based on indices before forming non-overlapping communities, resulting in the loss of the position of the hub element (the one similar to all others). We identified and rectified it, increasing the cluster coherency by 20%. Further, we merged non-overlapping communities based on similarity, creating crisply defined clusters.

3.5 System Architecture

We now provide the end-to-end architecture for GENWISE. The block diagram is shown in Figure 1.

1. **GenAI Issue/Action Capture:** This stage is critical for the initial processing of input text. The generative AI model scans the provided text, identifying and extracting key information segments. Our custom prompt and RAG are enabled at this stage to capture the information precisely. As seen from the first block in Figure 1, these key segments are usually issues or actions relevant to the text’s subject matter. For each extracted segment, we prompt the model to list a *succinct description* and a *concise label*.
2. **Vector Conversion:** Post extraction, each segment is converted into a vector form using its succinct description. For this conversion, we use state-of-the-art models as per the Massive Text Embedding Benchmark (MTEB)⁵.
3. **Multi-Level Clustering:** In this phase, the vectorized data undergoes hierarchical clustering. This method groups the vectors into clusters

⁵The Massive Text Embedding Benchmark (MTEB) is an extensive benchmark developed to assess the performance of text embedding models on many tasks and datasets. <https://huggingface.co/spaces/mteb/leaderboard>

Table 3: Overall Quantitative comparison of BERTopic and GENWISE-predicted labels

Data Source	documents	Label Similarity		Increase
		BERTopic	GENWISE	
CFPB	2000	65%	69%	6.2%
Bills	1000	59%	62%	5.1%

based on their similarities. The multi-level aspect of this clustering allows the system to organize the data at various levels of granularity, facilitating a more nuanced understanding and categorization of the themes within the text.

- 4. Representative Label Extraction:** The final step of the pipeline is the extraction of representative labels for each cluster. This process involves identifying the data point that accurately encapsulates each cluster’s core theme or idea. We found that the hub element serves as a good representative for the cluster as it acts as the central element for forming a community. We choose the concise label of the hub element to label the cluster. This process is repeated for each level of the constructed hierarchy.

4 Experiments

This section provides three findings:-

- Dataset - A brief introduction to the datasets used for experimentation.
- Label comparison – We compared BERTopic and GENWISE-predicted labels quantitatively and qualitatively.
- Quantitative evaluation of Enhanced Community Detection algorithm and time complexity of related algorithms
- Industry data-based study on Fixed Term Effort (FTE) reduction with GENWISE

4.1 Dataset

For experiments, we considered two datasets from different domains.

CFPB⁶ is a consumer complaints database by the Consumer Financial Protection Bureau (CFPB), which contains two kinds of labels for a complaint – Issue and Sub-issue (the issue & sub-issue mentioned by the consumer in the complaint). We have

⁶<https://www.consumerfinance.gov/data-research/consumer-complaints/>

taken a sample of 2000 complaints from CFPB for our experiment.

Bills is a generic dataset summarising the bills discussed in the U.S. Congress Bills (Adler and Wilkerson, 2018). This dataset has 21 high-level and 114 low-level human annotated labels. A sample of 1000 summaries has been considered for our experiment.

4.2 Label comparison

The ground truth (provided in the annotated datasets) and predicted labels (from LDA, BERTopic) are compared qualitatively and quantitatively with the labels generated by GENWISE. The results are mentioned in Table 3 and Table 4. In contrast to current clustering algorithms that produce a single label after processing, GENWISE automatically generates a hierarchy of labels, providing a more nuanced understanding of the data. Furthermore, GENWISE generates more informative labels than those produced by either BERTopic, LDA or the annotated labels. We evaluated it quantitatively by comparing the semantic similarity of the annotated label and the complaint. This process involved a manual comparison of a subset of the labels (presented in Table 4). Bge-large-en-v1.5⁷ has been used to compute the embedding of labels and complaints, and dot product has been used to compute the similarity. For both datasets, on average, the semantic similarity of GENWISE-predicted labels is 5% higher than that of BERTopic labels (mentioned in Table 3). For the examples provided in Table 4, Labels derived from GENWISE predictions have a higher semantic similarity than BERTopic.

4.3 Quantitative comparison

We compared the performance of old and Enhanced Community Detection algorithms using commonly used metrics, such as topic coherence and topic diversity. Both measures are based on the hub element (label) of community detection, as it represents that cluster. They are calculated for the last level of the hierarchy. For a cluster, topic coherence is implemented as normalized pointwise mutual information for n points closest to the hub element (n is taken as 3 for our study) (Bouma, 2009). Its value ranges from (-1, 1), where higher values show a more significant intracluster correlation. Topic

⁷<https://huggingface.co/BAAI/bge-large-en-v1.5>

Table 4: Qualitative & Quantitative comparison of BERTopic and GENWISE-predicted labels

Data Source	Complaint/Bill Summary	Issue -> Sub-issue	BERTopic		GENWISE	
			Label	Similarity	Label	Similarity
CFPB	(ID 7284263) I submitted a letter to the Credit Bureaus to correct these erroneous accounts. I think you have not validated these accounts in accordance with Sections 609, and I will pursue legal action against them.	Problem with a credit reporting company's investigation into an existing problem -> Their investigation did not fix an error on your report	34_in- clude_pursue_suspicious_prior	56%	Unauthorized credit report ->Request for removal of erroneous items under Fair Credit Reporting Act	77%
CFPB	(ID 7317133) I filed a dispute for incorrect information on my credit report.I received an email from the credit bureau stating that they are assuming the disputes are coming from a 3rd party.They in fact did not come from a 3rd party.I even called them to verify it was me and they still refused to process my disputes.	Problem with a credit reporting company's investigation into an existing problem -> Their investigation did not fix an error on your report	32_dis- pute_verify_incorrect_information	69%	Unauthorized credit report -> Dispute Not Processed by Credit Bureau	82%
Bills	(112-S-3595) Amends the Internal Revenue Code to exempt from passive loss rules any activity of a taxpayer carried on by a high technology research small business pass-thru entity. Defines "high technology research small business pass-thru entity"	Domestic Commerce -> Small Businesses	1_tax_credit_revenue_internal	59%	Exempting High Technology Research Small Businesses -> Exemption from Passive Loss Rules for High Technology Research Small Business	87%
Bills	(110-HR-614) Amends titles XI and XIX (Medicaid) of the Social Security Act (SSA) to remove the cap on Medicaid payments for Puerto Rico, the Virgin Islands, Guam, the Northern Mariana Islands, and American Samoa.	Public Lands ->Dependencies & Territories	2_health_medicare_service_care	52%	Removing Cap on Medicaid Payments ->Amendment to Social Security Act	73%

Table 5: Cohesion for Old & Enhanced community detection

Similarity threshold	Cohesion (Old)	Cohesion (Enhanced)	In-crease
60%	58%	76%	30%
65%	63%	74%	16%
70%	68%	76%	11%

diversity measures the intercluster correlation and is calculated by computing the pairwise similarity between the most representative members of every cluster. A larger diversity score indicates clusters that are distinct with the least overlap.

On the sample of the CFPB dataset mentioned above, we calculated cluster cohesion and diversion across a range of similarity thresholds for both old and Enhanced Community Detection algorithms. On average, cohesion increased by 20% for all such experiments. Topic diversity was similar across both old and Enhanced Community Detection algorithms.

Table 5 shows the coherence across different similarity thresholds.

End-to-end time comparison: We also noted the time taken to perform different components of the end-to-end pipeline in Table 6. Overall, BERTopic took the least time to run the complete pipeline. Since GENWISE uses a large language model to generate the descriptions from the document, it takes much more time than BERTopic end-to-end. Another thing to note is the time taken in an online setting. GENWISE takes the same time

Table 6: Time taken (in minutes) to generate themes on CFPB end-to-end using BERTopic, TopicGPT and GENWISE.

Time Taken (in min)	BERTopic	TopicGPT	GENWISE
Input Processing	0	0	5
Embedding + Clustering	0.7	0	2
Topic Generation	0.1	5	0.1
Topic Assignment	0.1	10.1	0.1
Total	0.9	15.1	7.2

as BERTopic in an online setting, as the first two pipeline stages for both BERTopic and GENWISE are pre-computed. However, each text snippet has multiple levels of granularity due to the themes provided by GENWISE at different granularity levels, which provides more information on the documents than BERTopic. Compared to TopicGPT, another LLM-based pipeline involving prompting, GENWISE is 50% faster as TopicGPT uses multiple prompts to run the complete pipeline.

4.4 Industry data-based study

Lastly, we report the experimental investigation on industry data, which is only a small part of complete unstructured data. The task involves analysing and extracting frequently appearing themes among customer text complaints received through the customer support helpline or email. The customer complaint dataset comprises a diverse set of complaints. The dataset comprises

Table 7: Data distribution

Time Period	Records	Customers
Aug-2022	132K	110K

Table 8: Results on industry based dataset

Records	Existing process	GENWISE
Total records (Aug 2022)	132K	132K
Complaints from high-risk category (Based on Complaint categorization)	12K	12K
Records for manual review	942	50
Actionable complaints (Opportunities) ⁹	8	8
Issue hit rate ¹⁰	<1%	16%

approximately 190 attributes⁸ for each complaint, which describes a customer’s spending history and other customer-specific information. One of these 190 fields is a complaint field in textual format. This complaint field describes major issues faced by the customer. This dataset is crucial for financial control and ensuring compliance with regulations. It helps promptly address high-risk complaints and issues to prevent potential legal or financial risks. In such cases, taking necessary actions as quickly as possible is essential. Table 7 provides details on the data distribution for the dataset.

In our research, we investigated the effectiveness of using GENWISE in reducing the amount of Fixed Term Effort (FTE) required in the existing complaint categorization process. FTE refers to the predetermined manual work or resources assigned to a specific task for a set duration. The existing process involves Customer Care Professionals (CCPs) manually filtering and reviewing complaints, which can be time-consuming and inefficient. GENWISE is a tool designed to automate this process and provide direct guidance to CCPs in identifying critical complaints, thus reducing the need for manual efforts.

As we discovered, the implementation of GENWISE led to a significant decrease of 95% in the FTE required for the existing process. Of 942 complaints in the industry-based dataset, 95% (892 complaints) were categorized as low-risk, enabling

⁸Due to privacy reasons, this dataset cannot be released.

⁹Throughout this paper, ‘actionable complaints’ (opportunities) refers to high-risk complaints that warrant quick action.

¹⁰The term “issue hit rate” refers to the percentage derived from the ratio of actionable complaints to the total number of complaints manually reviewed.

CCPs to concentrate only on the 50 high-risk-themed complaints that required manual review for actionable items. For a clear presentation of the outcomes obtained from the industry-based dataset, please refer to Table 8.

Additionally, the implementation of GENWISE significantly increased the issue hit rate. Previously, the rate was less than 1% , with only eight instances identified out of 942 complaints. However, with GENWISE, the rate dramatically rose to 16%, with 8 cases identified from a smaller sample size of 50 complaints.

5 Observations and Lessons Learnt

This paper outlines our method for extracting themes from unorganized and unlabeled textual data using specific knowledge in the field. Our approach is particularly effective in quickly identifying main themes in extensive data collections. Additionally, by using Enhanced Community Detection, we attained more connected and refined outcomes. The flexibility of this inclusive process enables smooth application on various datasets, with minimal adjustments and parameter tuning needed.

- Hierarchical clustering representation suits broad user community:** We started with regular clustering algorithms, which gave us a single label. Moreover, these labels usually fall in the medium range of granularity, i.e., they convey the subject of the cluster but not exactly what the cluster is about. Through ongoing engagement with our stakeholders, we discerned that they require labels at multiple levels - a broader label and then a label that tells precisely about the cluster. Depending upon the use case and the team utilizing the clustering output, multiple granularity levels would be required in the clustering. Thus, we designed a hierarchical clustering pipeline with three levels of clustering. The granularity of the clustering increased with each level. This helped us create a product aligned with the business requirements.
- Streamlined computation for real-time analysis:** Initially, we tried to run the entire approach in real-time, which was slow and sub-optimal. Maximizing computational tasks through batch processes is crucial to optimize the overall pipeline’s latency, minimizing the load during inference. This objective was accom-

plished by conducting various steps, such as pre-processing, RAG, LLM issue detection, embedding generation, and clustering labelling as offline batch processes. During inference, the focus was narrowed to efficient clustering analysis through Enhanced Community Detection, ensuring streamlined pipeline performance.

3. Importance of appropriately labelling a cluster: Following identifying a cluster, it becomes imperative to aptly label it, allowing users to grasp its essence succinctly. As highlighted in the prompt engineering section, we engaged various stakeholders to achieve this, tapping into their domain expertise. This collaboration proved invaluable in prompt engineering and the RAG-mentioned steps above. These methods underwent meticulous refinement, yielding verbose and succinct labels tailored to the specific requirements articulated in the prompt. Notably, we emphasized phrases near the cluster's centroid to discern the cluster's optimal semantic essence as mentioned in the Topic Generation step.

4. Evaluation of the pipeline: The critical challenge in our process stems from the subjective and domain-specific nature of generating content using Large Language Models (LLMs). To overcome this challenge, we carefully examined specific data segments, sometimes using a list of keywords related to particular customer issues. We used the ground truths obtained from these segments as benchmarks to assess the performance of our pipeline. By comparing the ground truths with the results generated by our pipeline, we created a confusion matrix for analysis. Additionally, incorporating some random data mixed with data containing a known set of themes allowed us to discover valuable insights in specific areas where improvements to the model were needed. This experiment was carried out with different random mixes and known theme distributions to check if the model could identify the themes independently.

6 Conclusion

Motivated by an industrial setting - going beyond accuracy and looking for trust and interactivity - we built and presented an end-to-end system, GENWISE, to derive themes from the text. Our system makes descriptive analytics and reporting much

easier and more natural for users. We made it trustworthy through features such as a clustering hierarchy and appropriate labels. During this journey, we encountered several gaps in academic solutions for the clustering. We presented lessons learnt while overcoming these challenges and supporting demands from business stakeholders.

References

- E Scott Adler and John Wilkerson. 2018. Congressional bills project: 1995-2018.
- David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.
- Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.
- Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. 2013. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer.
- Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.
- Konstantinos G Derpanis. 2005. Mean shift clustering. *Lecture Notes*, 32:1–4.
- Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.
- Cédric Févotte and Jérôme Idier. 2011. Algorithms for nonnegative matrix factorization with the β -divergence. *Neural computation*, 23(9):2421–2456.
- Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.
- John A Hartigan and Manchek A Wong. 1979. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108.
- Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. 2009. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.
- Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

- Alexander Hoyle, Rupak Sarkar, Pranav Goel, and Philip Resnik. 2023. [Natural language decompositions of implicit content enable better text representations](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13188–13214, Singapore. Association for Computational Linguistics.
- Kamran Khan, Saif Ur Rehman, Kamran Aziz, Simon Fong, and Sababady Sarasvady. 2014. Dbscan: Past, present and future. In *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)*, pages 232–238. IEEE.
- Philippe Laban, Wojciech Kryscinski, Divyansh Agarwal, Alexander Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. 2023. [SummEdits: Measuring LLM ability at factual reasoning through the lens of summarization](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9662–9676, Singapore. Association for Computational Linguistics.
- Sengjie Liu and Christopher G Healey. 2023. [Abstractive summarization of large document collections using gpt](#). *arXiv preprint arXiv:2310.05690*.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. [Efficient estimation of word representations in vector space](#). *arXiv preprint arXiv:1301.3781*.
- Christopher E Moody. 2016. [Mixing dirichlet topic models and word embeddings to make lda2vec](#). *arXiv preprint arXiv:1605.02019*.
- Chau Minh Pham, Alexander Hoyle, Simeng Sun, and Mohit Iyyer. 2023. [Topicgpt: A prompt-based topic modeling framework](#). *arXiv preprint arXiv:2311.01449*.
- Arik Reuter, Anton Thielmann, Christoph Weisser, Sebastian Fischer, and Benjamin Säfken. 2024. [Gp-topic: Dynamic and interactive topic representations](#). *arXiv preprint arXiv:2403.03628*.
- Vijay Viswanathan, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. 2023. [Large language models enable few-shot clustering](#). *arXiv preprint arXiv:2307.00524*.
- Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W White, Longqi Yang, et al. 2024. [Tnt-llm: Text mining at scale with large language models](#). *arXiv preprint arXiv:2403.12173*.
- Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023a. [Openchat: Advancing open-source language models with mixed-quality data](#). *arXiv preprint arXiv:2309.11235*.
- Zihan Wang, Jingbo Shang, and Ruiqi Zhong. 2023b. [Goal-driven explainable clustering via language descriptions](#). *arXiv preprint arXiv:2305.13749*.
- Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. [Clusterllm: Large language models as a guide for text clustering](#). *arXiv preprint arXiv:2305.14871*.

FINALE : Finance Domain Instruction-Tuning Dataset with High-Quality Rationales via Chain-of-Thought Prompting

Sangmin Lee

Korea University

sangmin_lee@korea.ac.kr

Suzie Oh

KT

suzie.oh@kt.com

Saeran Park

Korea University

saeran_park@korea.ac.kr

Gyujin Son

Yonsei University

spthsrbls123@yonsei.ac.kr

Pilsung Kang*

Korea University

pilsung_kang@korea.ac.kr

Abstract

Recent research on financial domain large language models (LLMs) progress by applying instruction tuning to general-domain LLMs, which are known for their powerful reasoning and generation capabilities. However, specialized domains such as finance and legal are replete with arcane terminology and require specialized knowledge, resulting in a diminished user understanding of the outputs generated by LLMs. Therefore, it is crucial to augment user comprehension by accompanying the model-generated responses with detailed rationales. Nevertheless, previous works focus primarily on training to generate the answer, failing to generate appropriate rationales in the financial context. Therefore, we propose FINALE, a financial instruction tuning dataset that includes high-quality rationales generated through the use of a Chain-of-Thought (CoT) prompting and quality filtering. A model trained on FINALE shows an average improvement of 9% across nine sub-tasks compared to models trained on other instruction tuning datasets. Additionally, human evaluation results show that the comprehensibility of outputs from models trained on FINALE is rated four times higher. Through various analytical experiments, we demonstrate the effectiveness of FINALE and emphasize the importance of training models to generate high-quality rationales.

1 Introduction

Recent advancements in Large Language Models (LLMs), such as LLaMA (Touvron et al., 2023), GPT-4 (OpenAI et al., 2024), and Alpaca (Taori et al., 2023), have generated significant interest in their application across diverse domains. Researchers are actively exploring how these models’ powerful generative capabilities can be leveraged to tackle various of tasks. Notably, using instruction tuning methodologies (Wei et al., 2022), LLMs

are fine-tuned to perform various tasks through domain adaptation (Bao et al., 2023; Yue et al., 2023a). In the financial domain, downstream tasks are defined, training data is compiled, and general-domain LLMs are fine-tuned to enhance performance across multiple tasks through by applying instruction tuning (Wu et al., 2023a; Wang et al., 2023; Xie et al., 2023).

Most tasks evaluated within the finance domain generate short answers. However, finance documents are characterized by specialized knowledge and rare words (Mik, 2017), making it difficult for users to comprehend the outputs generated by LLMs in financial tasks (Misheva and Osterrieder, 2023; Hicham Sadok and Maknouzi, 2022). Therefore, it is essential to verify 1) whether the financial knowledge is understood and 2) whether the answers are derived based on correct reasoning. However, previous studies overlook these considerations.

We think adding reasoning steps that serve as rationales to short answers is a feasible alternative. In the general domain, enhancing the quality and length of reasoning used in training data improves model performance (Wang et al., 2022; Sanh et al., 2022; Mukherjee et al., 2023). Similarly, InvestLM (Yang et al., 2023b) enhances performance by utilizing long answers in its training data. However, it does not apply the concept of rationales that consider the constructed questions and context. Additionally, this method requires substantial human resources because it relies on manually curated selections.

In this study, we introduce FINALE, a dataset that provides rationales for texts generated in the finance domain. Additionally, we present a construction pipeline that aims to develop high-quality rationales with minimal human effort.

When trained with the developed FINALE, it exhibits an average performance enhancement of 9% over other instruction-tuned models. This aligns

* Corresponding author.

with findings in the general domain that training models to generate reasoning steps improves performance (Wei et al., 2023; Nye et al., 2021; Zhou et al., 2023b; Gao et al., 2023), and marks the first study to demonstrate its applicability in the financial domain. Figure 3 shows that models trained on FINALE exhibit an average win rate four times higher than the baseline model in human evaluation comparisons of rationale quality. This demonstrates that training with high-quality rationales enhances user comprehension.

Additionally, Son et al. (2024); Bi et al. (2024) report a degradation in general performance when domain-specific LLMs are trained with instruction tuning. Yet, models trained with FINALE exhibit less degradation in performance compared to the baseline model. The contributions of this research can be summarized as follows:

- We propose FINALE, a dataset that enriches short answers in the finance domain with high-quality rationales.
- We provide a pipeline for constructing rationales with minimal human effort.
- Our evaluation results indicate that models trained with FINALE demonstrate improved performance and enhanced comprehensibility of generated text compared to those trained with other instruction-tuning datasets.

2 Background

2.1 Finance-Specific LLMs

Traditional financial language models have been studied in various scenarios based on BERT (Devlin et al., 2019), a representative encoder-based pre-trained model. These studies have focused on Named Entity Recognition (Nakayama and Wan, 2017), News Sentiment Analysis (Araci, 2019), and Text Summarization (La Quatra and Cagliero, 2020).

The advancement of LLMs such as ChatGPT (OpenAI, 2022), GPT-4, Alpaca, and LLaMA has led to research applying their reasoning and generation capabilities to the financial domain. One of the primary methods for domain adaptation of these LLMs is instruction tuning (Wei et al., 2022), which involves fine-tuning LLMs using various instructional data to achieve desired behavioral patterns (Bao et al., 2023; Yue et al., 2023a). A notable example related to this is BloombergGPT

(Wu et al., 2023b), which proposes a 50 billion parameter language model trained on a finance-specific corpus. However, the non-disclosure of the dataset poses challenges for further development of financial LLMs. To address this issue, research has focused on training on small LLMs (Son et al., 2023) and multi-task training (Wang et al., 2023; Xie et al., 2023) for specific financial tasks using publicly available financial data. Specifically, FinGPT adopts a data-centric approach and trains using LoRA (Hu et al., 2021), providing useful resources for researchers to develop their financial LLMs. InvestLM constructs its training data by manually collecting long answers based on results that extended reasoning steps significantly improve performance (Wang et al., 2022; Sanh et al., 2022; Mukherjee et al., 2023).

Our study deviates from previous research in the form of the target answers for finance LLMs. Most studies train models to generate short answers, which do not provide the rationales before the final answers. We further construct rationales suitable for the financial context. Additionally, we enhance quality by using filtering methods rather than manually selecting all data, minimizing human resources.

2.2 Chain-of-Thought Fine-tuning

Chain-of-Thought prompting (CoT) (Wei et al., 2023) is a method that encourages the model to generate reasoning steps before providing an answer. This approach effectively enhances the reasoning capabilities of language models (Nye et al., 2021; Zhou et al., 2023b; Gao et al., 2023). Furthermore, CoT-Collection (Kim et al., 2023a) has shown that training models to generate reasoning steps improves both zero-shot and few-shot performance in the general domain. Inspired by this, we aim to construct an instruction tuning dataset that includes high-quality rationales to assist the model’s reasoning capabilities in the financial domain. Through this approach, we seek to enhance both the performance of the model and the quality of the generated rationales, improving user comprehension.

3 FINALE

3.1 Task Overview

Our work extensively addresses whether including rationales for diverse forms, such as numbers and tables, in addition to the traditional text-based

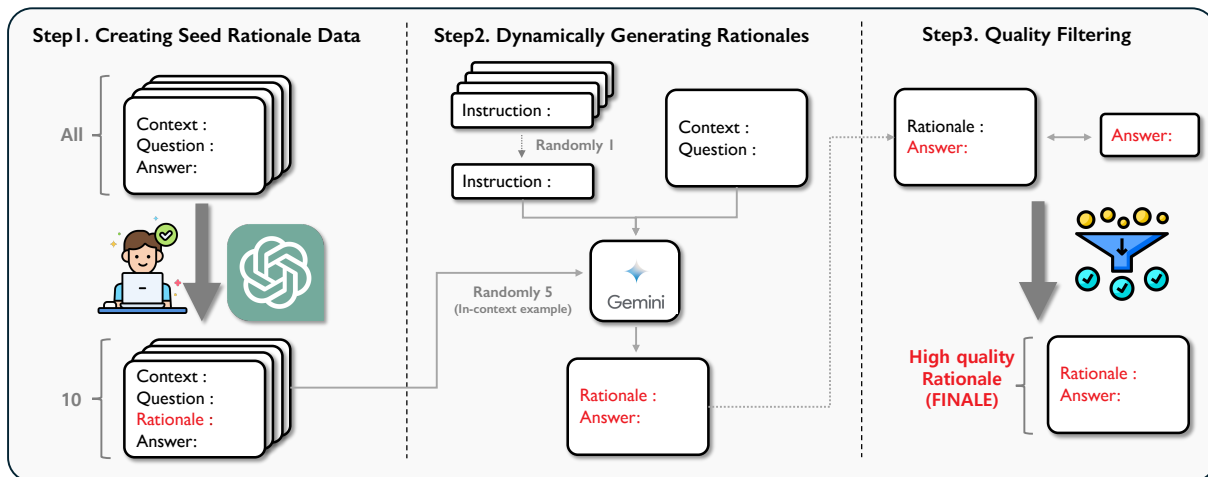


Figure 1: Overview of the FINALE creation pipeline consisting of 1) Creating seed rationale data and 2) Dynamically generating rationales and 3) Quality filtering.

Dataset	Sub-Task	# of Rationale (BF)	# of Rationale (AF)
<i>Sentiment Analysis</i>			
(Malo et al., 2014) (Organizers)	Sentiment Analysis (FPB)	4836	2888 (-41%)
	Impact Type Prediction (ESG)	790	289 (-63%)
<i>Numerical Reasoning</i>			
(AiHUB, 2023b)	Arithmetic (Arith.)	23064	5368 (-76%)
	Extraction (Extract.)	21000	11061 (-47%)
	Comparison (Comp.)	23016	14844 (-35%)
<i>Question Answering</i>			
(AiHUB, 2023a)	Multiple-Choice Question Answering (MCQA)	5265	4715 (-20%)
	Extractive Question Answering (EQA)	8248	652 (-92%)
	Binary Question Answering (BQA)	6368	4730 (-25%)
(AiHUB, 2023c)	Table Question Answering (TQA)	50000	31886 (-36%)
Total		98681	76433 (-23%)

Table 1: An overview of the FINALE dataset. # of Rationale (BF) denotes the total number of rationales generated, # of Rationale (AF) denotes the total number of rationales after the filtering process.

instruction tuning dataset, enhances generative capabilities in the financial domain. To this end, the task selection criteria adhere to these principles.

Firstly, instead of selecting homogeneous tasks to collect data, we diversify by including different data types, such as numbers and tables. The downstream tasks are divided into four categories, including Numerical Reasoning (AiHUB, 2023b), Question Answering (AiHUB, 2023a), and Table Question Answering (AiHUB, 2023c), which require proficiency within the financial domain.

Secondly, nine sub-tasks are selected, and data is collected to ensure coverage across diverse areas. For example, the Sentiment Analysis task includes: classifying the sentiment of news sentences using a financial phrase bank and classifying risk from an ESG perspective, considering detailed scopes. Each sub-task comprises a dataset ranging from a mini-

mum of 790 to a maximum of 50,000 instances. For more information about FINALE, refer to Table 1.

3.2 Dataset Creation

The objective of FINALE is to construct high-quality rationales in addition to the existing answers corresponding to contexts and questions. Inspired by Chung et al. (2022), we generate rationales for all instances in a CoT prompt format. In this study, the reasoning steps generated by the model are considered as rationales for the answers. Additionally, a three-step filtering process is applied to select high-quality rationales.

3.2.1 Creating Seed Rationale Data

Initially, the authors use GPT-4 to craft high-quality rationales, which are later used to guide following generations as in-context examples. Therefore, it is necessary to select a variety of high-quality ra-

tionale types. The selection criteria adhere to two main principles. Firstly, we select ten examples per sub-task while ensuring diversity. For instance, in the Arithmetic sub-task, Arithmetic is categorized into addition, subtraction, multiplication, and division. The authors reviews whether instances fall into these types and selects them in equal numbers.

Secondly, to ensure high-quality rationales, rationales are generated using GPT-4. We employ answer-based filtering and manual review to further filter for quality. If the final answer of a generated rationale differs from the gold answer, it is considered low quality. Therefore, only those with matching answers are selected. Furthermore, incorrect arithmetic operations and brief rationales, considered inadequate for explaining answers, are removed.

As discussed in LIMA (Zhou et al., 2023a), manually evaluating the quality and diversity of rationales ensures higher data quality when humans select sentence-form data. For more information about the prompts for generating seed rationales, refer to Appendix A.

3.2.2 Dynamically Generating Rationales

The seed rationale data is used as an in-context example within Gemini-Pro to generate rationales for all instances. We opt to use Gemini-Pro due to cost constraints. Nevertheless, it provides generative capabilities comparable to GPT-4, making it a reasonable alternative (Team et al., 2023). When generating rationales, we adhere to the following protocol:

The diversity in instruction data, as evidenced by WizardLM (Xu et al., 2023), has shown significant performance improvements. Therefore, instead of using fixed in-context examples, we dynamically change the in-context examples and instructions according to the principles of dynamic prompting (Yang et al., 2023a). Five out of ten seed data are randomly selected for the in-context examples, and one out of five instructions is chosen at random. This approach effectively prevent the monotony of rationales within the dataset. For more information about in-context example prompts for generating rationales, refer to Appendix B.

3.2.3 Quality Filtering

We designed a selection process to guarantee the selection of only high-quality rationales for all generated instances. While manual review of each instance is the most effective method to en-

sure fidelity and relevance, this approach is time-consuming, costly, and inefficient. Therefore, an automatic filtering method is applied to all instances. This filtering technique involves selecting rationales where the final answer included in the generated rationale matches the gold answer. For EQA and TQA, we filtered by ROUGE score because the answers were very long. Filtering by EM significantly reduced the data size. Instances are selected only if their ROUGE scores exceed the threshold. The criterion for choosing the threshold value was established empirically through manual inspection, determining that quality is assured when the value is 0.6 or higher.

3.3 Dataset Analysis

As illustrated in Figure 2, FINALE proposed in this paper consists of data that is, on average, longer and has a more varied distribution compared to the KOR-OpenOrca-Playti-V3 (KyujinHan) and CoT-Collection-Ko (Kim et al., 2023b) datasets. Liu et al. (2023) has demonstrated that the length of instruction data is crucial in building better models. Table 2 compares the datasets’ total number of instances and the count of unique words. FINALE contains the highest number of instances and unique words among these datasets. Moreover, unlike the other two datasets that cover various domains, FINALE includes a diverse vocabulary within the confined domain of finance. A diverse vocabulary can help improve performance (Choe et al., 2023).

Dataset	Instances	Unique Words	Rate
FINALE (Ours)	78k	96k	123%
KOR-Platypus-v3 ¹	34k	73k	213%
CoT-Collection-Ko	77k	89k	155%

Table 2: Comparison of number of instances and unique words. unique words is the number of unique words after tokenize the Mecab-Tokenizer (Kudo et al., 2004), and rate is the ratio of the total number of unique tokens divided by the total number of instances.

4 Experiments

4.1 Experimental setting

Baseline Models We utilize two open-source models that show high performance on Korean tasks as the foundation and baseline model, selecting models with 7 billion parameters due to resource limitations.

¹KOR-OpenOrca-Playti-V3

Models	FPB	ESG	Arith.	Ext.	Comp.	MCQA	EQA	BQA	TQA	Avg
Yi-6B-Ko	28.0	9.6	9.2	<u>55.2</u>	58.0	44.0	43.2	39.6	21.6	34.3
Ko-Platyi-6B	54.0	23.6	9.2	52.4	60.0	65.2	56.8	58.0	22.8	44.7
FINALE (OURS)										
Single (per100)	56.8	15.2	10.0	52.0	60.0	70.8	41.2	85.6	15.6	45.2
Single (per400)	54.4	<u>21.6</u>	<u>14.8</u>	53.6	57.2	83.2	46.8	83.2	15.2	47.8
Single (All)	<u>71.2</u>	15.2	<u>14.8</u>	54.8	68.8	89.2	<u>50.0</u>	<u>86.8</u>	24.8	<u>52.8</u>
Multi (All)	76.0	19.6	15.6	55.6	<u>68.4</u>	<u>83.6</u>	<u>50.0</u>	87.2	24.8	53.4
Δ	+22.0%	-4.0%	+6.4%	+3.2%	+8.4%	+18.4%	-6.8%	+29.2%	+2.0%	+8.7%

Table 3: Performance of models with different samples of FINALE. The highest-scoring model per task is highlighted in **bold**, and the second-highest is underlined. Δ values indicate the percentage change in performance of FINALE trained with Full data, compared to the Ko-Platyi-6B model.

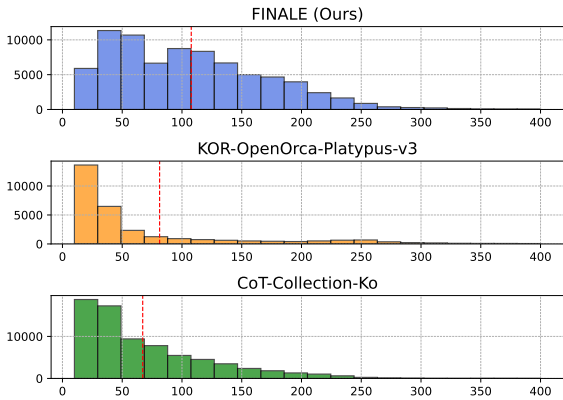


Figure 2: Length distribution of different datasets. The X-axis represents the number of tokens in each instance after tokenization, and the Y-axis represents the number of instances.

Yi-6b-ko is a model that is further pre-trained on Korean and English datasets using the Yi-6B (Yue et al., 2023b). We chose this model because it is recognized for its superior Korean language comprehension among smaller-sized models, as confirmed by the Korean LLM Leaderboard (Park et al., 2023) and KMMLU (Son et al., 2024).

Ko-Platyi-6B is a model that is instruction-tuned using the KOR-OpenOrca-Playti-V3 dataset. The KOR-OpenOrca-Playti-V3 dataset is a Korean translation of the Open-Platypus dataset. The reason for selecting Ko-Platyi-6B as the baseline is that it is an instruction-tuned model, which allows for an equivalent comparison. For more information about the training method and hyperparameters, see Appendix C.

Evaluation Method Given the varying sizes of the evaluation datasets across different datasets, we randomly sample 250 instances from the original evaluation dataset for each sub-task. Consequently, the evaluation dataset comprises 2250 instances

across the nine sub-tasks.

The evaluation prompt is assessed in a few-shot setting rather than a zero-shot setting to measure the maximum performance of the model. In the few-shot setting, three in-context examples are used, selected considering for the model’s maximum input length. To ensure a fair comparison of model performance, the same prompt is used for all models, although the instructions are different for each task. The details of the evaluation prompt can be found in Appendix D.

To ensure precise performance measurement of the model, sophisticated post-processing is applied to each sub-task. For instance, in the Arithmetic sub-task, the number following the "=" symbol in expressions like "3+5=2" is extracted as the final answer. For generated text that is not addressed by post-processing, we apply the Cover EM method (Rosset et al., 2021), a technique also used in the existing financial LLM, FinGPT.

4.2 Experimental Results

4.2.1 Performance Comparison

The results in Table 3 demonstrate that the model trained using FINALE (Multi) outperforms the Ko-Platyi-6b in most sub-tasks, with an average performance that is 20% higher than Yi-6B-Ko and 9% higher than Ko-Platyi-6B. This indicates that training the model to generate rationales before generating the final answers leads to more accurate answer generation.

4.2.2 Rationale Quality Comparison

Methodology To demonstrate the efficacy of FINALE, which is constructed to ensure high quality, we measure the quality of generated text from models trained using FINALE. We select Ko-Platyi-6b, an instruction-tuned model, as the

baseline. Three human annotators evaluate the answer generated by both models according to the assessment guidelines provided in Appendix E. Human annotators are composed of experts in economics. Ten samples for each sub-task are randomly selected for assessment.

Annotators are directed to determine the more comprehensible rationale from different models. If no rationale is deemed superior, evaluators score it as a "Tie" (2). Ultimately, the average score calculated by the three annotators for each sub-task determines the final score. The inter-annotator agreement shows a high consistency rate of 83% on average across sub-tasks.

Results Figure 3 indicates that outputs from the model trained with FINALE average scored four times higher than the baseline, demonstrating that FINALE was significantly helpful. This finding suggests that high-quality rationales improve the model’s ability to reason correct answers and significantly enhance comprehension for the users viewing the generated text. Particularly in tasks like ESG, EQA, and TQA, where quantitative performance is low, the comprehensibility of the generated text is higher than that of the baseline. This indicates that even if the model generates incorrect answers, a high-quality rationale increases the likelihood of human understanding. Therefore, we emphasize that in the finance domain, the rationale quality is as crucial as the correctness of the answers generated by the model.

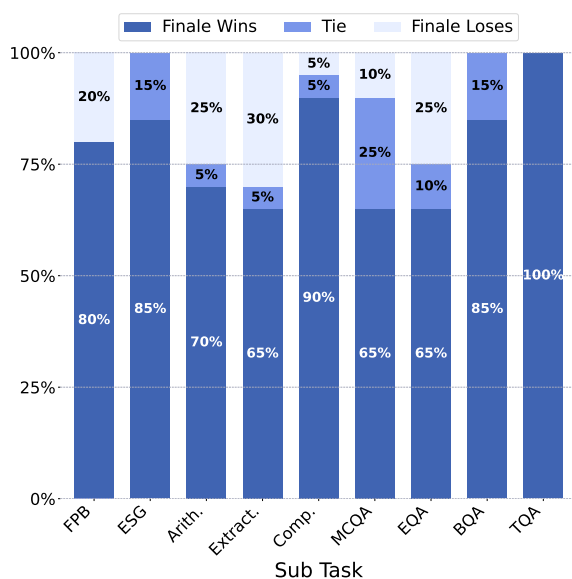


Figure 3: Result of the human evaluation of rationales generated by the model trained on Yi-6B using FINALE and the rationales generated by the Ko-Playti-6B.

5 Demonstrate the Effectiveness of Data Construction

To demonstrate the effectiveness of the FINALE construction method, we conduct additional experiments.

5.1 Compare other learning methods

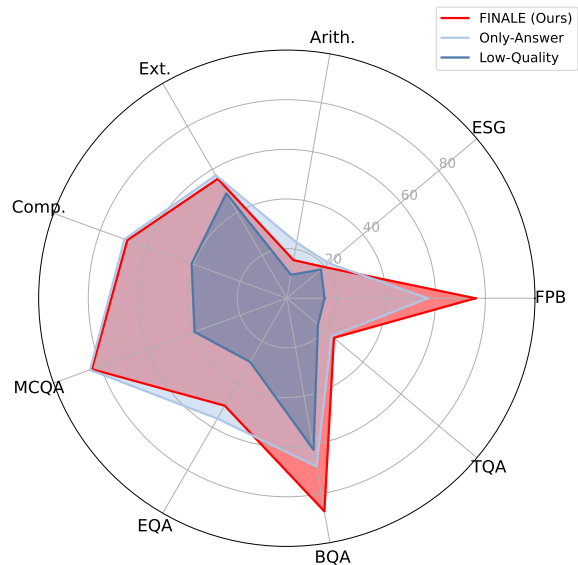


Figure 4: "FINALE" indicates training of the Yi-6B model with full data. "Only-Answer" indicates the results of training exclusively on answers, excluding rationales. "Low-Quality" indicates the results of training solely with data that has been removed through a filtering method.

Figure 4 compares the performance when using the same dataset as FINALE but excluding rationales and using only answers (light blue area in Figure 4) and when using data identified as low quality during the filtering process (blue area in Figure 4). The results show that FINALE significantly outperforms models trained solely on answers in the binary classification tasks of FPB and BQA. However, similar or significantly degraded performances are observed in other sub-tasks. This can be interpreted due to two factors.

Firstly, when the rationale length is excessively long, there is a tendency to generate new labels not specified in the task. For example, in the ESG sub-task, the average rationale length is 475, which is considerably longer than the average of around 100 in other tasks, leading to the generation of different answers, such as "economic" and "environmental" instead of the final gold answers like "indistinct," "opportunity" or "risk". The EQA sub-task, with an average rationale length of 486, also indiscriminately generated answers that differ from the gold

answer. This suggests that contrary to existing studies showing that reasoning steps improve model generate capabilities (Wei et al., 2023; Nye et al., 2021; Zhou et al., 2023b; Gao et al., 2023), excessively long rationales may hinder the model’s ability to generate the final gold answer.

Secondly, the rationale for the second numeric reasoning task emphasizes the need for specialized explanations. The rationales of FINALE tend to be short and concise, which do not sufficiently reflect the challenging nature of the numeric reasoning task. Therefore, numerical reasoning must consider the specialized rationale of arithmetic operations and the characteristics of the finance domain, suggesting that a sufficiently lengthy rationale is needed to explain the final answer adequately.

Moreover, models trained with datasets classified as low-quality (blue area in Figure 4) show significantly reduced performance across all sub-tasks, as evidenced by Figure 4. Despite the minimal difference in data quantity between Low-Quality and High-Quality datasets for most sub-tasks. In particular, in Extractive Question Answering (EQA), it is observed that performance significantly declines despite the notably large quantity of training data from low-quality cases. This emphasizes that rationales containing incorrect answers can substantially hinder model training. Therefore, the filtering process has a significant impact on enhancing the model’s performance.

5.2 Compare training data quantity

Table 3 presents a performance comparison based on the quantity of training data. Specifically, Single (per100) and Single (per400) represent results from models trained with 100 and 400 randomly sampled instances in each sub-task, respectively. Despite the very small amount of data, these models surpass the performance of the baseline Ko-Platy-6B. This indicates that the superior performance of FINALE is more than merely due to the large data size.

Furthermore, when comparing the results of training only on a single task (All) versus training on all tasks simultaneously multi-task (All), it is found that training in a multi-task (All) setting yields higher performance in all tasks except MCQA and Comp. This indicates that more extensive training on financial domain data simultaneously enhances performance across various tasks. Especially, FPB and ESG are classification tasks based on financial terminology. By training with other sub-tasks, additional financial knowledge is

acquired, which consequently has been observed to enhance performance.

5.3 Compare General Performance

Appendix F presents the effects of using the domain-specific dataset FINALE on general performance. Performance changes are observed using the Korean benchmark dataset KMMLU (Son et al., 2024), leading to the following key findings:

Models trained on Ko-Platy-6B and FINALE (Ours) exhibit lower performance compared to Yi-6B-Ko. This aligns with previous studies suggesting that Instruction Tuning can degrade general performance and negatively impact knowledge-based benchmarks (Son et al., 2024; Bi et al., 2024). However, despite being a finance domain instruction tuning dataset, FINALE shows a less performance decline than Ko-Platy-6B.

This can be analyzed for two reasons. Firstly, Ko-Platy-6B, derived from translated English datasets, may suffer from quality degradation due to translation errors (Xia et al., 2019; Riley et al., 2023; Yao et al., 2024). In contrast, FINALE is not a translated dataset and ensures high data quality through a filtering process that removes 54% of the original dataset. Secondly, while Ko-Platy-6B focuses on generating diverse instructions and answers, FINALE emphasizes training models on high-quality rationales before the final answers. These results underscore the importance of data quality management and stringent filtering processes, indicating that methodologies like instruction tuning that consider rationales are essential to minimize declines in general performance.

6 Conclusion

In this paper, we propose FINALE, an instruction-tuning dataset with high-quality rationales for the financial domain. Furthermore demonstrate that training the foundation model on FINALE enhances the generative capabilities of LLMs. Notably, the performance is approximately 9% better than the baseline. Human comprehension of the model-generated text is shown to be four times better. Additionally, despite using only a very small amount of data (100 or 400 instances), the performance exceeded that of the baseline, with minimal degradation in general performance. Through this study, we anticipate an increased recognition of the importance of rationales in the finance domain.

Limitations This research is conducted solely in Korean. However, our data generation method applies to all languages, leaving research in other languages as a future work. Additionally, we utilized Gemini-Pro for rationale generation due to the high API prices. However, by using more powerful models such as GPT-4, the quality of the rationales and performance can be further enhanced. Lastly, due to the lack of specific criteria for rationale quality in the financial domain, this study utilizes an automatic metric for filtering based on the final answers. Consequently, we plan future research to establish criteria for rationale quality suitable for the finance domain.

7 Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2022R1A2C2005455). This work was also supported by Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2021-0-00471, Development of Autonomous Control Technology for Error-Free Information Infrastructure Based on Modeling Optimization). We express our sincere gratitude to Yukyung Lee for her thoughtful advice. We also appreciate Jinwoo Park, Hanwool Lee, and Inyeol Choi for their invaluable feedback and evaluation. Additionally, we are thankful to the researchers at the DSBA lab for their comments on the paper.

References

- AiHUB. 2023a. Aihub; financial & legal machine reading comprehension dataset. <https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=&topMenu=&aihubDataSe=data&dataSetSn=71610>. Accessed: 2024-02-21.
- AiHUB. 2023b. Aihub; numerical machine reading comprehension dataset. <https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=data&dataSetSn=71568>. Accessed: 2024-02-21.
- AiHUB. 2023c. Aihub; tabular machine reading comprehension dataset. <https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=data&dataSetSn=71565>. Accessed: 2024-02-21.
- Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *Preprint*, arXiv:1908.10063.
- Zhijie Bao, Wei Chen, Shengze Xiao, Kuang Ren, Jiaao Wu, Cheng Zhong, Jiajie Peng, Xuanjing Huang, and Zhongyu Wei. 2023. *Disc-medllm: Bridging general large language models and real-world medical consultation*. *Preprint*, arXiv:2308.14346.
- Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qishi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.
- Jaeyoung Choe, Keonwoong Noh, Nayeon Kim, Seyun Ahn, and Woohwan Jung. 2023. *Exploring the impact of corpus diversity on financial pretrained language models*. *Preprint*, arXiv:2310.13312.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. *Bert: Pre-training of deep bidirectional transformers for language understanding*. *Preprint*, arXiv:1810.04805.
- Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. *Pal: Program-aided language models*. *Preprint*, arXiv:2211.10435.
- Fadi Sakka Hicham Sadok and Mohammed El Hadi El Maknoui. 2022. *Artificial intelligence and bank credit analysis: A review*. *Cogent Economics & Finance*, 10(1):2023262.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. *Lora: Low-rank adaptation of large language models*. *Preprint*, arXiv:2106.09685.
- Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023a. *The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning*. *Preprint*, arXiv:2305.14045.
- Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023b. *The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning*. *arXiv preprint arXiv:2305.14045*.
- Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. *Applying conditional random fields to Japanese morphological analysis*. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237, Barcelona, Spain. Association for Computational Linguistics.
- KyujinHan. *Kor-openorca-platypus-v3*. <https://huggingface.co/datasets/kyujinpy/KOR-OpenOrca-Platypus-v3>. Accessed: 2024-03-01.

- Moreno La Quatra and Luca Cagliero. 2020. End-to-end training for financial report summarization.
- Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint arXiv:2312.15685*.
- Ilya Loshchilov and Frank Hutter. 2019. **Decoupled weight decay regularization**. *Preprint*, arXiv:1711.05101.
- P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65.
- Eliza Mik. 2017. **Smart contracts: terminology, technical limitations and real world complexity**. *Law, Innovation and Technology*, 9:1–32.
- Branka Hadji Misheva and Joerg Osterrieder. 2023. **A hypothesis on good practices for ai-based systems for financial time series forecasting: Towards domain-driven xai methods**. *Preprint*, arXiv:2311.07513.
- Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*.
- Makoto Nakayama and Yun Wan. 2017. **Exploratory study on anchoring: Fake vote counts in consumer reviews affect judgments of information quality**. *Journal of theoretical and applied electronic commerce research*, 12:1–20.
- Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. 2021. **Show your work: Scratchpads for intermediate computation with language models**. *Preprint*, arXiv:2112.00114.
- OpenAI. 2022. Openai: Introducing chatgpt. <https://openai.com/blog/chatgpt>.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altmenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O’Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Ji-

- ayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. [Gpt-4 technical report](https://arxiv.org/abs/2303.08774). *Preprint*, arXiv:2303.08774.
- FinNLP Shared Task Organizers. Multi-lingual esg impact duration inference (ml-esg-3). <https://sites.google.com/nlg.csie.ntu.edu.tw/finnlp-kdf-2024/shared-task-ml-esg-3>. Accessed: 2024-02-21.
- Chanjun Park, Hwalsuk Lee, Hyunbyung Park, Hyeonwoo Kim, Sanghoon Kim, Seonghwan Cho, Sunghun Kim, and Sukyung Lee. 2023. Open ko-llm leaderboard. <https://huggingface.co/spaces/upstage/open-ko-llm-leaderboard>.
- Parker Riley, Timothy Dozat, Jan A. Botha, Xavier Garcia, Dan Garrette, Jason Riesa, Orhan Firat, and Noah Constant. 2023. [Frmt: A benchmark for few-shot region-aware machine translation](https://arxiv.org/abs/2210.00193). *Preprint*, arXiv:2210.00193.
- Corby Rosset, Chenyan Xiong, Minh Phan, Xia Song, Paul Bennett, and Saurabh Tiwary. 2021. [Knowledge-aware language model pretraining](https://arxiv.org/abs/2007.00655). *Preprint*, arXiv:2007.00655.
- Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. [Multi-task prompted training enables zero-shot task generalization](https://arxiv.org/abs/2110.08207). *Preprint*, arXiv:2110.08207.
- Guijin Son, Hanearl Jung, Moonjeong Hahm, Keonju Na, and Sol Jin. 2023. Beyond classification: Financial reasoning in state-of-the-art language models. *arXiv preprint arXiv:2305.01505*.
- Guijin Son, Hanwool Lee, Sungdong Kim, Seungone Kim, Niklas Muennighoff, Taekyoon Choi, Cheonbok Park, Kang Min Yoo, and Stella Biderman. 2024. [Kmmmlu: Measuring massive multitask language understanding in korean](https://arxiv.org/abs/2402.11548). *arXiv preprint arXiv:2402.11548*.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models*. <https://crfm.stanford.edu/2023/03/13/alpaca.html>, 3(6):7.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul R. Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, Alexandre Fréchet, Charlotte Smith, Laura Culp, Lev Proleev, Yi Luan, Xi Chen, James Lottes, Nathan Schucher, Federico Lebron, Alban Rustemi, Natalie Clay, Phil Crone, Tomas Kocisky, Jeffrey Zhao, Bartek Perz, Dian Yu, Heidi Howard, Adam Bloniarz, Jack W. Rae, Han Lu, Laurent Sifre, Marcello Maggioni, Fred Alcober, Dan Garrette, Megan Barnes, Shantanu Thakoor, Jacob Austin, Gabriel Barth-Maron, William Wong, Rishabh Joshi, Rahma Chaabouni, Deeni Fatiha, Arun Ahuja, Ruibo Liu, Yunxuan Li, Sarah Cogan, Jeremy Chen, Chao Jia, Chenjie Gu, Qiao Zhang, Jordan Grimstad, Ale Jakse Hartman, Martin Chadwick, Gaurav Singh Tomar, Xavier Garcia, Evan Senter, Emanuel Taropa, Thanumalayan Sankaranarayanan Pillai, Jacob Devlin, Michael Laskin, Diego de Las Casas, Dasha Valter, Connie Tao, Lorenzo Blanco, Adrià Puigdomènech Badia, David Reitter, Mianna Chen, Jenny Brennan, Clara Rivera, Sergey Brin, Shariq Iqbal, Gabriela Surita, Jane Labanowski, Abhi Rao, Stephanie Winkler, Emilio Parisotto, Yiming Gu, Kate Olszewska, Yujing Zhang, Ravi Addanki, Antoine Miech, Annie Louis, Laurent El Shafey, Denis Teplyashin, Geoff Brown, Elliot Catt, Nithya Attaluri, Jan Balaguer, Jackie Xi-ang, Pidong Wang, Zoe Ashwood, Anton Briukhov, Albert Webson, Sanjay Ganapathy, Smit Sanghavi, Ajay Kannan, Ming-Wei Chang, Axel Stjerngren, Josip Djolonga, Yuting Sun, Ankur Bapna, Matthew Aitchison, Pedram Pejman, Henryk Michalewski, Tianhe Yu, Cindy Wang, Juliette Love, Junwhan Ahn, Dawn Bloxwich, Kehang Han, Peter Humphreys, Thibault Sellam, James Bradbury, Varun Godbole, Sina Samangooei, Bogdan Damoc, Alex Kaskasoli, Sébastien M. R. Arnold, Vijay Vasudevan, Shubham Agrawal, Jason Riesa, Dmitry Lepikhin, Richard Tanburn, Srivatsan Srinivasan, Hyeontaek Lim, Sarah Hodkinson, Pranav Shyam, Johan Ferret, Steven Hand, Ankush Garg, Tom Le Paine, Jian Li, Yujia Li, Minh Giang, Alexander Neitz, Zaheer Abbas, Sarah York, Machel Reid, Elizabeth Cole, Aakanksha Chowdhery, Dipanjan Das, Dominika Rogozińska, Vitaly Nikolaev, Pablo Sprechmann, Zachary Nado, Lukas Zilka, Flavien Prost, Luheng He, Marianne Monteiro, Gaurav Mishra, Chris Welty, Josh Newlan,

Dawei Jia, Miltiadis Allamanis, Clara Huiyi Hu, Raoul de Liedekerke, Justin Gilmer, Carl Saroufim, Shruti Rijhwani, Shaobo Hou, Disha Shrivastava, Anirudh Baddepudi, Alex Goldin, Adnan Ozturk, Albin Cassirer, Yunhan Xu, Daniel Sohn, Devendra Sachan, Reinald Kim Amplayo, Craig Swanson, Dessie Petrova, Shashi Narayan, Arthur Guez, Siddhartha Brahma, Jessica Landon, Miteyan Patel, Ruizhe Zhao, Kevin Villela, Luyu Wang, Wenhao Jia, Matthew Rahtz, Mai Giménez, Legg Yeung, Hanzhao Lin, James Keeling, Petko Georgiev, Diana Mincu, Boxi Wu, Salem Haykal, Rachel Saputro, Kiran Vodrahalli, James Qin, Zeynep Cankara, Abhanshu Sharma, Nick Fernando, Will Hawkins, Behnam Neyshabur, Solomon Kim, Adrian Hutter, Priyanka Agrawal, Alex Castro-Ros, George van den Driessche, Tao Wang, Fan Yang, Shuo yin Chang, Paul Komarek, Ross McIlroy, Mario Lučić, Guodong Zhang, Wael Farhan, Michael Sharman, Paul Natsev, Paul Michel, Yong Cheng, Yamini Bansal, Siyuan Qiao, Kris Cao, Siamak Shakeri, Christina Butterfield, Justin Chung, Paul Kishan Rubenstein, Shivani Agrawal, Arthur Mensch, Kedar Soparkar, Karel Lenc, Timothy Chung, Aedan Pope, Loren Maggiore, Jackie Kay, Priya Jhakra, Shibo Wang, Joshua Maynez, Mary Phuong, Taylor Tobin, Andrea Tacchetti, Maja Trebacz, Kevin Robinson, Yash Katariya, Sebastian Riedel, Paige Bailey, Kefan Xiao, Nimesh Ghelani, Lora Aroyo, Ambrose Slone, Neil Houlsby, Xuehan Xiong, Zhen Yang, Elena Gribovskaya, Jonas Adler, Mateo Wirth, Lisa Lee, Music Li, Thais Kagohara, Jay Pavagadhi, Sophie Bridgers, Anna Bortsova, Sanjay Ghemawat, Zafarali Ahmed, Tianqi Liu, Richard Powell, Vijay Bolina, Mariko Iinuma, Polina Zablotskaia, James Besley, Da-Woon Chung, Timothy Dozat, Ramona Comanescu, Xiance Si, Jeremy Greer, Guolong Su, Martin Polacek, Raphaël Lopez Kaufman, Simon Tokumine, Hexiang Hu, Elena Buchatskaya, Yingjie Miao, Mohamed Elhawaty, Aditya Siddhant, Nenad Tomasev, Jinwei Xing, Christina Greer, Helen Miller, Shereen Ashraf, Aurko Roy, Zizhao Zhang, Ada Ma, Angelos Filos, Milos Besta, Rory Blevins, Ted Klimentko, Chih-Kuan Yeh, Soravit Changpinyo, Jiaqi Mu, Oscar Chang, Mantas Pajarskas, Carrie Muir, Vered Cohen, Charline Le Lan, Krishna Haridasan, Amit Marathe, Steven Hansen, Sholto Douglas, Rajkumar Samuel, Mingqiu Wang, Sophia Austin, Chang Lan, Jiepu Jiang, Justin Chiu, Jaime Alonso Lorenzo, Lars Lowe Sjösund, Sébastien Cevey, Zach Gleicher, Thi Avrahami, Anudhyan Boral, Hansa Srinivasan, Vittorio Selo, Rhys May, Konstantinos Aisopos, Léonard Hussenot, Livio Baldini Soares, Kate Baumli, Michael B. Chang, Adrià Recasens, Ben Caine, Alexander Pritzel, Filip Pavetic, Fabio Pardo, Anita Gergely, Justin Frye, Vinay Ramasesh, Dan Horgan, Kartikeya Badola, Nora Kassner, Subhrajit Roy, Ethan Dyer, Víctor Campos, Alex Tomala, Yunhao Tang, Dalia El Badawy, Elspeth White, Basil Mustafa, Oran Lang, Abhishek Jindal, Sharad Vikram, Zhitaogong, Sergi Caelles, Ross Hemsley, Gregory Thornton, Fangxiaoyu Feng, Wojciech Stokowiec, Ce Zheng, Phoebe Thacker, Çağlar Ünlü, Zhishuai Zhang, Mohammad Saleh,

James Svensson, Max Bileschi, Piyush Patil, Ankesh Anand, Roman Ring, Katerina Tsihlias, Arpi Vezer, Marco Selvi, Toby Shevlane, Mikel Rodriguez, Tom Kwiatkowski, Samira Daruki, Keran Rong, Allan Dafoe, Nicholas FitzGerald, Keren Gu-Lemberg, Mina Khan, Lisa Anne Hendricks, Marie Pellat, Vladimir Feinberg, James Cobon-Kerr, Tara Sainath, Maribeth Rauh, Sayed Hadi Hashemi, Richard Ives, Yana Hasson, YaGuang Li, Eric Noland, Yuan Cao, Nathan Byrd, Le Hou, Qingze Wang, Thibault Sottiaux, Michela Paganini, Jean-Baptiste Lespiau, Alexandre Moufarek, Samer Hassan, Kaushik Shivakumar, Joost van Amersfoort, Amol Mandhane, Pratik Joshi, Anirudh Goyal, Matthew Tung, Andrew Brock, Hannah Sheahan, Vedant Misra, Cheng Li, Nemanja Rakićević, Mostafa Dehghani, Fangyu Liu, Sid Mittal, Junhyuk Oh, Seb Noury, Eren Sezener, Fantine Huot, Matthew Lamm, Nicola De Cao, Charlie Chen, Gamaleldin Elsayed, Ed Chi, Mahdis Mahdiah, Ian Tenney, Nan Hua, Ivan Ptrychenko, Patrick Kane, Dylan Scandinaro, Rishub Jain, Jonathan Uesato, Romina Datta, Adam Sadovsky, Oskar Bunyan, Dominik Rabiej, Shimu Wu, John Zhang, Gautam Vasudevan, Edouard Leurent, Mahmoud Alnahlawi, Ionut Georgescu, Nan Wei, Ivy Zheng, Betty Chan, Pam G Rabinovitch, Piotr Stanczyk, Ye Zhang, David Steiner, Subhajt Naskar, Michael Azzam, Matthew Johnson, Adam Paszke, Chung-Cheng Chiu, Jaume Sanchez Elias, Afroz Mohiuddin, Faizan Muhammad, Jin Miao, Andrew Lee, Nino Vieillard, Sahitya Potluri, Jane Park, Elnaz Davoodi, Jiageng Zhang, Jeff Stanway, Drew Garmon, Abhijit Karmarkar, Zhe Dong, Jong Lee, Aviral Kumar, Luowei Zhou, Jonathan Evens, William Isaac, Zhe Chen, Johnson Jia, Anselm Levskaya, Zhenkai Zhu, Chris Gorgolewski, Peter Grabowski, Yu Mao, Alberto Magni, Kaisheng Yao, Javier Snaider, Norman Casagrande, Paul Suganthan, Evan Palmer, Geoffrey Irving, Edward Loper, Manaal Faruqui, Isha Arkatkar, Nanxin Chen, Izhak Shafran, Michael Fink, Alfonso Castaño, Irene Giannoumis, Wooyeol Kim, Mikołaj Rybiński, Ashwin Sreevatsa, Jennifer Prendki, David Soergel, Adrian Goedeckemeyer, Willi Gierke, Mohsen Jafari, Meenu Gaba, Jeremy Wiesner, Diana Gage Wright, Yawen Wei, Harsha Vashisht, Yana Kulizhskaya, Jay Hoover, Maigo Le, Lu Li, Chimezie Iwuanyanwu, Lu Liu, Kevin Ramirez, Andrey Khorlin, Albert Cui, Tian LIN, Marin Georgiev, Marcus Wu, Ricardo Aguilar, Keith Pallo, Abhishek Chakladar, Alena Repina, Xihui Wu, Tom van der Weide, Priya Ponnappalli, Caroline Kaplan, Jiri Simsa, Shuangfeng Li, Olivier Dousse, Fan Yang, Jeff Piper, Nathan Ie, Minnie Lui, Rama Pasumarthi, Nathan Lintz, Anitha Vijayakumar, Lam Nguyen Thiet, Daniel Andor, Pedro Valenzuela, Cosmin Paduraru, Daiyi Peng, Katherine Lee, Shuyuan Zhang, Somer Greene, Duc Dung Nguyen, Paula Kurylowicz, Sarmishta Velury, Sebastian Krause, Cassidy Hardin, Lucas Dixon, Lili Janzer, Kiam Choo, Ziqiang Feng, Biao Zhang, Achintya Singhal, Tejasi Latkar, Mingyang Zhang, Quoc Le, Elena Allica Abellan, Dayou Du, Dan McInnon, Natasha Antropova, Tolga Bolukbasi, Orgad Keller, David Reid, Daniel Finchelstein, Maria Abi

Raad, Remi Crocker, Peter Hawkins, Robert Dadashi, Colin Gaffney, Sid Lall, Ken Franko, Egor Filonov, Anna Bulanova, Rémi Leblond, Vikas Yadav, Shirley Chung, Harry Askham, Luis C. Cobo, Kelvin Xu, Felix Fischer, Jun Xu, Christina Sorokin, Chris Alberti, Chu-Cheng Lin, Colin Evans, Hao Zhou, Alek Dimitriev, Hannah Forbes, Dylan Banarse, Zora Tung, Jeremiah Liu, Mark Omernick, Colton Bishop, Chintu Kumar, Rachel Sterneck, Ryan Foley, Rohan Jain, Swaroop Mishra, Jiawei Xia, Taylor Bos, Geoffrey Cideron, Ehsan Amid, Francesco Piccinno, Xingyu Wang, Praseem Banzal, Petru Gurita, Hila Noga, Premal Shah, Daniel J. Mankowitz, Alex Polozov, Nate Kushman, Victoria Krakovna, Sasha Brown, MohammadHossein Bateni, Dennis Duan, Vlad Firoiu, Meghana Thotakuri, Tom Natan, Anhad Mohananey, Matthieu Geist, Sidharth Mudgal, Sertan Girgin, Hui Li, Jiayu Ye, Ofir Roval, Reiko Tojo, Michael Kwong, James Lee-Thorp, Christopher Yew, Quan Yuan, Sumit Bagri, Danila Sinopalnikov, Sabela Ramos, John Mellor, Abhishek Sharma, Aliaksei Severyn, Jonathan Lai, Kathy Wu, Heng-Tze Cheng, David Miller, Nicolas Sonnerat, Denis Vnukov, Rory Greig, Jennifer Beattie, Emily Cave-ness, Libin Bai, Julian Eisenschlos, Alex Korchemniy, Tomy Tsai, Mimi Jasarevic, Weize Kong, Phuong Dao, Zeyu Zheng, Frederick Liu, Fan Yang, Rui Zhu, Mark Geller, Tian Huey Teh, Jason Sanmiya, Evgeny Gladchenko, Nejc Trdin, Andrei Sozanschi, Daniel Toyama, Evan Rosen, Sasan Tavakkol, Linting Xue, Chen Elkind, Oliver Woodman, John Carpenter, George Papamakarios, Rupert Kemp, Sushant Kafle, Tanya Grunina, Rishika Sinha, Alice Talbert, Abhimanyu Goyal, Diane Wu, Denese Owusu-Afriyie, Cosmo Du, Chloe Thornton, Jordi Pont-Tuset, Pradyumna Narayana, Jing Li, Sabaer Fatehi, John Wieting, Omar Ajmeri, Benigno Uria, Tao Zhu, Yeongil Ko, Laura Knight, Amélie Héliou, Ning Niu, Shane Gu, Chenxi Pang, Dustin Tran, Yeqing Li, Nir Levine, Ariel Stolovich, Norbert Kalb, Rebecca Santamaria-Fernandez, Sonam Goenka, Wenny Yustalim, Robin Strudel, Ali Elqursh, Balaji Lakshminarayanan, Charlie Deck, Shyam Upadhyay, Hyo Lee, Mike Dusenberry, Zonglin Li, Xuezhi Wang, Kyle Levin, Raphael Hoffmann, Dan Holtmann-Rice, Olivier Bachem, Summer Yue, Sho Arora, Eric Malmi, Daniil Mirylenka, Qijun Tan, Christy Koh, Soheil Hassas Yeganeh, Siim Põder, Steven Zheng, Francesco Pongetti, Mukarram Tariq, Yanhua Sun, Lucian Ionita, Mojtaba Seyedhosseini, Pouya Tafti, Ragha Kotikalapudi, Zhiyu Liu, Anmol Gulati, Jasmine Liu, Xinyu Ye, Bart Chrzaszcz, Lily Wang, Nikhil Sethi, Tianrun Li, Ben Brown, Shreya Singh, Wei Fan, Aaron Parisi, Joe Stanton, Chenkai Kuang, Vinod Koverkathu, Christopher A. Choquette-Choo, Yunjie Li, TJ Lu, Abe Ittycheriah, Prakash Shroff, Pei Sun, Mani Varadarajan, Sanaz Bahargam, Rob Willoughby, David Gaddy, Ishita Dasgupta, Guillaume Desjardins, Marco Cornero, Brona Robenek, Bhavishya Mittal, Ben Albrecht, Ashish Shenoy, Fedor Moiseev, Henrik Jacobsson, Alireza Ghaffarkhah, Morgane Rivière, Alanna Walton, Clément Crepey, Alicia Parrish, Yuan Liu, Zongwei Zhou, Clement Farabet, Carey Radebaugh, Praveen

Srinivasan, Claudia van der Salm, Andreas Fidje-land, Salvatore Scellato, Eri Latorre-Chimoto, Hanna Klimczak-Plucińska, David Bridson, Dario de Cesare, Tom Hudson, Piermaria Mendolicchio, Lexi Walker, Alex Morris, Ivo Penchev, Matthew Mauger, Alexey Guseynov, Alison Reid, Seth Odoom, Lucia Loher, Victor Cotruta, Madhavi Yenugula, Dominik Grewe, Anastasia Petrushkina, Tom Duerig, Antonio Sanchez, Steve Yadlowsky, Amy Shen, Amir Globerson, Adam Kurzrok, Lynette Webb, Sahil Dua, Dong Li, Preethi Lahoti, Surya Bhupatiraju, Dan Hurt, Haroon Qureshi, Ananth Agarwal, Tomer Shani, Matan Eyal, Anuj Khare, Shreyas Rammohan Belle, Lei Wang, Chetan Tekur, Mihir Sanjay Kale, Jinliang Wei, Ruoxin Sang, Brennan Saeta, Tyler Liechty, Yi Sun, Yao Zhao, Stephan Lee, Pandu Nayak, Doug Fritz, Manish Reddy Vuyyuru, John Aslanides, Nidhi Vyas, Martin Wicke, Xiao Ma, Taylan Bilal, Evgenii Eltyshev, Daniel Balle, Nina Martin, Hardie Cate, James Manyika, Keyvan Amiri, Yelin Kim, Xi Xiong, Kai Kang, Florian Luisier, Nilesch Tripuraneni, David Madras, Mandy Guo, Austin Waters, Oliver Wang, Joshua Ainslie, Jason Baldridge, Han Zhang, Garima Pruthi, Jakob Bauer, Feng Yang, Riham Mansour, Jason Gelman, Yang Xu, George Polovets, Ji Liu, Honglong Cai, Warren Chen, Xi-angHai Sheng, Emily Xue, Sherjil Ozair, Adams Yu, Christof Angermueller, Xiaowei Li, Weiren Wang, Julia Wiesinger, Emmanouil Koukoumidis, Yuan Tian, Anand Iyer, Madhu Gurumurthy, Mark Goldenson, Parashar Shah, MK Blake, Hongkun Yu, Anthony Urbanowicz, Jennimaria Palomaki, Chrisantha Fernando, Kevin Brooks, Ken Durden, Harsh Mehta, Nikola Momchev, Elahe Rahimtoroghi, Maria Georgaki, Amit Raul, Sebastian Ruder, Morgan Redshaw, Jinhyuk Lee, Komal Jalan, Dinghua Li, Ginger Perng, Blake Hechtman, Parker Schuh, Milad Nasr, Mia Chen, Kieran Milan, Vladimir Mikulik, Trevor Strohman, Juliana Franco, Tim Green, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. 2023. [Gemini: A family of highly capable multimodal models](#). *Preprint*, arXiv:2312.11805.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#). *Preprint*, arXiv:2302.13971.

Neng Wang, Hongyang Yang, and Christina Dan Wang. 2023. [Fingpt: Instruction tuning benchmark for open-source large language models in financial datasets](#). *Preprint*, arXiv:2310.04793.

Yizhong Wang, Swaroop Mishra, Pegah Alipoor-molabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krma Doshi, Maitreya Patel, Kuntal Kumar Pal, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza,

- Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddhartha Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. [Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks](#). *Preprint*, arXiv:2204.07705.
- Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. [Finetuned language models are zero-shot learners](#). *Preprint*, arXiv:2109.01652.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. [Chain-of-thought prompting elicits reasoning in large language models](#). *Preprint*, arXiv:2201.11903.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kam-badur, David Rosenberg, and Gideon Mann. 2023a. [Bloomberggpt: A large language model for finance](#). *Preprint*, arXiv:2303.17564.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kam-badur, David Rosenberg, and Gideon Mann. 2023b. [Bloomberggpt: A large language model for finance](#). *arXiv preprint arXiv:2303.17564*.
- Mengzhou Xia, Xiang Kong, Antonios Anastasopoulos, and Graham Neubig. 2019. [Generalized data augmentation for low-resource translation](#). *Preprint*, arXiv:1906.03785.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. [Pixiu: A large language model, instruction data and evaluation benchmark for finance](#). *Preprint*, arXiv:2306.05443.
- Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. [Wizardlm: Empowering large language models to follow complex instructions](#). *Preprint*, arXiv:2304.12244.
- Xianjun Yang, Wei Cheng, Xujiang Zhao, Wenchao Yu, Linda Petzold, and Haifeng Chen. 2023a. [Dynamic prompting: A unified framework for prompt tuning](#). *Preprint*, arXiv:2303.02909.
- Yi Yang, Yixuan Tang, and Kar Yan Tam. 2023b. [Investlm: A large language model for investment using financial domain instruction tuning](#). *Preprint*, arXiv:2309.13064.
- Binwei Yao, Ming Jiang, Diyi Yang, and Junjie Hu. 2024. [Benchmarking llm-based machine translation on cultural awareness](#). *Preprint*, arXiv:2305.14328.
- Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and Zhongyu Wei. 2023a. [Disc-lawllm: Fine-tuning large language models for intelligent legal services](#). *Preprint*, arXiv:2309.11325.
- Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhui Chen. 2023b. [Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi](#). *Preprint*, arXiv:2311.16502.
- Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyang Luo, and Yongqiang Ma. 2024. [Llamafactory: Unified efficient fine-tuning of 100+ language models](#). *arXiv preprint arXiv:2403.13372*.
- Chunting Zhou, Pengfei Liu, Puxin Xu, Srinu Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023a. [Lima: Less is more for alignment](#). *Preprint*, arXiv:2305.11206.
- Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023b. [Least-to-most prompting enables complex reasoning in large language models](#). *Preprint*, arXiv:2205.10625.

A Rationale Generation Prompt

Below are examples of the prompts we used to generate a seed rationale for two sub-tasks: the sentiment analysis task, which receives only one sentence as input, and the multiple-choice question answering task, which receives context and a question. These prompts aim to instruct the model to generate appropriate rationales for each sub-task.

Instruction: Please classify the sentiment of the sentence as positive, negative, or neutral, and explain the reasons step by step in Korean. Finish your answer in the following format. "Therefore, the answer is X."

Input: During the past decade, POSCO has gradually divested noncore assets and bought several sports equipment makers, including California-based Fitness Products International and Sparks, Nevada-based ATEC, a leading maker of baseball and softball pitching machines.

Response:

Figure 5: Sentiment Analysis Prompt used in Rationale Generation.

Instruction: Please answer the question based on the context, and explain the reasons step by step in Korean. Finish your answer in the following format. "Therefore, the answer is X."

Context: During the first quarter, capital investment was surveyed to have increased slightly compared to the fourth quarter of the previous year. The semiconductor sector saw an increase due to the import of manufacturing equipment such as EUV (Extreme Ultraviolet) lithography equipment. Additionally, steel saw a slight increase due to facility maintenance during the Lunar New Year holidays, and machinery and equipment due to the anticipation of economic recovery. On the other hand, automobiles, displays, petrochemicals, and refining showed levels similar to the fourth quarter of the previous year. The monitoring results indicate that capital investment is expected to increase slightly in the future. Semiconductors are expected to see a slight increase due to the introduction of new production equipment, and petrochemicals and refining due to the gradual recovery in demand for petroleum products. Meanwhile, automobiles, displays, steel, and machinery and equipment are expected to maintain levels similar to the first quarter.

Question: What is expected to slightly increase in capital investment due to the anticipated gradual recovery in demand for petroleum products according to monitoring results?

Options: A. Steel B. Petrochemicals/Refining C. Automobiles D. Displays

Response:

Figure 6: Multi-Choice Question Answering Prompt used in Rationale Generation.

B Rationale Generation In-Context Example Prompt

Below is the prompt for a sentiment analysis task for rationale generation. The purpose of the prompt is to generate a high-quality rationale by providing five in-context examples.

Instruction: Please classify the sentiment of the sentence as positive, negative, or neutral, and explain the reasons step by step in Korean. Finish your answer in the following format. "Therefore, the answer is X."

Input: During the past decade, POSCO has gradually divested noncore assets and bought several sports equipment makers, including California-based Fitness Products International and Sparks, Nevada-based ATEC, a leading maker of baseball and softball pitching machines.

Response: This sentence describes how POSCO has disposed of non-core assets over the past decade and acquired a major sports equipment manufacturer. These strategic actions reflect the strengthening of the company's portfolio and focus on its business, receiving positive evaluations. Therefore, the answer is positive.

Input: Net sales of Finnish Sanoma Learning & Literature, of Finnish media group Sanoma, decreased by 3.6% in January-June 2009 totalling EUR 162.8 mn, down from EUR 168.8 mn in the corresponding period in 2008.

Response: The decrease in sales can be considered negative because it is generally interpreted as a negative indicator for a company. A decline in sales can suggest a decrease in profitability, market share, and economic soundness, which can cause concern among relevant stakeholders. This sentence is classified as having a negative sentiment because the decrease in sales is perceived as an event that can negatively affect the performance of a company. Therefore, the answer is negative.

Input: Vaisala Oyj Stock exchange release 26.03.2010 klo 09.00 1 (1) The Annual General Meeting of Vaisala Oyj decided on March 25, 2010 to approve the Company's annual accounts for 2009.

Response: This sentence reports on the approval of the annual accounts for 2009 at the annual general meeting of Vaisala Oyj. This reporting of facts is intended to convey information rather than evoke emotions. It does not explicitly contain elements that are emotionally positive or negative, but simply provides information about the event of the company's annual accounts approval. Therefore, classifying this sentence as neutral is due to the content not being biased towards any specific emotion and being faithful to the purpose of providing information. Therefore, the answer is neutral.

Input: 26 October 2010 - Finnish environmental management company Lassila & Tikanoja Oyj (HEL: LAT1V), or L&T, said today its net profit declined to EUR 20.9 m for the first nine months of 2010 from EUR 27.6 m for the same period a year earlier.

Response: A decrease in net profit generally indicates a deterioration in corporate performance, so such news can lead to a negative evaluation of the company. Therefore, the reason for the negative classification is that the decrease in net profit represents concerns about the financial soundness of the company. Therefore, the answer is negative.

Input: Finnish sports equipment maker Amer Sports Oyj (HEL: AMEAS) said today that its net loss narrowed to EUR 16.9 million (USD 22.3m) in the second quarter of 2010 from EUR 23.2 million in the corresponding period a year earlier.

Response: The reason this statement is classified as having a positive sentiment is because the decrease in net loss signifies an improvement in the company's financial performance, which can send a positive signal to investors and the market. Therefore, the answer is positive.

Input: Over the course of 6 months, operating profit and net sales increased from 0.4 million euros to 3.2 million euros compared to the same period in 2005.

Response:

Figure 7: Sentiment Analysis Prompt used in Rationale Generation.

C Training Method and Hyper parameters

We conducted fine-tuning on the Yi-Ko-6B model using a single A100 80GB GPU. For training, we selected the AdamW (Loshchilov and Hutter, 2019) optimizer along with a linear learning rate scheduler, setting the learning rate at 3e-4 and a batch size of 16 with a gradient accumulation of 4. Additionally, to optimize the training process, LLaMA-Factory (Zheng et al., 2024) was utilized as the code base during fine-tuning.

Hyperparameter	Value
Model	Yi-6B-Ko
Number of epochs	3
Devices	1 Tesla A100 GPU (80GB)
Learning rate	3e-4
Scheduler	Linear
Batch size	16
Input max length	2,048
Output max length	512
Temperature	0.0

Table 4: Hyperparameters used for training and inference.

D Evaluation In-Context Example Prompt

Instruction: Please classify the sentiment of the sentence as positive, negative, or neutral.

Input: Over a three-month period, both operating profit and net sales increased to 160,000 euros and 1.39 million euros respectively, compared to the same quarter in 2006.

Response: Positive tone: The use of the word "increased" suggests that the company's performance is improving, which conveys a positive meaning. Such increases imply the company's profitability and potential for growth. Positive outcome: Increases in operating profit and net sales generally indicate good financial performance and overall health of a company. This can elicit positive responses from investors and market participants. Taking these factors into consideration, this sentence contains positive content that reflects the improvement and financial growth of the company. Therefore, the answer is positive.

Input: In Finland, sales decreased by 10.5% in January, while international sales fell by 17%.

Response: Mention of sales decrease: The sentence states, "In Finland, sales decreased by 10.5% in January, while international sales fell by 17%." A decrease in sales can negatively impact a company's revenue and prospects for growth. Deterioration in market conditions: The decrease in sales can occur due to a worsening of market conditions, increased competition, or internal problems within the company. These factors can negatively affect the future growth and profitability of the company. Therefore, the answer is negative.

Input: According to Gran, while the company is growing, there are no plans to move all production to Russia.

Response: The sentence conveys information about whether the company will relocate its production to Russia but does not express a clear opinion or emotion. This sentence is interpreted as neutral since it lacks a positive or negative tone. Therefore, the answer is neutral.

Input: {sentence}

Response:

Figure 8: Sentiment Analysis Prompt used in evaluation.

E Human annotation guideline

Read the question and the context, and choose which of the two rationales given is more helpful in reasoning the correct answer. Your choice should not be based on whether you got the answer right but on whether the rationale helps you make an inference even if you got the answer wrong.

Rationale 1 -> 1 in the selection box
Rationale 2 -> 2 in the selection box
Neither rationale is helpful -> 0 in the selection box

Context: {context}
Question: {question}
Answer: {answer}

Rationale 1: {*model_A* rationale}
Rationale 2: {*model_B* rationale}

Selection:

Figure 9: Human Annotation Guideline.

F Compare Generation Performance

Models	HUMSS	STEM	Applied Science	Other	Total
Yi-6B-Ko	39.76	40.49	39.51	41.62	40.33
Ko-PlatYi-6B	39.13	36.94	37.21	39.00	38.05
FINALE (Ours)	39.71	40.14	38.96	40.36	39.77

Table 5: The result of the General Performance comparison using the KMMLU dataset

Capturing Analysts’ Questioning Strategies in Earnings Calls via a Question Cornering Score (QCS)

Giulia D’Agostino, Andrea Rocci
IALS
Università della Svizzera italiana
Switzerland
{name.surname}@usi.ch

Chris Reed
Centre for Argument Technology
University of Dundee
Scotland, UK
c.a.reed@dundee.ac.uk

Abstract

The study of questions in the setting of dialogical interactions in corporate communication has the purpose of understanding and capitalizing on the opinions that the questioner has with respect to the questioned company. Particularly, financial analysts have the maximal incentive to be right in their forecasts about the company’s performance, but they are also incentivized and expected to maintain a good relationship with the management – and therefore, not to be too challenging in their questions. While avoiding overt adversarialness, analysts adopt alternative strategies to seek the desired information; among which modulating the *cornering* quality of questions. This paper presents a way of measuring such cornering property, automatically extracting feature scores, and comparing the results with a manually annotated gold standard. Results encourage further research along this stream, particularly towards the study of replies and their degree of *answerhood* with respect to the cornering quality of the prompting question.

1 Introduction

Multiple strategies can be put in place to make a question more effective, that is, to enhance its likelihood of eliciting a reply maximally compliant with the type of answer that the questioner wished to obtain.

In political press conferences (henceforth: PPCs) the questioner role is held by journalists. In this context, questioners have the incentive of being on the side *opposite* to the politician they are interviewing, if need be; this because the institutional role of journalists is to gain information *to the benefit of the public*. An effective question, in this environment, may feature an openly confrontational stance towards the interlocutor in order to obtain the sought after information and/or hold the interlocutor accountable. Therefore, adversarialness (Clayman and Heritage, 2002; Clayman et al.,

2007) is a measure that captures the confrontational attitude with which journalists load their questions.

In earnings conference calls (henceforth: ECCs), on the other hand, questioners are financial analysts. Whilst representing the interests of investors they also need to maintain a good relationship with the interlocutor (Palmieri et al., 2015; Koller and Wu, 2023). Therefore, to model effective questions analysts need to concoct tactics that are functionally equivalent to adversarialness but rooted in distinct mechanisms, since their role discourages them from showing any confrontational attitude; such tactics modulate what we call the *cornering* quality of questions. The idea is that “to corner” means to constrain the set of possible moves and to limit the ability to perform evasive maneuvers; thus, cornering questions constrain what counts as a valid answer and are difficult to evade.

The present contribution proposes a metric for the evaluation of the degree of such a cornering attitude in ECCs; we call it Question Cornering Score (QCS). A baseline QCS is calculated on the basis of manually annotated features, and later GPT-4 performance in the scoring of the same features is tested against the baseline. We conduct a case-study, comparing the variation of call-QCS in the annotated sample with financial data describing the evolution of the corresponding stock price, as well as forecasted and actual earnings per share (EPS), their delta and their surprise component for each financial quarter. Results show that the score has a correspondence in the likely stance that the questioner developed towards the issuing corporation due to event-external relevant factors. Encouraged from these results, we finally present some future goals that extend from the (automatic) assessment of the QCS towards the estimation of the answerhood degree of the corresponding replies.

Extracting the cornering degree of questions, therefore, represents a fundamental advancement towards a systematic study of Q&A pairs, captur-

ing answerhood and cooperation vs. evasion in corporate-side answers – which will arguably produce insights into corporate performance and market reaction.

1.1 Earnings Conference Calls

ECCs are quarterly public events with a formulaic and predefined structure, during which companies present financial results of the previous quarter, explain the current situation and share an outlook for the following quarters (Crawford Camiciottoli, 2010). They are voluntary events which are however held by the vast majority of listed companies, since they represent one of the few public interactive moments between corporate representatives and financial analysts, among the investor relations activities (Rocci and Raimondo, 2017). The significance of such an activity is widely recognized from both sides, and this is testified by the invariable attendance of high-level managers, typically even the CEO and/or the CFO.

ECCs comprise a Question-and-Answer (Q&A) session, during which analysts can seek to elicit as much contextual information around the disclosed results as it is possible to obtain without creating the obligation for the company to provide a supplementary disclosure of *material information* (Clark, 2021). To do so, questioners apply multiple rhetorical strategies to get the right understanding about whether the company is really worth investing in, and later communicate their valuation and recommendation in reports made available to investors (Palmieri et al., 2015). The tone, content and interaction of Q&A sessions have already been shown to have an impact on stock price (Chen et al., 2018), particularly due to analyst intervention.

Analysts' questions in ECCs, which make the object of this contribution, are therefore powerful carriers of information. Additionally, the measure here developed and its analysis would be applicable to comparable Q&A interaction schemes, both in the financial domain (e.g., interviews with the top management) and outside it (e.g., press conferences).

2 Theoretical background

2.1 NLP for finance

The use of Natural Language Processing (NLP) techniques in finance is mostly of the text-mining kind (Kumar and Ravi, 2016) and caters for the needs of both investors and traders, and of the

firm's Investor Relations. NLP is employed to extract information about what is explicitly stated in documents, disclosures or exchanges, or to explore the implicit content that lies behind the statements – whether sentiment, opinions, or argumentation.

FinTech applications mostly deal with explicit content and are typically developed for (or from) the corporate side, to enhance the effectiveness of the communication with clients or investors (see Chen et al., 2020).

The mining of implicit content, on the other hand, is more commonly a domain that potentially helps clients or investors making informed decisions. The goal in this case is to acquire insights about the past performance and inferentially predict the future course of a company. Results drawn from such studies could arguably be exploited by companies as well, to check the soundness of their current approach to investors communication and possibly improve it. Techniques of this kind traditionally involve the assessment of the sentiment (see Kearney and Liu, 2014 for a review), but notably also include opinion mining and argument mining (Garcia Villalba and Saint-Dizier, 2012; Liu, 2012; Chen et al., 2021), which deal with inferential connections between (often material) premises and (often evaluative) conclusions.

NLP applied to ECCs has either a descriptive or a predictive approach. Description is aimed at the retrieval of certain trends or patterns (Davis et al., 2015; Rocci et al., 2019), possibly correlating them with financial data; not necessarily confined to *text* analysis (Chen et al., 2023). Prediction is forward-oriented, prognosticating for instance post-event analysts' recommendations on the basis of questions formulation and answers tone (Keith and Stent, 2019; Paziienza et al., 2020).

The current contribution is text-based and primarily descriptive, but as part of a planned pipeline including answerhood evaluation and argument mining it has the potential to feature in a range of NLP application for companies and investors, including those aimed at forecasting.

2.2 Adversarialness in political press conferences

Structurally similar to ECCs, PPCs are a field in which descriptive research on question design and questioning strategies flourished for years (Heritage, 2003; Clayman et al., 2006, 2012; Heritage and Clayman, 2013). One development from which this study draws inspiration is the study and mea-

sure of the *adversarialness* of journalists' questions in PPCs (Clayman and Heritage, 2002; Clayman et al., 2007). For ECCs, question reformulations and their relative adversarial strength has been investigated only qualitatively by de Oliveira and Pereira (2017).

The original proponents of the adversarialness measure were primarily data-driven in the decomposition of the concept into relevant features and the scores attributed to each of them. The way of computing the total, however, was convoluted and opaque to the reader.

The property and score that are proposed in the current contribution differ from the concept of adversarialness in the following regards:

- a. The property is not a characteristic of a single question but of a wider textual unit called MIU, presented below in §2.3
- b. The property does not describe how hostile a question is, but how much difficult (i.e., reputationally costly) it is to evade a proper answer to the question
- c. The score computed to evaluate the property derives from the plain sum of the scores attributed to its constituent features

2.3 Text segmentation: Maximal Interrogative Units

In ECCs there is a conventional limit on the number of turns an analyst is granted before ceding the floor to the next questioner. Analysts typically have a number of issues they aim to resolve and a number of questions to ask, and so construct individual turns such that they introduce multiple questions which in another activity type, such as spontaneous informal conversation, could be spread out over a series of shorter turns (D'Agostino et al., 2024b).

These multi-issue question turns are segmented by speakers into topically homogeneous sequences of utterances, called Maximal Interrogative Units (MIUs): questioning units typically below the level of the turn, but above the level of the clause or individual speech act. ECC speakers at times make explicit reference to MIU segmentation; a case of this is illustrated by Example (1), further discussed below, in which the speaker, analyst Jeremy Sigeo, explicitly and repeatedly marks two sections of their turn as forming a first and second 'question'.

- (1) Morning. Thank you very much. Apologies for taking on the painful bits, but I still

think there's more clarification that we need.
I wanted to just ask two things.

{**One** is on Greensill. [You've got about CHF5 billion cash, but also about CHF5 billion remaining exposure in those funds.]}*preface1* [And I just wondered if you could put a number on how much of that CHF5 billion remaining exposure is to doubtful borrowers, including, obviously, Gupta, but also some of the other doubtful borrowers who seem reluctant to pay.]}*question 1* **So, that's my first question.**}*MIU 1*

{**And my second question** is on the other painful, like I said, I'm afraid, on the Archegos situation. [Could you walk us through the mechanics of how that loss came about in terms of what the outstanding gross exposure was at the moment of problem?]}*question 2* [How much margin you had and the sequence of events in terms of, were you slow to sell down or how do you assess what happened?]}*question 3*}*MIU 2*

Those are my two questions please.

In Example (1), the speaker emphasises the fact that they are asking two 'questions' both at the beginning of the turn (before the first MIU, "I wanted to just ask two things") and at the end of the turn (after the second MIU, "Those are my two questions please"). The closing remark, moreover, also plays the role of concluding the turn, leaving the stage to the management for a reply.

Also the "So, that's my first question" remark that concludes the first MIU engages in a similar double purpose: it both reiterates the enumeration of 'questions' and declares the conclusion of the first questioning act.

Finally, both MIUs are introduced by a discourse marker ("One", "And my second question") that serves the purpose of counting the progression of 'questions'.

3 Question cornering score

The core contribution of the current work is of theory development, paired with an exploratory study on the application of such a theory to the context of ECCs and the automatic replication of the measure that the theory proposes. The theoretical construct

presented here is the *cornering* property of an MIU and the score (QCS) that is assigned to the MIU on the basis of six discrete, topic independent features that are selected as realistic means of estimating such a property.

An MIU is evaluated to be cornering the more it raises the cost to which the respondent is exposed for not answering properly and fully to it. This means that the higher the cornering score of an MIU performed by an analyst is, the heavier the burden of compliance with cooperation that is cast upon the management is. The cost associated with uncooperativeness can typically be a decreased perception of accountability, reliability, and ultimately value associated with the management and, subsequently, the company overall.

Cooperation is here to be intended in terms of a high degree of answerhood: how much the reply approximates the *principal possible answer*, i.e., can be regarded as logically sufficient and immediate with respect to the prompting question (Wiśniewski, 2015).

The QCS sums up the scoring of six independent features that describe the relevant structural properties of MIUs with respect to their ability to shape such a constriction:

$$QCS|_{MIU} = \sum_{i \in \text{features}} (score)_i$$

where $i = 1, \dots, 6$ are the features under consideration.

The following cut-off criterion was adopted for the scoring: for each feature, the MIU gets assigned the score of the highest-ranking type that is contains, independently from how many tokens it contains.

Following, an overview of the six features and their individual scores.

(1) Framing preface → score {0, 2}

This feature tracks the presence of a prefatory statement (Lucchini et al., 2022). If not present, the score is 0. In case there is at least one preface introducing the question(s): score 1 if the statement is neutral or positive; score 2 if it is negative.

(2) Complexity → score {0, 2}

Complexity counts the number of questions in the MIU. It assigns 0 if one question is present; 1 if questions are 2 or 3; 2 if there are 4 questions or more.

(3) Directness → score {0, 1}

This feature recognizes whether the MIU contains elements of indirectness (score 0) or whether questions are formulated in a direct way (score 1). Indirectness is both related to hedging and the modality of a question, i.e., formulations such as “*I would be glad to hear something*”, but also “*Could you say something*” (see Crawford Camiciottoli, 2009) would be both assigned score 0.

(4) Assertiveness → score {0, 1}

The assertiveness of questions relates to their formulation. Assuming that each question can be formulated as open or closed, 0 is given if all questions in the MIU are open; 1 if at least one question of the MIU is closed.

(5) Request type → score {0, 2}

This feature depends on a two-step annotation of each question. First, each question is attributed a certain request type – according to a speech-act typology described and operationalized in Lucchini and D’Agostino (2023). Based on that, the score is assigned to the highest-ranking request in the MIU, according to the following scheme:

- score 0 to requests for elaboration or data
- score 1 to requests for opinion, explanation, clarification or of confirmation of some material data
- score 2 to requests for justification, commitment, or the confirmation of an inference

(6) Time orientation → score {0, 2}

The time orientation score is 0 if the topic of the question(s) is placed in the present; 1 if in the future; 2 if in the past.

The final cornering score is the sum of the individual scores assigned to an MIU and lies in the range {0, 10}. An MIU is considered to be (increasingly) ‘cornering’ if its QCS is equal to or higher than 5. Concrete examples showing the application of this scoring are presented in Appendix A.

Beside plain QCS, attributed to MIUs, we name call-QCS the sum of all QCS values of a call. Call-QCS is defined as follows over the $j = 1, \dots, n$ MIUs of a call:

$$\text{call}QCS = \sum_{j \in \text{MIUs}} (QCS)_j$$

Call-QCS is not weighted with the length of the

call; this means that it is not calculated considering the number of MIUs per call as a biasing factor. On the contrary, the number of MIUs is acknowledged to be an underlying additional factor that determines the cornering nature of the call.

4 Research questions

Two research questions are addressed in this study:

- RQ1. Does the QCS reflect an inquisitive attitude of the speaker, motivated by noteworthy external factors?
- RQ2. Is the QCS a measure that can reliably be reproduced by AI tools?

RQ1 will be answered by comparing call-QCS with financial data such as the estimated performance of the company and its actual results over time. RQ2 will be answered by measuring the agreement rate between manual and automatic scoring.

The hypotheses against which the results will be tested are:

- H1. The QCS correlates with financial results and news that have a clear impact on such results; particularly, the call-QCS is expected to be higher, the more challenging and potentially disrupting the situation is for the company (and vice-versa).
- H2. The measure can be assessed reliably by LLMs insofar as it is decomposed into constitutive features. Some features are harder to score than others.

5 Data and method

The dataset for the current study are the four ECCs held in 2021 by the Swiss bank *Credit Suisse* (CS), for a total of 111 MIUs (483 sentences; 9,853 words; language: English). CS was chosen as a case study because of the poor performance and the sequence of critical issues that the company faced in 2021. The most remarkable features taken into consideration are:

- CS steadily reported losses along the whole financial year
- although a certain variability in CS stock prices can be traced, their value drops around each ECC, with an overall decline over the fiscal year

- CS incurred in at least two major scandals during the period considered (financial and reputational crises); namely the Archegos-Greensill double bankruptcy and the “tuna bonds” fraudulent issuance

In a precarious environment, financial analysts are expected to ask questions that are straight to the point. As a consequence, a study about the cornering degree of questions to CS representatives in 2021 seemed an ideal environment to start testing the soundness of the score. Following, a sketch of the methodological approach.

Setup The first step is the segmentation of question turns into MIUs. This is currently performed manually to ensure precision. The measured inter-annotator reliability for this task (Krippendorff’s alpha (1995) for the unitizing of textual continua among three annotators) is $U\alpha = 0.933$ (see D’Agostino et al., 2024a for further details).

Manual assessment To collect manually annotated data, four trained annotators¹ are instructed to manually score all the 111 pre-segmented MIUs for each feature. Krippendorff’s alpha coefficient for nominal data is the measure employed to evaluate their annotation agreement. Over the single features, the agreement rate ranges from $\alpha = 0.38$ (Request type) to $\alpha = 0.84$ (Assertiveness), as shown in Table 1. Better scores are consistently measured considering only annotators A and B; the remaining two (C and D in Table 1) introduce excessive outliers. Therefore, only the scores by annotators A and B will be used hereafter. The agreement rate among the two best annotators for the QCS (the sum of the single scores) is $\alpha = 0.57$.

Baseline values are calculated as the statistical mode of the manual assessment by the two best annotators. They are determined both at feature (*baseline feature score*) and at the QCS level (*baseline QCS*).

RQ1 The call-QCS is calculated as the sum of the baseline QCS values of a call.

¹Annotators are student assistants, employed with a part-time contract by the project that funds the current contribution. They are second-year Master’s students in investor relations with a background in languages/linguistics. Their tasks include, but are not limited to, data annotation and the assessment of the current score. For any task, their training is carried jointly by the two PhD students who work on the project. The annotation guidelines for this task that were provided to the annotators are those presented in §3.

Feature	α (4 annotators)	α (A + B)	α (A + B + C)	α (A + B + D)	α (C + D)
Framing preface	0.39	0.58	0.43	0.40	0.46
Complexity	0.75	0.99	0.86	0.75	0.64
Directness	0.49	0.93	0.68	0.47	0.38
Assertiveness	0.84	1.00	0.88	0.84	0.90
Request type	0.38	0.89	0.46	0.53	0.29
Time orientation	0.61	0.89	0.86	0.55	0.37
QCS	0.27	0.57	0.30	0.36	0.19

Table 1: Inter-rater reliability for the manual annotation of the QCS and its constitutive features, measured as Krippendorff’s alpha (nominal)

The financial data and relevant news are retrieved from the Bloomberg terminal. These are qualitatively compared with call-QCS measures.

RQ2 The MIUs are passed to GPT-4 via API. The model is instructed with zero-shot prompting to assign a score to each feature of every MIU. The best performing prompt is found through four cycles of instruction-tuning, and holding the best-performing prompt (evaluated in terms of F1 scoring with respect to the baseline) for each feature independently. The set of final prompts can be found in Appendix B. The LLM was tested beforehand and determined to be capable of discerning each value under observation without further instructions or the need for context-specific examples; therefore few-shot prompting was not considered necessary for this exploratory study.

The automatic scoring of both the features and the resulting QCS is tested against the corresponding manual baseline.

6 Results

RQ1 The manual assessment of the call-QCS across the four financial quarters of 2021 is reported in Table 2.

Quarter	call-QCS
Q1	174
Q2	149
Q3	50
Q4	116

Table 2: Manual assessment of call-QCS across the four financial quarters of 2021

Stock prices (closing price every day at 4 p.m.) and main events for the year 2021 are summarized in the line chart of Figure 1, where the values of Table 2 are also displayed as a bar chart. Table 3

reports the analysts’ earnings per share (EPS) estimate consensus, the actual EPS results at the end of the period considered, and the resulting surprise; for each financial quarter and annually.

Time period	EPS estimate	EPS result	EPS surprise (%)
Q1	0.86	-0.07	n.a.
Q2	0.37	0.18	-28.8
Q3	0.11	0.15	56.1
Q4	0.03	-0.76	8.8
year	-0.64	-0.61	4.75

Table 3: Earnings per share (estimate consensus, result, percentage surprise) across the four financial quarters of 2021 and for the full year

RQ2 GPT-4 API was called for each QCS feature independently. The feature results were first tested against the feature-baseline and then summed up to form the QCS; the latter was tested against the baseline QCS. GPT-4 predictive performance is measured in terms of F-score; particularly, balanced accuracy F1. Results are presented in Table 4.

Feature	F1
Framing preface	0.59
Complexity	0.62
Directness	0.37
Assertiveness	0.80
Request type	0.53
Time orientation	0.64
QCS	0.20

Table 4: F1 assessment over the entire dataset, testing GPT-4 (best zero-shot prompt) vs. manual QCS-baseline

Manually and automatically assessed QCS was

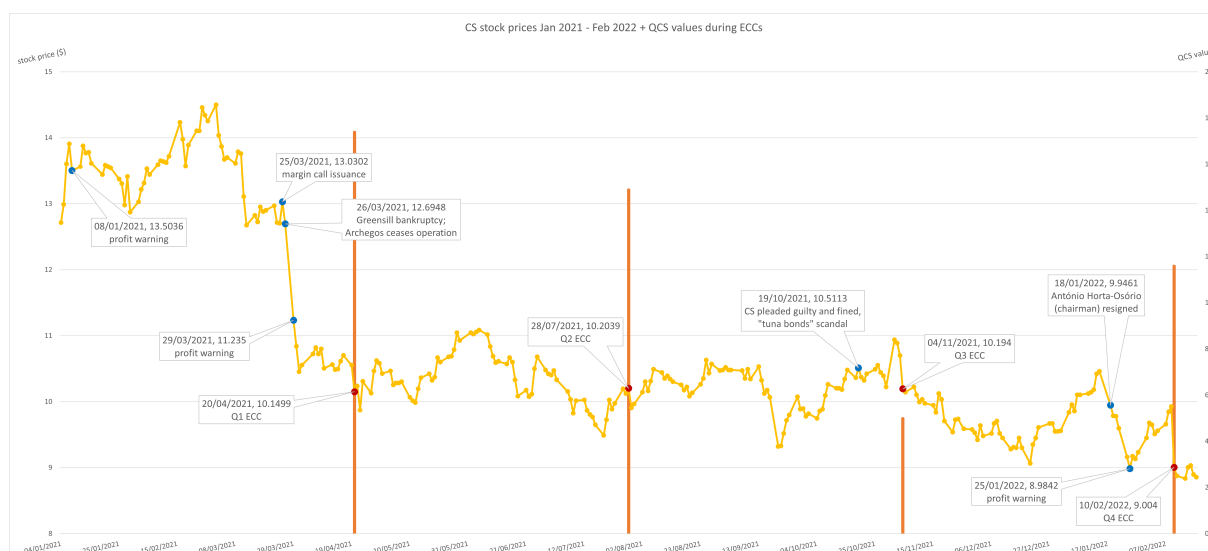


Figure 1: *Line chart*: CS stock prices chart and call-out tags on main events (date, price, event type) occurred to the company between January 2021 and February 2022. Red dots: ECC day; blue dots: disclosure of a negative episode. *Bar chart*: call-QCS values of the four ECCs (as reported in Table 2).

again compared for accuracy, resulting in $F1 = 0.20$ – as shown in the last line of Table 4.

7 Discussion

RQ1 The call-QCS value for Q1 is equal to 174; the highest of the year. Stock prices, on the other hand, were the lowest of the semester on the day of the call and the ones immediately preceding. The context is presumably critical due to the Greensill Capital and Archegos Capital Management scandals, both of which took place about a month before the issue of the Q1 report. Credit Suisse’s misconduct with respect to such events was ascertained by several inquires and the company was pleaded guilty of breaches on risk management and governance; this hugely affected its reliability. Moreover, despite the judiciary issues and two profit warnings since the beginning of the year, the analyst consensus towards Q1 earnings was decidedly positive; results, however, were negative and, most importantly, the difference between estimate and result was almost a point apart ($\Delta = 0.93$): a strong negative surprise. This means that analysts were optimistic and their trust was not rewarded; an optimal ground for a highly cornering tone, aimed at understanding what went wrong.

During Q2, stock prices settled only a little higher than the slump that preceded Q1 announcements: the market was starting to realise that the crisis was not temporary as one might have originally expected. Investigations, especially for the

Archegos case, were still ongoing and many points were still unclear. The difference between expected and actual earnings is again negative, but results are positive and the delta is lower ($\Delta = 0.19$). This semblance of recovery may have been the reason for lowering the Q2 call-QCS slightly, but not a reason strong enough to let it drop. In fact, it is the second highest score of the year.

Q3 was marred by the “tuna bonds” scandal which, however, did not seemingly have a huge impact on either the reliability or the profitability of the company: stock prices do not appear affected in the period following the accusations, and earnings were even greater than expected. This correlates with a rather low cornering score for the call; the lowest of the year.

Quarterly results for Q4 are decidedly negative, as it is the difference between estimation and results; their delta is the second highest of the year ($\Delta = 0.79$). With respect to stock market data, on ECC day the stock price reached a historical low among the ECC days of 2021 (\$ 9.004); besides, it constitutes the second lowest price of the year up to that point. Such a critical environment would seem to call for a high degree of cornering in the questions of analysts; the call-QCS for Q4, however, is 116, lower than in Q2.

Two additional factors need to be acknowledged:

- As shown in Table 3, the release of quarterly results for Q4 was paired with year-on-year earnings data. Although the quarter was nega-

tive, annual results were better than estimated and lead to a earnings-per-share surprise equal to 4.75%; the first positive surprise for this metric in years. This may have softened the stance of some analysts.

- The ECC event was closely followed by an absolute low in value for the company, that reached a cost per share equal to \$ 8.836; Credit Suisse's stock price keep decreasing until the acquisition by UBS at the beginning of 2023. Insider knowledge and intuition both may have contributed to a general slacking in the questioning tone of some analysts: if the belief is that the company is not worth investing in anymore, there is no use in asking cornering questions.

Ultimately, the score appears to follow the financial trends, thus confirming hypothesis H1.

RQ2 Among the six parameters that were submitted for classification to GPT, four obtained a satisfactory F1 result (Assertiveness, Time orientation, Complexity, and Framing preface), one a borderline result (Request type), and one did not reach sufficiency (Directness). This confirms part of the hypothesis, although Directness was not the feature that was expected to perform the lowest.

Rather striking is however the assessment of the predictive performance of the overall QCS (i.e., the sum of the individual feature scores), which resulted in $F1 = 0.20$ – an underwhelming result that evaluates the performance as insufficient. The interpretation of such a measure acknowledges the cumulative nature of F1 scores with respect to each task: whereas feature prediction generally appears to be good enough *on average*, summing the single predictions to evaluate the complex score for each example reveals that they are most often wrong.

In conclusion, the performance of GPT in the classification of constitutive parameters of QCS cannot be assessed as reliable and, subsequently, it cannot be deemed as a valid alternative to the manual assessment of the cornering score. This is in contrast with hypothesis H2.

8 Conclusions and future work

The present study introduces the notions of cornering attitude of a questioner and the Question Cornering Score (QCS) that measures it. It argues for the significance of the QCS in assessing the tone of questions performed by financial analysts

over the course of Q&A sessions of Earnings Conference Calls, it shows that the score correlates with the company's financial performance, and it evaluates the reliability of a GPT model in predicting such a score while decomposed into independent constitutive features.

With respect to the purposefully sampled dataset under observation, the QCS appears to be a descriptive measure of the market stance towards a company over time. Besides the extension of the corpus and verification of such results, following steps will include the assessment of whether the QCS can also work as a predictive indicator.

Given the descriptive power of the property and its related score, the automatic measuring of the QCS on text segments constituting a macro questioning act (MIUs) is a critical goal. The LLM GPT-4 is employed to evaluate MIUs with respect to six independent features. Results appear to be generally good by feature, but unsatisfying on the complete score; consequently, the model is not adequate for the assessment of such a score with the proposed methodology. Further research will investigate new ways of using GPT in the assessment of the cornering quality of questions; however, aware of the fact that GPT is not the adequate tool for mimicking sophisticated logical activities such as inference, we might argue that the subtle clues that suggest the cornering tone of a question also fall into this category. More specialized AI engineering will therefore be required to obtain satisfactory results, as for the argument mining domain.

In the perspective of future work featuring the QCS, here are some goals for our research:

- To identify a reliable way to assess the QCS automatically.
- To correct the score by the possible influence of personal style of the questioner and/or casual noise independent from the ECC event.
- To verify whether the score has a predictive value, other than descriptive.
- To identify patterns within the answers provided to cornering MIUs.
- To measure the degree by which an answer to a cornering MIU is cooperative, i.e., provides the desired type of response.

References

- Chung Chi Chen, Hen Hsen Huang, and Hsin Hsi Chen. 2020. *NLP in FinTech Applications: Past, Present and Future*. *arXiv*. ArXiv: 2005.01320.
- Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. *From Opinion Mining to Financial Argument Mining*. SpringerBriefs in Computer Science. Springer Singapore, Singapore.
- Jason V. Chen, Venky Nagar, and Jordan Schoenfeld. 2018. *Manager-analyst conversations in earnings conference calls*. *Review of Accounting Studies*, 23(4):1315–1354.
- Yuan Chen, Dongmei Han, and Xiaofeng Zhou. 2023. *Mining the emotional information in the audio of earnings conference calls : A deep learning approach for sentiment analysis of securities analysts' follow-up behavior*. *International Review of Financial Analysis*, 88:102704.
- Cynthia E. Clark. 2021. *How do standard setters define materiality and why does it matter?* *Business Ethics, the Environment & Responsibility*, 30(3):378–391.
- Steven E. Clayman, Marc N. Elliott, John Heritage, and Megan K. Beckett. 2012. *The President's Questioners: Consequential Attributes of the White House Press Corps*. *The International Journal of Press/Politics*, 17(1):100–121.
- Steven E. Clayman, Marc N. Elliott, John Heritage, and Laurie L. McDonald. 2006. *Historical Trends in Questioning Presidents, 1953-2000*. *Presidential Studies Quarterly*, 36(4):561–583.
- Steven E. Clayman and John Heritage. 2002. *Questioning presidents: Journalistic deference and adversarialness in the press conferences of U.S. Presidents Eisenhower and Reagan*. *Journal of Communication*, 52(4):749–775. ArXiv: 1011.1669v3 ISBN: 1460-2466.
- Steven E. Clayman, John Heritage, Marc N. Elliott, and Laurie L. McDonald. 2007. *When Does the Watchdog Bark? Conditions of Aggressive Questioning in Presidential News Conferences*. *American Sociological Review*, 72(2005):23–41. ISBN: 0003-1224.
- Belinda Crawford Camiciottoli. 2009. *"Just wondering if you could comment on that": Indirect requests for information in corporate earnings calls*. *Text and Talk*, 29(6):661–681.
- Belinda Crawford Camiciottoli. 2010. *Earnings calls: Exploring an emerging financial reporting genre*. *Discourse & Communication*, 4(4):343–359.
- Giulia D'Agostino, Chris Reed, and Daniele Puccinelli. 2024a. *Segmentation of Complex Question Turns for Argument Mining: A Corpus-based Study in the Financial Domain*. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14524–14530, Torino, Italia. ELRA and ICCL.
- Giulia D'Agostino, Ella Schad, Eimar Maguire, Costanza Lucchini, Andrea Rocci, and Chris Reed. 2024b. *Superquestions and some ways to answer them*. *Journal of Argumentation in Context*. In press.
- Angela K. Davis, Weili Ge, Dawn Matsumoto, and Jenny Li Zhang. 2015. *The effect of manager-specific optimism on the tone of earnings conference calls*. *Review of Accounting Studies*, 20(2):639–673.
- Maria do Carmo Leite de Oliveira and Silvia Maura Rodrigues Pereira. 2017. *Formulations in Delicate Actions: A Study of Analyst Questions in Earnings Conference Calls*. *International Journal of Business Communication*, 55(3):293–309.
- Maria Paz Garcia Villalba and Patrick Saint-Dizier. 2012. *Some facets of argument mining for opinion analysis*. *Frontiers in Artificial Intelligence and Applications*, 245(1):23–34. ISBN: 9781614991106.
- John Heritage. 2003. *Designing Questions and Setting Agendas in the News Interview*. In Phillip Glenn, Curtis D. LeBaron, and Jenny Mandelbaum, editors, *Studies in Language and Social Interaction: In Honor of Robert Hopper*, pages 57–90. Lawrence Erlbaum, Mahwah, NJ.
- John Heritage and Steven E. Clayman. 2013. *The changing tenor of questioning over time tracking a question form across us presidential news conferences, 1953-2000*. *Journalism Practice*, 7(4):481–501.
- Colm Kearney and Sha Liu. 2014. *Textual sentiment in finance: A survey of methods and models*. *International Review of Financial Analysis*, 33:171–185.
- Katherine A. Keith and Amanda Stent. 2019. *Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 493–503, Florence, Italy. Association for Computational Linguistics. ArXiv: 1906.02868.
- Veronika Koller and Xiaoxi Wu. 2023. *Analysts' identity negotiations and politeness behaviour in earnings calls of US firms with extreme earnings changes*. *Corporate Communications: An International Journal*, 28(5):769–787.
- Klaus Krippendorff. 1995. *On the Reliability of Utilizing Continuous Data*. *Sociological Methodology*, 25:47.
- B. Shraavan Kumar and Vadlamani Ravi. 2016. *A survey of the applications of text mining in financial domain*. *Knowledge-Based Systems*, 114:128–147.
- Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool. ISSN: 1947-4040.
- Costanza Lucchini and Giulia D'Agostino. 2023. *Good answers, better questions. Building an annotation scheme for financial dialogues*. Technical report. Ark:/12658/srd1326777.

Costanza Lucchini, Andrea Rocci, and Giulia D'Agostino. 2022. *Annotating argumentation within questions. Prefaced questions as genre specific argumentative pattern in earnings conference calls*. In *Proceedings of the 22nd Edition of the Workshop on Computational Models of Natural Argument (CMNA 22)*, volume vol. 3205, pages 61–66, Cardiff. CEUR.

Rudi Palmieri, Andrea Rocci, and Nadzeya Kudraut-sava. 2015. *Argumentation in earnings conference calls. Corporate standpoints and analysts' challenges*. *Studies in communication sciences*, 15, 2015(1):120–132.

Andrea Pazienza, Davide Grossi, Floriana Grasso, Rudi Palmieri, Michele Zito, and Stefano Ferilli. 2020. *An abstract argumentation approach for the prediction of analysts' recommendations following earnings conference calls*. *Intelligenza Artificiale*, 13(2):173–188.

Andrea Rocci and Carlo Raimondo. 2017. *Conference calls: A communication perspective*. In Alexander V. Laskin, editor, *The Handbook of Financial Communication and Investor Relations*, pages 293–308. John Wiley & Sons, New York, NY. <https://doi.org/10.1002/9781119240822.ch26>.

Andrea Rocci, Carlo Raimondo, and Daniele Puccinelli. 2019. *Evidentiality and Disagreement in Earnings Conference Calls : Preliminary Empirical Findings*. In *Proceedings of the 3rd Workshop on Advances In Argumentation In Artificial Intelligence co-located with the 18th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2019)*, pages 1–5.

Andrzej Wiśniewski. 2015. *Semantics of Questions*. In Shalom Lappin and Chris Fox, editors, *The Handbook of Contemporary Semantic Theory*, 1 edition, pages 271–313. Wiley.

A Examples of QCS attribution

Showcasing both the single feature-level scores and the overall cornering score of the MIU.

- (2) Firstly is just on the strategy again. Obviously, you had a very detailed presentation in December last year and probably we were talking about the 10% to 12% and the investing for growth. Should we assume by this fact the strategy stands at 10% to 12% RoTE or should we expect, as you indicated, post the Investment Banking review and the new chairman arriving that we will get a new strategic update?

Example (2) (analyst Anke Reingen, CS Q1 2021): Framing preface: 1 (preface present; the preface has a neutral-to-positive tone); Complexity: 0 (one question); Directness: 1 (no hedging or modalization); Assertiveness: 1 (closed formulation of

the question); Request type: 2 (confirmation of an inference); Time orientation: 1 (future-oriented); **QCS: 6** → *the MIU is cornering*

- (3) Sorry, on the prime brokerage business, you indicated two-thirds of the, I think, balances down, but you also gave an indication of 600 million of revenues, 400 million of cost to be reduced in 2022. I just wonder – I assume some of that is already in the numbers or should we think about the numbers getting bigger?

Example (3) (analyst Kian Abouhossein, CS Q4 2021): Framing preface: 2; Complexity: 0; Directness: 0; Assertiveness: 1; Request type: 1; Time orientation: 1; **QCS: 5** → *the MIU is slightly cornering*

- (4) The first one, just trying to get a sense, I appreciate you don't prejudge the outcome, but – so the strategic review, just if we can get a bit more color in terms how the process works, how that's being conducted, how decisions will be made and the kind of trade-offs and the processes involved in that, that would be helpful.

Example (4) (analyst Amit Goel, CS Q2 2021): Framing preface: 1; Complexity: 0; Assertiveness: 0; Request type: 0; Time orientation: 0; **QCS: 1** → *the MIU is not cornering*

B GPT prompts

Framing preface messages=[
{"role": "system", "content": "You are a helpful assistant designed to output JSON."},
{"role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a quantitative assessment called 'framing preface', relative to the presence and the nature of background statements that precede or follow questions in the input. A background statement is an assertive sentence used to give some background, context or justification to the question proper. The quantitative assessment must come in the form of a single integer number in the range {0, 2}, where 0 means that there is no background statement, 1 means that there is at least one background statement that presents a positive or neutral situation, 2 means that there is at least one background statement that presents a negative situation."},

```
{ "role": "system", "content": "The scores must be considered as in ascending order of importance: if there were two background statements, of which one positive and one negative, only the negative one will be considered." },
{ "role": "user", "content": text } ]
```

Complexity messages=[
 { "role": "system", "content": "You are a helpful assistant designed to output JSON." },
 { "role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a score called 'complexity', which refers to the number of interrogative sentences present in the input. The score 'complexity' must come in the form of a single integer number in the range {0, 2}, where 0 means that the input displays one interrogative sentence, 1 means that the input displays two to three interrogative sentences, 2 means that the input displays four or more interrogative sentences." },
 { "role": "system", "content": "The answer must be based on the total number of interrogative sentences, including multiple instances of the same one." },
 { "role": "user", "content": *text* }]

Directness messages=[
 { "role": "system", "content": "You are a helpful assistant designed to output JSON." },
 { "role": "system", "content": "You examine the text of questions asked by financial analysts in earnings conference calls. Sometimes analysts ask questions in an indirect, tentative and polite manner, sometimes they are blunt and to the point." },
 { "role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a score called 'directness'. The score 'directness' must come in the form of a single integer number in the range {0, 1}, where 0 means that the input contains at least one indicator of politeness, tentativeness, indirectness or hedging, 1 means that the input does not contain any indicator of politeness, tentativeness, indirectness or hedging." },
 { "role": "user", "content": *text* }]

N.B. the second "system" message can be omitted and the results, i.e., the F1 measure of the performance over the 111 MIUs, is not affected up to the 15th decimal position.

Assertiveness messages=[
 { "role": "system", "content": "You are a helpful

assistant designed to output JSON." },
 { "role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a quantitative assessment called 'assertiveness', which refers to the formulation of questions in the input. The quantitative assessment 'assertiveness' must come in the form of a single integer number in the range {0, 1}, which rates only once the element with the highest score in the input." },
 { "role": "system", "content": "Score 0 means that the input contains only open questions. Open questions are wh- questions (beginning with what, why, how) or questions asking to describe, elaborate or explain something in an open ended way. Score 1 means that the input contains polar questions or choice questions. Polar questions are yes/no questions or questions asking whether someone can confirm or agrees with a statement, a comment, a forecast, an explanation or a piece of reasoning. Choice questions posit a closed list of alternatives asking to choose between them, it could be alternative descriptions, evaluations, outlooks, explanations or reasons" },
 { "role": "user", "content": *text* }]

Request type messages=[
 { "role": "system", "content": "You are a helpful assistant designed to output JSON." },
 { "role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a quantitative assessment called 'request type', which refers to the kind of answer that is sought by the questions in the input. The quantitative assessment 'request type' must come in the form of a single integer number in the range {0, 2}, which rates only once the element with the highest score in the input." },
 { "role": "system", "content": "For the quantitative assessment 'request type', follow these criteria: questions that challenge the respondent to provide a justification grant the score 2; questions that seek a commitment to action from the respondent grant the score 2; questions that ask the respondent to confirm or disconfirm a hypothesis, inference, guess or calculation grant the score 2; questions that ask for an evaluative or predictive opinion or some kind of assessment grant the score 1; questions that seek an explanation, query about the causes of an event or the motives of an action grant the score 1; questions that request a clarification of what has been said grant the score 1;

questions that ask to confirm material data grant the score 1; questions that ask to elaborate on a topic or ask for details grant the score 0; questions that merely ask for data grant the score 0."},
{"role": "user", "content": *text* }]

Time orientation messages=[
{"role": "system", "content": "You are a helpful assistant designed to output JSON."},
{"role": "system", "content": "You take as an input some text drawn from a question turn of an earnings conference call and output a quantitative assessment called 'time orientation'. The quantitative assessment 'time orientation' must come in the form of a single integer number in the range {0, 2}, which rates only once the element with the highest score in the input."},
{"role": "system", "content": "Score 0 is attributed when the entire input asks questions about the present; score 1 is attributed when the input contains questions about the future; score 2 is attributed when the input contains questions about the past."},
{"role": "user", "content": *text* }]

FinNLP-AgentScen-2024 Shared Task: Financial Challenges in Large Language Models - FinLLMs

Qianqian Xie¹ and Jimin Huang¹ and Dong Li⁹ and Zhengyu Chen⁹ and Ruoyu Xiang¹

Mengxi Xiao⁹ and Yangyang Yu⁷ and VijayaSai Somasundaram⁸ and Kailai Yang²

Chenhan Yuan² and Zheheng Luo² and Zhiwei Liu² and Yueru He¹¹ and Yuechen Jiang⁷

Haohang Li⁷ and Duanyu Feng⁵ and Xiao-Yang Liu^{3,11} and Benyou Wang⁴ and Hao Wang⁵

Yanzhao Lai⁶ and Jordan Suchow⁷ and Alejandro Lopez-Lira⁸ and Min Peng⁹

Sophia Ananiadou^{2,10}

¹The Fin AI, Singapore; ²University of Manchester, UK; ³Open Finance, USA;

⁴Chinese University of Hong Kong, Shenzhen, China; ⁵Sichuan University, China;

⁶Southwest Jiaotong University, China; ⁷Stevens Institute of Technology, USA;

⁸University of Florida, USA; ⁹Wuhan University, China; ¹⁰Archimedes RC, Greece;

¹¹Columbia University, USA;

Abstract

Despite the promise of large language models (LLMs) in finance, their capabilities for comprehensive analysis and decision-making remain largely unexplored, particularly in areas such as financial text analysis, generation, and decision-making. To evaluate the capabilities of LLMs in finance, we introduce an LLMs-based financial shared task featured at IJCAI FinNLP-AgentScen-2024, FinLLMs Challenge. This challenge includes three subtasks: financial classification, financial text summarization, and single stock trading. In this paper, we provide an overview of these tasks and datasets, summarize participants' methods, and present their experimental evaluations, highlighting the effectiveness of LLMs in addressing diverse financial challenges. To the best of our knowledge, the FinLLMs Challenge is one of the first challenges for assessing LLMs in the financial area. In consequence, we provide detailed observations and take away conclusions for future development in this area.

1 Introduction

FinNLP workshop is a platform committed to promoting international cooperation and the exchange of knowledge in applying Natural Language Processing (NLP) within the ever-evolving realm of

FinTech. In recent years, the FinNLP series has delved into the intersection of FinTech and NLP, uncovering significant challenges and guiding future research directions, along with proposing a series of diverse share task in financial domain, involving Sentence boundary detection (Azzi et al., 2019; Au et al., 2020), learning semantic representations (Maarouf et al., 2020) and semantic similarities (Kang et al., 2021; Kang and El Maarouf, 2022; Chen et al., 2023).

Recent studies (Xie et al., 2024b, 2023; Lopez-Lira and Tang, 2023; Liu et al.; Xie et al., 2024a) have highlighted the significant potential of advanced large language models (LLMs) in finance, particularly for tasks involving financial text analysis and prediction. These models can transform traditional methodologies by boosting efficiency and enhancing the accuracy of predictive models. Although several approaches have achieved remarkable performance with LLMs, their capabilities of comprehensive analysis and decision-making for finance remain largely unexplored.

To explore the ability of LLMs from these facets, we propose a LLMs-based financial shared task, **FinLLMs Challenge**. This challenge includes three published datasets designed to address a range

of financial challenges effectively and comprehensively. These tasks include financial classification, financial text summarization, and single stock trading. For financial classification tasks, we utilize the FinArg AUC dataset (Chen et al.), which provides financial texts paired with two opinions. Using this data, we provide a prompt template to classify the text as either a claim or a premise. For financial text summarization tasks, we introduce the EDTSum dataset (Zhou et al., 2021), which is used to summarize given financial news articles, along with a recommended prompt template. For decision-making tasks, we provide the fintrade dataset (Xie et al., 2024a), which can be leveraged by FinMem (Yu et al., 2023) agent framework, allowing LLMs to generate one of three trading decisions from “buy”, “sell” or “hold.”

This paper overviews three subtasks and datasets in the FinLLMs Challenge, summarizes participant methods, and evaluates their experiments to explore LLM’s capabilities in financial analysis and prediction. Our comprehensive evaluation highlights the strengths and limitations of current methodologies, showcasing the effectiveness of LLMs across various financial tasks and the potential of domain-specific instruction tuning in the financial sector.

2 Tasks and Datasets

We provide three tasks for assessing the performance of LLMs in finance, as shown in Table 1.

Task 1: Financial Classification. This task, derived from FinBen (Xie et al., 2024a), concentrates on argument unit classification to identify and categorize individual units or segments of arguments within the discourse found in earnings conference call data. The objective of this task is to evaluate the capability of LLMs to distinguish and classify texts as premises or claims. The dataset (Chen et al.) includes 7.75k training examples and 969 testing examples for sentence categorization into claims or premises. We use two metrics to evaluate classification capability, including Macro F1 and Accuracy. Macro F1 score is used as the final ranking metric.

Task 2: Financial Text Summarization. Derived from FinBen (Xie et al., 2024a), this task aims to evaluate the ability of LLMs in producing coherent summaries. The dataset (Zhou et al., 2021) includes 8,000 training instances and 2,000 test instances for summarizing financial news articles suc-

cinctly. We utilize two metrics including ROUGE (1, 2, and L) (Lin, 2004) and BERTScore (Zhang et al., 2020), to evaluate generated summaries in terms of relevance. ROUGE-1 score is used as the final ranking metric.

Task 3: Single Stock Trading. Building on the Trading task in FinBen (Xie et al., 2024a), this evaluation aims to rigorously assess the ability of LLMs to execute complex trading decisions, addressing the critical challenge of human limitations in processing large volumes of data rapidly. We construct and provide the first public dataset of 291 distinct data points, which allows to test the models’ decision-making capabilities in stock trading based on the agent framework. Participants are required to analyze the dataset, adapt or develop LLM frameworks for financial data interpretation, and implement algorithms to generate sophisticated trading strategies based on the FinMem agent framework (Yu et al., 2023).

We employ the following prompt for model inputs:

Instruction: [task prompt] **Context:** [input context] **Response:** [output].

[input text] represents the financial investment information provided in the prompt. The [output] must adhere strictly to the following JSON format, without any additional content:

```
{
  "investment_decision": string,
  "summary_reason": string,
  "short_memory_index": number,
  "middle_memory_index": number,
  "long_memory_index": number,
  "reflection_memory_index": number
}
```

We offer a comprehensive assessment of profitability, risk management, and decision-making prowess by a series of metrics, including Sharpe Ratio (SR) (Sharpe, 1994), Cumulative Return (CR), Daily (DV) and Annualized volatility (AV), and Maximum Drawdown (MD). Sharpe Ratio (SR) score is used as the final ranking metric, which is calculated by dividing the portfolio’s average excess return (R_p) over the risk-free rate (R_f) by its volatility (δ_p).

$$SharpeRatio = \frac{R_p - R_f}{\delta_p} \quad (1)$$

Where R_p represents the portfolio’s average excess return, R_f is the risk-free rate, δ_p is the port-

Task	Dataset	Size	Types	License
Financial classification	FinArg (Chen et al.)	8,719	Earnings calls	CC BY-NC-SA 4.0
Financial text summarization	EDTSum (Zhou et al., 2021)	10,000	Financial News	Public
Single stock trading	Fintrade (Xie et al., 2024a)	291	Financial News, Company Fillings, Historical prices	MIT License

Table 1: Summary of the tasks and datasets in FinNLP-AgentScen-2024

folio’s volatility.

3 Model Cheating Detection

To assess the risk of *model cheating*, where models improperly access test data during training (Zhou et al., 2023), we introduce a new metric called the Data Leakage Test (DLT). This metric builds on previous research (Wei et al., 2023; Xu et al., 2024) and aims to quantify the likelihood that a model is exposed to the test set during its training process.

The DLT measures the risk by comparing how well the LLM performs on the training data versus the test data. We feed the training and test sets separately into the model and measure its perplexity on each. The DLT score is then calculated by subtracting the perplexity on the training set from the perplexity on the test set:

$$DLT = PPL(D_{test}) - PPL(D_{train}) \quad (2)$$

where PPL is the perplexity given the dataset inputs.

A larger DLT score suggests the LLM is less likely to have been exposed to the test data during training. Conversely, a smaller DLT score implies the LLM is more likely to have seen the test data during training, suggesting a higher likelihood of cheating.

4 Participants and Automatic Evaluation

35 teams have registered for the FinLLMs Challenge, out of which 8 teams have submitted their LLMs solution papers. In this section, we provide a detail overview of the LLMs based solutions for each paper. For task 1 and 2, we employ two baseline models from (Xie et al., 2024a): GPT-4 (OpenAI et al., 2024) and LLaMA3-8B¹. GPT-4, developed by OpenAI, is the state-of-the-art commercialized large language model, whereas LLaMA3-8B, created by MetaAI, is an open-source large language model built with more training data than its predecessor, LLaMA2.

¹<https://llama.meta.com/llama3/>

4.1 Task 1: Financial Classification

Table 2 presents the experimental results of task 1. BAI-Arg LLM (Srivastava, 2024) leverages LLaMA3-8B which is fine-tuned via QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023). L3iTC (Pontes et al., 2024), utilizes Mistral-7BInst-v0.3 to be finetuned with 4-bit quantization and LoRA (Hu et al., 2021) to reduce the memory usage of LLMs. Wealth Guide (Das et al., 2024) fine-tuned DistilBERT for financial text classification. CatMemo (Cao et al., 2024) finetuned Mistral-7B with fused datasets of both task 1 and task 2 via LoRA. Upaya (Jindal et al., 2024) utilizes distillation-based fine-tuning of the LLaMA3-8B method to learn the rationale generated by LLaMA-3 (70B parameters) and labels.

4.2 Task 2: Financial Text Summarization

Table 3 presents the experimental results of task 2. University of Glasgow (Guo et al., 2024) investigated three common strategies: few-shot learning, fine-tuning, and reinforcement learning, to adapt LLMs to abstract news into concise summaries, with the fine-tuned model ranked first on the leaderboard. Upaya (Jindal et al., 2024) also utilized distillation-based fine-tuning of the LLaMA3-8B method, which leveraged the augmented datasets with a maximum of 5 relevant sentences from the original news text that are relevant to the given summary via LLaMA3-70B. Finance Wizard (Lee and Lay-Ki, 2024) introduced a pipeline approach. Based on LLaMA3-8B foundation, they first continual pretrained the model with the financial corpus, then they tailored it to the finance domain with multi-task instruction data, and finally fine-tune it for specific tasks. Revelata (Kawamura et al., 2024) first designed a set of prompts by systematically changing parts of the prompts and then fine-tuning Meta-LLaMA3-8B-Instruct on each of these prompts separately. L3iTC (Pontes et al., 2024) introduced Mistral-7B-Inst-v0.3 model, a finetuning model combining 4-bit quantization and LoRA to optimize the finetuning process.

Team	Method	F1	Accuracy
BAI-Arg LLM	LlaMA3-8B + QLoRA + Finetuning	0.7612	0.7626
Albatross	–	0.7575	0.7575
L3iTC	Mistral-7B + 4 Bit + Lora + Finetuning	0.7543	0.7544
Wealth Guide	DistilBERT + Finetuning	0.7509	0.7513
Finance Wizard	–	0.7262	0.7286
CatMemo	Mistral-7B + Task 1 + Task 2 + Qlora + Finetuning	0.7086	0.7110
Upaya	LlaMA3-8B + Distillation + Finetuning	0.7083	0.7090
Vidra	–	0.7070	0.7079
jt	–	0.4630	0.4933
Baseline (Xie et al., 2024a)	GPT-4	0.6000	–
Baseline (Xie et al., 2024a)	LlaMA3-8B	0.5100	–

Table 2: Evaluation Results of Task 1 - Financial Classification.

Team	Method	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
University of Glasgow	LlaMA3-8B + 4 bit	0.5346	0.3581	0.4922	0.9117
	+ QLoRA + Instruction tuning				
Upaya	LlaMA3-8B	0.5295	0.3582	0.4860	0.9106
	+ Distillation + Finetuning				
Finance Wizard	LlaMA3-8B	0.5210	0.3406	0.4735	0.9084
	+ Continual pretraining				
Revelata	+ Multi-task tuning + Specific tuning	0.5004	0.3330	0.4644	0.9070
	LlaMA3-8B-Instruct				
Albatross	+ Finetuning + Lead-in phrase	0.3691	0.2011	0.3227	0.8720
L3iTC	–	0.3661	0.1872	0.3046	0.8750
	Mistral-7B-Inst-v0.3				
Wealth Guide	+ Lora + Finetuning	0.3089	0.1795	0.2819	0.8596
Vidra	–	0.2850	0.1348	0.2286	0.8587
Baseline	GPT-4	0.2000	–	–	0.6700
Baseline	LlaMA3-8B	0.1400	–	–	0.6000

Table 3: Evaluation results of Task 2 - Financial Text Summarization.

4.3 Task 3: single stock trading

Table 4 presents the experimental results of task 3. Wealth Guide (Das et al., 2024) utilizes the LLaMA2-13B model with zero-shot and few-shot fine-tuning, integrating sentiment scores and stock prices for trading predictions. CatMemo (Cao et al., 2024) also utilizes the Mistral-7B model fine-tuned using PEFT and LoRA techniques, integrating datasets from Task 1 and Task 2.

5 Discussion

5.1 Task 1: financial classification

As shown in Table 2, the experimental results highlight the remarkable performance of various teams in the financial text classification task, all of which employed fine-tuning with task-specific training data. Notably, BAI-Arg LLM, utilizing the LLaMA3-8B model with fine-tuning, carefully designed prompts, and semantically similar examples, achieved the best performance with an F1 score of 0.7612 and an accuracy of 0.7626. This performance surpasses both GPT-4 and the backbone model LLaMA3-8B, fully demonstrating the benefits of fine-tuning with task-specific data in

financial classification tasks based on LLMs.

Compared to L3iTC and other teams, BAI-Arg LLM’s performance underscores the importance of both prompt templates and semantically similar examples for fine-tuning LLMs on financial classification tasks. This indicates the necessity for LLMs to be adapted to financial classification tasks through prompt engineering and few-shot learning. Moreover, their performance surpasses that of DistilBERT, proving the potential of LLMs compared to traditional BERT-based methods.

5.2 Task 2: financial text summarization

As shown in Table 3, the experimental results highlight the potential of LLMs in financial text summarization. Leveraging LLMs facilitates the generation of high-quality summaries, thereby enhancing both efficiency and quality. Similar to financial classification tasks, performance improves significantly with task-specific fine-tuning.

Notably, methods employing LLMs generally achieve high scores across various metrics. For instance, the University of Glasgow team achieved a ROUGE-1 score of 0.5346 using the instruc-

Teams	Method	SR	CR	SD	AV	MD
Wealth Guide	LLaMA2-13B + Finetuning	0.9264	0.0727	0.0085	0.1353	0.0605
Albatross	–	0.4838	0.0280	0.0081	0.1399	0.1158
Upaya	–	0.4675	0.0308	0.0097	0.1547	0.1112
CatMemo	Mistral-7B + Task 1 + Task 2 + Qlora + Finetuning	-0.6199	0.0450	0.0083	0.1311	0.1056

Table 4: Evaluation results of Task 3 - Single Stock Trading.

tion tuning method, while the Upaya team scored 0.5295 with a distillation-based fine-tuning approach. These results indicate that LLMs, when fine-tuned with appropriate methods, can effectively capture and condense the main information from financial texts into clear and concise summaries. The Finance Wizard team employed continual pretraining, multi-task fine-tuning, and specific task fine-tuning with LLaMA3-8B, demonstrating substantial benefits in overall performance. These approaches outperform GPT-4 and the backbone model LLaMA3-8B, underscoring that fine-tuning and continual pretraining can lead to significant improvements in financial text summarization tasks.

5.3 Task 3: single stock trading

Table 4 presents the performance of various teams using different LLMs in single stock trading tasks. The experimental results indicate that our challenge and provided resources have indeed contributed to advancements in financial investment decision-making. Participants utilized these resources to develop effective strategies and models, thereby improving their performance in this domain. The results reveal the potential of LLMs in financial investment decision-making, especially when integrated within an agent framework.

Notably, methods employing LLMs have achieved remarkable performance in key metrics. For instance, the Wealth Guide team achieved the highest Sharpe Ratio score of 0.9264 using a sentiment-score-based trading prediction model, indicating the effectiveness of LLMs in predicting market trends. In terms of Cumulative Return, the Wealth Guide team’s model also showed significant promise. These findings underscore the potential of LLMs to enhance trading strategies and improve investment outcomes when fine-tuned and applied within an agent framework. However, the CatMemo team’s use of the Mistral-7B method recorded lower performance, highlighting the variability in effectiveness depending on the specific model and approach used. Despite this, the overall results suggest that with proper tuning and integration, LLMs can be powerful tools in financial stock

trading based on the agent framework.

5.4 Model Cheating Detection

We further conducted a Model Cheating Detection analysis using our Data Leakage Test (DLT) on teams that disclosed their training procedures in Task 1 and Task 2. The results, summarized in Table 5, reveal no evidence of model cheating among these teams.

Team	Task	Rank	DLT
BAI-Arg LLM	Task1	1	38.90
L3iTC	Task1	3	2.26
Upaya	Task2	2	0.83
Finance Wizard	Task2	3	1.74

Table 5: Evaluation—Model Cheating Detection

For instance, “BAI-Arg LLM”, the top-performing team in Task 1, exhibited a DLT score of 38.90, significantly above zero, effectively ruling out any data leakage concerns. Similarly, teams like “L3iTC” and “Finance Wizard” consistently displayed DLT scores exceeding 1.5, indicating a negligible risk of data leakage.

These findings suggest that the majority of the participating teams adhered to the competition’s ethical guidelines. Furthermore, even with this strict adherence, the impressive performance improvements these teams achieved, exceeding the original benchmarks, underscore the immense potential of LLMs within the financial realm.

6 Conclusion

In this paper, the FinLLMs Challenge has demonstrated the efficacy and potential of LLMs in the domain of financial investment decision-making. Our challenge, along with the resources provided, has significantly contributed to advancing this field. Participants utilized these resources to develop effective strategies and models, which led to improved performance across various tasks. The experimental results from tasks such as financial classification, text summarization, and single stock trading highlight the considerable value of LLMs-

based approaches. The overall trend indicates that performance improves with increasing model size and advancements in fine-tuning and prompt engineering. These findings offer valuable insights for future research in financial tasks using LLMs. The success of this challenge underscores the importance and impact of collaborative efforts in pushing the boundaries of AI applications in finance.

Acknowledgments

We would like to thank all the anonymous reviewers and area chairs for their comments. This work is supported by the project JPNP20006 from New Energy and Industrial Technology Development Organization (NEDO). This work has also been partially supported by project MIS 5154714 of the National Recovery and Resilience Plan Greece 2.0 funded by the European Union under the NextGenerationEU Program. This is also supported by National Science and Technology Major Project (No.2021ZD0113304), National Natural Science Foundation of China (U23A20316), Key R&D Project of Hubei Province (2021BAA029), General Program of Natural Science Foundation of China (NSFC) (Grant No.62072346), and founded by Joint&Laboratory on Credit Technology.

References

Willy Au, Bianca Chong, Abderrahim Ait Azzi, and Dialekti Valsamou-Stanislawski. 2020. [FinSBD-2020: The 2nd shared task on sentence boundary detection in unstructured text in the financial domain](#). In *Proceedings of the Second Workshop on Financial Technology and Natural Language Processing*, pages 47–54, Kyoto, Japan. -.

Abderrahim Ait Azzi, Houda Bouamor, and Sira Ferradans. 2019. [The FinSBD-2019 shared task: Sentence boundary detection in PDF noisy text in the financial domain](#). In *Proceedings of the First Workshop on Financial Technology and Natural Language Processing*, pages 74–80, Macao, China.

Yupeng Cao, Zhiyuan Yao, Zhi Chen, and Zhiyang Deng. 2024. [Catmemo@ijcai 2024 finllm challenge: Fine-tuning large language models using data fusion in financial applications](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.

Chung-Chi Chen, Chin-Yi Lin, Chr-Jr Chiu, Hen-Hsen Huang, Alaa Alhamzeh, Yu-Lieh Huang, Hiroya Takamura, and Hsin-Hsi Chen. [Overview of the](#)

[ntcir-17 finarg-1 task: Fine-grained argument understanding in financial analysis](#).

- Chung-Chi Chen, Yu-Min Tseng, Juyeon Kang, Anaïs Lhuissier, Yohei Seki, Min-Yuh Day, Teng-Tsai Tu, and Hsin-Hsi Chen. 2023. [Multi-lingual ESG impact type identification](#). In *Proceedings of the Sixth Workshop on Financial Technology and Natural Language Processing*, pages 46–50, Bali, Indonesia. Association for Computational Linguistics.
- Sarmistha Das, R E Zera Marveen Lyngkhai, Sriparna Saha, and Alka Maurya. 2024. [Wealth guide: A sophisticated language model solution for financial trading decisions](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [Qlora: Efficient finetuning of quantized llms](#). *Preprint*, arXiv:2305.14314.
- Lubingzhi Guo, Javier Sanz-Cruzado, and Richard McCreadie. 2024. [University of glasgow at the finllm challenge task: Adapting llama for financial news abstractive summarization](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#). *Preprint*, arXiv:2106.09685.
- Ashvini Kumar Jindal, Pawan Kumar Rajpoot, and Ankur Parikh. 2024. [Upaya at the finllm challenge task 1 and 2: Distfin: Distillation based fine-tuning for financial tasks](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Juyeon Kang and Ismail El Maarouf. 2022. [FinSim4-ESG shared task: Learning semantic similarities for the financial domain. extended edition to ESG insights](#). In *Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP)*, pages 211–217, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Juyeon Kang, Ismail El Maarouf, Sandra Bellato, and Mei Gan. 2021. [FinSim-3: The 3rd shared task on learning semantic similarities for the financial domain](#). In *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, pages 31–35, Online. -.

- Ken Kawamura, Zeqian Li, Chit-Kwan Lin, and Bradley McDanel. 2024. [Revelata at the finllm challenge task: Improving financial text summarization by restricted prompt engineering and fine-tuning](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Meisin Lee and Soon Lay-Ki. 2024. [‘finance wizard’ at the finllm challenge task: Financial text summarization](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Xiao-Yang Liu, Guoxuan Wang, Hongyang Yang, and Daochen Zha. [Fingpt: Democratizing internet-scale data for financial large language models](#). In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.
- Alejandro Lopez-Lira and Yuehua Tang. 2023. [Can chatgpt forecast stock price movements? return predictability and large language models](#). *arXiv preprint arXiv:2304.07619*.
- Ismail El Maarouf, Youness Mansar, Virginie Mouilleron, and Dialekti Valsamou-Stanislawski. 2020. [The FinSim 2020 shared task: Learning semantic representations for the financial domain](#). In *Proceedings of the Second Workshop on Financial Technology and Natural Language Processing*, pages 81–86, Kyoto, Japan. -.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O’Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong

- Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. [Gpt-4 technical report](#). *Preprint*, arXiv:2303.08774.
- Elvys Linhares Pontes, Carlos-Emiliano González-Gallardo, Mohamed Benjannet, Caryn Qu, and Antoine Doucet. 2024. [L3itc at the finllm challenge task: Quantization for financial text classification summarization](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- William F. Sharpe. 1994. [The sharpe ratio](#).
- Varad Srivastava. 2024. [Bai-arg llm at the finllm challenge task: Earn while you argue - financial argument identification](#). In *proceedings of Joint Workshop of the 8th Financial Technology and Natural Language Processing (FinNLP) and the 1st Agent AI for Scenario Planning (AgentScen): FinNLP-AgentScen @ IJCAI 2024*. International Joint Conference on Artificial Intelligence.
- Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, Chenxia Li, Liu Yang, Xilin Luo, Xuejie Wu, Lunan Liu, Wenjun Cheng, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, Lei Lin, Xiaokun Wang, Yutuan Ma, Chuanhai Dong, Yanqi Sun, Yifu Chen, Yongyi Peng, Xiaojuan Liang, Shuicheng Yan, Han Fang, and Yahui Zhou. 2023. [Skywork: A more open bilingual foundation model](#). *Preprint*, arXiv:2310.19341.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyang Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. 2024a. [The finben: An holistic financial benchmark for large language models](#). *Preprint*, arXiv:2402.12659.
- Qianqian Xie, Weiguang Han, Yanzhao Lai, Min Peng, and Jimin Huang. 2023. [The wall street neophyte: A zero-shot analysis of chatgpt over multimodal stock movement prediction challenges](#). *arXiv preprint arXiv:2304.05351*.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2024b. [Pixiu: A comprehensive benchmark, instruction dataset and large language model for finance](#). *Advances in Neural Information Processing Systems*, 36.
- Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. 2024. [Benchmarking benchmark leakage in large language models](#). *arXiv preprint arXiv:2404.18824*.
- Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Su-chow, and Khaldoun Khashanah. 2023. [Finmem: A performance-enhanced llm trading agent with layered memory and character design](#). *Preprint*, arXiv:2311.13743.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. [Bertscore: Evaluating text generation with bert](#). *Preprint*, arXiv:1904.09675.
- Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. [Don't make your llm an evaluation benchmark cheater](#). *arXiv preprint arXiv:2311.01964*.
- Zhihan Zhou, Liqian Ma, and Han Liu. 2021. [Trade the event: Corporate events detection for news-based event-driven trading](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.

University of Glasgow at the FinLLM Challenge Task: Adapting Llama for Financial News Abstractive Summarization

Lubingzhi Guo and Javier Sanz-Cruzado and Richard McCreddie

University of Glasgow

l.guo.1@research.gla.ac.uk and javier.sanz-cruzadopuig@glasgow.ac.uk
and richard.mccreadie@glasgow.ac.uk

Abstract

In this paper, we explore different approaches for aligning Large Language Models (LLMs) with the objectives of the financial abstractive summarization shared task. This shared task focuses on using LLM to abstract news into concise summaries. We investigate three common strategies: few-shot learning, fine-tuning, and reinforcement learning, to adapt LLMs for this purpose, with the fine-tuned model ranked first on the leaderboard.

1 Introduction

Text summarization aims to create coherent and concise summaries from input documents using either extractive and abstractive methods. The extractive approach identifies the most important sentences from the source text(s) and concatenates them into a summary, while the abstractive method focuses on generating novel sentences and words (Widyassari et al., 2022). With the advent of large language models (LLM) in text generation, summaries generated by state-of-the-art LLMs, specifically with instruction tuning, perform comparably to those written by human annotators (Zhang et al., 2024).

The financial text summarization shared task is designed to explore the capabilities of LLMs in the finance domain; the task is focused on generating abstractive news summaries using LLMs. We approach the task by exploring the current tuning strategies for LLMs with the goal of generating concise financial summaries.

2 Dataset

The provided training dataset consists of 8,000 news articles from the EDT corpus (Zhou et al., 2021), which is designed for news event detection and financial domain adaption. For the purpose of financial abstractive summarization, the gold

summaries are constructed through distant supervision, using the corresponding news headlines and sub-headlines.

To gain a clear understanding of the summary requirements, we further analyze the provided gold summaries. Figure 1 presents a histogram of the word counts for the gold summaries in the training dataset. The x-axis represents the word count for each summary, calculated using the nltk library (Bird et al., 2009), while the y-axis shows the frequency of summaries at the corresponding word count. From the histogram, it is evident that the majority of summaries have between 10-25 words, with very few extending beyond 100 words. This indicates that the reference summaries are generally brief. Despite the gold summaries comprising both headlines and sub-headlines, the evaluation of this task primarily focuses on headline generation.

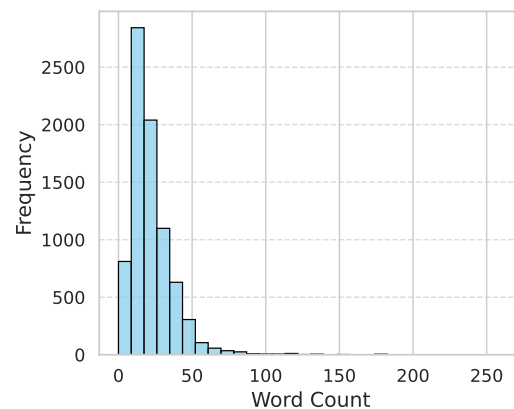


Figure 1: Frequency Distribution of Word Counts in Gold Summaries

3 Methods

We investigate the three techniques described below to adapt the pre-trained LLM from the general domain for this specific summarization task. Figure 2 shows the overall procedure for the three methods described below.

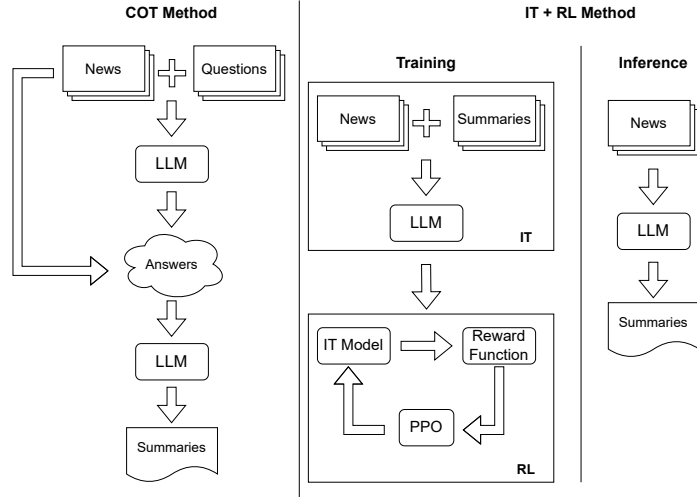


Figure 2: A diagram of the three applied methods

3.1 Chain of Thoughts

LLMs exhibit strong few-shot learning capabilities, effectively using a few demonstrations to perform a wide range of downstream tasks through in-context learning (Brown et al., 2020). Chain of thought (COT) prompting further augments the approach via step-by-step reasoning examples instead of standard question-answer pairs (Wei et al., 2022b; Nye et al., 2022). Wang et al. (2023) improved the summarization ability of LLMs by employing guiding questions as prompts to generate step-by-step, by adapting 5W1H (who, what, when, where, why and how) framework to represent semantic elements of news events, the answers to these key questions are considered to result in summaries with more fine-grained elements. Therefore, following the SumCOT (Wang et al., 2023) approach, we first employ the same set of questions to prompt LLM to generate answers that identify elements such as entities, dates, and events. Moreover, given that the summaries for this task need to be concise, we adjust the questions to align more closely with 5W rules, as shown in Table 1. These answers, along with the corresponding news articles and questions, are then used as input to generate the final summary.

3.2 Instruction Tuning

The above method enables task adaption for LLMs without updating any parameters. However, fine-tuning can be a more effective method to align with desired downstream tasks when the examples from the target domain are available (Ouyang et al.,

SumCOT
What are the important entities in this document? What are the important dates in this document? What events are happening in this document? What is the result of these events?
5WCOT
Who is involved? (Identify all key entities.) What happened? (Describe the main event or action.) Where did it occur? (Provide the location or setting.) When did it take place? (Specify the date and time, if applicable.) Why did it happen? (Explain the causes, reasons, or purposes behind the event.)

Table 1: Guiding Questions for COT Method

2022; Taori et al., 2023). Specifically, instruction tuning (IT) is the process of fine-tuning LLMs with instruction-response pairs that use labeled data to improve performance (Wei et al., 2022a). Since full-model fine-tuning requires significant computational resources, parameter-efficient fine-tuning (PEFT) has been introduced, which allows for training only on a small set of additional parameters (Houlsby et al., 2019). Therefore, in this work, we use the QLoRA (Dettmers et al., 2023) method for supervised instruction tuning on the given query-answer pairs using the labeled dataset, which allows for the fine-tuning of a quantized 4-bit model with low-rank adapter weights (Hu et al., 2022). To construct the input prompt for training, we follow the provided instruction in the dataset (Xie et al., 2024), as detailed in Table 2, where the '{text}' and '{answer}' denote the corresponding fields in the dataset. During the inference phase, we exclude the content after 'Answer:[/INST]' to prompt the model to generate summaries.

Training Prompt Template
<pre><s>[INST]You are given a text that consists of multiple sentences. Your task is to perform abstractive summarization on this text. Use your understanding of the content to express the main ideas and crucial details in a shorter, coherent, and natural sounding text. Text:{text} Answer:[/INST]{answer}</s></pre>

Table 2: Template for IT Method

3.3 Reinforcement Learning

The IT method can improve performance (Wei et al., 2022a), however, there’s still room for improvement using the reinforcement learning. Lambert et al. (2022) proposed training a language model using proximal policy optimization (PPO) to further align the model human feedback. Recently, approaches that adapt the final result as the reward signal as outcome supervision has been to solve math problems (Lightman et al., 2024). Inspired by these works, we further investigate the outcome supervision for this task by using a combination of final performance metrics as the reward function to provide reward signals in PPO training. We construct the summary-level reward S by averaging the ROUGE-1/2/L scores and BERTScore, as detailed in Equation 1. Moreover, we incorporate a length penalty L , derived from the BLEU score method (Papineni et al., 2002), to constrain the length of the generated summary relative to the reference summary. As shown in Equation 2, c and r represent the word counts of the generated candidate summary and the reference summary, respectively, with words separated by blank spaces.

$$S = L \times \left[\frac{\text{ROUGE-1} + \text{ROUGE-2}}{4} + \frac{\text{ROUGE-L} + \text{BERTScore}}{4} \right] \quad (1)$$

$$L = \begin{cases} e^{(1-c/r)} & \text{if } c > r, \\ e^{(1-r/c)} & \text{if } c \leq r, \\ 0 & \text{if } c = 0 \end{cases} \quad (2)$$

4 Experimental Setup

Data Preprocessing We split the dataset into two subsets for model training and validation, allocating 80% training and 20% for validation via random seed.

Baseline Journalists commonly address the 5W questions within the first few sentences of an article to highlight the core event (Hamborg et al., 2018). Hence, we use the first sentence of each

news article as a summary to serve as our baseline for analysis (First Sentence).

Implementation For the few-shot method, following the SumCOT approach, we use the GPT-3.5 model through the OpenAI API¹. For instruction fine-tuning, we employ 4bit quantized Llama3-8b model (AI@Meta, 2024)², using the unsloth library³. This model training is conducted using a rank and alpha of 16 across all applicable modules, with a learning rate of $2e-4$. As for the final submission, we choose the fine-tuned checkpoint with the highest performance on the validation set to generate the results, which is trained for 600 steps. For the reinforcement learning approach, we employ the PPOTrainer from the trl (von Werra et al., 2020) library based on the best fine-tuned checkpoint with the learning rate of $5e-6$. Since the reward continues to decrease throughout the training process, we only report the model performance after 200 steps.

Evaluation In this shared task, We evaluate the quality of generated summaries through unigram (ROUGE-1) and bigram (ROUGE-2) overlap as well as the longest common subsequence (ROUGE-L) comparison to reference summaries (Lin, 2004). Besides using the n-gram based metrics, BERTScore (Zhang et al., 2020) is also employed, which computes the cosine similarity between their textual embeddings from a BERT-based model. Specifically, we use the evaluate library⁴ to calculate the performance scores, and use the multilingual BERT model⁵ for BERTScore F1 measurement.

5 Results

In this section, we compare the results of the three different approaches for generating financial abstractive summaries. In particular, we investigate the following research question:

- RQ: How effective are the three different methods in adapting LLMs for abstractive summarization in the financial domain?

To answer this question, we evaluate the three methods by comparing the generated summaries

¹<https://platform.openai.com/docs/models/gpt-3-5-turbo>

²<https://huggingface.co/unsloth/llama-3-8b-bnb-4bit>

³<https://github.com/unslothai/unsloth>

⁴<https://github.com/huggingface/evaluate>

⁵<https://huggingface.co/google-bert/bert-base-multilingual-cased>

Table 3: Overall Performance

Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
First Sentence	0.3443	0.1808	0.2872	0.6992
SumCOT	0.3002	0.1453	0.2397	0.69
5WCOT	0.3082	0.1511	0.2439	0.6923
IT	0.5348	0.358	0.4924	0.8074
RL	0.4944	0.3294	0.4577	0.7906

against reference summaries using evaluation metrics such as ROUGE-1, ROUGE-2, ROUGE-L, and BERTScore, with higher scores indicating better performance. In addition to LLM-based models, we have included a baseline denoted First Sentence. As we can see from Table 3, this model establishes foundational performance with a ROUGE-1 score of 0.3443, ROUGE-2 score of 0.1808, ROUGE-L score of 0.2872, and a BERTScore of 0.6992.

Chain-of-Thought Techniques: In few-shot scenarios, both SumCOT and 5WCOT show comparable performance, with 5WCOT slightly improving overall performance when refined guiding questions are used. However, in terms of ROUGE scores, COT methods perform worse than the baseline, with the highest ROUGE-1 score achieved by a COT method being only 0.3082, the highest ROUGE-2 score being only 0.1511, the best ROUGE-L score being 0.2439 and the best BERTScore only reaching 0.6923.

Instruction tuning (IT): When using this approach, we observe a considerable improvement over the baseline (between 15% and 99% improvements, depending on the metric), chain-of-thought and reinforcement learning methods, achieving the highest performance scores. Therefore, we have submitted this result as our submission to the shared task, with the detailed performance on the leaderboard shown in Table 4.

Reinforcement learning (RL): Finally, reinforcement learning achieves a notorious improvement over the First Sentence baseline and the chain-of-thought approaches. However, it lowers the performance of the best fine-tuned checkpoint (IT) in all the studied metrics (around 7-8% in the case of ROUGE metrics, 2.1% for BERTScore). This suggests that employing standard metrics with reference summaries as reward signals may not effectively guide the model toward developing better strategies for generating financial news summaries.

Conclusions: Overall, when assessing abstractive summarization with headlines as the gold standard,

Table 4: Our Submission on Leaderboard

Metrics	Performance
ROUGE-1	0.5346
ROUGE-2	0.3581
ROUGE-L	0.4922
BERTScore	0.9117
BARTScore	-3.4076

it is clear that using the first sentence as a summary forms a strong baseline. Additionally, instruction tuning is essential to ensure that the model’s output aligns with the desired summaries.

6 Conclusions

This work mainly explores the application of COT, IT and PPO method to adapting the LLM for financial abstractive summarization task. Surprisingly, the IT method surpasses both COT and PPO methods, achieved the highest performance and the 1st rank in this shared task. Although the other two approaches failed at this task, their results also indicate that relying only on the standard performance metrics based on a single reference summary to evaluate the quality of the LLM-generated summary may be insufficient and may not provide a useful signal for the LLM to learn more effective summarization strategies. Additionally, the brevity of headline contents may limit the evaluation in terms of informativeness and user interest. The effectiveness of automatic metrics is closely dependent on the quality of reference summaries and the preferences of different annotators can vary when evaluating the same summary (Zhang et al., 2024). Therefore, particularly in the domain of finance, it is beneficial to identify the target consumer and their preferences. For example, previous tracks on temporal and crisis summarization (Aslam et al., 2014; McCreadie and Buntain, 2023) evaluated performance by assessing the coverage of information and the similarity to the user query. Similarly, Böhm et al. (2019); Lambert et al. (2022) suggest constructing the reward function directly from human ratings instead of the existing metrics. Overall, for the future direction, we would employ human-in-loop approaches that evaluate summaries based on the aspects that are important to the target user. By integrating human feedback into the evaluation process, it becomes more possible that the summaries capture essential topics while addressing the specific interests of the financial domain.

References

- AI@Meta. 2024. [Llama 3 model card](#).
- Javed A. Aslam, Fernando Diaz, Matthew Ekstrand-Abueg, Richard McCreddie, Virgil Pavlu, and Tet-suya Sakai. 2014. TREC 2014 Temporal Summarization Track Overview. In *23rd Text REtrieval Conference (TREC 2014)*, Gaithersburg, Maryland, USA.
- Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O’Reilly Media, Inc.
- Florian Böhm, Yang Gao, Christian M. Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. 2019. [Better rewards yield better summaries: Learning to summarise without references](#). In *2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*, pages 3110–3120, Hong Kong, China. Association for Computational Linguistics.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language Models are Few-Shot Learners](#). In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, pages 1877–1901, Virtual Event. Curran Associates, Inc.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [QLoRA: Efficient Finetuning of Quantized LLMs](#). In *37th Conference on Neural Information Processing Systems (NeurIPS 2023)*, New Orleans, LA, USA.
- Felix Hamborg, Soeren Lachnit, Moritz Schubotz, Thomas Hepp, and Bela Gipp. 2018. [Giveme5W: Main Event Retrieval from News Articles by Extraction of the Five Journalistic W Questions](#). In *iConference 2018*, pages 356–366, Sheffield, United Kingdom.
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. [Parameter-efficient transfer learning for NLP](#). In *36th International Conference on Machine Learning (ICML 2019)*, pages 2790–2799. PMLR.
- Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *10th International Conference on Learning Representations (ICLR 2022)*, Virtual Event.
- Nathan Lambert, Louis Castricato, Leandro von Werra, and Alex Havrilla. 2022. [Illustrating Reinforcement Learning from Human Feedback \(RLHF\)](#). *Hugging Face Blog*.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. [Let’s Verify Step by Step](#). In *12th International Conference on Learning Representations (ICLR 2024)*, Vienna, Austria.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Richard McCreddie and Cody Buntain. 2023. [Crisis-facts: Buidling and evaluating crisis timelines](#). In *20th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2023)*, pages 320–339, Omaha, NE, USA.
- Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2022. [Show your work: Scratchpads for intermediate computation with language models](#). In *Deep Learning for Code Workshop (DLAC 2022) at the 10th International Conference on Learning Representations (ICLR 2022)*, Virtual Event.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). In *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, New Orleans, LA, USA.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [BLEU: a method for automatic evaluation of machine translation](#). In *40th Annual Meeting on Association for Computational Linguistics (ACL 2002)*, page 311–318, Philadelphia, Pennsylvania. Association for Computational Linguistics.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. [Stanford alpaca: An instruction-following llama model](#). https://github.com/tatsu-lab/stanford_alpaca.
- Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, and Shengyi Huang. 2020. [TRL: Transformer Reinforcement Learning](#). <https://github.com/huggingface/trl>.
- Yiming Wang, Zhuosheng Zhang, and Rui Wang. 2023. [Element-aware summarization with large language](#)

- models: Expert-aligned evaluation and chain-of-thought method. In *61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8640–8665, Toronto, Canada. Association for Computational Linguistics.
- Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022a. [Finetuned language models are zero-shot learners](#). In *10th International Conference on Learning Representations (ICLR 2022)*, Virtual Event.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022b. [Chain-of-thought prompting elicits reasoning in large language models](#). In *36th Annual Conference on Neural Information Processing Systems (NeurIPS 2022)*, pages 24824–24837, New Orleans, LA, USA. Curran Associates, Inc.
- Adhika Pramita Widyassari, Supriadi Rustad, Guruh Fajar Shidik, Edi Noersasongko, Abdul Syukur, Afandy Affandy, and De Rosal Ignatius Moses Setiadi. 2022. [Review of automatic text summarization techniques & methods](#). *Journal of King Saud University - Computer and Information Sciences*, 34(4):1029–1046.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2024. Pixiu: a large language model, instruction data and evaluation benchmark for finance. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23*, Red Hook, NY, USA. Curran Associates Inc.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. [BERTScore: Evaluating Text Generation with BERT](#). In *8th International Conference on Learning Representations (ICLR 2020)*, Virtual Event.
- Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B. Hashimoto. 2024. [Benchmarking Large Language Models for News Summarization](#). *Transactions of the Association for Computational Linguistics*, 12:39–57.
- Zhihan Zhou, Liqian Ma, and Han Liu. 2021. [Trade the Event: Corporate Events Detection for News-Based Event-Driven Trading](#). In *Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021)*, pages 2114–2124, Virtual Event. Association for Computational Linguistics.

Wealth Guide at the FinLLM Challenge Task: A Sophisticated Language Model Solution for Financial Trading Decisions

Sarmistha Das¹, R E Zera Marveen Lyngkhai¹, Sriparna Saha¹ and Alka Maurya²

¹Department of Computer Science and Engineering, Indian Institute of Technology Patna

² CRISIL Limited

sarmistha_2221cs21@iitp.ac.in, zeramarveenlyngkhai@gmail.com

Abstract

Single stock trading (SST) is surging in the Fintech era, fueled by tech advances and enhanced trading platforms. Meanwhile, natural language processing (NLP) is revolutionizing finance, with advanced AI and large language models (LLMs) leading the charge. This paper represents our participation in FinNLP-AgentScen-IJCAI 2024 (Joint Workshop of the 8th Financial Technology and Natural Language Processing and the 1st Agent AI for Scenario Planning). The primary objective of this task is to assess the capacity of LLMs to execute sophisticated trading decisions ("buy", "sell", "hold") grounded on a fusion of open-source stock and ETF (exchange traded fund) data. Our methodology centers on integrating news articles with their sentiment scores and correlating them with the stock price on the corresponding day, leveraging LLaMA-2-13 billion. Furthermore, we have explored various LLMs, including Mistral, Gemma, subjecting them to knowledge transfer and additional fine-tuning procedures in zero-shot and few-shot settings. Our model secured the first position in the SST task with a 0.926 Sharpe ratio. Our resultant findings underscore, with sufficient context and information, LLMs can perform these tasks effectively without including historical data.

1 Introduction

Single stock trading (SST) is gaining significant traction due to its strong correlation with making money. Due to persistent inflation and the transition to higher trend inflation and interest rates, the global equity markets are expected to face challenges in 2024¹. Despite the challenges posed by persistent inflation and the transition to higher trend rates, the New York Stock Exchange (NYSE) maintains its position as the world's largest stock exchange, boasting an equity market capitalization exceeding 28 trillion U.S. dollars as of March

¹<https://www.troweprice.com>

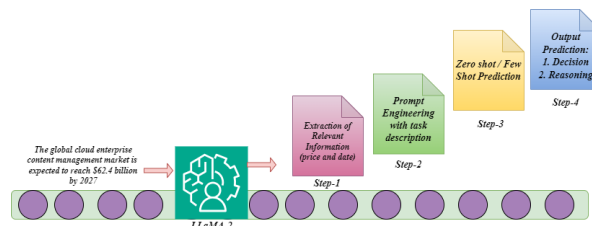


Figure 1: Workflow of our proposed approach with LLaMA-2 in Single Stock Trading prediction task

2024². The proliferation of stock-related information disseminated through various channels, such as news outlets and Twitter, has been instrumental in helping investors analyze market trends. This abundance of data has spurred the adoption of Natural Language Processing (NLP) techniques to explore the intricate relationship between textual data and fluctuations in stock prices, as evidenced by the studies conducted by Xu et al. (Xu and Cohen, 2018) and Oliveira et al. (Oliveira et al., 2017). In 2021, Zhou et al. (Zhou et al., 2021) demonstrated the importance of textual features (e.g., bag-of-words) and sentiments to directly make stock predictions. The advancement of Large Language Models (LLMs) has revolutionized trading agents by addressing many concerns in NLP (Achiam et al., 2023). Models like GPT-3 (Brown et al., 2020), LLaMA (Touvron et al., 2023), and Flan-T5 (Chung et al., 2022) demonstrate unique architectures and diverse language pipelines (Raffel et al., 2020a; Zhuang et al., 2021). These LLMs, with carefully designed prompts, can articulate reasoning and outcomes, enabling immediate scrutiny and adjustment of their decision-making processes. LLMs enhance decision-making by integrating extensive pre-trained knowledge with insights from diverse textual and numerical data sources, overcoming the limitations of isolated environments (Wang et al., 2024). Prompt-guided reasoning

²<https://www.statista.com>

has been shown to significantly improve problem-solving across various domains (Wang et al., 2023). Financial LLMs like FinGPT (Liu et al., 2023), BloombergGPT (Wu et al., 2023), and InvestLM (Yang et al., 2023) are increasingly researched for informed trading decisions, interacting dynamically with financial data and leveraging large parameter configurations. FinMA (Xie et al., 2023), designed for financial instruction tuning, utilizes a dataset of 136K financial samples to enhance its performance in financial decision-making. FinBen (Xie et al., 2024) became the first open-source financial benchmark, encompassing 35 datasets across 23 financial tasks. Subsequently, FINMEM (Yu et al., 2024) is proposed, a novel LLM-based framework for financial decision-making comprising profiling, memory with layered message processing, and decision-making modules.

We participated in the FinLLM challenge, inspired by recent advancements in large language models (LLMs) for finance. It was organized by FinNLP-AgentScen-2024 IJCAI and supported by project JPNP20006 commissioned by NEDO; the competition featured three tasks: financial classification, financial text summarization, and single-stock trading. Our team secured the top position for Task 3, the single stock trading task with a Sharpe ratio of 0.926 by leveraging our model³. Subsequently, we secured 4th place in Task 1 based on the financial text classification task and 5th position in Task 2 focused on the financial text summarization task.

2 FinLLM Challenge-Shared Tasks

The FinLLM challenge focuses on evaluating the capabilities of large language models (LLMs) in the financial domain across three primary tasks: 1) Financial classification, which aims to categorize sentences as claims or premises; 2) Financial text summarization, which involves abstracting financial texts into concise summaries; and 3) Single stock trading, which aims to make predictable decisions regarding stock trading. This challenge builds upon the FinLLM (Xie et al., 2024) shared tasks. Our participation concentrated on all three tasks, starting with Task 1, which involved financial text prediction for identifying "premise" or "claim," Task 2 focused on concisely summarising abstract financial news articles with 8,000 training samples and 2,000 test samples and Task 3, which focused

on single stock prediction with the primary objective of making informed decisions to "buy," "hold," or "sell" a stock. For Task 1, we were provided with 7,750 training samples and 969 testing samples. In contrast, Task 3 encompassed a dataset of 291 data points, integrating both open-source stock and ETF data. The dataset included separate training and test sets with attributes such as stock_id, price, date, query, and news. Table 1 depicts the instances of the dataset we received for Task 3.

3 Methodology

In this section, we briefly discuss two tasks, with our prime focus on Task 3, followed by Task 1 and Task 2.

3.1 Problem Statement for Task-3

Given a dataset D consisting of financial news texts and associated metadata attributes such as price, date, trading_id, and news as $D = \{(p_i, d_i, t_i, n_i)\}_{i=1}^N$; our objective is to predict the trading decision for a particular stock for each record; where, $p_i \in \mathbb{R}$ is the price of the stock at time i , $d_i \in \mathbb{D}$ is the date at time i , $t_i \in \mathbb{T}$ is the trading identifier at time i , $n_i \in \mathbb{N}$ is the financial news text associated with the stock at time i . The prime goal is to learn a function $f : \mathbb{N} \times \mathbb{R} \times \mathbb{D} \times \mathbb{T} \rightarrow \{\text{"buy"}, \text{"sell"}, \text{"hold"}\}$ using an LLM such that for a given set of inputs, the function f predicts the appropriate trading decision.

3.2 Approaches

For single stock prediction, we utilize summarized news N_i and their sentiment scores in conjunction with the stock price P_i . These elements are appended as context to the query Q_i , forming the prompt $\text{Prompt}_i = N_i + P_i + Q_i$. Prompt_i is then fed into the model for zero-shot prediction. The model output Y is preprocessed to extract both the decision d and the reasoning r behind it. The textual decision is converted into a numerical representation with mappings: buy = 1, sell = -1, hold = 0.5, and no decision = 0. To evaluate performance, the score is calculated by multiplying the action taken on each day (action_i) with the return of that day (daily_i), formally expressed as $\text{returns}_i = \text{daily}_i \times \text{action}_i$, where returns_i represents the return on day i , daily_i is the daily return on day i , and action_i is the action taken on day i . The Sharpe ratio is then computed as

³Codes are available here: https://github.com/sarmistha-D/Wealth_Guide-FinLLM2k24

Table 1: Sample instance of the given Single Stock Trading dataset

id	jinj_test0
date	"2020-10-09"
price	{ "DRIV": 17.52210235595703 }
filing_k	"FORM": "null"
filing_q	{ "FORM": "null" }
news	"DRIV": ["The global cloud enterprise content management market is expected to reach \$62.4 billion by 2027, driven by a CAGR of 25.6% and significant growth in the U.S. and China. The positive score for this news is 2.3659735504111268e-08. The neutral score for this news is 0.9999990463256836. The negative score for this news is 9.636863751438796e-07.", "The global emergency lighting batteries market is expected to reach \$2.8 billion by 2027, growing at a CAGR of 10.8% despite the COVID-19 pandemic's impact. The positive score for this news is 1.166244 1465887241e-05. The neutral score for this news is 0.9995514750480652. The negative score for this news is 0.000436866597738117.", "Despite the impact of the COVID-19 pandemic, the global market for two-wheeler spark plugs is expected to reach 86.2 million units by 2027, growing at a CAGR of 4.9%. The positive score for this news is 1.1285221262369305e-05. The neutral score for this news is 0.998855113983 1543. The negative score for this news is 0.0011336031602695584.", "Despite pandemic setbacks, the global market for two-wheeler upside-down forks is expected to reach 701.8 thousand units by 2027, driven by growth in China and the U.S. The positive score for this news is 9.909140175068387e-08. The neutral score for this news is 0.9999970197677612. The negative score for this news is 2.81238385468896e-06.", "The global embedded analytics market is expected to reach \$84.6 billion by 2027, driven by a 13% CAGR."]

Sharpe Ratio = $\frac{R_p - R_f}{\sigma_p}$, where R_p is the portfolio's average excess return, R_f is the risk-free rate, and σ_p is the portfolio's volatility, with a Sharpe ratio of 1 being considered good as mentioned in (Yu et al., 2024). For the competition, we tested the llama2-13B model, which was fine-tuned on a financial summarization dataset using zero-shot prompting, as well as other models such as Gemma-7b and Mixtral-7b.

3.3 Sentiment Consideration

In 2021, Zhou et al. (Zhou et al., 2021) showed that textual features (e.g., bag-of-words) and sentiments are crucial for stock predictions, leveraging corporate events as key drivers of stock movements to profit from temporary mispricing. Inspired by this notion, we incorporated sentimental consideration into our model. For the sentiment score of the news the model used was FinBert (Yu et al., 2024) where we have summarized news N_i as input and output is positive score pos_i , negative score neg_i and neutral score neu_i , which are appended to the summarized news $News_i$ to get $N_i = N_i + pos_i + neu_i + neg_i$ as final news input to the model.

3.4 Definition of Task-1

In the FinLLM challenge, we actively participated in Task 1, which entailed a financial text classification task. Our objective was to determine whether a given financial text constitutes a *premise* or a *claim*. For a given input text T_i , we aim to learn a function, $C : \mathbb{N} \times \mathbb{T} \rightarrow \{"premise", "claim"\}$. We received 7,750 training data samples (Please refer to the appendix section D) To this end, we fine-tuned several state-of-the-art language models across 5-10 epochs each to perform this task.

Table 2: Ablation studies among different generative based language models on Financial Text classification(Task-1) task

MODEL	F1	Accuracy
BERT	72.42	56.75
T5-Small	68.43	52.01
Bart	64.29	47.37
DistilBert	75.13	50.14

3.5 Definition of Task-2

Our participation in Task 2 explores the capabilities of large language models (LLMs) in summarizing financial documents. Using a specific prompt template, we framed the input text as multi-sentence financial news and the output as its abstractive summary. Our goal is to learn a function $S : \mathbb{T} \rightarrow \mathbb{S}$, mapping input texts \mathbb{T} to their summaries \mathbb{S} . We conducted fine-tuning of a leading-edge language model over 3-5 epochs to achieve this objective.

3.6 Experimental Setup

For the Task-3 experiments, we utilized three language models: Using a zero-shot and few-shot setting, we combined news articles with sentiment scores and stock prices to generate decision predictions using reasoning. These experiments were conducted on an NVIDIA GeForce RTX 3090 24GB, operating with 4-bit precision. All the experiments were conducted using the same hyperparameter settings: temperature =1, top_k=5,do_sample=True, and max_new_token=350. For the financial text classification task(Task-1), the learning rate was set to 2e-05, and the optimizer was Adam, with a weight decay of 0.01.

4 Experimental Results and Discussions

In this section, we briefly discuss the experiments and research findings of Task 1 and Task 3; Task

Table 3: Ablation studies among different language models on SST (Single Stock Trading) prediction task (Task-3); DRIV, JNJ, MSFT are stock names and SR-stands for Sharpe Ratio

Experimental Settings		Models			
		Phi-1.3b	Gemma-7b	Mistral-7b	LLaMA2-13b
Zero shot settings	SR	0.1643	0.1252	0.1037	0.6976
	SR- DRIV	-0.1038	0.1329	0.07715	0.48563
	SR- JNJ	-0.3207	0.1176	0.09508	0.078737
	SR- MSFT	-0.0685	0.1250	0.13916	1.55556
Few shot settings	SR	0.89321	0.1132	0.9132	0.6756
	SR- DRIV	0.13109	0.1129	0.06715	0.42563
	SR- JNJ	0.0867	0.1147	0.09528	0.077437
	SR- MSFT	0.1210	0.1390	0.14976	1.23556

Input	Gemma-7B	Mistral-7B	LLaMA-2-13B (Wealth Guide)
Price = 11.237374305725098 News= [The global recloser control market is expected to grow significantly, driven by increasing demand for power quality and reliability, especially in the electric segment and emerging economies like China. The positive score for this news is 1.1491235518690246e-08. The neutral score for this news is 0.999998807907104. The negative score for this news is 6.358970239261907e-08.]	Decision: Reasoning: ... python buy ...	{'sentiment': 0.14141012581704443, 'sentiment_score': {'buy': {'value': 0.14141012581704443, 'label': 'buy'}, 'hold': {'value': 0.0, 'label': 'hold'}, 'sell': {'value': 1.0, 'label': 'sell'}}, 'score': {'value': 1.1491235518690246e-08, 'label': 'buy'}, 'text': 'The global recloser control market is expected to grow significantly, driven by increasing demand for power quality and reliability, especially in the electric segment and emerging economies like China. The positive score for this news is 1.1491235518690246e-08. The neutral score for this news is 0.999998807907104. The negative score for this news is 6.358970239261907e-08.'}, 'text_score': {'value': 0.999998807907104, 'label': 'neutral'}, 'text_type': 'news'}	Decision: Hold Reasoning: The news indicates that the global recloser control market is expected to grow significantly, which could lead to an increase in demand for power quality and reliability. However, the positive score is relatively low, indicating that the market may not grow as much as expected. Additionally, the neutral score is close to the positive score, indicating that the market may be relatively stable. Therefore, it is best to hold onto the investment and monitor the market trends before making any decisions.

Figure 2: Qualitative Analysis between our proposed LLaMA-2, Gemma-7b and Mistral-7b

2 results are mentioned in appendix section E. We conducted the experiments for Task 1, a binary classification to classify "premise" and "claim" class labels with F1-score and accuracy metrics. We compared the results with four generative language-based models, for instance, BERT (Devlin et al., 2018), BART (Lewis et al., 2019), T5-small (Raffel et al., 2020b) and DistilBERT (Sanh et al., 2019). To conduct Task 3 of the SST task experiment, we compared the results of our model, LLaMA-2-13B (Touvron et al., 2023), with three other LLM models, Gemma-7B (Team et al., 2024), Phi-1.3B (Gunasekar et al., 2023) and Mixtral-7B (Jiang et al., 2023) under two conditions: "few-shot" and "zero-shot" settings.

4.1 Resultant Discussion

Evidently, in the financial text classification task (Task-1) detailed in Table 2, the DistilBERT model demonstrated superior performance compared to other models fine-tuned under identical configurations. Notably, BART exhibited the poorest performance, which can be attributed, in part, to its propensity for accelerated overfitting relative to DistilBERT. This overfitting issue is less pronounced in DistilBERT, likely due to its more compact architecture. Subsequently, Table 3

illustrates the performance comparison between DistilBERT, popular LLM models, and our model, LLaMA-2-13B. Our model significantly outperformed the other models across most metrics, except for the Sharpe Ratio for JNJ in both zero-shot and few-shot settings. In these metrics, Gemma-7B outperformed LLaMA-2-13B in both settings. However, Mistral-7B performed lower than the other LLMs but still better than Phi-1.3b. Conclusively, it is evident that LLMs are a better choice for large datasets than generative language models due to their superior ability to make context-aware inferences. Additionally, Mistral-7B and Gemma-7B exhibited inconsistent outputs when tested multiple times on the same data inputs. In contrast, LLaMA-2-13B consistently provided accurate decisions and reasoning for all test inputs, achieving an overall Sharpe ratio of 0.6976. This consistency and higher performance can be attributed to LLaMA-2-13B's larger model size and greater number of parameters. Figure 2 depicts two output instances where our model LLaMA-2 performed significantly better compared to Mistral-7b and Gemma-7b.

5 Conclusion

In this paper, we presented our approach for the FinNLP-2024 shared task, FinNLP-AgentScen-IJCAI 2024, evaluating LLM capabilities in financial tasks. Using 291 news & price data points without historical data, we leveraged the LLaMA-2-13B model, known for generating coherent and contextually accurate text. We developed sentiment-score-based trading (SST) prediction model to take "hold", "sell", and "buy" decisions by adapting LLaMA-2 to predict news sentiment and determine decisions based on the sentiment score. This approach helped us to secure the top position in Task -3 of the FinLLM-2024 challenge.

Acknowledgement

This work is a collaborative effort between the Indian Institute of Technology Patna and CRISIL. We thank Dr. Sriparna Saha for her thorough review throughout the competition and the writing process. Additionally, the authors acknowledge the assistance provided by CRISIL Limited.

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*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.
- 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*.
- Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. [Textbooks are all you need](#).
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b](#).
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Xiao-Yang Liu, Guoxuan Wang, and Daochen Zha. 2023. Fingpt: Democratizing internet-scale data for financial large language models. *arXiv preprint arXiv:2307.10485*.
- Nuno Oliveira, Paulo Cortez, and Nelson Areal. 2017. The impact of microblogging data for stock market prediction: Using twitter to predict returns, volatility, trading volume and survey sentiment indices. *Expert Systems with applications*, 73:125–144.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020b. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
- Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shrutu Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.
- Peiyi Wang, Lei Li, Liang Chen, Feifan Song, Binghuai Lin, Yunbo Cao, Tianyu Liu, and Zhifang Sui. 2023. Making large language models better reasoners with alignment. *arXiv preprint arXiv:2309.02144*.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kam-badur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. 2024. The finben: An holistic financial benchmark for large language models. *arXiv preprint arXiv:2402.12659*.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *arXiv preprint arXiv:2306.05443*.
- Yumo Xu and Shay B Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979.
- Yi Yang, Yixuan Tang, and Kar Yan Tam. 2023. Investlm: A large language model for investment using financial domain instruction tuning. *arXiv preprint arXiv:2309.13064*.

Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W Su-chow, and Khaldoun Khashanah. 2024. Finmem: A performance-enhanced llm trading agent with layered memory and character design. In *Proceedings of the AAAI Symposium Series*, volume 3, pages 595–597.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. [Trade the event: Corporate events detection for news-based event-driven trading](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized bert pre-training approach with post-training. In *Proceedings of the 20th chinese national conference on computational linguistics*, pages 1218–1227.

A Ethical Consideration

This paper includes limited information about various stock names, values, and financial organization names, primarily provided by the FinLLM challenge organizers. The authors do not endorse or recommend investing in any specific stock. The primary objective of this research is to explore the capabilities of large language models (LLMs) in the financial domain and to contribute to the advancement of research in the FinTech sector. The authors have deliberately refrained from offering specific trading advice.

B Limitations and Future work

LLM tend to hallucinate in zero-shot tasks and even few shot task when no proper context are given. Using bigger LLM require a lot of resources to run and train them as opposed to smaller llm. Using smaller llm though has its limitation like sacrificing better performance in terms of output as compared to bigger model. Also different LLM responds differently to the same prompts so Prompt engineering might be required for zero shot and few shot task. Also for task like classification Bigger llm tend to overfit faster as oppose to smaller model during finetuning.

Future Work: Our future endeavours encompass enhancing Task 3 by incorporating historical data, as such data provides crucial insights. Additionally, we aim to extend the task to support multilingualism, thereby creating new opportunities and avenues for research and application.

C Prompt Construction

For efficient results, we tested with multiple prompts, and we found the following two prompt variants to work for multiple models .

Prompt-1:

Instruction:

Given this context: {context} and price: {price}, output only one decision from the square bracket [buy, sell, hold] and provide reasoning on why.

Response:

Decision:

Reasoning:

Prompt-2:

Instruction:

Given this context: {context} and price: {price}, output only one decision from the square bracket [buy, sell, hold] and provide reasoning on why. The output of the Decision should be only one of [buy, sell, hold].

Response:

Decision:

Reasoning:

D Hidden details about Task-1 and Task-3

In the FinLLM challenge, The main focus of Task 1 is on classifying financial texts as either 'premise' or 'claim'. The mapping should be done from input features to the classes 'premise' or 'claim' for a given input text. We utilized 7,500 training samples for fine-tuning various advanced language models over 5-10 epochs each to achieve optimal performance on this task. Table 5 illustrates a sample instance of the given dataset. In Task 1, a total of 8 different teams participated. Our team, Wealth Guide, secured 4th place in this competition. Table 4 depicts the overall performance of the participated teams for task-1.

Table 4: Performance Comparison between the proposed model, Wealth Guide and other competitive teams.

Team Name	Accuracy	F1-Score	MCC
Team Barclays	0.7626	0.5237	0.7427
Albatross	0.7575	0.5174	0.7555
L3iTC	0.7544	0.5149	0.7581
Wealth Guide	0.7513	0.5018	0.7406
Finance Wizard	0.7286	0.4554	0.7008
CatMemo	0.711	0.4199	0.6818
Upaya	0.709	0.4166	0.6941
jt	0.4933	0.0141	0.590

Table 5: The sample instances received for Task-1; Financial text Classification task

Id	query	answer	text	choices
finargeccauc0	Analyze sentences from earnings conference calls and identify their argumentative function. Each sentence is either a premise, offering evidence or reasoning, or a claim, asserting a conclusion or viewpoint. Return only premise or claim. Text: I mean, sometimes it's not that you came up with some brilliant strategy, it's just like really good work consistently over a long period of time. Answer:	premise	I mean, sometimes it's not that you came up with some brilliant strategy, it's just like really good work consistently over a long period of time.	["premise", "claim"]
finargeccauc1	Analyze sentences from earnings conference calls and identify their argumentative function. Each sentence is either a premise, offering evidence or reasoning, or a claim, asserting a conclusion or viewpoint. Return only premise or claim. Text: Even while in International, we're continuing to invest in a lot of areas, we continue to frontload Prime benefits for the newer geographies, we continue to launch new countries as we launch Prime in Australia recently. Answer:	claim	Even while in International, we're continuing to invest in a lot of areas, we continue to frontload Prime benefits for the newer geographies, we continue to launch new countries as we launch Prime in Australia recently.	["premise", "claim"]

Table 6: The performance comparison between the proposed model, Wealth Guide, and other competitive teams for the Summary Generation oriented Task-2

Team	Metrics					
	Rouge-1	Rouge-2	Rouge-L	BertScore	BartScore	DLT
Finance Wizard	0.521037018	0.34060938	0.473530112	0.90836845	-3.497988865	1.7346
Upaya	0.529459817	0.358203218	0.486046685	0.910644962	-3.45155009	0.8332
Wealth Guide	0.308893532	0.179468097	0.281924302	0.85959909	-4.961457408	-
Albatross	0.369077581	0.201058395	0.322684316	0.872049115	-3.933526929	-
LBZ	0.534616211	0.358105428	0.492179554	0.911732047	-3.407560172	-
L3iTC	0.366093426	0.187210467	0.304610677	0.875037043	-4.257126737	-
Revelata	0.500411369	0.333023818	0.464356474	0.907018743	-3.805486962	-

For task-3, we received the dataset of financial news articles along with information such as stock prices, dates, and trading IDs. Our goal is to predict whether to "buy," "sell," or "hold" a stock based on this data. Each record in our dataset includes a) The stock price at a certain time, b) The date, c) A unique trading identifier, and d) The financial news text related to the stock at that time. To achieve significant predictions, we used a model called FinBert to analyze the sentiment of the news articles. FinBert summarizes each news article and provides three scores: positive, negative, and neutral. We then combine these scores with the summarized news to create a comprehensive input for our model. This combined input enhances our model's ability to accurately predict the appropriate trading decision. In Task 3, a total of four teams participated, and our proposed model, Wealth Guide, secured the top position. The detailed results are presented in Table 7.

Table 7: For Task-3, the performance comparison between the proposed model, Wealth Guide and other competitive teams

Metrics	Team			
	Wealth Guide	Upaya	Albatross	CatMemo
Sharpe Ratio	0.926385228	-0.467489019	-0.48383204	-0.619939784
Sharpe Ratio- DRIV	0.485625528	-0.380232272	-0.251306057	-1.393291177
Sharpe Ratio- FORM	1.585611423	-0.108506918	-1.435471054	0.175932289
Sharpe Ratio- JNJ	0.078737051	-1.102831656	-1.558522674	-0.383243051
Sharpe Ratio- MSFT	1.555566911	-0.278385232	1.309971626	-0.879157198

E Task 2 in brief

Task 2 aims to evaluate the effectiveness of large language models (LLMs) in summarizing financial documents. We were provided with 8,000 training samples and 2,000 test samples to transform financial news articles into concise summaries Using a specific prompt template—"Instruction: [task prompt] Context: [input context] Response: [output].". For this task, we finetuned the LLAMA2-13B model. The evaluation metrics utilized were the ROUGE score and BERT score. In the competition, LLAMA2-13B achieved a ROUGE-1 (R1) score of 0.3088, which was used for the final competition ranking, as indicated in Table 6. The

LLAMA2-13B model was finetuned with a learning rate of $2e-03$ using the AdamW optimizer on a 24GB NVIDIA RTX 3090 GPU.

L3iTC at the FinLLM Challenge Task: Quantization for Financial Text Classification & Summarization

Elvys Linhares Pontes¹ and Carlos-Emiliano González-Gallardo²
and Mohamed Benjannet¹ and Caryn Qu¹ and Antoine Doucet²

¹ Trading Central Labs, Trading Central, Paris, France

² University of La Rochelle, L3i, La Rochelle, France

{elvys.linhares_pontes,mohamed.benjannet,caryn.qu}@tradingcentral.com

{carlos.gonzalez_gallardo,antoine.doucet}@univ-lr.fr

Abstract

This article details our participation (L3iTC) in the FinLLM Challenge Task 2024, focusing on two key areas: Task 1, financial text classification, and Task 2, financial text summarization. To address these challenges, we fine-tuned several large language models (LLMs) to optimize performance for each task. Specifically, we used 4-bit quantization and LoRA to determine which layers of the LLMs should be trained at a lower precision. This approach not only accelerated the fine-tuning process on the training data provided by the organizers but also enabled us to run the models on low GPU memory. Our fine-tuned models achieved third place for the financial classification task with an accuracy of 75.44% and sixth place in the summarization task on the official test datasets.

1 Introduction

Financial markets are characterized by their complexity and the vast volume of unstructured data they generate daily. The use of Large Language Models (LLMs) in finance has brought significant focus to tasks involving the analysis, generation, and decision-making related to financial texts. Indeed, LLMs have demonstrated remarkable performance in a large range of applications, from conversational agents to complex decision-making systems. Despite the advances, their potential for thorough analysis and decision-making in finance is still unexplored.

The Financial Challenges in Large Language Models (FinLLM)¹ aims to investigate and enhance the role of LLMs in advancing financial analysis and decision-making processes (Xie et al., 2024). More precisely, it focuses on three applications: financial classification of sentences (Sy et al., 2023), financial news summarization (Zhou et al., 2021), and single stock trading (Yu et al., 2023).

¹<https://sites.google.com/nlg.csie.ntu.edu.tw/finllm-agent scen/shared-task-finllm>

Motivated by these challenges, we participated (L3iTC) on the financial text classification and financial text summarization tasks. We proposed a fine-tuning process that combine 4-bit quantization and LoRA to optimize several LLMs for each task. This approach accelerated the fine-tuning process on the training data provided by the organizers but also enabled us to run the models on low GPU memory. Our results secured the third place for the financial classification task with an accuracy of 75.44% and sixth place in financial text summarization on the official test datasets.

2 FinLLM Challenge Task

With the advent of LLMs in finance, tasks related to financial text analysis, generation, and decision-making have garnered increasing attention. Key applications in this domain include financial classification, financial text summarization, and single-stock trading. While several approaches utilizing LLMs have demonstrated remarkable performance in these areas, their capabilities for comprehensive analysis and decision-making in finance remain largely unexplored.

FinLLM aims to investigate and enhance the role of LLMs in advancing financial analysis and decision-making processes (Xie et al., 2024).

2.1 Task 1: Financial Classification

The first task aims to evaluate the capabilities of LLMs in identifying and categorizing texts as either premises or claims (Sy et al., 2023). This task is particularly challenging due to financial texts' nuanced and complex nature, where distinguishing between these concepts (claims and premises) requires sophisticated understanding and contextual analysis. The organizers provided a training dataset with 7.75k data examples and the official test dataset composed of 969 examples.

2.2 Task 2: Financial Text Summarization

This task is designed to test the capabilities of LLMs in generating coherent and concise summaries (Zhou et al., 2021). The challenge lies in the ability to accurately capture the essential points and nuances of complex financial news, ensuring that the summary remains both informative and coherent (Li et al., 2023). The organizers provided a training dataset with 8k data examples and the official test dataset composed of 2k examples.

2.3 Model Leakage Detection

To measure the risk of data leakage from the test set during model training, organizers have developed a new metric called the Data Leakage Test (DLT), building on existing research (Wei et al., 2023). DLT assesses the risk of data leakage by calculating the difference in perplexity between training and test data for large language models (LLMs). A larger DLT value indicates a lower likelihood of the LLM having seen the test set during training, suggesting a lower risk of model cheating, while a smaller DLT value suggests a higher risk of data leakage and model cheating.

2.4 Evaluation Metrics

For the financial text classification task, the organizers employed two primary evaluation metrics to gauge the performance of the participants' models: F1-score and accuracy. F1-score considers both precision and recall, providing a balanced measure of a model's accuracy. Accuracy represents the ratio of correctly predicted instances to the total instances.

For the financial text summarization task, organizers used ROUGE (1, 2, and L), BERTScore, and BARTScore metrics. ROUGE-n measures the overlap of n-grams between the generated summaries and the reference summaries. BERTScore calculates the similarity between the generated and reference summaries using sentence representation. Finally, BARTScore compares the generated summaries against a reference summary to determine how well the generated summaries capture the reference summaries' meaning, fluency, and coherence.

3 L3iTC Approaches

We participated in the first two tasks. We developed the following architecture to address these tasks to generate our fine-tuned LLM for the FinLLM shared task (Figure 1).

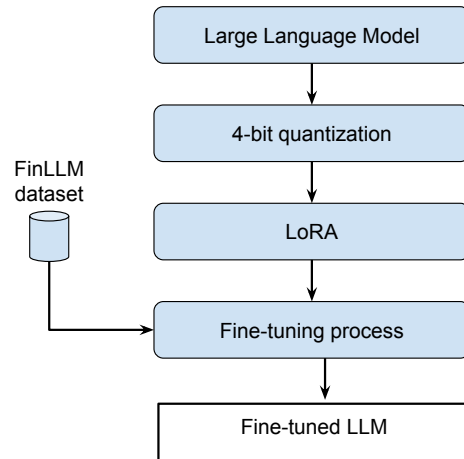


Figure 1: Fine-tuning methodology

3.1 Large Language Models

LLMs can perform a variety of natural language processing tasks such as translation, summarization, and conversational dialogue (Chang et al., 2024). They are trained on diverse datasets encompassing a wide range of topics, enabling them to generate coherent and contextually relevant responses. Among prominent LLMs available today, we selected the following Instruct models due to their high performance and relative small size: Mistral-7B-Instruct-v0.2², Mistral-7B-Instruct-v0.3³, and Meta-Llama-3-8B-Instruct⁴.

3.2 Fine-tuning

In the classic fine-tuning of LLMs, a significant portion of model weights is typically modified, necessitating substantial computational resources. To alleviate GPU memory requirements during fine-tuning, we employed quantization techniques as proposed by Dettmers et al. (2022). Specifically, we utilized 4-bit quantization to reduce the memory footprint of LLMs prior to fine-tuning.

To make fine-tuning more efficient, LoRA (Hu et al., 2021) improves efficiency by using two smaller matrices, known as update matrices, to represent weight updates via low-rank decomposition. These matrices are trained to adjust to new data while minimizing the total number of changes. The original weight matrix stays unchanged and is not further modified. The final results are

²<https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2>

³<https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3>

⁴<https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct>

achieved by combining the original weights with the adapted ones. In our study, we focused on training parameters within specific modules, including “q_proj”, “k_proj”, “v_proj”, “o_proj”, “gate_proj”, “up_proj”, “down_proj”, and “lm_head” employing a dropout rate of 0.05.

For both tasks undertaken, we partitioned the training dataset into three subsets: train, validation, and test. The validation and test subsets each comprised 10% of the examples, with the remaining 80% constituting the training data. We set the learning rate to 5×10^{-5} and the batch size to 4. The models underwent fine-tuning over 2,000 steps.

4 Preliminary results

Table 1 summarizes the performance of fine-tuning each LLM for Task 1. Initially, the models predicted more than just the target word, attempting to choose the correct class and justify their selection. This approach led to poor results. The best model without fine-tuning was Mistral-7B-Inst-v0.2, which achieved an accuracy of 54% and an F1-score of 0.39.

Despite the limitations encountered during the fine-tuning process, particularly those related to LoRA configuration and 4-bit quantization, all fine-tuned models showed improved performance by generating only the target predicted class. Notably, the best fine-tuned model was FT-Clas-Mistral-7B-Inst-v0.3, which achieved an accuracy of 78% and an F1-score of 0.78. Therefore, we selected it to compete on the first task on the official test dataset.

Team	Accuracy	F1
Mistral-7B-Inst-v0.2	54%	0.39
Mistral-7B-Inst-v0.3	46%	0.36
Meta-Llama-3-8B-Inst	52%	0.48
FT-Clas-Mistral-7B-Inst-v0.2	76%	0.76
FT-Clas-Mistral-7B-Inst-v0.3	78%	0.78
FT-Clas-Meta-Llama-3-8B-Inst	67%	0.67

Table 1: Preliminary fine-tuning results for financial classification task.

Table 2 summarizes the performance of fine-tuning each LLM for Task 2. Unfortunately, the fine-tuning process did not yield significant improvements in the ROUGE score and even resulted in a decline in BERTScore performance. The main reasons for the poor results are mainly related to our finetuning process. More precisely, the quantization process of 4-bits indeed reduces the amount of GPU memory necessary to fine-tune the model;

however, this quantization limited the precision of the learning process which also affected the quality of our models.

Although FT-Sum-Mistral-7B-Inst-v0.2 obtained the best ROUGE-1 score, which is used as the final ranking metric, we found that BertScore better correlates summary quality and human judgment (Table ?? lists some summaries generated by FT-Sum-Mistral-7B-Inst-v0.2 and Mistral-7B-Inst-v0.3). Thus, we selected the Mistral-7B-Inst-v0.3 model for the second task.

Team	ROUGE-1	BertScore
Mistral-7B-Inst-v0.2	22.45	0.5373
Mistral-7B-Inst-v0.3	22.48	0.5374
Meta-Llama-3-8B-Inst	22.40	0.5333
FT-Sum-Mistral-7B-Inst-v0.2	23.12	0.5097
FT-Sum-Mistral-7B-Inst-v0.3	22.50	0.502
FT-Sum-Meta-Llama-3-8B-Inst	22.31	0.488

Table 2: Preliminary fine-tuning results for financial text summarization task.

5 Official Results

The organizers created a test dataset consisting of 969 test cases for the first task and 2,000 test cases for the second task. The official results are listed in Tables 3 and 4 for tasks 1 and 2, respectively. For the financial classification task, our fine-tuned model achieved notable results, with an accuracy of 75.44% and an F1-score of 0.5149. Remarkably, we secured third place, trailing the first place by less than one percentage point in accuracy.

Team	Accuracy	F1	MCC	DLT
Team Barclays	76.26%	0.5237	0.7427	38.9
Albatross	75.75%	0.5174	0.7555	–
L3iTC	75.44%	0.5149	0.7581	2.2
Wealth Guide	75.13%	0.5018	0.7406	–
Finance Wizard	72.86%	0.4554	0.7008	–
CatMemo	71.10%	0.4199	0.6818	–
Upaya	70.90%	0.4166	0.6941	–

Table 3: Official results for the Financial classification task on the test dataset (Task 1).

As expected, our models did not achieve the best results, primarily due to limitations encountered during the fine-tuning process. Specifically, the use of 4-bit quantization constrained the learning process and adversely affected the quality of summary generation. Consequently, our model ranked sixth for the second task, with a ROUGE-1 score of 0.3661 and a BERTScore of 0.875.

Team	ROUGE-1	ROUGE-2	ROUGE-L	BertScore	BartScore	DLT
LBZ	0.5346	0.3581	0.4921	0.9117	-3.41	
Upaya	0.5294	0.3582	0.4860	0.9106	-3.45	-
Finance Wizard	0.5210	0.3406	0.4735	0.9083	-3.49	0.8332
Revelata	0.5004	0.3330	0.4643	0.9070	-3.80	1.7346
Albatross	0.3691	0.2010	0.3227	0.8720	-3.93	-
L3iTC	<i>0.3661</i>	<i>0.1872</i>	<i>0.3046</i>	<i>0.8750</i>	<i>-4.25</i>	-
Wealth Guide	0.3089	0.1794	0.2819	0.8595	-4.96	-

Table 4: Official results for the Financial text summarization task on the test dataset (Task 2).

Id	Mistral-7B-Inst-v0.3	FT-Sum-Mistral-7B-Inst-v0.2
edtsum0	The all-season tire market in Europe is projected to grow by USD 3.42 billion during 2020-2024, with a CAGR of nearly 9% ...	All-Season Tire Market in Europe 2020-2024 Evolving Opportunities with Bridgestone Corp. and Continental AG Technavio ...
edtsum2	The chocolate market is projected to grow by USD 41.15 billion from 2020 to 2024, with a CAGR of over 5% ...	Chocolate Market to Grow by USD 41.15 Billion During 2020-2024 Increasing Premiumization of Chocolates to boost the Market Growth Technavio Report English USA - English USA - English USA - ...
edtsum15	Aon PLC has experienced dealings with Farallon Capital Management, L.L.C. on behalf of funds managed jointly with Farallon Partners, L.L.C.	Form 8.3 - Aon PLC - 12 April 2021 - Farallon Capital Management, L.L.C. on behalf of funds managed jointly with Farallon Partners, L.L.C. - Amended - 13 April 2021 - 10:00 am - 10:00 am
	...	

Table 5: Examples of the generated summaries on the official test dataset. These examples demonstrate that the readability and informativeness are superior in the Mistral-7B-Inst-v0.3 model compared to the FT-Sum-Mistral-7B-Inst-v0.2 model.

6 Conclusion

This article presents our participation (L3iTC) in the FinLLM Challenge Task 2024, concentrating on two primary tasks: Task 1, financial text classification, and Task 2, financial text summarization. To tackle these challenges, we fine-tuned several LLMs to enhance their performance for each specific task.

For Task 1, our fine-tuning efforts led to a third-place finish, achieving an accuracy of 75.44%, just 0.82 percentage points behind the first place. In Task 2, we secured sixth place on the official test datasets. These outcomes demonstrate that combining LoRA configuration and 4-bit quantization allows for the efficient fine-tuning of LLMs minimizing GPU memory and processing time, yielding notable results for tasks that do not require the generation of numerous tokens. In addition, combining quantization and LoRA enables the possibility of fine-tuning LLMs in smaller infrastructures that

demand less energy thus reducing their carbon footprint (Samsi et al., 2023). However, when the number of tokens generated increases, as in the case of financial text summarization, this approach reveals its limitations. The quality of the generated summaries declines compared to those produced by the original LLMs without fine-tuning, highlighting the trade-offs involved in using this combination for tasks requiring extensive text generation.

Future work will focus on enhancing the fine-tuning process by employing 8-bit or 16-bit quantization and evaluating their performance on complex tasks such as summarization. Additionally, we aim to perform data augmentation for these datasets and integrate datasets from different tasks. This approach aims to develop a more robust model capable of handling various tasks without compromising the quality of generation.

Acknowledgments

This work has been supported by the ANNA (2019-1R40226), TERMITRAD (2020-2019-8510010), Pypa (AAPR2021-2021-12263410), and Actuadata (AAPR2022-2021-17014610) projects funded by the Nouvelle-Aquitaine Region (France); as well as the France Relance (ANR-21-PRRD-0010-01) project funded by the French National Recherche Agency (ANR). We would like to also thank José G. Moreno for the insightful discussions.

References

- Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.
- Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. *Gpt3.int8(): 8-bit matrix multiplication for transformers at scale*. In *Advances in Neural Information Processing Systems*, volume 35, pages 30318–30332. Curran Associates, Inc.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. *Lora: Low-rank adaptation of large language models*. *Preprint*, arXiv:2106.09685.
- Haozhou Li, Qinke Peng, Xu Mou, Ying Wang, Zeyuan Zeng, and Muhammad Fiaz Bashir. 2023. *Abstractive financial news summarization via transformer-bilstm encoder and graph attention-based decoder*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:3190–3205.
- Siddharth Samsi, Dan Zhao, Joseph McDonald, Baolin Li, Adam Michaleas, Michael Jones, William Bergeron, Jeremy Kepner, Devsh Tiwari, and Vijay Gadeppally. 2023. From words to watts: Benchmarking the energy costs of large language model inference. In *2023 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–9. IEEE.
- Eugene Sy, Tzu-Cheng Peng, Shih-Hsuan Huang, Heng-Yu Lin, and Yung-Chun Chang. 2023. *Fine-grained argument understanding with BERT ensemble techniques: A deep dive into financial sentiment analysis*. In *Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023)*, pages 242–249, Taipei City, Taiwan. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).
- Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, Chenxia Li, Liu Yang, Xilin Luo, Xuejie Wu, Lunan Liu, Wenjun Cheng, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, Lei Lin, Xi-aokun Wang, Yutuan Ma, Chuanhai Dong, Yanqi Sun, Yifu Chen, Yongyi Peng, Xiaojuan Liang, Shuicheng Yan, Han Fang, and Yahui Zhou. 2023. *Skywork: A more open bilingual foundation model*. *Preprint*, arXiv:2310.19341.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyang Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. 2024. *The finben: An holistic financial benchmark for large language models*. *Preprint*, arXiv:2402.12659.
- Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah. 2023. *Finmem: A performance-enhanced llm trading agent with layered memory and character design*. *Preprint*, arXiv:2311.13743.
- Zhihan Zhou, Liqian Ma, and Han Liu. 2021. *Trade the event: Corporate events detection for news-based event-driven trading*. *Preprint*, arXiv:2105.12825.

Revelata at the FinLLM Challenge Task: Improving Financial Text Summarization by Restricted Prompt Engineering and Fine-tuning

Ken Kawamura¹, Zeqian Li¹, Chit-Kwan Lin¹, Bradley McDanel²

¹Revelata, Inc., ²Franklin and Marshall College

Correspondence: ken@revelata.com

Abstract

Fine-tuning large language models (LLMs) is a promising approach for domain-specific tasks such as financial text summarization. However, the role of prompt design in fine-tuning LLMs, especially on limited training data, remains under-explored. In this paper, we examine the impact of instruction complexity and restricted prompt engineering on fine-tuning instruction-tuned LLMs for financial headline generation. Surprisingly, we find that restricting modifications to a specific portion of the prompt (the “lead-in phrase” for the LLM assistant role) significantly influences the quality of the generated outputs, even outperforming models fine-tuned on more complex instructions. Our results underscore the pivotal role of prompt design in adapting LLMs to specialized domains, and suggest that carefully crafting specific portions of an instruction-tuned LLM’s prompt can yield substantial performance gains even with minimal training data.

1 Introduction

Recent advancements in LLMs (Sanh et al., 2021; Brown et al., 2020; et al., 2022, 2023b) are finding wider adoption in finance (Wu et al., 2023; Xie et al., 2024; Yu et al., 2023a), driven in part by shared task challenges such as FinLLM. Here, we discuss our submission to FinLLM Task 2: Financial Text Summarization, in which we investigate how the quality of LLM-generated financial news summaries can be improved by modifying specific parts of conversational prompts when fine-tuning instruction-tuned LLMs.

Surprisingly, when fine-tuning Meta-Llama-3-8B-Instruct¹ on a small financial news article dataset (Zhou et al., 2021) with a variety of prompts, the complexity of the prompt instructions given to the model has relatively little impact on summarization performance.

¹<https://ai.meta.com/blog/meta-llama-3/>

In contrast, fine-tuning with the “right” lead-in phrase (i.e., the portion of the prompt immediately before the model begins generating) outperforms models that are fine-tuned on complex instructions, as measured with ROUGE-1 (Lin, 2004).

2 Related Work

Financial news articles can have critical impacts on the stock market (Tetlock, 2005). Prior research has explored the use of sentiment analysis (Araci, 2019; Sy et al., 2023) on financial news articles to predict stock movement (Zhou et al., 2021; Kalyani et al., 2016; Shah et al., 2018; Mohan et al., 2019). However, sentiment paints news articles with a broad brush, and is less suitable for nuanced analyses. For such analyses, news article summarization plays a critical role.

In general, the goal of news article summarization is to generate a concise text that captures the key points of a longer news article. Prior works have relied on datasets such as the CNN/Daily Mail Corpus (Nallapati et al., 2016) and XSum (Narayan et al., 2018) for evaluations of various summarization methods, which can range from those (Liu and Lapata, 2019) based on BERT (Devlin et al., 2019) to more recent ones based on GPT (Brown et al., 2020) models (Zhang et al., 2023; Goyal et al., 2022). This recent adoption of LLMs (Wu et al., 2023; Yang et al., 2023b; Lee et al., 2024; Yu et al., 2023b) has opened up many possibilities of LLM-based financial news summarization (Xie et al., 2024). In this work, we explore the interaction between prompt design and fine-tuning LLMs for financial news summarization.

3 Task and Dataset

3.1 Task Description

FinLLM Task 2 centers around training an LLM to generate coherent and concise summaries of financial news articles. This task is formulated as

an abstractive summarization problem, where the model is asked to generate a compact summary that captures the essence of the article. In order to guide the model to output such summaries, the participants in the task are allowed to create their own prompt and perform fine-tuning on custom datasets. The organizers detect model cheating when perplexities on training and test data are too close in value, following existing work (Wei et al., 2023) on data leakage.

3.2 Dataset

We were provided a dataset of 8,000 training samples and 2,000 test samples from the EDTSUM dataset² (Zhou et al., 2021; Xie et al., 2024). Each sample consisted of two elements: (1) the text of a financial news article from a source such as Businesswire or PRNewswire; and (2) the article’s corresponding title, which served as an approximation of an abstractive summary. Thus, the true task could be better described as “title generation”, rather than a more broadly-construed summarization task; this distinction informed our prompt design. Lastly, along with the dataset, the organizers provided a baseline prompt template (Xie et al., 2024) (see Appendix A).

3.2.1 Data Cleaning

Through manual inspection of the training dataset we found that there existed titles that were too short or too long to be qualitatively good titles (Table 1). This led us to examine the distribution of the title lengths (Figure 1), which we found to be long-tailed. We reasoned that outliers were likely to harm model training, and decided to remove samples with titles shorter than four words and titles longer than 69 words (99th percentile). We emphasize we only removed extreme outliers; the remaining samples still reflected the broad spectrum of title complexity in original dataset (e.g., we retained the vast majority of titles that contained subtitles and bullets).

We also found duplicate titles and samples where company names used in the title could not be found in their corresponding article. We filtered out these samples since they would likely negatively impact model training as well. Lastly, we found punctuation missing in many of the titles. While missing punctuation does not impact ROUGE-1, we reasoned that such titles would have a lower probability of being generated by any LLM, since such se-

²https://huggingface.co/datasets/TheFinAI/edtsum_train

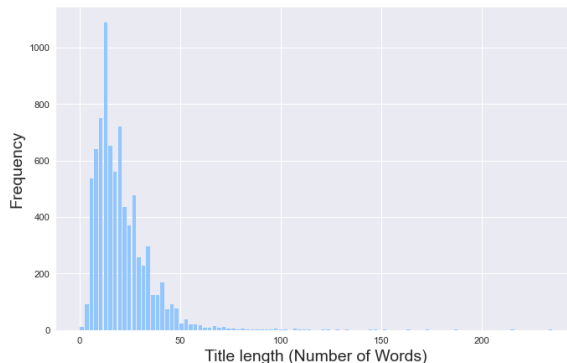


Figure 1: The distribution of news article title length is long-tailed.

quences are likely out-of-distribution with respect to the LLM’s pre-training data. To avoid this problem, we used GPT-4 (et al., 2023b) to impute title punctuation (see Appendix B). After cleaning, we had 7,803 training samples remaining.

4 Approach

We first designed a set of prompts by systematically changing parts of the prompts, and then fine-tuned Meta-Llama-3-8B-Instruct on each of these prompts separately. We conjectured that when training data is limited, as in this task, the choice of which part of the prompt to modify could have a large impact on the fine-tuning result.

4.1 Baselines

Our baselines are Gemini Pro (Team et al., 2023) and LLaMA2-70B (et al., 2023a) as reported in FinBen (Xie et al., 2024). They were evaluated in a zero-shot fashion with PIXIU (Xie et al., 2023) using the baseline prompt.

4.2 Prompt Tuning

4.2.1 Parts of a Prompt

LLMs trained on instructions for chat applications switch between the two roles, user and assistant (et al., 2023b,a; Jiang et al., 2023; Roller et al., 2020), allowing a single model to simulate the conversation between two parties and to act as one or the other, depending on the role.

One natural way to engineer a prompt is to refine the instruction given by a user to an assistant *before* the latter’s response. In our scenario, this would involve carefully defining the summarization task and giving detailed guidelines for the assistant’s response. Figure 2 shows the instruction portion of an example prompt in orange.

Long/Short	Title	Word Count
Short	Annual Financial Report	3
Long	Henry Schein Reports Record First-Quarter 2021 Financial Results from Continuing Operations Total net sales of \$2.9 billion up 20.4% versus prior year GAAP diluted EPS from continuing operations of \$1.16 versus prior-year GAAP diluted EPS from continuing operations of \$0.91 Non-GAAP diluted EPS from continuing operations of \$1.24 versus prior-year non-GAAP diluted EPS from continuing operations of \$0.94 Reflecting strong first-quarter results, the Company raises guidance for 2021 non-GAAP diluted EPS from continuing operations to be at or above \$3.70	80

Table 1: Examples of titles found in the dataset that are either too short or too long to be qualitatively good titles.

<p>Instruction You are a seasoned marketing PR professional brainstorming a captivating headline for a press release at BUSINESS WIRE and PRNewswire</p> <p>Write a headline with strong SEO potential. Article: {Body of News Article}</p> <p>Just write a title.</p> <p>Assistant Title: {Title}</p>

Figure 2: An example prompt illustrating the different parts of the prompt. The orange text is the instruction a user provides, and the blue text is a lead-in phrase for the assistant’s generation. The violet text is the final title that the model generates. Here, we use a simple instruction and “Title: ” as a lead-in phrase.

Another way to tune a prompt is to control the assistant’s *lead-in* phrase, just before it generates its response. Figure 2 shows an example of a lead-in phrase in blue. Prior works (Kojima et al., 2022; Wei et al., 2022) have shown that zero-shot LLM predictions can be improved by adding “Let’s think step by step.” to the prompt immediately before the response. Along these lines, we manipulated the conversational *lead-in* phrase of the assistant response; e.g., we controlled the start of the assistant response to be “Title:” or “Here is a headline with strong SEO potential:”. As instruction-tuned models such as Meta-Llama-3-8B-Instruct are trained to be conversational, we found that certain lead-in responses are better suited for chat interactions than others, even when the instruction portion remains the same.

4.2.2 Prompt Design

First, we manually created relatively simple prompts (Figure 2). For example, we simply changed the lead-in phrase from “Answer:” in the baseline prompt provided by the organizers to

“Title: ” because it better aligns with the task. From this simple prompt, we crafted additional prompts by modifying the instruction and the lead-in phrase parts of the prompt.

Instruction. We designed four different prompts by replacing the simple instruction with more complex instructions, while keeping the lead-in phrase “Title: ” fixed. These complex prompts had much more detailed instructions than just asking the model to “Write a headline with strong SEO potential.”. In Table 2, we present the best-performing complex instruction alongside a baseline instruction and a simple instruction. For a comprehensive list of all complex instructions tried, see Appendix C.

Lead-In Phrase. In total, we devised three lead-in phrases: (1) “Title: ”, (2) “” (empty string), and (3) “Here is a headline with strong SEO potential: ”. When the lead-in is empty, the model is free to start its response in whatever manner it chooses. Phrase 3 originates from our initial prompt exploration efforts; Meta-Llama-3-8B-Instruct sometimes started its generations with this phrase. Since the model already produced this lead-in phrase on its own, we conjectured that it could improve the quality of generated titles and kept it.

4.2.3 Model Fine-tuning

We fine-tuned Meta-Llama-3-8B-Instruct hosted on huggingface hub³ using AutoTrain⁴, with default settings. To determine the number of epochs to train the model, we first split the training set 9:1, to create a small validation split. We found that six epochs gave the best ROUGE-1 score on the validation split, and subsequently fine-tuned Meta-Llama-3-8B-Instruct for six epochs over the entire dataset, resulting in the model we submitted.

³<https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct>

⁴<https://github.com/huggingface/autotrain-advanced>

Type	Instruction
<i>Baseline</i>	You are given a text that consists of multiple sentences. Your task is to perform abstractive summarization on this text. Use your understanding of the content to express the main ideas and crucial details in a shorter, coherent, and natural sounding text. {Body of News Article}
<i>Simple</i>	You are a seasoned marketing PR professional brainstorming a captivating headline for a press release at BUSINESS WIRE and PRNewswire Write a headline with strong SEO potential. Article: {Body of News Article} Just write a title.
<i>Complex</i>	You are a helpful assistant. You are given a challenge. Below is the text of a press release article. The title has been hidden from you. The goal is to figure out the exact title based on the body of the article. You know that articles such as these can have either simple titles or complex titles that include subtitles in a bullet list. However, it is tricky to determine whether an article should have a simple or complex title, so you need to pay careful attention to the content of the article for any hints or clues. Do your best to write the exact title that was hidden from you. {Body of News Article}

Table 2: Instruction Variations. The baseline instruction is provided by the organizers.

Model	Zero-Shot/Fine-tune	Instruction	Lead-in	ROUGE-1
<i>Baselines</i>				
Gemini Pro	Zero-shot	Baseline	“Answer: ”	0.39
LLaMA2-70B	Zero-shot	Baseline	“Answer: ”	0.25
<i>Ours</i>				
Meta-Llama-3-8B-Instruct	Zero-shot	Simple	“Title: ”	0.402
	Fine-tune	Simple	“Title: ”	0.446
	Fine-tune	Complex	“Title: ”	0.441
	Fine-tune	Simple	“”	0.412
	Fine-tune	Simple	“Here is a headline with strong SEO potential:”	0.500

Table 3: Test ROUGE-1 Score on EDTSUM. We only show the best performing result for the prompts with complex instructions.

5 Results

As shown in Table 3, zero-shot title prediction by Meta-Llama-3-8B-Instruct with a simple instruction (ROUGE-1: 0.402) already outperforms both Gemini Pro (ROUGE-1: 0.39) and LLaMA2-70B (ROUGE-1: 0.25). Fine-tuning further improves the ROUGE-1 score from 0.402 to 0.446, using a simple instruction and “Title: ” as a lead-in phrase. This result underscores the significance of fine-tuning for adapting foundation models to a specific downstream task.

Surprisingly, we observed that varying the instruction has marginal effect on ROUGE-1 score when the model is fine-tuned. In fact, even a best performing complex instruction with detailed guidelines and a careful task definition (ROUGE-1: 0.441) performed worse than a simple instruction (ROUGE-1: 0.446) by 0.005.

In contrast, varying the lead-in phrase has a substantive impact on performance. Among the fine-tuned models, the model that performed worst (ROUGE-1: 0.412) had an empty lead-in phrase. Meanwhile, by simply replacing “Title: ” with “Here is a headline with strong SEO potential: ”, and keeping the simple instruction, we achieved

our best result (ROUGE-1: 0.500). This suggests that when fine-tuning a model trained for chat applications, tailoring how the assistant starts its conversational response (i.e., the lead-in phrase) is substantially more important than giving complex instructions, if we want the model to achieve high ROUGE-1 performance.

6 Conclusions

Our study highlights the crucial role of prompt engineering in fine-tuning LLMs. Specifically, we find that refining the lead-in phrase of the assistant response significantly improves performance when fine-tuning instruction-tuned models such as Meta-Llama-3-8B-Instruct.

However, manually crafting these prompts can be resource-intensive in practical deployments. In future works, we plan to explore automated approaches to optimize lead-in phrases using frameworks such as Optimization by Prompting (OPRO) (Yang et al., 2023a).

References

- Dogu Araci. 2019. [Finbert: Financial sentiment analysis with pre-trained language models](#). *ArXiv*, abs/1908.10063.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). *ArXiv*, abs/2005.14165.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hugo Touvron et al. 2023a. [Llama 2: Open foundation and fine-tuned chat models](#). *ArXiv*, abs/2307.09288.
- Josh Achiam et al. 2023b. [GPT-4 Technical Report](#).
- Teven Le Scao et al. 2022. [BLOOM: A 176B-Parameter Open-Access Multilingual Language Model](#). *ArXiv*, abs/2211.05100.
- Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. [News summarization and evaluation in the era of gpt-3](#). *ArXiv*, abs/2209.12356.
- Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L'elio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *ArXiv*, abs/2310.06825.
- Joshi Kalyani, Prof. H. N. Bharathi, and Prof. Rao Jyothi. 2016. [Stock trend prediction using news sentiment analysis](#). *ArXiv*, abs/1607.01958.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. [Large language models are zero-shot reasoners](#). *ArXiv*, abs/2205.11916.
- Jean Lee, Nicholas Stevens, Soyeon Caren Han, and Minseok Song. 2024. [A survey of large language models in finance \(finllms\)](#). *ArXiv*, abs/2402.02315.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Yang Liu and Mirella Lapata. 2019. [Text summarization with pretrained encoders](#). *ArXiv*, abs/1908.08345.
- Saloni Mohan, Sahitya Mullapudi, Sudheer Sammeta, Parag Vijayvergia, and D. Anastasiu. 2019. [Stock price prediction using news sentiment analysis](#). *2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)*, pages 205–208.
- Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çağlar Gülçehre, and Bing Xiang. 2016. [Abstractive text summarization using sequence-to-sequence rnns and beyond](#). In *Conference on Computational Natural Language Learning*.
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. [Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization](#). *ArXiv*, abs/1808.08745.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2020. [Recipes for building an open-domain chatbot](#). In *Conference of the European Chapter of the Association for Computational Linguistics*.
- Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan D. Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng-Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. [Multi-task prompted training enables zero-shot task generalization](#). *ArXiv*, abs/2110.08207.
- Dev Shah, Haruna Isah, and Farhana H. Zulkernine. 2018. [Predicting the effects of news sentiments on the stock market](#). *2018 IEEE International Conference on Big Data (Big Data)*, pages 4705–4708.
- Eugene Sy, Tzu-Cheng Peng, Shih-Hsuan Huang, and Yung-Chun Chang Heng-Yu Lin. 2023. [Fine-grained argument understanding with bert ensemble techniques: A deep dive into financial sentiment analysis](#). In *Taiwan Conference on Computational Linguistics and Speech Processing*.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. [Gemini: a family of highly capable multimodal models](#). *arXiv preprint arXiv:2312.11805*.
- Paul C. Tetlock. 2005. [Giving content to investor sentiment: The role of media in the stock market](#). *The Journal of Finance*, 62(3):1139–1168.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. [Chain of thought prompting elicits reasoning in large language models](#). *ArXiv*, abs/2201.11903.

Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, Chenxia Li, Liu Yang, Xilin Luo, Xue Gang Wu, Lunan Liu, Wenjun Cheng, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, Lei Lin, Xiaokun Wang, Yutuan Ma, Chuanhai Dong, Yanqi Sun, Yifu Chen, Yongyi Peng, Xiaojuan Liang, Shuicheng Yan, Han Fang, and Yahui Zhou. 2023. [Skywork: A more open bilingual foundation model](#). *ArXiv*, abs/2310.19341.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambar, David Rosenberg, and Gideon Mann. 2023. [Bloomberggpt: A large language model for finance](#). *ArXiv*, abs/2303.17564.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyang Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. 2024. [The finben: An holistic financial benchmark for large language models](#). *Preprint*, arXiv:2402.12659.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. [Pixiu: A large language model, instruction data and evaluation benchmark for finance](#). *Preprint*, arXiv:2306.05443.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2023a. [Large language models as optimizers](#). *ArXiv*, abs/2309.03409.

Hongyang Yang, Xiao-Yang Liu, and Chris Wang. 2023b. [Fingpt: Open-source financial large language models](#). *ArXiv*, abs/2306.06031.

Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah. 2023a. [Finmem: A performance-enhanced llm trading agent with layered memory and character design](#). In *AAAI Spring Symposia*.

Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Suchow, and Khaldoun Khashanah. 2023b. [Finmem: A performance-enhanced llm trading agent with layered memory and character design](#). *Preprint*, arXiv:2311.13743.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori Hashimoto. 2023. [Benchmarking large language models for news summarization](#). *Transactions of the Association for Computational Linguistics*, 12:39–57.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. [Trade the event: Corporate events detection for news-based event-driven trading](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.

A Baseline Prompt Provided By Organizers

You are given a text that consists of multiple sentences. Your task is to perform abstractive summarization on this text. Use your understanding of the content to express the main ideas and crucial details in a shorter, coherent, and natural sounding text. {Body of News Article}
Answer:

B GPT-4 Instruction Used to Add Punctuation

You are a helpful proofreader. The text below has no period punctuation. Please add it back. Respond with only the updated text. \n\nText:

C Prompts with Various Instructions

Instruction

You are a helpful assistant.

You have written a press release for your employer. The text of it follows these instructions. You need to now write a suitable title for the press release. You know that some press releases in the past have had a single title, while others have had a main title accompanied by subtitles. Taking that into account, you should write a title that is appropriate for this article. In any case, do your best to write a title that will make the reader feel interested in reading the article itself, and to ensure that your title has high SEO potential.

Here is the article:

{Body of News Article}

Assistant

Title:

{Title}

D Prompts with Various Lead-in Phrases

Instruction

You are a financial research analyst.

You ran a web scraper script that scrapes press release articles from company and newswire websites. However, there was a bug in the script that accidentally left out all the titles. You know some articles have a single title, while others have a main title followed by subtitles. Knowing this, do your best to write an appropriate title for the scraped article below.

{Body of News Article}

Assistant

Title:

{Title}

Instruction

You are a large language model.

You are given a prompt to generate the title of a financial news article, e.g., a press release. Even though you aren't allowed to know the actual title of the article, the title you generate must have a high ROUGE-1 score with respect to the actual title of the article. Since it's a ROUGE-1 score, you want to maximize the number of words that overlap with the actual title, regardless of the order in which they appear in the title.

Here is the article:

{Body of News Article}

Assistant

Title:

{Title}

Instruction

You are a helpful assistant.

You are given a challenge. Below is the text of a press release article. The title has been hidden from you. The goal is to figure out the exact title based on the body of the article. You know that articles such as these can have either simple titles or complex titles that include subtitles in a bullet list. However, it is tricky to determine whether an article should have a simple or complex title, so you need to pay careful attention to the content of the article for any hints or clues. Do your best to write the exact title that was hidden from you.

{Body of News Article}

Assistant

Title:

{Title}

Instruction

You are a seasoned marketing PR professional brainstorming a captivating headline for a press release at BUSINESS WIRE and PRNewswire

Write a headline with strong SEO potential. Article: *{Body of News Article}*

Just write a title.

Assistant

{Title}

Instruction

You are a seasoned marketing PR professional brainstorming a captivating headline for a press release at BUSINESS WIRE and PRNewswire

Write a headline with strong SEO potential. Article: *{Body of News Article}*

Just write a title.

Assistant

Here is a headline with strong SEO potential:

{Title}

‘Finance Wizard’ at the FinLLM Challenge Task: Financial Text Summarization

Meisin Lee and Soon Lay-Ki

Monash University, Bandar Sunway, Selangor Malaysia

{lee.meisin, soon.layki}@monash.edu

Abstract

This paper presents our participation under the team name ‘Finance Wizard’ in the FinNLP-AgentScen¹ 2024 shared task 2: Financial Text Summarization. It documents our pipeline approach of fine-tuning a foundation model into a task-specific model for Financial Text Summarization. It involves (1) adapting Llama3 8B, a foundation model, to the Finance domain via continued pre-training, (2) multi-task instruction-tuning to further equip the model with more finance-related capabilities, (3) finally fine-tuning the model into a task-specific ‘expert’. Our model, FinLlama3_sum, yielded commendable results, securing the third position in its category with a ROUGE-1 score of 0.521.

1 Introduction

Since the release of Large Language Models (LLMs), they have been swiftly fine-tuned into specialized LLMs in various domains such as biomedical, legal, finance and economics. For the Finance domain, a number of FinLLMs have been released: **FinMA** (also known as PIXIU) (Xie et al., 2023a), **InvestLM** (Yang et al., 2023), **FinGPT** (Wang et al., 2023) and **BloombergGPT** (Wu et al., 2023). As part of expanding the capabilities of Financial LLM, organizers of the FinLLM challenge prepared three subtasks, namely financial classification, financial text summarization and stock trading prediction. For subtask 2 (financial text summarization), participants are given a dataset of financial news article and the task is to produce abstractive summaries for each piece of news.

While all the FinLLMs listed above are trained to be multi-task models capable of a wide variety of finance-related tasks, this paper differs from these FinLLMs where we aim to produce a task-specific

model to achieve the best possible score for Financial Text Summarization. The training approach we took is influenced by the findings in (Jang et al., 2023; Lee et al., 2024) that *expert* Language Models (LM) fine-tuned on just a single task can outperform a multi-task LM trained with numerous different tasks. We aim to produce a task-specific model, similar to FinPythia(Xie et al., 2023b) - a model trained specifically for financial sentiment analysis.

This paper documents the considerations and planning that went into producing the task-specific model for the financial text summarization task. This includes the selection of the Foundation Model (both evaluating existing FinLLMs and new LLMs), the selection of training corpus and datasets, as well as the design of the end-to-end fine-tuning approach.

2 Related Work

To the best of our knowledge, there are two survey papers written on Financial Large Language Models (FinLLMs): (1) *Large Language Models in Finance: A Survey*(Li et al., 2023) and (2) *A survey of Large Language Models in Finance (FinLLMs)* (Lee et al., 2024). Based on their findings, the list of FinLLMs and their properties are captured in Table 1. For brevity, only LLMs² are listed, finance PLMs (Pretrained Language Models) such as FinBERT(Araci, 2019; Yang et al., 2020; Liu et al., 2021) and FLANG(Chung et al., 2024) are omitted.

These FinLLMs are trained on multiple tasks, which includes Sentiment Analysis, Text Classification, Structure Boundary Detection, Named Entity Recognition, Question Answering and more. Just like the foundation LLMs (Radford et al., 2019), these multi-task FinLLMs are also used in in-context learning and in unseen tasks via few-shot

¹<https://sites.google.com/nlg.csie.ntu.edu.tw/finnlp-agentscen/shared-task-finllm>

²According to (Zhao et al., 2023), Large LM are models that have 7 billion or more parameters

Type	Model	Foundation	Parameters	Techniques	Task(s)	Context Len.
Multi-task	BloombergGPT (Wu et al., 2023)	BLOOM	50B	PT, PE	SA, TC NER,QA	2048
	InvestLM (Yang et al., 2023)	LLaMa	65B	IFT, PE,	SA,TC QA,Summ	2048
	FinMA (Xie et al., 2023a)	LLaMa2	7B, 30B	IFT, PE	SA,TC,NER QA,SMP	2048
	FinGPT (Wang et al., 2023)	LlaMa2, Falcon, MPT, BLOOM ChatGLM2, QWEN	7B and various sizes	IFT, PE	SA,TC NER,RE	4096
1-Task	Fin-Pythia-1.4B (Rodriguez Inserte et al., 2023)	Pythia	1.4B	PT, PE	SA	2048

Table 1: A summary of the list of Instruction Fine-tuned FinLLMs. List of tasks that these models are fine-tuned on are: SA (Sentiment Analysis), TC (Text Classification), SBD (Structure Boundary Detection), NER (Named Entity Recognition), QA (Question Answering), SMP (Stock Movement Prediction), Summ (Text Summarization), and RE (Relation Extraction).

and zero-shot prompting.

3 Approach

We designed a pipeline approach where it first starts with a foundation LLM, followed by adapting it to the finance domain, and then instruction-tuning it into a task-specific ‘expert’. This approach is illustrated in Figure 1, and also described below:

1. Continual pre-training: in the first stage, we further pre-trained a foundation LLM on a collection of Finance-related corpus on the text completion task (causal language modeling) via unsupervised training. This produced **FinLlama3**.
2. Multi-task instruction-tuning: the second stage involves instruction-tuning FinLlama3 on multiple financial tasks via supervised fine-tuning (SFT) to obtain **FinLlama3_mt**.
3. Task-specific Instruction-tuning: in the last stage, the model is instruction-tuned specifically on just one-task - text summarization to produce **FinLlama3_sum**.

3.1 Choice of Foundation Model

For the task of Financial Text Summarization, an LLM’s contextual window length plays an important role because the context window need to be large enough to fit the entire news passage so that crucial information is not lost. The open-source Multi-task FinLLMs (FinMA, InvestLM, FinGPT) listed in Table 1 act as good candidates as ‘foundation model’ for instruction-tuning for Financial Text Summarization. However, their context window length is constrained to be at a maximum of

4,096 (see column ‘Context Len’ in Table 1 for information). At the point of writing, Llama3 (8B, 70B) is the newest open-source LLM that has the largest context window (8,192 tokens), it is boasted to outperform all of its predecessors. Hence, we chose Llama3 8B as the foundation model.

3.2 Continual Pre-training

Continual pre-training of LLMs aims to adapt an existing general LLM, which is also known as Foundation Models, to a new domain and be enriched with new domain knowledge, in the same way that FinBERT was trained. With Llama3 8B identified as the foundation model, we further pre-trained it on a Finance Corpus we have identified. The corpus is made up of financial texts such as financial news, financial statements, company annual reports, financial research reports, financial literature, market data, etc. Specifically, the corpus is made up of:

1. News from Reuters³: The datasets contains 55,700 records of financial news stories from Reuters.
2. News from CNBC, Reuters, WSJ⁴: This dataset contains 18,400 records of financial news from various sources,.
3. Finance and economic glossary (Investopedia)⁵: This dataset contains 6,310 records of financial and economic glossary

³<https://huggingface.co/datasets/rjjan/reuters21578>

⁴<https://huggingface.co/datasets/Lettria/financial-articles>

⁵https://huggingface.co/datasets/infCapital/investopedia_terms_en

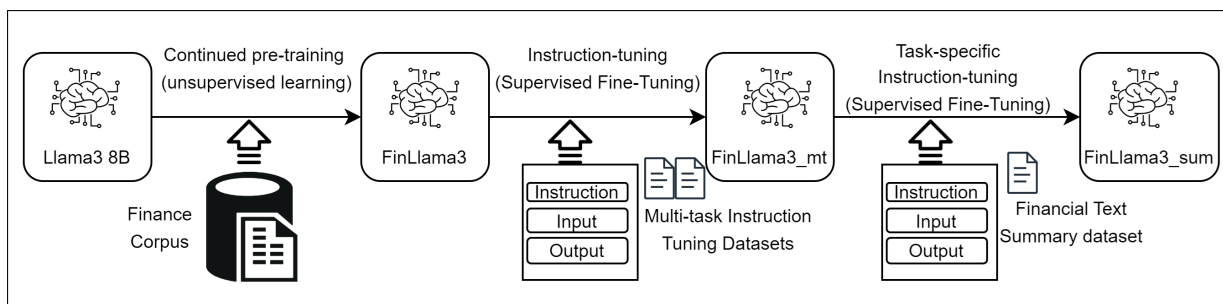


Figure 1: The design of our end-to-end fine-tuning approach. This shows the evolution of a foundation model to the final task-specific ‘expert’ for financial text summarization.

4. Edgar SEC filings (1993 - 2020)⁶: This dataset comprises financial reports (10-K) submitted to the U.S. Securities and Exchange Commission (SEC). We only used reports for the year 2019.
5. FinWeb⁷: The dataset is a website that provides economic knowledge and information on finance, loans and products.

3.3 Instruction Fine-tuning

Instruction Fine-tuning fine-tunes LLMs on a stream of supervised instruction-following data, this step aims to empower LLMs to follow users’ instruction. Instruction-tuning dataset comes in a specific template, consisting of (1) instruction, (2) text/input and (3) answer/output as shown in Table 3.

3.3.1 Multi-task Instruction Tuning

For multi-task instruction tuning, we aligned closely with established financial benchmarks such as the pioneering FLUE benchmark (Shah et al., 2022) and the Fundamental Tasks of the FinBen (Xie et al., 2024) benchmark. We used a public dataset Sujet-Finance-Instruct-177k⁸ containing about 177,000 records that covers the following tasks in the finance domain:

1. Sentiment Analysis: Financial Phase Bank (FPB)(MALO et al., 2014), FiQA-SA(Maia et al., 2018)
2. Text classification: News Headlines Classification(Sinha and Khandait, 2021)
3. Named Entity Recognition (NER) (Alvarado et al., 2015)

⁶<https://huggingface.co/datasets/eloukas/edgar-corpus>

⁷https://github.com/deep-over/FiLM/tree/main/pretraining/document/train_dataset/finweb

⁸<https://huggingface.co/datasets/sujet-ai/Sujet-Finance-Instruct-177k>

4. Question Answering FiQA(FiQA), ConvFinQA(Chen et al., 2022)

For more information about this multi-task dataset refer Appendix B.1.

The Multi-Task Instruction-tuning phase aims to train the LLMs on multiple concurrent Finance-NLP tasks, thus improving their versatility and usability on a variety of financial tasks.

3.3.2 Task-Specific Instruction Tuning

In this stage, we fine-tuned the model on a single, specific task - Abstractive Text Summarization. Apart from solely using the training dataset (EDT-SUM) provided as part of the shared task challenge, we have also identified other text summarization datasets in the Finance domain. These datasets are combined for the instruction-tuning step:

- Training set (EDTSum) (8,000 records) of financial news summarization
- Earnings call summarization (2,424 records)⁹
- Long earnings calls bullet point summarization (ECTSum)(Mukherjee et al., 2022)¹⁰
- Edgar Annual Reports (10k) SEC filings summarization (1,000 records)¹¹
- Financial news summarization (27,000 records)¹²

4 Experiments and Results

4.1 Training Details

1. **Stage 1: Continual pre-training on Financial Corpus** - we further pre-train Llama3 8B¹³ on the financial corpus for 2 epochs to

⁹https://huggingface.co/datasets/soumakchak/earnings_call_dataset

¹⁰<https://github.com/rajdeep345/ECTSum>

¹¹<https://huggingface.co/datasets/wyx-ucl/SUM-DATASET-BASED-EDGAR-CORPUS>

¹²https://huggingface.co/datasets/kdave/Indian_Financial_News

¹³<https://huggingface.co/unsloth/llama-3-8b-Instruct-bnb-4bit>

	Model	Approach	ROUGE-1	ROUGE-2	ROUGE-L	BertScore	BartScore
(A)	Llama3 8B	zero-shot	0.2020	0.0921	0.1572	0.6506	-4.0342
	FinLlama3	zero-shot	0.2020	0.0923	0.1600	0.6701	-4.0012
(B)	FinMA-7B	zero-shot	0.2021	0.0942	0.1621	0.5935	-4.0941
	FinGPT(LLaMa2)	zero-shot	0.0005	0.0001	0.0005	0.5128	-4.9231
	FinLlama3_mt	zero-shot	0.2044	0.0945	0.1604	0.6517	-3.6981
(C)	FinLlama3_sum	<i>instruction-tuned</i>	0.5210	0.3406	0.4735	0.9084	-3.4980

Table 2: Text Summarization results tested on the EDTSum test set. (A): foundation models, (B): multi-tasks FinLLMs, (C): task-specific model. Access to InvestLM model was not available, hence not listed in the table above.

produce **FinLlama3**.

- Stage 2: Multi-task instruction-tuning** - FinLlama3 is fine-tuned on multiple financial tasks for another 15 epochs to produce **FinLlama3_mt**. Results and evaluation metrics for each tasks are found in Table 4.
- Stage 3: Task-specific Instruction-tuning** - FinLlama3_mt is then fine-tuned on text summarization task for 9 epochs. We utilize three metrics, such as ROUGE (1, 2, and L) (Lin, 2004), BERTScore (Zhang et al., 2019) and BARTScore (Yuan et al., 2021), to evaluate the quality of the generated summaries.

All training was done using Unsloth’s FastLanguageModel¹⁴ that uses the PEFT (Parameter Efficient Fine-Tuning) library, 4-bit quantization and QLoRA (Quantized Low-Rank Adaptation). The execution was two times faster and used 60% less memory compared to Huggingface transformer library, significantly lower the computational requirements. As for hardware, we used one A100 GPU with 80GB memory, and with auto-find-batch-size set to true.

4.2 Results and Discussion

Apart from testing our model using the test set (EDTSUM_test), we also ran other multi-task FinLLMs listed in Table 1 against the same test, the results are captured in Table 2. Llama3 8B is the foundation model that serves as a baseline comparison with the other FinLLMs. Among the models, only two, InvestLM (Yang et al., 2023) and our FinLlama3_sum were specifically instruction-tuned on the financial text summarization task. The rest of the models were evaluated for their text summarization ability via zero-shot prompting.

As expected, multi-task financial models in Group B show better performance compared to

Group A’s (foundation models) baseline performance. This shows that instruction-tuning a model in multiple Finance-related tasks somewhat improves the model’s overall ability to generalize to unseen tasks via zero shot prompting. However, only marginal improvement is observed here. Instead, significant improvement is observed for FinLlama3_sum that was specifically instruction-tuned for the financial text summarization task.

The results also prove that our pipeline approach is successful as the models (FinLlama3 → FinLlama3_mt → FinLlama3_sum) show progressive improvements in financial text summarization. While the final model, FinLlama3_sum, performs well in text summarization, it might suffer from catastrophic forgetting on earlier fine-tuned tasks. As part of a rigorous testing, this model should be subjected to benchmark tests such as the FinBen benchmark (Xie et al., 2024), which consist of 35 datasets across 23 financial tasks. It is important to determine if the model suffers from any catastrophic forgetting, and also to test its ability to generalize to a wider suite of financial tasks.

5 Conclusion

Our pipeline approach of turning a foundational model Llama3 8B into a FinLLM (LLM adapted to the Finance domain) and finally to a task-specific ‘expert’ in Text Summarization has proven to be effective, as the final model is capable of generating summaries with a Rouge-1 score of 0.521 and is ranked 3rd in the challenge. As part of continuous improvement, we intend to explore other open-source LLMs such as Llama3 70B, and other newer models such as Gwen2 72B.

6 Availability

We made our model available here: https://huggingface.co/meisin123/llama3_FinLLM_textsum and codes here: https://github.com/meisin/IJCAI_FinLLMChallenge.

¹⁴Unsloth is a lightweight library for faster LLM fine-tuning, which is fully compatible with the Huggingface ecosystem (Hub, transformers, PEFT, TRL), <https://unsloth.ai/>

References

- Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90.
- Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.
- Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. Convinqa: Exploring the chain of numerical reasoning in conversational finance question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6279–6292.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.
- FiQA. <https://sites.google.com/view/fiqa>.
- Joel Jang, Seungone Kim, Seonghyeon Ye, Doyoung Kim, Lajanugen Logeswaran, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2023. Exploring the benefits of training expert language models over instruction tuning. In *International Conference on Machine Learning*, pages 14702–14729. PMLR.
- Jean Lee, Nicholas Stevens, Soyeon Caren Han, and Minseok Song. 2024. A survey of large language models in finance (finllms). *arXiv preprint arXiv:2402.02315*.
- Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. Large language models in finance: A survey. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 374–382.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.
- Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2021. Finbert: A pre-trained financial language representation model for financial text mining. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, pages 4513–4519.
- Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www’18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.
- Pekka MALO, Ankur SINHA, Pekka KORHONEN, Jyrki WALLENUS, and Pyy TAKALA. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796.
- Rajdeep Mukherjee, Abhinav Bohra, Akash Banerjee, Soumya Sharma, Manjunath Hegde, Afreen Shaikh, Shivani Shrivastava, Koustuv Dasgupta, Niloy Ganguly, Saptarshi Ghosh, et al. 2022. Ectsum: A new benchmark dataset for bullet point summarization of long earnings call transcripts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10893–10906.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.
- Pau Rodriguez Inserte, Mariam Nakhlé, Raheel Qader, Gaetan Caillaut, and Jingshu Liu. 2023. Large language model adaptation for financial sentiment analysis. In *Proceedings of the Sixth Workshop on Financial Technology and Natural Language Processing*, Bali, Indonesia. Association for Computational Linguistics.
- Raj Shah, Kunal Chawla, Dheeraj Eidnani, Agam Shah, Wendi Du, Sudheer Chava, Natraj Raman, Charese Smiley, Jiaao Chen, and Diyi Yang. 2022. When flue meets flang: Benchmarks and large pretrained language model for financial domain. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2322–2335.
- Ankur Sinha and Tanmay Khandait. 2021. Impact of news on the commodity market: Dataset and results. In *Advances in Information and Communication: Proceedings of the 2021 Future of Information and Communication Conference (FICC), Volume 2*, pages 589–601. Springer.
- Neng Wang, Hongyang Yang, and Christina Wang. 2023. Fingpt: Instruction tuning benchmark for open-source large language models in financial datasets. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhajan Kam-badur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. 2024. The finben: An holistic financial benchmark for large language models. *arXiv preprint arXiv:2402.12659*.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023a. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *arXiv preprint arXiv:2306.05443*.
- Yong Xie, Karan Aggarwal, and Aitzaz Ahmad. 2023b. Efficient continual pre-training for building domain

specific large language models. *arXiv preprint arXiv:2311.08545*.

Yi Yang, Yixuan Tang, and Kar Yan Tam. 2023. Investlm: A large language model for investment using financial domain instruction tuning. *arXiv preprint arXiv:2309.13064*.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

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

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

A Prompt template

Template:
Instruction: [task prompt]
Text: [input context]
Answer: [output]
Example for Text Summarization
Instruction: "You are given a text that consist of multiple sentences. Your task is to perform abstractive summarization on this text. Use your understanding of the content to express the main ideas and crucial details in a shorter, coherent, and natural sounding text."
Text: <i>financial news article</i>
Answer: <i>abstractive summary of Text</i>

Table 3: Instruction-tuning Template for Text Summarization

B Multi-task Instruction-Tuning

B.1 Dataset

This dataset, `sujeet-ai/Sujeet-Finance-Instruct-177k`, can be accessed here: <https://huggingface.co/datasets/sujeet-ai/Sujeet-Finance-Instruct-177k>. This dataset is made up of

- **Sentiment Analysis:** 44,209 entries. This involves analyzing financial texts to categorize sentiments as positive, negative, neutral, bearish, or bullish.

- **QA (Question Answering):** 38,801 entries. Direct-answer finance questions that don't require additional context.
- **QA with Context:** 40,475 entries. Finance questions necessitating context for answers.
- **QA Conversation:** 15,613 entries. This category includes questions following a conversational history between a user and an LLM assistant.
- **Yes/No Question:** 20,547 entries. Questions necessitating a simple yes or no answer.
- **Topic Classification:** 16,990 entries. Tasks requiring the classification of financial texts into specific finance-related categories.
- **NER (Named Entity Recognition) Sentiment Analysis:** 962 entries. This involves conducting sentiment analysis at the entity level within texts.

B.2 Multi-task Instruction-tuning results

Dataset	Metrics	Results
Sentiment Analysis	F1	0.69
	Acc	0.71
Text Classification	AvgF1	0.77
NER	Entity F1	0.57
QA	EmmAcc	0.46

Table 4: Multi-task Instruction Tuning Results based on each tasks.

Upaya at the FinLLM Challenge Task 1 and 2: DistFin: Distillation based Fine-Tuning for Financial Tasks

Ashvini Kumar Jindal
LinkedIn AI USA

Pawan Kumar Rajpoot
SCB DataX Thailand

Ankur Parikh
UtilizeAI Research India

Abstract

With the advent of Large Language Models (LLM) in finance, financial text analysis and generation tasks have received growing attention. Financial text classification and financial text summarization are some of the very important text analysis and generation tasks, respectively. Adapting LLMs to these tasks is very crucial for domain adaptation. This paper presents a method to fine-tune LLMs to Financial Argument Classification and Financial Abstractive Summarization. The argument classification task focuses on argument unit classification to test the capabilities of LLMs to identify and categorize texts as premises or claims. The summarization task aims to abstract financial texts into concise summaries. The dataset was released along with shared tasks as a part of the 8th Financial Technology and Natural Language Processing (FinNLP), co-located with IJCAI 2024. In this paper, we employed a distillation-based fine-tuning of Llama-3 (8B parameters) to learn the rationale/step generated by Llama-3 (70B parameters) along with labels. In the argument classification task, we achieved an F1-score (evaluation metric) of 0.4166. In the summarization task, we got the 2nd rank with the Rouge-1 score (evaluation metric) of 0.5294.

1 Introduction

Recently, Large Language Models (LLMs) (Brown et al., 2020) such as GPT-2 and GPT-4 (OpenAI et al., 2024), have reshaped the field of natural language processing (NLP) and exhibited remarkable capabilities in specialized domains across mathematics, coding, medicine, law, and finance (Bubeck et al., 2023). Within the financial domain, recent several studies (Xie et al., 2023a; Lopez-Lira and Tang, 2023; Li et al., 2023; Xie et al., 2023b) have shown the great potential of advanced LLMs such as GPT-4 on financial text analysis/prediction and generation tasks. Examples of financial text analysis tasks are sentiment analysis, news headline

classification, hawkish-dovish classification, argument unit classification, argument relation classification, ESG issue identification, deal completeness classification, etc. and instances of financial text generation tasks are text summarization, financial report generation, etc. (Xie et al., 2024). This paper focuses on one such analysis task - argument unit classification and one such generation task - abstractive summarization.

The primary objective of the argument unit classification is to categorize argumentative sentences found in earnings conference call text into 'claim' and 'premise' classes (Sy et al., 2023). This classification is a foundational step, enabling a granular breakdown of financial narratives. The precision in isolating these units paves the way for deeper comprehension and subsequent analysis. Recognizing the distinct units of arguments means that investors and stakeholders can better interpret the sentiments conveyed in these financial discussions. (Sy et al., 2023) employed voting-based ensemble of various fine-tuned language models such as BERT (Devlin et al., 2019), ROBERTA (Liu et al., 2019), ETCE-TRA (Araci, 2019), FINBERT (Clark et al., 2020) etc.

The summarization task aims to abstract financial texts into concise summaries. Summarizing news articles is very useful in trading strategies. By shaping investors' perceptions and assessments of companies, financial news significantly impacts the stock market (Engle and Ng, 1993; Tetlock, 2007). News-based stock prediction models are thus developed to automatically discover signals of stock market movements from the countless news articles that are generated every moment (Kalyani et al., 2016; Shah et al., 2018; Mohan et al., 2019; Zhou et al., 2021). Summaries of company 10-q and 10-k reports also form the working memory of an LLM-based trading agent (Yu et al., 2023). The other summarization task focuses on earnings call transcripts (Mukherjee et al., 2022). (Yang et al.,

2023) has instruction-tuned Llama-65B (Touvron et al., 2023) on various financial tasks including financial summarization task.

In this paper, we utilized a distillation-based fine-tuning of Llama-3 (8B parameters) (AI@Meta, 2024) to learn the rationale/step generated by Llama-3 (70B parameters) (AI@Meta, 2024) along with labels. For the argument unit classification task, we prompted Llama-3 (70B) to generate a rationale for the given argumentative sentence and label premise, claim pair. We prompted Llama-3 (70B) to identify the main ideas/sentences given the financial news text and summary pair. We achieved F1-score (evaluation metric) of 0.4166 in the argument classification task. In the summarization task, we got the 2nd rank with the Rouge-1 score (evaluation metric) of 0.5294. Both of our models are available on HuggingFace ¹

2 Preliminary Background

2.1 Argument Unit Classification

2.1.1 Task

Given an input argumentative sentence S , the objective is to adopt an LLM M that accurately categorizes S into either the argument unit $A=\{\text{claim, premise}\}$ class.

2.1.2 Data

The dataset released with this task contains 7.75k and 969 data points in training and test data, respectively. These data points represent financial text along with labels - premise and claim.

2.1.3 Evaluation

The prompt template used to evaluate the LLM submission is - Instruction: [task prompt] Text: [input text] Response: [output]. The instruction is - 'Analyze sentences from earnings conference calls and identify their argumentative function. Each sentence is either a premise, offering evidence or reasoning, or a claim, asserting a conclusion or viewpoint. Return only premise or claim'. The evaluation metric is F1-score.

2.2 Abstractive Summarization

2.2.1 Task

Given an input financial news text T , the task is to adapt an LLM M that accurately generates an abstractive summary A .

¹https://huggingface.co/upaya07/finnlp_task_1,
https://huggingface.co/upaya07/finnlp_task_2

Table 1: Prompt for generating rationale for Argument Unit Classification task

```
## Task
We are working on a Text, which is from earnings conference calls and identify their argumentative function. This text can be classified as either a premise or a claim. A premise sentence offers evidence or reasoning, while a claim sentence asserts a conclusion or viewpoint. Analyse following sentence and assume that you secretly know the provided Answer, write a clear one or two line max reasoning that concludes provided with final Answer. Return only the Reasoning part.

## Text
""""{Text}""""

## Reasoning
```

2.2.2 Data

The dataset released with this task contains 8k and 2k data points in training and test data, respectively. These data points represent financial news text along with an abstract summary.

2.2.3 Evaluation

The prompt template used to evaluate the LLM submission is - Instruction: [task prompt] Context: [input context] Response: [output]. The instruction is - 'You are given a text that consists of multiple sentences. Your task is to perform an abstractive summarization of this text. Use your understanding of the content to express the main ideas and crucial details in a shorter, coherent, and natural-sounding text'. The evaluation metric is the ROUGE-1 score.

3 Argument Unit Classification

To add more context to the training data, we prompted the bigger Llama-3 70B model to get rationale behind the gold label. Further, we added this rationale in the training data, reformatted it, and then fine-tuned the smaller Llama-3 8B chat model to generate rationale and an answer/a label based on it.

3.1 Rationale Generation

We used Llama-3-70B-Instruct model to generate rationale. We provide prompt in Table 1.

3.2 Supervised Fine Tuning

We augmented the training data with the generated rationale and LoRA (Hu et al., 2021) fine-tuned the Llama-3 8b Instruct model for generating rationale and the answer in a defined order. We fine-tuned our model for 3 epochs using a Nvidia 4090

Table 2: Example record with reasoning augmented training data for Argument Unit Classification task

Task
Analyze sentences from earnings conference calls and identify their argumentative function. Each sentence is either a premise, offering evidence or reasoning, or a claim, asserting a conclusion or viewpoint. Return the Reason first and then Answer that is premise or claim.
Text
So, now that with a SaaS approach, you can reach a much broader base of business customers all over the world, is one opportunity.
Reason
This sentence asserts a conclusion or viewpoint about the opportunity presented by a SaaS approach, rather than providing evidence or reasoning to support a larger argument. Hence claim.
Answer
Claim

Table 3: Results on Argument Unit Classification Task. F1 score is used for final ranking.

Team	Accuracy	F1-Score
Team Barclays	0.762	0.523
Albatross	<u>0.757</u>	<u>0.517</u>
L3iTC	0.754	0.514
Upaya(ours)	0.709	0.416

GPU system with 40 GB RAM and 24GB VRAM. The hyper-parameters for the fine-tuning are shared along with the model on the huggingface ². We provide an example record of reasoning augmented training data in Table 2.

3.3 Results

Overall, we got 7th rank in the task with an accuracy score of around 71 and an f1 score of 41. Table 3 shows results from top-3 teams.

4 Abstractive Summarization

The financial text summarization aims to summarize financial news articles into concise summaries. The task provides 8k training data and 2k test data. Metrics such as such as ROUGE (1, 2, and L) and BERTScore are computed for all submission and ROUGE-1 score is used for the final rankings. For this task, we used the similar approach as the task explained in previous section with few modifications outlined below.

²https://huggingface.co/upaya07/finnlp_task_1

Table 4: Results on Financial News Summarization task. ROUGE-1 is used for final ranking.

Team	ROUGE-1	BERTScore
LBZ	0.535	0.912
Finance Wizard	0.521	0.908
Upaya(ours)	<u>0.529</u>	<u>0.911</u>

4.1 Relevant Sentence Extraction

Our approach is based on an intuition that there are a few sentences in original news article that play an important role in writing a coherent summary. Following the intuition, we prompted Llama-3 70B Instruct model to extract maximum of 5 relevant sentences from the original news text that are relevant to the given summary. Along with prompting the model to extract relevant sentences to the summary, we also prompted it to generate a rationale behind importance of each extracted sentence. This scheme helps to extract relevant sentences conditioned on ground truth summary. We provided prompt in Table 5 that we applied to extract relevant sentences from the training data provided in the task.

4.2 Supervised Fine-tuning

Once relevant sentences are extracted using the approach described in the previous section, original 8k training data is augmented and the new output contains extracted sentences along with summary. Next, we fully fine-tuned Llama-3 (8B parameters) model on top of the augmented training data. Specifically, during fine-tuning, the model takes original news text as input and learns to generate both relevant sentences and final summary. In this work, we did not explore adding rationale for model fine-tuning. The hyper-parameters for the fine-tuning are shared along with the model on the huggingface ³. We provided the prompt in Table 6 that we applied to fine-tune Llama-3-8B-Instruct model.

4.3 Results

We achieved 2nd rank in Financial News Summarization task with ROUGE-1 score of 0.529. Table 4 shows results from top-3 teams.

³https://huggingface.co/upaya07/finnlp_task_2

Table 5: Prompt for extracting relevant sentence(s) from financial text

```

## Task
You are given a financial text under "## Financial Text" section. Assuming that you secretly have access to the summary of the financial text under "## Summary" section, you need to extract maximum 5 relevant sentences from original financial text following below instructions:
- Each extracted sentence should be important and contributes to the given summary.
- Rank relevant sentences in order of high to low importance. Each relevant sentence should contain rationale behind its importance on a scale of 1 to 10 where 1 being least important and 10 being most important.
- Do not modify the original sentence and keep rationale limited to one line only.
- Rationale should not contain phrases that directly or indirectly reveal that you have access to the summary.
- There can be less than 5 relevant sentences, hence, you need to only provide relevant ones instead of always providing 5 sentences.

## Financial Text
""#{Financial Text}""

Summary
""#{ summary }""

[RESPONSE FORMAT]
Generate response as JSON with following schema. Each entry contains extracted sentence, rationale, and importance score on scale of 1-10.

[
  {
    "sentence": <sentence>,
    "rationale": <rationale>,
    "importance": <importance>
  },
]

[JSON RESPONSE]

```

Conclusion

This paper explores distillation based fine-tuning of Llama-3 models for two of the financial tasks: 1. Argument Unit Classification 2. Abstractive Summarization. For the first task, we used Llama-3 70B model to distill the rationale behind the label given the financial sentence and the label pair. In the second task, we prompted Llama-3 70B model to distill main ideas behind the summary given the financial text and summary. In both tasks, we augmented training data with this distilled information and performed instruction-tuning to adapt Llama-3 8B model on these tasks. We achieved F1-score of 0.4166 in the argument classification task. In the summarization task, we got the 2nd rank with the Rouge-1 score of 0.5294.

Table 6: Prompt for training model for financial text summarization task

```

## Task
You are given a financial text under "## Financial Text" section and you need to write a summary of the given text.
- First, extract relevant sentences from the given text that you think are important for summary.
- Next, write a summary focusing on extracted sentences and optionally given text.

## Financial Text
""#{Financial Text}""

```

Acknowledgements

We would like to thank Lambda Labs⁴ for providing the compute resources required for this work.

References

- AI@Meta. 2024. [Llama 3 model card](#).
- Dogu Araci. 2019. [Finbert: Financial sentiment analysis with pre-trained language models](#).
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#).
- Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. [Sparks of artificial general intelligence: Early experiments with gpt-4](#).
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. [Electra: Pre-training text encoders as discriminators rather than generators](#).
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [Bert: Pre-training of deep bidirectional transformers for language understanding](#).
- Robert F Engle and Victor K Ng. 1993. Measuring and testing the impact of news on volatility. *The journal of finance*, 48(5):1749–1778.

⁴<https://lambdalabs.com/>

- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#).
- Joshi Kalyani, Prof Bharathi, Prof Jyothi, et al. 2016. Stock trend prediction using news sentiment analysis. *arXiv preprint arXiv:1607.01958*.
- Xianzhi Li, Samuel Chan, Xiaodan Zhu, Yulong Pei, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2023. [Are chatgpt and gpt-4 general-purpose solvers for financial text analytics? a study on several typical tasks](#).
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized bert pretraining approach](#).
- Alejandro Lopez-Lira and Yuehua Tang. 2023. [Can chatgpt forecast stock price movements? return predictability and large language models](#).
- Saloni Mohan, Sahitya Mullapudi, Sudheer Sammeta, Parag Vijayvergia, and David C Anastasiu. 2019. Stock price prediction using news sentiment analysis. In *2019 IEEE fifth international conference on big data computing service and applications (Big-DataService)*, pages 205–208. IEEE.
- Rajdeep Mukherjee, Abhinav Bohra, Akash Banerjee, Soumya Sharma, Manjunath Hegde, Afreen Shaikh, Shivani Shrivastava, Koustuv Dasgupta, Niloy Ganguly, Saptarshi Ghosh, and Pawan Goyal. 2022. [Ectsum: A new benchmark dataset for bullet point summarization of long earnings call transcripts](#).
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altmenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O’Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. [Gpt-4 technical report](#).
- Dev Shah, Haruna Isah, and Farhana Zulkernine. 2018. Predicting the effects of news sentiments on the stock

- market. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 4705–4708. IEEE.
- Eugene Sy, Tzu-Cheng Peng, Shih-Hsuan Huang, Heng-Yu Lin, and Yung-Chun Chang. 2023. Fine-grained argument understanding with bert ensemble techniques: A deep dive into financial sentiment analysis. In *Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023)*, pages 242–249.
- Paul C Tetlock. 2007. Giving content to investor sentiment: The role of media in the stock market. *The Journal of finance*, 62(3):1139–1168.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. [Llama: Open and efficient foundation language models](#).
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyang Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. 2024. [The finben: An holistic financial benchmark for large language models](#).
- Qianqian Xie, Weiguang Han, Yanzhao Lai, Min Peng, and Jimin Huang. 2023a. [The wall street neophyte: A zero-shot analysis of chatgpt over multimodal stock movement prediction challenges](#).
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023b. [Pixiu: A large language model, instruction data and evaluation benchmark for finance](#).
- Yi Yang, Yixuan Tang, and Kar Yan Tam. 2023. [Investlm: A large language model for investment using financial domain instruction tuning](#).
- Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W. Sutchow, and Khaldoun Khashanah. 2023. [Finmem: A performance-enhanced llm trading agent with layered memory and character design](#).
- Zhihan Zhou, Liqian Ma, and Han Liu. 2021. [Trade the event: Corporate events detection for news-based event-driven trading](#).

BAI-Arg LLM at the FinLLM Challenge Task: Earn While You Argue - Financial Argument Identification

Varad Srivastava

Barclays

varad.srivastava@barclays.com

Abstract

Previous studies have shown that analyst decisions that can influence investors to buy or sell in markets, are based on statements in Earnings Conference Calls (ECC). In this study, we present our LLMs (BAI-Arg Alpha and Beta) dedicated to the task of financial argument identification in sentences from ECC transcripts. Our experiments involved using in-context zero-shot and semantically similar few-shot learning, along with QLoRA-based fine-tuning methods. Our model BAI-Arg Alpha was able to out-perform all other proposed models, to achieve 1st rank on the leaderboard of FinLLM challenge (IJCAI'24). Furthermore, using our BAI-Arg Beta model, we were able to achieve micro-F1 and macro-F1 scores of 76.68% and 76.66% respectively, which are state-of-the-art, and out-perform all previously proposed models and approaches on the task. By being able to categorize arguments in ECC with a high degree of accuracy through our model, we hope to offer stakeholders enhanced clarity on financial sentiments, which can enable them to make more informed decisions in the economic markets.

1 Introduction

Predicting movements in market is a challenging problem, even with the recent growth of data and advance algorithms in the field of finance. This is because several factors and environments can influence its movements, which makes it difficult to get a very accurate estimate of stock prices in the future. According to the "efficient market hypothesis" (Fama, 1970), since the market is efficient (everything is fairly priced according to their value), it is not possible to outperform the overall market all the time even by using technical analysis to predict trends and select market timings. However, it is a widely accepted view that most of the investment decisions are influenced by cognitive bias and experience of a person, as humans are not known to

be rational decision-makers (Tversky and Kahneman, 1974). Past research works have extensively studied the impact of sentiments and events from online news, and social media platforms like tweets, as well as the semantics of language and recommendations used by forecasters and professional analysts, which can influence investors decision to buy or sell in markets. Findings by Keith and Stent (2019) in particular have shown that statements on Earnings Conference Calls (ECC) are reflective of analysts' decisions.

ECC are organized during every fiscal quarter and consist of the following three parts: a safe harbor statement, a presentation and question answering (Q&A) session. During presentations, executives present their statements about the performance of the company in last quarter as well as expectations about the future quarters. Professional analysts posit their questions and demand clarifications from the company's representatives during the Q&A session. The company executives present their arguments as answers in order to justify their opinions and convince people to believe in them. Previous studies have shown that discussions during the Q&A session have the most influence on the shifts in market (Matsumoto et al., 2011; Price et al., 2012).

While most of the previous works have encapsulated the use of semantic or syntactic analyses, argument mining can be used to extract a deeper interpretation of the language used to make statements in these sessions which in turn can help understand what people expect of the markets. This information can be used to drive investment decisions.

ECC transcripts are more favourable to extract arguments for two reasons. One, social media platforms are often restricted by number of characters. Two, people tend to post their opinions and views rather than structured premises or claims. For example, most of the tweets only have claims, which

	Train	Test	Whole
Premise	4,062	508	4,570
Claim	3,691	461	4,152
Total	7,753	969	8,722

Table 1: Data statistics

assert a conclusion or viewpoint without providing the required reasoning or evidence.

Additionally, even though language models have been extensively used for this task, use of LLMs on financial tasks such as these is still under-explored and under-utilized. Therefore in this paper, we experiment with various LLMs, utilizing methods like in-context learning and fine-tuning to investigate the arguments stated in the answers of company executives to questions of analysts. We finally propose LLM models (BAI-Arg), to leverage their state-of-the-art capabilities to classify these statements on the basis of argumentative function they represent - premise, or claim.

2 Dataset

The FinArg dataset (Alhamzeh et al., 2022) was used for the task of argument unit classification and was made available as part of shared task of the Fin-LLM challenge. See Appendix:A.1 for examples from the dataset. Here, the task is to use the capabilities of LLMs to interpret the argument units in statements from ECC transcripts by classifying them into "premise" or "claim". 7,753 statement texts and their gold labels were provided as training data, and the models were evaluated on 969 texts of test data. See Table 1 for more details.

3 Related Work

On the FinArg-1 challenge task of argument unit identification in NTCIR-17 (Chen et al., 2023), various language models were examined with either prompting or fine-tuning. The best model was submitted by TMUNLP (Lin et al., 2023) which was based on assembling outputs of ELECTRA and Roberta models using a voting mechanism, and achieved 76.55% macro-F1 score. The second ranked model by IDEA (Tang and Li, 2023) combined BERT hidden state embeddings with a Convolutional Neural Network (CNN), while the third ranked model by TUA1 (Chen et al., 2023) used the T5 model with prompt-based learning and instruction tuning. Other submitted approaches included leveraging GPT-3.5 Turbo for in-context

learning as well as generating more similar data to augment the dataset.

Sy et al. (2023) experimented with a BERT-based ensemble learning approach using a majority-voting mechanism to achieve a macro-F1 score of 76.62% on the task. More recently, Xie et al. (2024) in their work on the FinBen benchmark, evaluated several state-of-the-art LLM models like GPT-4, Gemini, LLaMA-70B, FinMA-7B, Falcon-7B, ChatGLM3-6B, FinGPT-7b-lora. InternLM-7B, Mixtral-7B, and CFGPTsft-7B-Full, on the Financial Argument Classification (FinArg-ACC) task, with GPT-4 out-performing all others with a macro-F1 score of 60.0%.

4 Methodology

This section provides descriptions of the various approaches we experimented for the challenge.

4.1 In-Context Learning

For in-context learning, we use LLMs like:

- Llama-3: We used Llama-3 8B parameter model (AI@Meta, 2024), which has context length of 8,192.
- Mistral: We used Mistral-7B model version 0.2 (Jiang et al., 2023; MistralAI) which has a context window of 32,768.
- Gemma: We used Gemma 7B model (Google), which has context length of 8,192.
- GPT: We used GPT-3.5 Turbo (OpenAI), which has a context window of 16,385 tokens.

These pre-trained chat models have been further fine-tuned to follow instructions with Reinforcement Learning from Human Preferences (RLHF). Therefore, we use the instruction-tuned versions of each of the models.

4.2 Prompt Engineering

Articulate prompt engineering is crucial in steering behaviour and response of the LLMs, by providing them the appropriate instructions and context for a task. Our prompt template, which went through various iterations of experiments, is provided in Appendix:A.3. The prompt starts with an instruction which encompasses the context of the task including a knowledge base detailing the classification criteria and short description of each of the classes. The test statement is then provided as an input by the user.

4.2.1 Zero Shot and Random Few Shot Learning

For our initial approach, we experimented with zero-shot learning and in-context learning with 1, 5 and 10 examples per class, chosen randomly from the training set.

4.2.2 Semantically Similar Few Shot Learning

In this approach, we select those examples for in-context learning from the training set, which are semantically similar to the test statement at inference. This is achieved by first training a sentence-transformer (MPNet) on the training set, which learns to encode the statements in the embedding space, based on whether their class is similar or dissimilar. In this work, we select one of its variations - 'all-mpnet-base-v2', which also ranks among the top in the HuggingFace sentence transformers leaderboard. Therefore based on this idea, for each test sentence to be classified, we use the all-mpnet-base-v2 vector embeddings and the cosine similarity metric (for distance calculation) to retrieve the 5,10 and 20 most similar examples at inference time, while performing in-context learning. For more details on the MPNet model, and its hyper-parameter tuning, refer to Appendix:A.2.

4.3 Fine-Tuning of Instruction Tuned LLMs

Based on the performance of models during in-context learning, we select Llama-3 8B model for fine-tuning to enhance model performance further. Each sample from the training set was converted into a prompt which included the test statement as a user input and the true label as the reply expected from the chat assistant. We experimented with two prompts here, with differences in only the structure and language of instruction (see Appendix:A.3).

QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023) was used to efficiently fine-tune the model. We first quantized the pre-trained model to 4-bit and then added a set of learnable low-rank adapter weight matrices with rank 64, that are tuned using backpropagation for upto 3 epochs. This was able to significantly reduce trainable parameters to 167M, hence reducing GPU memory requirements. The details of hyper-parameters are shown in Table 2. For the metrics reported in the Section 5, the model with "Prompt-1" was trained for three epochs, while the model with "Prompt-2" for two epochs. Hereafter, we refer to earlier model as "BAI-Arg Alpha", and the latter as "BAI-Arg Beta".

Hyperparameters	Value
Gradient Accumulation Steps	4
Learning Rate	2e-4
Epoch	2
LoRA-Rank	64
LoRA-Alpha	128
LoRA-Dropout	0
Optimizer	Adam

Table 2: QLoRA Hyper-parameter Details

5 Results

5.1 Performance on FinLLM Challenge Task

We report the performance of our modelling approaches through the metrics: micro-F1 (μ -F1) and macro-F1 (m-F1), as shown in Table 3. We observed that although all models perform poorly on zero-shot and random few-shot in-context learning, Llama-3 here still edges out Mistral and Gemma models. Additionally, when Retrieval-Augmented Generation (RAG) is used by augmenting in-context learning with semantically similar examples, there is a significant increase in performance for all models. Notably, here as well, Llama-3 is able to outperform other models, barring the 10-shot similar examples setting, where GPT-3.5 outperforms even Llama-3, by 0.33 percentage points (pp) in macro-F1. Nevertheless, since this does not hold on other settings like the 20-shot, we selected Llama-3 model for fine-tuning to investigate if performance can be further enhanced. Indeed, fine-tuning was able to enhance performance significantly, with macro-F1 increasing by upto 4.39 pp on the BAI-Arg Beta¹ model. Additionally, see Appendix:A.4 and A.5 for details on ablation studies conducted on few-shot learning and fine-tuning approaches respectively, and A.6 for error analysis.

Our model was ranked 1st on the FinLLM challenge leaderboard² for this task, when compared against the performance of other submitted models, as shown in the first section of the Table 4.

5.2 Performance Comparison with Existing Models

We also compared the performance of our model against the performance of LLMs in previous works such as that of Xie et al. (2024), and performance of the top-3 models proposed during

¹https://huggingface.co/varadsrivastava/BAI_Arg_Beta

²<https://huggingface.co/spaces/TheFinAI/IJCAI-2024-FinLLM-Leaderboard>

Table 3: Classification results for all models on the test data, with N-Shot indicating the number of samples used during learning. FT-n indicates fine-tuned using Prompt 'n'

Methods	Setting	$\mu - F_1$	m-F ₁
Gemma	0-shot	55.41	50.93
Llama-3	0-shot	59.44	56.74
Mistral	0-shot	53.56	47.26
Gemma (random)	1-shot	50.57	40.04
Llama-3 (random)	1-shot	58.93	54.21
Mistral (random)	1-shot	58.10	53.76
Gemma (random)	5-shot	53.97	49.13
Llama-3 (random)	5-shot	61.61	60.16
Mistral (random)	5-shot	53.56	39.35
Gemma (similar)	5-shot	64.09	62.22
Llama-3 (similar)	5-shot	71.00	70.87
Mistral (similar)	5-shot	67.91	66.65
GPT-3.5 (similar)	5-shot	69.04	68.83
Gemma (random)	10-shot	52.94	45.83
Llama-3 (random)	10-shot	61.09	57.55
Mistral (random)	10-shot	55.73	47.88
Gemma (similar)	10-shot	66.98	66.20
Llama-3 (similar)	10-shot	70.69	70.65
Mistral (similar)	10-shot	70.90	70.13
GPT-3.5 (similar)	10-shot	71.10	70.98
Gemma (similar)	20-shot	69.35	68.58
Llama-3 (similar)	20-shot	72.34	72.27
Mistral (similar)	20-shot	71.93	71.36
GPT-3.5 (similar)	20-shot	70.69	70.51
BAI-Arg Alpha	FT-1	76.26	76.12
BAI-Arg Beta	FT-2	76.68	76.66

NTCIR-17 (2023) (Chen et al., 2023). These comparison results are shown in Table 4.

We observe that our model BAI-Arg Beta outperforms all others in its ability to identify the argument unit, achieving micro-F1 and macro-F1 scores of 76.68% and 76.66%.

5.3 Model Cheating Detection

Due to concerns around data leakage in LLMs, a perplexity-based metric - Data Leakage Test (DLT), has been proposed by the FinLLM challenge organizers building on existing research (Wei et al., 2023). For details about the metric, refer to Appendix:A.7.

The DLT values are shown in Table 5. We observed that both of our models have a high enough DLT value, and even though there's a drop in the Beta version, the DLT metric value is still significantly higher than the reference baseline from the

Table 4: Comparison of our model's performance against other proposed models

Models	$\mu - F_1$	m-F ₁
Albatross ²	75.75	-
L3iTC ²	75.44	-
Wealth Guide ²	75.13	-
GPT-4 (Xie et al., 2024)	60.0	-
Gemini (Xie et al., 2024)	31.0	-
LLaMA2-70B (Xie et al.)	58.0	-
FinMA-7B (Xie et al., 2024)	27.0	-
Falcon-7B (Xie et al., 2024)	23.0	-
TMUNLP-1 (Lin et al., 2023)	76.57	76.55
IDEA-1 (Tang and Li, 2023)	76.47	76.46
TUA1-1 (Chen et al., 2023)	76.37	76.36
Sy et al. (2023)	-	76.62
BAI-Arg Alpha (Ours)	76.26	76.12
BAI-Arg Beta (Ours)	76.68	76.66

Table 5: Data Leakage Test Results

Models	DLT
L3iTC ²	2.2565
BAI-Arg Alpha	28.8399
BAI-Arg Beta	14.6049

leaderboard. This indicates that our models have a very low likelihood of cheating from data leakage.

6 Conclusion

In the rapidly evolving field of research using LLMs in finance domain, this shared task of FinLLM presented a unique opportunity to leverage LLM-based approaches for financial argument identification in quarterly Earnings Conference Calls (ECC) as premise or claim. In this paper, we presented our model, BAI-Arg LLM, based on well-articulated instruction prompts and fine-tuned Llama-3 8B model, which ranked first on the task in the IJCAI'24 FinLLM challenge leaderboard. It is able to out-perform all the other model submissions in the challenge, as well as the models proposed in previous literature. Therefore, by being able to categorize arguments in ECC with a high degree of accuracy through our model, we are able to offer stakeholders enhanced clarity on financial sentiments, which can enable them to make more informed decisions in the economic markets.

References

AI@Meta. 2024. [Llama 3 model card](#).

- Alaa Alhamzeh, Romain Fonck, Erwan Vermée, Elöd Egyed-Zsigmond, Harald Kosch, and Lionel Brunie. 2022. *It's time to reason: Annotating argumentation structures in financial earnings calls: The FinArg dataset*. In *Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP)*, pages 163–169, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Chung-Chi Chen, Chin-Yi Lin, Chr-Jr Chiu, Hen-Hsen Huang, Alaa Alhamzeh, Yu-Lieh Huang, Hiroya Takamura, and Hsin-Hsi Chen. 2023. *Overview of the ntcir-17 finarg-1 task: Fine-grained argument understanding in financial analysis*. NII Institutional Repository.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. *Qlora: Efficient finetuning of quantized llms*. *arXiv preprint arXiv:2305.14314*.
- Eugene F. Fama. 1970. *Efficient capital markets: A review of theory and empirical work*. *The Journal of Finance*, 25(2):383–417.
- Google. Gemma 7b instruct model card on huggingface. <https://huggingface.co/google/gemma-7b-it>. [Accessed 21-06-2024].
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. *Mistral 7b*. *arXiv preprint arXiv:2310.06825*.
- Katherine Keith and Amanda Stent. 2019. *Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 493–503, Florence, Italy. Association for Computational Linguistics.
- Heng-Yu Lin, Eugene Sy, Tzu-Cheng Peng, Shih-Hsuan Huang, and Yung-Chun Chang. 2023. *Tmunlp at the ntcir-17 finarg-1 task*. NII Institutional Repository.
- Dawn Matsumoto, Maarten Pronk, and Erik Roelofsen. 2011. *What Makes Conference Calls Useful? The Information Content of Managers' Presentations and Analysts' Discussion Sessions*. *The Accounting Review*, 86(4):1383–1414.
- MistralAI. Mistral 7b instruct v0.2 model card on huggingface. <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2>. [Accessed 21-06-2024].
- OpenAI. <https://platform.openai.com/docs/models/gpt-3-5-turbo>. [Accessed 21-06-2024].
- S. McKay Price, James S. Doran, David R. Peterson, and Barbara A. Bliss. 2012. *Earnings conference calls and stock returns: The incremental informativeness of textual tone*. *Journal of Banking and Finance*, 36(4):992–1011.
- Eugene Sy, Tzu Cheng Peng, Shih Hsuan Huang, Hen You Lin, and Yung Chun Chang. 2023. *Fine-grained argument understanding with bert ensemble techniques: A deep dive into financial sentiment analysis*. ROCLING 2023 - Proceedings of the 35th Conference on Computational Linguistics and Speech Processing, pages 242–249. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).
- Shaopeng Tang and Lin Li. 2023. *Idea at the ntcir-17 finarg-1 task: Argument-based sentiment analysis*. NII Institutional Repository.
- Amos Tversky and Daniel Kahneman. 1974. *Judgment under uncertainty: Heuristics and biases*. *Science*, 185(4157):1124–1131.
- Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, Chenxia Li, Liu Yang, Xilin Luo, Xuejie Wu, Lunan Liu, Wenjun Cheng, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, Lei Lin, Xiaokun Wang, Yutuan Ma, Chuanhai Dong, Yanqi Sun, Yifu Chen, Yongyi Peng, Xiaojuan Liang, Shuicheng Yan, Han Fang, and Yahui Zhou. 2023. *Skywork: A more open bilingual foundation model*. *Preprint*, arXiv:2310.19341.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyang Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. 2024. *The finben: An holistic financial benchmark for large language models*. *Preprint*, arXiv:2402.12659.

A Appendix

A.1 Dataset Examples

Humans often use argumentations to express themselves during a communication, and to think or deliberate about a situation or choice, which forms the core part of human decision making. A simple form of argument has two parts: a 'premise' (which provides some evidence or reason), and it supports a 'claim' (which is a conclusion).

An example of each from the provided dataset is provided as follows.

Premise:

"But another area that's growing incredibly quickly is private messaging, right, where between Messenger and WhatsApp, I think we're around 60 billion messages a day, which I think is something like three times more than the peak of global SMS traffic."

Claim:

"And what we're doing on Messenger and on WhatsApp are really making sure that businesses can connect with people, and then in the early stages of testing messaging."

Recognizing arguments from a text involves two sub-tasks: firstly, identifying and separating the argumentative units from the non-argumentative text units; secondly, classifying argument units into premises and claims. However, it is possible that a sentence is not a separate argument unit, rather encompasses several argument units. Because of this, argument units in the dataset were originally annotated at a minimum of clause-level and a maximum of sentence-level. Various clauses within the same sentence were considered different argument components if there was an inference relation between them (for e.g., appeared in forms like "claim because of premise", "Since premise then claim.", "In view of the fact premise that it follows that claim"), rather than a conjunction (for e.g. "and", "or"), or conditional (for e.g. "if, then"). However, this resulted in a few counter-intuitive clauses in the dataset, which might not make much sense in themselves, unless seen together with their original sub-clauses. Some examples of these instances are:

Premises:

"The second thing is video."
"because of the FX situation, right."

Claims:

"So, first on head count."
"One is just the format."

Therefore, these noisy examples make the task more challenging than it seems. Assuming the distribution of such instances in the test set to be similar to the training set, in our prompts - we decided to rely on instructions based on the function of the argument unit (premise or claim), rather than its structure.

A.2 MPNet Model and Hyper-parameter Tuning

MPNet is a transformer-based model which uses permuted language modelling to learn dependency among predicted tokens, as well as uses auxiliary position information as input. It is pre-trained on a text corpora of over 160 GB and fine-tuned on downstream tasks like GLUE, and SQuAD. Hyper-parameter tuning for MPNet-v2 was performed using Optuna framework. Over 10 trials, validation micro-F1 was maximized by having search spaces

over body's learning rate (1e-5, 5e-3), as well as the batch size [4,8,16,32].

A.3 Prompt Templates

The prompt template for the BAI-Arg Beta model is shown below.

```
<|begin_of_text|>
<|start_header_id|>system<|end_header_id|>
You are an expert assistant which can analyze
sentences from earnings conference call and
identify their argumentative function. Your
task is to classify the sentence after
<<<>>> into one of the following predefined
classes:
premise
claim
A sentence is a premise if it offers an evidence
or reasoning. A sentence is a claim if it
asserts a conclusion or viewpoint. You will
only respond with the name of the class. In
case you reply with something else, you will
be penalized. Do NOT provide explanations
or notes.<|eot_id|><|start_header_id|>user<|
end_header_id|>
<<<
Sentence: {Text}
>>>
<|eot_id|><|start_header_id|>assistant<|
end_header_id|>
Class: {Class}<|eot_id|>
```

A similar template was used to generate final results (shown in Table 3) for all other models (Mistral, Gemma and GPT) as well. The prompt for each of the specific models only differed in the special tokens they use to identify the instructions, user input or model's reply. For e.g. in Gemma, we use "`<start_of_turn>userinstruction<end_of_turn>`" to specify the instruction and context, and "`<start_of_turn>model`" to indicate that we expect a reply from the model.

For the initial challenge submissions, we had worked on an earlier version of prompt, which we refer to as "Prompt-1". This was used in the model, BAI-Arg Alpha. However, during our later experimentations, we came up with "Prompt-2" (which was used in the BAI-Arg Beta model). We were able to achieve significantly better performance on in-context learning with this prompt. The performance of the Llama-3 model on the 'Prompt-1' is shown in Table 6, for reference.

The prompt template for the BAI-Arg Alpha³ model is shown below.

```
<|begin_of_text|>
<|start_header_id|>system<|end_header_id|>
```

³https://huggingface.co/varadsrivastava/BAI_Arg_Alpha

Table 6: Classification results comparison for Llama-3 on the test data using Prompts 1 and 2, with N-Shot indicating the number of samples used during in-context learning.

Prompt	Setting	$\mu - F_1$	m-F ₁
Prompt-1	0-shot	54.80	51.44
Prompt-2	0-shot	59.44	56.74
Prompt-1 (random)	1-shot	57.48	55.91
Prompt-2 (random)	1-shot	58.93	54.21
Prompt-1 (random)	5-shot	59.65	58.66
Prompt-2 (random)	5-shot	61.61	60.16
Prompt-1 (similar)	5-shot	61.61	60.68
Prompt-2 (similar)	5-shot	71.00	70.87
Prompt-1 (random)	10-shot	60.06	57.21
Prompt-2 (random)	10-shot	61.09	57.55
Prompt-1 (similar)	10-shot	64.81	64.16
Prompt-2 (similar)	10-shot	70.69	70.65

You are an expert assistant, helping to analyze sentences from earnings conference calls and identify their argumentative function. Given a sentence which will be provided to you by the user from a earnings conference call, decide whether it is a premise or claim, described respectively as follows:
premise: A sentence which offers evidence or reasoning.
claim: A sentence which asserts a conclusion or viewpoint.
Reply with only one word (premise or claim).<|eot_id|>|start_header_id|>user<|end_header_id|>
Sentence: {Text}<|eot_id|>|start_header_id|>assistant<|end_header_id|>
Class: {Class}<|eot_id|>

A.4 Few Shot Learning: Ablation Study

Since we observed significantly better results when using semantically similar few-shot learning as compared to random few-shot learning, we investigated if the models are doing better because of the inherent biasness in examples that were retrieved (based on semantic similarity). To test this, we analysed how likely it is for the majority of labels of the few examples to match the target class.

We observed that the majority of the classes of the most semantically similar examples matched the target class on upto 76.68% of the test instances at inference. See Table 7 for more details. At first look, this does hint that the similar examples might be biasing the model into doing better. However, since the model performance of all models (except GPT-3.5) increases considerably as the similar examples are increased from 5 to 20, with the number of biased examples falling down. Hence, it could

Table 7: Analysis of biasness in Few-shot learning approach: The table shows how likely it is for the majority of the labels of the few examples to match the target class of test sentence.

Setting	Instances of biased examples	% of Test
5-shot (similar)	743	76.68%
10-shot (similar)	713	73.58%
20-shot (similar)	729	75.23%

be possible that the higher number of examples are also improving the argument understanding of the model, and the model might not just be resorting to the biasness of the examples for its good performance.

To investigate how significant of a role the biasness of the semantically similar examples are playing in few-shot learning, we perform an ablation study, wherein we investigated the performance of models by 'de-biasing' the example classes by sampling top-k examples from each class. For this, we sampled examples from top 500 semantically similar sentences (using the same methodology as described in Section 4.2.2), to retrieve top-5 and top-10 examples from each class, for 10-shot and 20-shot learning, respectively.

Table 8 shows the scores of the models when this equitable distribution of examples by class were retrieved for each test sentence at inference.

We observed that the performance of models drop significantly when the example classes are 'de-biased', by sampling top-k examples from each class. Therefore, this indicates that the biasness of the examples had a major role to play in the significant gains in performance of the models. Since, retrieval of such semantically-similar examples could be difficult in a noisy, real world use-case of this approach which could make the model less stable. Therefore, this provides support to the fine-tuning approach we used subsequently and the robustness of our final proposed model, BAI-Arg Beta.

A.5 Fine-tuning: Ablation Study

We performed an ablation study with the fine-tuning approach as well. We investigated two questions - One, whether training the BAI-Arg model with few examples improves the performance or the few shots are only helpful before fine-tuning; Two, if few examples do help in fine-tuning, whether there is a difference if the model is trained on ran-

Table 8: Classification results for all models using similar and de-biased examples for in-context learning, with N-Shot indicating the number of samples used during learning.

Methods	Setting	$\mu - F_1$	m-F ₁
Gemma (similar)	10-shot	66.98	66.20
Llama-3 (similar)	10-shot	70.69	70.65
Mistral (similar)	10-shot	70.90	70.13
GPT-3.5 (similar)	10-shot	71.10	70.98
Gemma (debiased)	10-shot	62.33	60.29
Llama-3 (debiased)	10-shot	65.63	64.94
Mistral (debiased)	10-shot	59.86	53.34
GPT-3.5 (debiased)	10-shot	67.39	67.39
Gemma (similar)	20-shot	69.35	68.58
Llama-3 (similar)	20-shot	72.34	72.27
Mistral (similar)	20-shot	71.93	71.36
GPT-3.5 (similar)	20-shot	70.69	70.51
Gemma (debiased)	20-shot	61.40	58.53
Llama-3 (debiased)	20-shot	65.94	63.78
Mistral (debiased)	20-shot	60.99	55.91
GPT-3.5 (debiased)	20-shot	67.70	67.68

Table 9: Classification results for models trained on few-shot (five) examples

Methods	$\mu - F_1$	m-F ₁
In-context (random ex)	61.61	60.16
In-context (similar ex)	71.00	70.87
Fine-tuned (random ex)	72.34	71.30
Fine-tuned (similar ex)	74.51	74.38

dom examples versus similar examples.

In order to investigate these, we performed an "active few-shot fine-tuning" where we included random and semantically similar (five) examples in the training of the Llama-3 model. The QLoRA hyper-parameters used were the same as shown in Table 2, and the model was trained for two epochs. The results obtained are shown in Table 9.

We observed that the "active few-shot fine-tuning" with randomly selected examples significantly improves the performance by upto 9 pp on micro-F1. as compared to the in-context learning with random examples. Additionally, semantically similar examples improve the performance even further, although the gains over in-context learning are not as significant, here. Interestingly, fine-tuning without few-shot examples still outperforms fine-tuning with examples, indicating that the examples might only help the model improve its understanding of arguments upto a limit.

Therefore, few examples do improve the performance and are helpful not just before, but during training as well. Additionally, here too, similar examples out-perform randomly selected ones in model performance.

A.6 Error Analysis

We performed a qualitative error analysis of our BAI-Arg Beta model to understand the model’s behaviour by observing what it gets wrong. This model made 122 errors on the 'premises' and 104 errors on the 'claims'.

Although its difficult to figure out the model’s exact heuristics for arriving at the decision, here are some observations we made:

- *Errors in premises being identified as claims:* These could be a result of the evidence or reasoning being expressed as view-points or lacking any key metrics. For e.g.:

"In terms of overall ad tech world, I think a lot is happening and there's a lot that's going to evolve in the whole ecosystem.

And if you go beyond that, I feel good about our gaming business sequentially ."

"The iPhone SE, we are thrilled with the response that we've seen on it."

"I feel confident in our ability to produce gross margin improvement across all those services."

"So we have great relationships with third party carriers."

"So we've said often that we think that virtual reality and augmented reality could be the next big computing platform."

Also, we observed certain errors where premises were rather expressed as past actions or planned ones in future, which might be the reason, the model classified them as claims. For e.g.:

"And we're continuing to invest across the board in terms of our core R&D and innovation efforts in terms of headcount growth there."

"And so we don't enter into those with no experience, although we will enter into them humbly."

- *Errors in claims being identified as premises:* These errors could have been caused due to addition of specific metrics, which the model might be mistaking for being part of evidence or reasoning. For e.g.:

"In the last 18 months, we've doubled the number of paid Prime member, which we're very excited about."
 "So all of those trailing 12-month metrics actually stayed the same or slightly declined in Q1."
 "But in general, inclusive of LinkedIn, I'm still around 100 bps."

Also, we observed certain errors where the missing context of the sentence might have confused the model in mistaking the sentences for facts, rather than a conclusion. For e.g.:

"We did see ARPU growth this quarter."

A claim like above might have been preceded by a context which is likely to include premise clauses, probably something like "although users decreased by xx%..." which might then make more sense for the below example to be perceived as a claim (with an inference relation like "in view of the fact premise").

Some other such examples are:

"And people had, again the ability to see the benefit that Prime membership save incremental dollars, because of it at Whole Foods."
 "And the other one is the on-premises server number which is very good in terms of hybrid demand this quarter also with high margin."
 "In the United States, which is usually the most advanced market, 35% of small businesses have no web presence at all."

Although, improving performance on such examples with somewhat overlapping argumentation intents is difficult without providing the context for each argument clause; for future work, we will try to leverage Chain-of-Thought reasoning in our prompts to mitigate them.

A.7 Data Leakage Test

The DLT metric calculates the difference in perplexity of the LLMs between the training and test data to determine its data generation tendencies. A larger difference implies that the LLM is less likely to have seen the test set during training compared to the training set and suggests a lower likelihood of the model cheating, and vice versa. The formula for the DLT metric is as follows:

$$\begin{aligned}
 DLT &= PPL(D_{\text{test}}) - PPL(D_{\text{train}}) \\
 PPL(D_{\text{train}}) &= \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} P(x)^{-\frac{1}{N}} \\
 &= \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} P(w_1 w_2 \cdots x_N)^{-\frac{1}{N}} \\
 &= \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} 2^{-\frac{1}{N} \log P(w_1 w_2 \cdots x_N)} \\
 &= \frac{1}{|D_{\text{train}}|} \sum_{x \in D_{\text{train}}} 2^{\text{Cross-Entropy}(x)}
 \end{aligned}$$

DLT values have been calculated for one other submitted model as well, to establish a reference baseline of Model Cheating, and minimize the impact of generalization on the metric.

CatMemo at the FinLLM Challenge Task: Fine-Tuning Large Language Models using Data Fusion in Financial Applications

Yupeng Cao*, Zhiyuan Yao*, Zhi Chen*, Zhiyang Deng*

*Equal Contribution

Stevens Institute of Technology, Hoboken, NJ

{ycao33, zyao9, zchen100, zdeng10}@stevens.edu

Abstract

The integration of Large Language Models (LLMs) into financial analysis has garnered significant attention in the NLP community. This paper presents our solution to IJCAI-2024 FinLLM challenge, investigating the capabilities of LLMs within three critical areas of financial tasks: financial classification, financial text summarization, and single stock trading. We adopted Llama3-8B and Mistral-7B as base models, fine-tuning them through Parameter Efficient Fine-Tuning (PEFT) and Low-Rank Adaptation (LoRA) approaches. To enhance model performance, we combine datasets from task 1 and task 2 for data fusion. Our approach aims to tackle these diverse tasks in a comprehensive and integrated manner, showcasing LLMs' capacity to address diverse and complex financial tasks with improved accuracy and decision-making capabilities.

1 Introduction

In recent years, FinTech research has increasingly focused on using textual information to aid investment decisions by analyzing various financial textual data (Allen et al., 2021). However, the complexity of financial documents makes it difficult to classify and summarize market information. Additionally, the intricate and volatile nature of financial markets poses significant challenges for making informed, sequential investment decisions. To address these challenges, advanced natural language processing techniques and models are necessary to process and interpret vast amounts of financial data accurately (Fisher et al., 2016). Lately, Large Language Models (LLMs) have demonstrated impressive capabilities in the field of finance (Bubeck et al., 2023; Li et al., 2023). These models excel in understanding and generating human-like text, making them ideal candidates for tackling complex financial tasks.

Although LLMs demonstrate significant promise

in the financial sector, their efficacy in specific financial tasks requires deeper investigation. The FinLLM challenge @ IJCAI-2024 initiative, as introduced in Xie et al. (2024), seeks to investigate the potential of LLMs in analyzing financial documents and enhancing decision-making processes. By leveraging the power of LLMs, the initiative aims to improve the accuracy and efficiency of financial information processing, ultimately aiding in improved investment strategies and a better market understanding.

This paper describes our technical solution for three diverse tasks provided by the FinLLM challenge: financial classification (Sy et al., 2023), text summarization (Zhou et al., 2021), and single stock trading (Yu et al., 2024). The classification task involves distinguishing between claims and premises in financial texts, the summarization task aims to distill extensive financial narratives into succinct summaries, and the trading task focuses on formulating predictive trading decisions based on algorithmic insights.

The core idea of our solution is to fine-tune pre-trained LLMs using PEFT (Mangrulkar et al., 2022) and LoRA (Hu et al., 2021) techniques, leveraging data fusion strategy on the provided datasets from task 1 & 2 in the FinLLM challenge. Specifically, we select Llama3-8B (AI@Meta, 2024) and Mistral-7B (Jiang et al., 2023) as the pre-trained base models due to their large number of parameters, which enable them to capture complex patterns and nuances in financial text data—essential for the three tasks in the challenge. Additionally, these models are pre-trained on vast and diverse datasets, providing a broad understanding of language that can be fine-tuned for financial domains, enhancing their versatility and adaptability to specific financial tasks. Furthermore, both models support PEFT and LoRA techniques, allowing efficient and effective specialization for the financial domain, even with limited labeled data.

Our extensive experiments conducted on the three shared tasks have yielded significant findings: 1) Mistral-7B outperforms Llama3-8B in terms of both overall performance and its ability to generate well-structured outputs; 2) the fine-tuned model by using the fused data, showed enhanced results on Task 1 and Task 2; 3) however, this fine-tuned model did not demonstrate improvement in the more complex single-stock trading task (Task 3). For this, we do a more detailed analysis of the results in Section 4.

2 Shared Task Description

The FinLLM challenge consists of three shared tasks: financial classification (task 1), text summarization (task 2), and single stock trading (task 3). Datasets description can be found in: https://huggingface.co/datasets/TheFinAI/flare-finarg-ecc-auc_test and https://huggingface.co/datasets/TheFinAI/flare-edtsum_test.

Task 1 in the FinLLM challenge focuses on the **financial classification**, specifically categorizing sentences within financial documents as either claims or premises. A claim is a statement that asserts a point of view or opinion, while a premise provides the supporting information or evidence for that claim. This task is fundamental for understanding and analyzing financial narratives, as it helps in structuring the information into coherent arguments, which is essential for various downstream applications such as sentiment analysis, risk assessment, and investment decision-making. The evaluation metric for Task 1 is the **F1 score**, which provides a balanced measure of the model’s precision and recall.

Task 2 in the FinLLM challenge focuses on **financial texts summarization**. The objective is to condense lengthy financial documents into concise summaries that capture the essential information and key insights while omitting redundant or less important details. This task is crucial for enabling quick and effective information processing, allowing stakeholders to make informed decisions without wading through extensive reports. Task 2 utilizes three metrics, namely ROUGE (1, 2, and L) and BERTScore, to evaluate generated summaries in terms of relevance, with the **ROUGE-1 score** serving as the final ranking metric.

Task 3 in the FinLLM challenge focuses on the application of LLMs to **single stock trading**, aiming to make informed and predictive trading decisions. The primary goal of this task is to develop a model that can analyze various financial texts and other relevant data to predict the future price movements of a single stock and make trading decisions based on these predictions. The evaluation metric includes Sharpe Ratio (SR), Cumulative Return (CR), Daily (DV) and Annualized Volatility (AV), and Maximum Drawdown (MD), with the **Sharpe Ratio (SR)** used as the final ranking metric.

3 Proposed Method

The success of large language models like GPT-4 (Achiam et al., 2023) and Llama3 demonstrates the benefits of integrating diverse data sources during pre-training, enhancing their capabilities and generalizability across various real-world applications. This approach not only broadens the model’s understanding of different data forms but also significantly boosts performance on specialized tasks through fine-tuning (Nguyen-Mau et al., 2024; Huang et al., 2024). Inspired by these advancements, our work employs a cross-task data fusion strategy for LLM fine-tuning, aiming to enhance the model’s effectiveness by combining insights from different financial tasks. Figure 1 illustrates the proposed fine-tuning method.

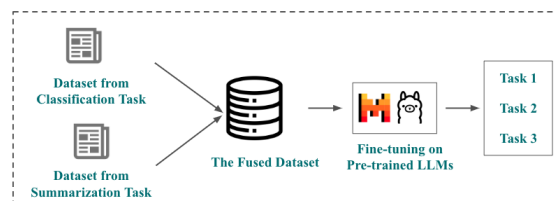


Figure 1: Schematic of proposed fine-tuning method.

We curated and preprocessed a robust training set from two tasks: financial text classification and financial text summarization, to cover a wide range of real-world financial scenarios. We excluded the dataset for task 3, which focuses on texts related to three specific stocks, due to its narrow company-specific content. This selective integration forms the basis for fine-tuning a pre-trained LLM, equipping it to effectively understand and generate nuanced financial texts. After fine-tuning, we applied the enhanced model to each of the three tasks to evaluate its practical utility and performance across various financial applications.

4 Experiment and Discussion

In this section, we present technical details of our implementation and numerical results of our fine-tuned models on tasks 1, 2, and 3. We also compare the performance of these models on different tasks and present our observations on discrepancies between the two base models.

4.1 Experiment Setup

Mistral-7B and Llama3-8B are employed as the base LLM in this study. Due to the limit of computational resources, we perform fine-tuning using Low-Rank Adaptation (LoRA, [Hu et al. \(2021\)](#)) with LoRA- α 16 and 4-bit quantization ([Jacob et al., 2018](#)) to reduce the usage of GPU memory and to accelerate training. The models were trained and inferenced on two NVIDIA RTX-A6000 GPUs (each has 48GB DRAM) with one epoch. Our implementation employs PEFT, Quantization libraries and other pipelines provided in Huggingface¹. We divided the training set portion of the validation set in the ratio of 80:20 for performance evaluation. The models are further tested and compared using the provided testing data sets.

4.2 Experiment Results on Validation Set

In preliminary experiments, we observed a significant difference in performance between the fine-tuned Mistral-7B and Llama3-8B models. Mistral-7B demonstrated superior predictive capabilities and produced well-formatted outputs that could be easily parsed to yield final predictions. In contrast, Llama3-8B required additional processing of its outputs through specific prompting, which could potentially alter the original outputs. Consequently, we decided to conduct all subsequent experiments using Mistral-7B.

4.2.1 Task 1

Table 1 illustrates that the fine-tuned LLMs have significantly improved reasoning for downstream-specific tasks. Furthermore, the LLMs, fine-tuned using the fused dataset, exhibit significant performance enhancements, where it achieves a 0.5634 F1 score. This evidence supports the notion that integrating different tasks can substantially enhance the reasoning capabilities of LLMs.

4.2.2 Task 2

Table 2 also demonstrates that the fine-tuned LLMs, by using the fused dataset, achieved signifi-

¹<https://huggingface.co/>

Dataset	ACC	F1
No Fine-tune	0.4997	0.1581
Task 1	0.3490	0.3913
Task 1 + Task 2	0.6259	0.5634

Table 1: The performance for two models tasked with classifying sentences as either "premise" or "claim". It includes two key metrics: Accuracy (ACC) and F1 Score (F1). Model "Task 1" was fine-tuned using only the dataset from Task 1, while Model Task 1 + Task 2 used datasets from both Task 1 and Task 2 for fine-tuning.

Dataset	Rouge-1	Rouge-2	BertScore
Task 1	0.4847	0.2921	0.6904
Task 1 + Task 2	0.4920	0.3015	0.6946

Table 2: The performance results for two models tasked with summarizing. It includes metrics for evaluating summarization: Rouge and Bert Score. Model "Task 1" was fine-tuned using only the dataset from Task 1, while Model Task 1 + Task 2 used datasets from both Task 1 and Task 2 for fine-tuning.

cant performance gains in the text summarization task. This reinforces the idea that integrating various tasks can notably enhance the generalization capabilities of LLMs across different applications.

4.2.3 Task 3

We compare the three fine-tuned models based on Mistral-7B in Task 3. We exclude the models fine-tuned from Llama3-8B in this comparison because Llama3-based models cannot consistently produce trading decisions in the correct format. We fine-tuned three models:

1. Model 1 is fine-tuned only using the training data from Task 1,
2. Model 2 is fine-tuned only using the training data from Task 2,
3. Model 3 is fine-tuned using the training data from Task 1 and Task 2.

The three models are implemented in the FinMem framework as described in [Yu et al. \(2024\)](#) to generate trading decisions.

We are interested in the performance discrepancies of these models trained on different datasets. Figure 2 shows the return changes of the three models across four stocks during the testing period. Table 3 details the performance metrics of the models on different stocks. The models generate distinct strategies for all four assets, indicating sensitivity to

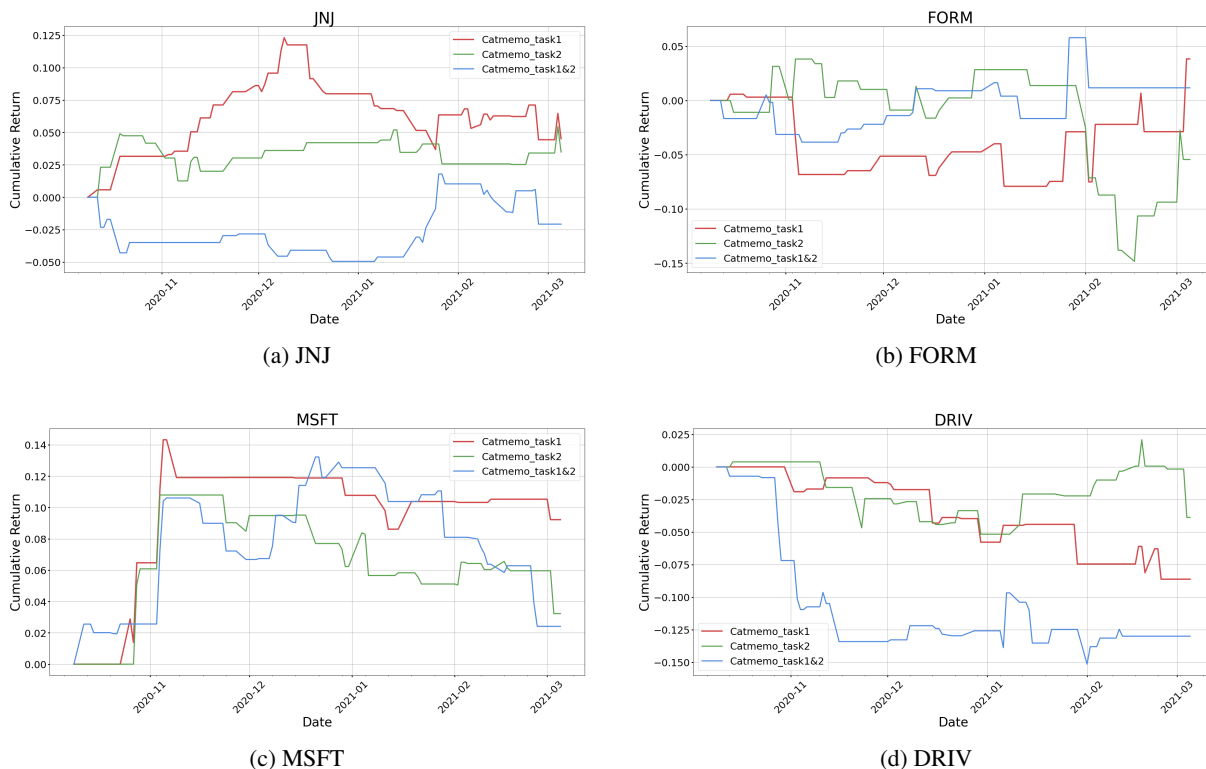


Figure 2: Comparison of Cumulative Returns in 4 Stocks

	FORM			JNJ			MSFT			DRIV		
	Model 1	Model 2	Model 3	Model 1	Model 2	Model 3	Model 1	Model 2	Model 3	Model 1	Model 2	Model 3
CR \uparrow	0.038	-0.054	0.012	0.045	0.035	-0.021	0.092	0.032	0.024	-0.086	-0.039	-0.130
SR \uparrow	0.440	-0.574	0.176	0.927	0.898	-0.506	1.594	0.564	0.418	-2.139	-0.834	-2.291
SD \downarrow	0.014	0.015	0.010	0.006	0.006	0.009	0.009	0.009	0.009	0.006	0.007	0.009
AV \downarrow	0.217	0.237	0.165	0.101	0.097	0.102	0.144	0.143	0.144	0.101	0.116	0.142
MD \downarrow	0.084	0.175	0.046	0.084	0.059	0.144	0.056	0.074	0.104	0.084	0.059	0.144

Table 3: Performance Metrics Comparison Across Different Models and Datasets.

the fine-tuning datasets. However, none of the models consistently produce profitable strategies. The Mistral-7B model is relatively small compared to state-of-the-art LLMs like OpenAI GPT-4 (Achiam et al., 2023) and Google Gemini (Team et al., 2023), limiting its ability to solve complex tasks such as trading decisions. This aligns with the reported performance of other LLMs in Xie et al. (2024). Additionally, Model 3, trained on both datasets, does not outperform the models trained on each dataset individually. This could be due to the introduction of noise or conflicting information from combining datasets. Given that tasks 1 and 2 are not directly related to trading, it is reasonable that all three models perform poorly in this task.

4.3 Experiment Results on Test Set

Based on the above analysis, we selected the Mistral-7B model, fine-tuned through data fusion, for the final challenge testing. In Task 1, the model

achieved an ACC of 0.711, an F1 score of 0.4199, and a Matthews correlation coefficient (MCC) of 0.6818. In Task 3, the integrated Sharp Ratio (SR) was -0.6199. These results are consistent with those observed in our validation set.

5 Conclusion

In this study, we fine-tuned LLMs using datasets that span multiple tasks, resulting in performance improvements in classification and summarization tasks. However, our approach did not yield positive results for the stock trading task. This outcome suggests that more complex financial tasks may require advanced data fusion steps. Furthermore, it underscores the need to explore the impact of incorporating larger datasets on the model’s performance after fine-tuning.

Limitation

Our work relies on the pre-trained large language model at 7B/8B level with 4-bit quantization, we have not considered other parameter-level pre-trained models like Llama3-70B which will be explored in the future.

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*.
- AI@Meta. 2024. *Llama 3 model card*.
- Franklin Allen, Xian Gu, and Julapa Jagtiani. 2021. A survey of fintech research and policy discussion. *Review of Corporate Finance*, 1:259–339.
- Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.
- Ingrid E Fisher, Margaret R Garnsey, and Mark E Hughes. 2016. Natural language processing in accounting, auditing and finance: A synthesis of the literature with a roadmap for future research. *Intelligent Systems in Accounting, Finance and Management*, 23(3):157–214.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.
- Hui Huang, Bing Xu, Xinnian Liang, Kehai Chen, Muyun Yang, Tiejun Zhao, and Conghui Zhu. 2024. Multi-view fusion for instruction mining of large language model. *Information Fusion*, page 102480.
- Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. 2018. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713.
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.
- Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. Large language models in finance: A survey. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 374–382.
- Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. <https://github.com/huggingface/peft>.
- Toan Nguyen-Mau, Anh-Cuong Le, Duc-Hong Pham, and Van-Nam Huynh. 2024. An information fusion based approach to context-based fine-tuning of gpt models. *Information Fusion*, 104:102202.
- Eugene Sy, Tzu-Cheng Peng, Shih-Hsuan Huang, Heng-Yu Lin, and Yung-Chun Chang. 2023. Fine-grained argument understanding with bert ensemble techniques: A deep dive into financial sentiment analysis. In *Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023)*, pages 242–249.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. 2024. The finben: An holistic financial benchmark for large language models. *arXiv preprint arXiv:2402.12659*.
- Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Denghui Zhang, Rong Liu, Jordan W Suchow, and Khaldoun Khashanah. 2024. Finmem: A performance-enhanced llm trading agent with layered memory and character design. In *Proceedings of the AAAI Symposium Series*, volume 3, pages 595–597.
- Zhihan Zhou, Liqian Ma, and Han Liu. 2021. Trade the event: Corporate events detection for news-based event-driven trading. *arXiv preprint arXiv:2105.12825*.