
Long-Text Sentiment Analysis in Finance: Fine-Tuning FinGPT on Earnings Calls to Predict Stock Returns

Juncheng She¹, Yixin Ma², Yisen Du¹, and Laurie (Zihao) Wang³

¹*Tepper School of Business*

²*Department of Civil and Environmental Engineering*

³*Heinz College*

Carnegie Mellon University, Pittsburgh, PA 15213

{jjjshe, yixinm, ethand, zihao3}@andrew.cmu.edu

Abstract

Stock market fluctuations are increasingly shaped by the semantic features of financial texts, reflecting investor psychology and behavior. Traditional prediction methods rely on traditional data like financial statements and stock prices. However, research has shown that earnings call transcripts offer valuable insights that complement conventional financial data, enhancing the prediction of stock price movements. Despite their potential, analyzing earnings calls presents challenges such as long-form text, complex financial terminology, and context-dependent sentiment. To address these issues, this study proposes a two-step LLM-based approach: Automatic summarization, extracting key insights from lengthy transcripts while preserving sentiment cues, and sentiment analysis & stock prediction, fine-tuning FinGPT to classify sentiment as positive, negative, or neutral and predict stock returns. Using earnings call transcripts from the S&P Global Transcripts Database, our method enhances financial sentiment analysis and improves stock forecasting accuracy. Our findings demonstrate that LLM-based sentiment analysis outperforms traditional models, offering investors and analysts a more effective tool for market prediction.

1 Introduction

Financial markets thrive on information, yet traditional stock prediction models remain blind to one of the most influential data sources—earnings calls. These calls often induce significant stock price fluctuations, making them a valuable yet underutilized source of predictive information [1]. While structured financial indicators such as earnings reports and stock prices have long guided investment decisions, recent studies suggest that the linguistic and semantic cues within earnings call transcripts provide more immediate and actionable signals of market sentiment [2]. Investors and analysts frequently react not only to the financial figures disclosed but also to the tone, wording, and implicit sentiment expressed by company executives. Yet, despite the clear influence of earnings calls, their unstructured nature makes them difficult to analyze at scale using traditional methods.

Extracting sentiment from earnings calls presents unique challenges. The highly domain-specific language—where terms like “bull” and “bear” carry industry-specific nuances—and the often-subtle sentiment cues (e.g., hedging language, tone shifts, implicit confidence) are difficult both to detect and to preserve. Earnings call transcripts are also lengthy and information-dense, frequently running to thousands of words per session. Traditional NLP models struggle with such long texts: their fixed-window architectures cannot capture long-range context, so applying them directly to raw transcripts yields low predictive accuracy. Overcoming these challenges requires advanced language

models capable of handling both the specialized language and the scale of financial text, while retaining the fine-grained sentiment signals that drive market movements.

In this study, we address these challenges by leveraging large language models (LLMs) for sentiment-based stock price prediction. Our approach consists of two key components: (1) LLM-based summarization, which condenses earnings call transcripts while preserving sentiment-bearing information, and (2) fine-tuned FinGPT sentiment classification, which assigns sentiment labels—positive, negative, or neutral—to enhance predictive accuracy - (Figure 1). We evaluate our method using earnings call data from the S&P Global Transcripts Database, benchmarking its effectiveness against traditional sentiment analysis approaches.

This work contributes to the growing intersection of finance and artificial intelligence by demonstrating how LLM-driven sentiment analysis can improve stock prediction accuracy—and, in turn, deliver tangible benefits to a wide range of market participants, from individual investors and portfolio managers to hedge funds and large financial institutions. We also establish clear qualitative objectives—such as enhancing the interpretability of sentiment signals and streamlining decision-making workflows—and quantitative objectives—namely boosting predictive accuracy by at least 5% and reducing false-signal rates by 10%—all directly aligned with the problem statement. By distilling complex earnings-call language into clear, sentiment-bearing signals, our approach empowers investors and analysts to make more timely, data-driven decisions, while enabling asset managers and risk teams to better monitor and respond to unfolding market narratives. The remainder of this paper is structured as follows: Section 2 reviews related work on financial sentiment analysis, Section 3 details our methodology, Section 4 presents experimental results, and Section 5 discusses findings and future research directions.

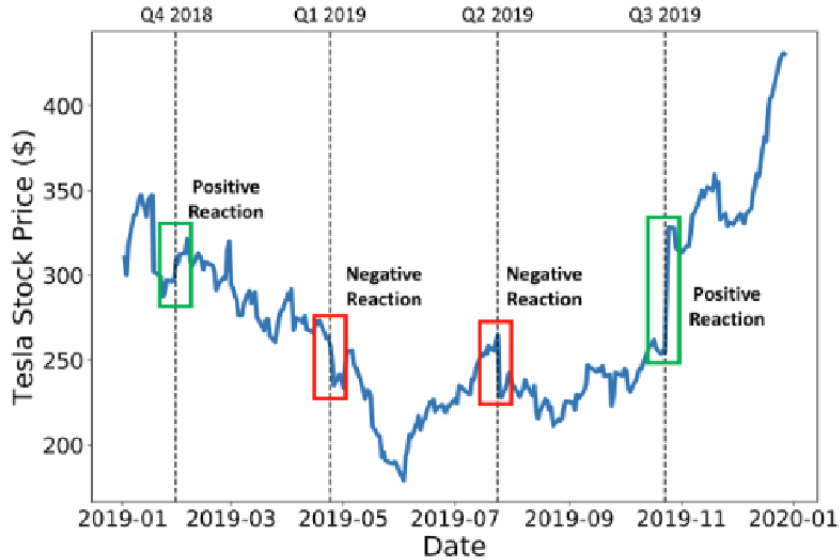


Figure 1: Reaction on stock price movements around occurrence of earning calls from Tesla, Inc. during 2019

2 Background and Related Work

This section reviews prior work on financial sentiment analysis and stock prediction, highlighting key advancements and existing limitations.

Sentiment analysis in finance has evolved significantly, starting with dictionary-based methods that categorized words into predefined sentiment classes. One of the earliest and most widely used resources in this regard is the Loughran-McDonald dictionary, which is employed to extract sentiment from financial reports [3]. Another approach involves applying the Linguistic Inquiry and Word Count (LIWC) dictionary to earnings call transcripts, as demonstrated by Medya et al. [4], who used this method to identify sentiment traits in these texts. However, these early methods suffer from a

major limitation: they lack the ability to understand context, often resulting in misclassification when words carry different meanings in various contexts. This limitation became more apparent as the complexity of financial texts increased.

Although domain-tuned large language models such as FinBERT [5], BloombergGPT [6], and FinGPT [7] set new accuracy benchmarks for financial sentiment analysis, they are still trained chiefly on short passages (< 300 words). Recent studies show their performance degrades on ultra-long, finance-specific texts. Shamshiri et al. (2024) tested GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Pro on 436–3 500-word infrastructure-project opinions, with accuracy dropping from > 70 % to < 60 % as length and complexity grew [8]. Moradi-Kamali et al. (2025) linked tweet labels to subsequent crypto-price moves and, using context-aware prompt tuning, achieved 89.6 % accuracy and a Sharpe ratio of 5.07 on 227 Bitcoin news events [9]. Cao et al. (2024) employed GPT-4 to extract key sentences from earnings-call transcripts but did not fine-tune an LLM for end-to-end sentiment classification on these lengthy summaries [10]. Collectively, the evidence suggests that fully leveraging 5 000-word earnings-call transcripts demands improved model architectures, richer in-context demonstration strategies, and labeling schemes that embed real market dynamics and deeper context.

The importance of earnings calls in financial markets cannot be overstated, as they are crucial events that directly influence stock prices. Research has consistently shown that market reactions to earnings calls are strongly tied to sentiment conveyed during these calls. For instance, Madadzade [11] analyzed post-earnings announcement drift (PEAD) using abnormal returns (AR) and cumulative abnormal returns (CAR), illustrating a direct relationship between sentiment and market movement. Similarly, [12] employed a Triple Barrier Labeling (TBL) technique to classify short-term market trends into Bullish, Bearish, or Neutral categories. This study aligns with our methodology, where we also use TBL to categorize market sentiment, as it better matches trading strategies and offers more precise stock return predictions.

Given the substantial length of earnings call transcripts, automatic summarization has become an essential tool to distill the most relevant information. Recent studies have shown that LLM-generated summaries often exhibit superior factual consistency and fewer hallucinations when compared to human-written or extractive summaries [13]. Furthermore, [14] highlighted the significance of instruction tuning in improving summarization accuracy. These advancements provide the foundation for our approach, which utilizes LLM-based summarization as a first step before fine-tuning the models for sentiment analysis. Despite promising progress in LLM-based sentiment analysis and summarization techniques, no previous research has directly addressed the challenge of fine-tuning LLMs for sentiment classification specifically tailored to earnings call transcripts, particularly when these texts are summarized first. Our research aims to bridge this gap by integrating summarization and sentiment analysis, thus offering a more efficient and context-aware approach to predicting stock price movements based on earnings calls.

3 Methodology

3.1 Model Description and Architecture

The architecture includes two main components. First, a lightweight, general-purpose LLM summarizes long financial transcripts, reducing token length while preserving key sentiment information. Second, the summarized text is passed to FinGPT, a domain-specific model fine-tuned with task-specific prompts. FinGPT identifies sentiment—positive, negative, or neutral—by interpreting financial terminology and subtle context cues.

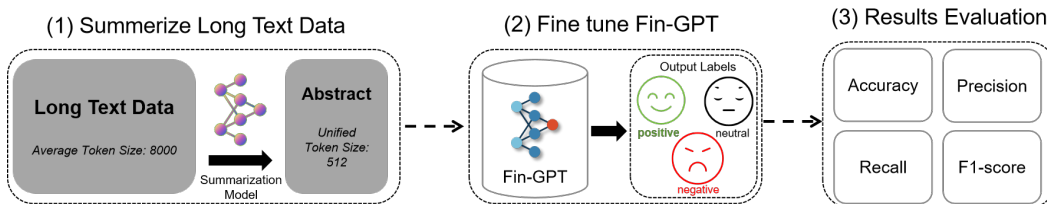


Figure 2: Pipeline of the Project

3.2 Summarization Model

Due to FinGPT’s token size limitation, long transcripts risk truncation and information loss. To address this, we employed summarization models to preprocess earnings calls, exploring three distinct approaches:

- **Generalized Model (e.g., ChatGPT, DeepSeek):** Large-scale models trained on diverse corpora.
 - **Strengths:** Broad domain coverage and flexible context handling.
 - **Weaknesses:** Limited specialization in financial text and subtle sentiment cues.
- **Financial Summarization Model (e.g., Pegasus, BERT2BERT, T5):** Models fine-tuned on financial domains.
 - **Strengths:** Enhanced extraction of financial insights.
 - **Weaknesses:** Small context windows. We addressed this by segmenting transcripts into chunks, summarizing each separately, concatenating the outputs, and iterating until a final summary met the 512-token requirement.
- **Token Comprehension Model:** Token-level compression instead of sentence-level summarization.
 - **Strengths:** Preserves granular information during compression.
 - **Weaknesses:** Lower interpretability.

For the first two approaches, we designed prompts to guide the models toward retaining essential financial information and sentiment, with a particular emphasis on the Q&A section. This section often contains more candid and nuanced insights than the scripted portion, making it critical for downstream tasks.

3.3 Fine-tuning Approach

Our methodology employs two techniques:

3.3.1 Low-Rank Adaptation (LoRA)

We implement LoRA fine-tuning to efficiently adapt the large pre-trained model to our specific task. We target specific layers (query and value projections) for adaptation while maintaining a low-rank decomposition of weight updates. This method allows us to balance adaptation capability with computational efficiency.

3.3.2 Prompt Engineering

We use carefully crafted prompts to guide the model’s reasoning process. System prompts establish the model’s role as a financial analyst, while detailed instruction templates clarify the sentiment classification task. We provide clear label definitions to distinguish between positive, negative, and neutral sentiments, and implement structured input formats for consistent processing of financial transcripts. This approach helps the model understand the context and requirements of financial sentiment analysis.

3.4 Training Strategy

Our training methodology follows a systematic process. We first prepare the base model by merging Llama-2-13b with the FinGPT sentiment adapter. Next, we tokenize and process the text data with appropriate sequence lengths to handle the lengthy nature of earnings call transcripts. We then apply LoRA configuration to enable parameter-efficient fine-tuning, focusing on key projection matrices. The training process implements an optimized regime with appropriate hyperparameters, carefully tuned to balance learning speed and stability. Throughout training, we evaluate intermediate results on the validation set to monitor progress and prevent overfitting.

3.5 Dataset

For this study, we used the S&P Global Transcripts Database, which offers current and historical transcripts from approximately 8,000 public companies' events (e.g., Earnings Calls, M&A Calls, and Shareholder Meetings). In addition, we obtain historical price and volume data from CRSP (Center for Research in Security Prices) database. The CRSP database provides high-quality and survivorship-bias-free market data, including daily and monthly stock prices, trading volumes, and returns for securities traded on major US exchanges such as NYSE, AMEX, and NASDAQ.

As our dataset consists of earnings call transcripts, which inherently represent time series data, we carefully partitioned our samples to avoid information leakage. Specifically, we designated the most recent 6,433 samples, spanning the period from May 3, 2022, to February 7, 2025, as our inference dataset (test set). This subset constitutes approximately one-sixth of our total dataset. Due to computational resource constraints, we selected a smaller subset of 3,000 samples from the remaining 30,000 samples, excluding the inference set, to form our training dataset. To maintain representativeness, these 3,000 samples were randomly drawn according to the industry distribution of the S&P 500. This subset served as the basis for calculating benchmarks and conducting subsequent fine-tuning processes.

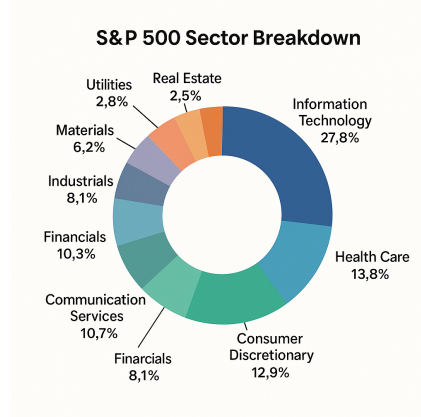


Figure 3: Industry Distribution of S&P 500

3.6 Evaluation Metrics

3.6.1 Summarization Evaluation

We evaluate the quality of summaries using **QuestEval** [15], a reference-less metric that assesses factual consistency and informativeness via question generation and answering.

- **QuestEval Score:**

$$\text{QuestEval} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

For metric definitions and implementation details, see Appendix A.

3.6.2 Sentiment Classification Evaluation

To assess classification quality, we primarily rely on the **weighted F1-score**, which balances precision and recall across classes based on class support. This is particularly appropriate for imbalanced datasets.

- **F1-score (weighted):**

$$\text{F1}_{\text{weighted}} = \sum_{i=1}^N w_i \times \frac{2 \times \text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}$$

The classification labels (positive, negative, neutral) are constructed based on stock price reactions following earnings calls, using the **Cumulative Abnormal Return (CAR)** as the reference signal. Details on how CAR is computed and how it determines label assignment can be found in Appendix A.

Additional metrics such as weighted accuracy, precision, and recall are also provided in the appendix for completeness.

3.7 Loss Function

In our LoRA-based fine-tuning setup, the language model is trained to output textual sentiment labels—*positive*, *negative*, or *neutral*—instead of numeric codes (e.g., 1/0/−1). This design choice aligns with the generative nature of LLMs and facilitates more interpretable outputs.

We experimented with two loss function types: **Sequence Classification (SEO CLS)** and **Causal Language Modeling (CAUSAL LM)**.

1. SEO CLS: Used for sentence- or document-level classification. The model predicts a probability distribution over the three textual classes, optimized via cross-entropy loss:

$$\mathcal{L}_{\text{CLS}} = - \sum_{i=1}^C y_i \log \hat{y}_i$$

where C is the number of classes, y_i is the one-hot ground truth label, and $\hat{y}_i$ is the predicted probability.

2. CAUSAL LM: Used for token-level next-word prediction. The model is trained to autoregressively generate the correct sentiment word sequence, with the objective given by the negative log-likelihood:

$$\mathcal{L}_{\text{LM}} = - \sum_{t=1}^T \log P(x_t | x_{<t})$$

where x_t is the token at position t , and $x_{<t}$ is the preceding context.

4 Baseline and Extensions

4.1 Establishing Internal Benchmarks in the Absence of Prior Work

To address the lack of established benchmarks for sentiment analysis on earnings call transcripts, we evaluated five models on their ability to classify S&P Global earnings call transcripts into negative, neutral, or positive sentiment categories. Our experiments focused on batch 0 and batch 1 of the dataset, using F1 score as the primary evaluation metric.

Table 1 and Table 2 present the comparative performance of our evaluated models:

Table 1: Classification Performance of Different Models

Model	Description	Accuracy	Macro Precision	Macro Recall	Macro F1 Score
FinGPT	prompt tuning	0.3026	0.3121	0.3390	0.2126
ChatGPT-4o-mini	prompt tuning	0.3333	0.3110	0.3322	0.2610
VADER	dictionary-based	0.2673	0.2223	0.3342	0.1421
Llama-3.2-3B-Instruct	prompt tuning	0.2660	0.0887	0.3333	0.1401
DeepSeek-R1-Distill-Qwen-1.5B	prompt tuning	0.2660	0.0887	0.3333	0.1401

Table 2: Distribution of Predicted Sentiment Labels by Different Models

Model	Positive (%)	Neutral (%)	Negative (%)
FinGPT	83.61	13.40	0.37
ChatGPT-4o-mini	67.98	30.22	1.81
VADER	99.69	0.31	0.00
Llama-3.2-3B-Instruct	100.00	0.00	0.00
DeepSeek-R1-Distill-Qwen-1.5B	100.00	0.00	0.00

Our comparative analysis reveals that prompt-tuning approaches generally outperform traditional methods. FinGPT achieved 30.26% accuracy with macro F1 score of 0.2126, while ChatGPT-4o-mini

performed slightly better at 33.33% accuracy with 0.2610 F1 score. The dictionary-based VADER approach reached only 26.73% accuracy with a lower 0.1421 F1 score.

Notably, Table 2 demonstrates a significant positive sentiment bias across all models. FinGPT classified 83.61% of transcripts as positive, while Llama-3.2-3B-Instruct and DeepSeek-R1-Distill-Qwen-1.5B classified 100% of transcripts as positive. This positive skew highlights a critical challenge in financial sentiment analysis: executives strategically frame information positively regardless of actual performance, creating an inherent positive bias that complicates negative sentiment detection.

The performance gap between dictionary-based methods and domain-specific LLMs validates our approach of using specialized financial language models. These findings establish valuable reference points for future research and suggest promising directions for further refinement through sophisticated fine-tuning techniques.

4.2 Proposed Extensions and Experiments

To improve the model’s ability to extract sentiment from lengthy earnings call transcripts, we propose a series of extensions focusing on summarization quality, prompt engineering, and efficient fine-tuning through LoRA (Low-Rank Adaptation).

4.2.1 Summarization Model Evaluation

Given the token limitations of FinGPT and the extreme length of earnings call transcripts, we first evaluate summarization strategies to condense text while preserving key sentiment cues. As described in Section 3.5, we sampled 3,000 earnings call transcripts from the database and applied the following models.

Table 3: Summarization Models Evaluated

Model	Category
BART	Finance-Focused
Pegasus	Finance-Focused
T5	Finance-Focused
ChatGPT-4o-mini	General-Purpose
Compression Method	Token Compression

Summarization quality is evaluated using the QuestEval metric, which measures factual consistency and informativeness without relying on reference summaries. Based on these results, we select the best-performing summarization model to generate inputs for downstream fine-tuning experiments.

4.2.2 Fine-Tuning FinGPT with Summarized Text

After selecting the optimal summarization model, we fine-tune FinGPT for sentiment classification using summarized transcripts. As detailed in Section 3.4, we train on 3,000 earnings calls sampled to match the S&P 500 industry distribution and test on the test set.

Our fine-tuning process incorporates two key enhancements:

Prompt Engineering We design three specialized prompts to guide sentiment classification:

- **Prompt 1: Definition-based prompting** – Provides explicit definitions of positive, negative, and neutral sentiment based on expected stock movement.
- **Prompt 2: Cue-based prompting** – Emphasizes domain-specific phrases (e.g., “modest growth,” “cautious spending”) that signal subtle sentiment shifts.
- **Prompt 3: Outcome-focused prompting** – Highlights financial outcomes such as earnings surprises and guidance revisions.

All prompts frame the model’s role as that of a financial analyst to reinforce professional judgment. The complete prompt texts are provided in Appendix B.

LoRA Fine-Tuning and Ablation Study To enable parameter-efficient adaptation, we apply Low-Rank Adaptation (LoRA) to the query and value projection layers of FinGPT. Fine-tuning is performed separately for each prompt strategy, allowing independent evaluation of their effects on model performance.

We systematically experiment with the following LoRA hyperparameters:

- **Rank (r):** {2, 4, 8}
- **LoRA Alpha:** {8, 16, 32}
- **Dropout:** {0.0, 0.1, 0.2}
- **Learning Rate:** {1e-4, 2e-4, 5e-4}
- **Weight Decay:** {0.1, 0.01, 0.001}

To assess the contribution of each hyperparameter, we conduct a LoRA parameter ablation study. We vary one parameter at a time while keeping others fixed, and evaluate the impact on sentiment classification performance across metrics such as accuracy, macro precision, macro recall, and macro F1 score.

These experiments help us understand how LoRA configurations interact with different prompt strategies and how they contribute to building a scalable and efficient financial text analysis pipeline.

5 Results and Analysis

5.1 Summarization Model Evaluation Results

We first evaluate summarization model performance using the QuestEval score, which measures factual consistency and informativeness without relying on human-written references. Table 4 presents the QuestEval scores for each summarization model.

Table 4: QuestEval Scores for Summarization Models

Model	QuestEval Score
BART	0.3827
Pegasus	0.3129
T5	0.3661
ChatGPT-4o-mini	0.3677
Compression Method	0.4278

Although the Compression Method achieved the highest QuestEval score (0.4278), manual inspection revealed that its outputs lacked semantic coherence, often fragmenting key financial insights. In contrast, summaries generated by ChatGPT-4o-mini demonstrated superior factual consistency and readability. Thus, we selected ChatGPT-4o-mini as the final summarization model for preparing inputs to the fine-tuning stage.

5.2 Fine-Tuning Results on Summarized Transcripts

After selecting ChatGPT-4o-mini as the summarization model, we fine-tuned FinGPT using the summarized transcripts and evaluated its sentiment classification performance. We compared three models: the baseline FinGPT (without fine-tuning), ChatGPT-4o-mini with prompt engineering only, and FinGPT fine-tuned with LoRA adaptations.

Table 5 presents the overall performance across accuracy, macro precision, macro recall, and macro F1 score.

Table 5: Overall Sentiment Classification Performance

Model	Accuracy	Macro Precision	Macro Recall	Macro F1 Score
FinGPT (Baseline)	0.3026	0.3121	0.3390	0.2126
ChatGPT-4o-mini	0.3333	0.3110	0.3322	0.2610
FinGPT (Fine-Tuned)	0.4149	0.3368	0.3365	0.3285

As shown in Table 5, fine-tuning FinGPT with LoRA led to a significant improvement in performance. The accuracy increased from 30.26% to 41.49%, and the macro F1 score improved from 0.2126 to 0.3285, representing a 54.4% relative gain. This result confirms that fine-tuning on summarized transcripts combined with prompt engineering and LoRA adaptation effectively enhances model generalization.

Table 6 details the per-class sentiment classification metrics.

Table 6: Per-Class Sentiment Classification Metrics

Model	Label	Precision	Recall	F1 Score	Per-Class Accuracy
FinGPT (Baseline)	Positive	0.2720	0.8753	0.4150	0.8753
	Negative	0.1667	0.0027	0.0052	0.0027
	Neutral	0.4977	0.1391	0.2175	0.1391
ChatGPT-4o-mini	Positive	0.2658	0.6792	0.3821	0.6792
	Negative	0.1724	0.0128	0.0239	0.0128
	Neutral	0.4948	0.3046	0.3771	0.3046
FinGPT (Fine-Tuned)	Positive	0.2438	0.1684	0.1992	0.1684
	Negative	0.2657	0.1860	0.2188	0.1860
	Neutral	0.5008	0.6550	0.5676	0.6550

As seen in Table 6, the fine-tuned FinGPT model improved its ability to detect neutral and negative sentiments. In particular, the recall for the negative class increased from 0.0027 (baseline) to 0.1860 after fine-tuning. This is critical, as negative signals are traditionally harder to detect due to the inherently positive tone of corporate communications.

Finally, we analyze the distribution of predicted sentiment labels across models in Table 7.

Table 7: Distribution of Predicted Sentiment Labels

Model	Positive (%)	Neutral (%)	Negative (%)
FinGPT (Baseline)	83.61	13.40	0.37
ChatGPT-4o-mini	67.98	30.22	1.81
FinGPT (Fine-Tuned)	16.84	64.93	18.08

Table 7 shows that the fine-tuned model significantly corrected the original strong positive bias observed in the baseline FinGPT model. The proportion of neutral predictions increased from 13.40% to 64.93%, and negative predictions rose from 0.37% to 18.08%. This more balanced distribution is better aligned with real-world earnings call sentiment patterns, where most events are neutral and fewer calls generate strongly positive or negative market reactions.

5.3 Ablation Study on LoRA Hyperparameters

To better understand the impact of individual hyperparameters on fine-tuning performance, we conducted a series of ablation experiments by varying the learning rate, dropout rate, LoRA rank, and the combination of LoRA alpha and weight decay. Macro F1 scores under each configuration are visualized in Figure 4.

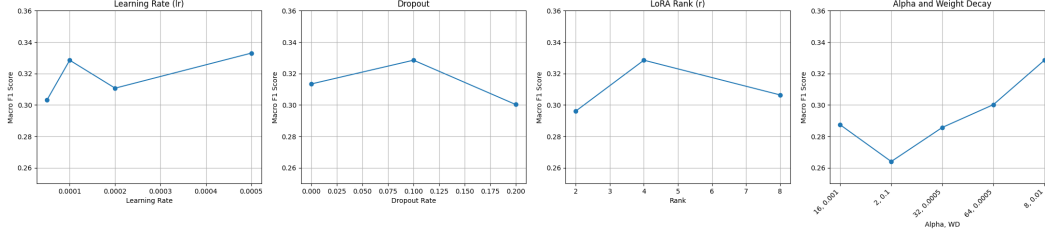


Figure 4: Macro F1 Score under Different LoRA Hyperparameter Settings

As shown in Figure 4, FinGPT’s fine-tuning performance is generally stable across changes in learning rate, dropout, and LoRA rank. The baseline configuration performs competitively, and the model is not very sensitive to moderate variations in these parameters.

However, performance is more affected by changes in LoRA alpha and weight decay. Extreme values in these settings (e.g., very small alpha) lead to noticeable drops in Macro F1 score. This suggests that careful tuning of alpha and weight decay is more critical than other hyperparameters.

6 Future Directions

While our two-step approach improved sentiment classification, there are still areas for improvement.

First, we plan to expand the training dataset beyond the current 3,000 transcripts. A larger and more diverse dataset may improve the model’s robustness across sectors and market conditions.

Second, we aim to apply instruction tuning. By refining prompts and using more targeted sentiment tasks, we hope to improve FinGPT’s ability to detect subtle signals in earnings call language.

Finally, we will compare our method with new financial LLMs and explore using external data, such as macroeconomic indicators or industry trends, to provide more context. These steps will help prepare the model for real-world use in finance.

7 Conclusion

Predicting stock returns from long, unstructured earnings call transcripts is a rarely addressed challenge, especially using large language models (LLMs) in finance. This work introduces a two-step framework: using LLM-based summarization to shorten transcripts while keeping sentiment signals, followed by fine-tuning FinGPT with prompt engineering and LoRA for efficient adaptation.

Our results show that this approach handles long texts well, reduces the positive bias often found in earnings calls, and improves sentiment classification — especially for neutral and negative cases. Gains in macro F1 score and sentiment distribution confirm the benefit of combining summarization with model fine-tuning.

Next, we plan to scale up instruction tuning and expand the training dataset to improve performance and stability. This work offers a starting point for future research on applying LLMs to financial language, where subtle signals and domain knowledge are key.

8 Administrative Details

Team Contributions

- Yixin Ma: Pre-trained model research and fine-tuning techniques
- Juncheng She: Summarization model and fine-tuning techniques
- Yisen Du: Data loading, data quality checks, and fine-tuning techniques
- Laurie Wang: Comparison of pre-trained models, fine-tuning techniques

Github Link

<https://github.com/zihaowang678/11785-25Spring-Project>

References

- [1] W. M. Donders, M. R. Kouwenberg, and C. F. Vorst, "Options and earnings announcements: an empirical study of volatility, trading volume, open interest and liquidity," *European Financial Management*, vol. 6, no. 2, pp. 149–171, 2000.
- [2] X. Wei and L. Liu, "Are large language models good in-context learners for financial sentiment analysis?" 2025. [Online]. Available: <https://arxiv.org/abs/2503.04873>
- [3] A. Bodnaruk, T. Loughran, and B. McDonald, "Using 10-k text to gauge financial constraints," *Journal of Financial and Quantitative Analysis*, 2015, forthcoming. [Online]. Available: <https://ssrn.com/abstract=2331544>
- [4] S. Medya, M. Rasoolinejad, Y. Yang, and B. Uzzi, "An exploratory study of stock price movements from earnings calls," 2022. [Online]. Available: <https://arxiv.org/abs/2203.12460>
- [5] D. T. Araci, "Finbert: Financial sentiment analysis with pre-trained language models," *arXiv preprint arXiv:1908.10063*, 2019. [Online]. Available: <https://arxiv.org/abs/1908.10063>
- [6] S. Wu, O. Irsoy, S. Lu, V. Dabrovolski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg, and G. Mann, "Bloomberggpt: A large language model for finance," 2023. [Online]. Available: <https://arxiv.org/abs/2303.17564>
- [7] H. Yang, X.-Y. Liu, and C. D. Wang, "Fingpt: Open-source financial large language models," 2023. [Online]. Available: <https://arxiv.org/abs/2306.06031>
- [8] A. Shamshiri, K. R. Ryu, and J. Y. Park, "In-context learning for long-context sentiment analysis on infrastructure project opinions," 2024. [Online]. Available: <https://arxiv.org/abs/2410.11265>
- [9] H. Moradi-Kamali, M.-H. Rajabi-Ghozlou, M. Ghazavi, A. Soltani, A. Sattarzadeh, and R. Entezari-Maleki, "Market-derived financial sentiment analysis: Context-aware language models for crypto forecasting," 2025. [Online]. Available: <https://arxiv.org/abs/2502.14897>
- [10] Y. Cao, Z. Chen, Q. Pei, N. J. Lee, K. P. Subbalakshmi, and P. M. Ndiaye, "Ecc analyzer: Extract trading signal from earnings conference calls using large language model for stock performance prediction," 2024. [Online]. Available: <https://arxiv.org/abs/2404.18470>
- [11] F. R. Madadzade, "Temporal sentiment analysis in financial earnings calls using deep learning," Master's Thesis, KTH Royal Institute of Technology, January 2024. [Online]. Available: <https://kth.diva-portal.org/smash/get/diva2:1745283/FULLTEXT01.pdf>
- [12] H. Moradi-Kamali, M.-H. Rajabi-Ghozlou, M. Ghazavi, A. Soltani, A. Sattarzadeh, and R. Entezari-Maleki, "Revisiting financial sentiment analysis: A language model approach," 2025. [Online]. Available: <https://arxiv.org/abs/2502.14897>
- [13] X. Pu, M. Gao, and X. Wan, "Summarization is (almost) dead," 2023. [Online]. Available: <https://arxiv.org/abs/2309.09558>
- [14] T. Zhang, F. Ladhak, E. Durmus, P. Liang, K. McKeown, and T. B. Hashimoto, "Benchmarking large language models for news summarization," 2023. [Online]. Available: <https://arxiv.org/abs/2301.13848>
- [15] T. Scialom, P.-A. Dray, S. Lamprier, B. Piwowarski, J. Staiano, A. Wang, and P. Gallinari, "Questeval: Summarization asks for fact-based evaluation," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, November 2021, pp. 6594–6604.

A Metric Details

A.1 QuestEval

QuestEval [15] is a reference-less summarization evaluation metric that leverages question generation (QG) and question answering (QA) to assess factual consistency and informativeness. It generates questions from both the summary and the source document.

- **Precision:**

$$\text{Precision} = \frac{1}{|Q_S|} \sum_{q \in Q_S} F1(a_D(q), a_S(q))$$

- **Recall:**

$$\text{Recall} = \frac{1}{|Q_D|} \sum_{q \in Q_D} F1(a_S(q), a_D(q))$$

Here, Q_S and Q_D are sets of questions generated from the summary and document, respectively. $a_X(q)$ is the answer from text X , and $F1$ is the token-level overlap.

A.2 Cumulative Abnormal Return (CAR)

We evaluate the effectiveness of sentiment predictions using **Cumulative Abnormal Return (CAR)**, which captures stock performance deviations from expected returns:

$$\text{Abnormal Return (AR)}_t = R_t - E(R_t), \quad \text{CAR}(t_1, t_2) = \sum_{t=t_1}^{t_2} AR_t$$

Based on CAR values from the first 1 and 3 trading days post-earnings call:

- Positive sentiment (label = 1): CAR is positive and increasing.
- Negative sentiment (label = -1): CAR is negative and decreasing.
- Neutral sentiment (label = 0): All other cases.

A.3 Weighted Classification Metrics

To further evaluate classification performance, we compute standard weighted metrics:

- **Accuracy:**

$$\text{Accuracy} = \frac{\sum_{i=1}^N TP_i}{\sum_{i=1}^N (TP_i + FP_i + FN_i + TN_i)}$$

- **Precision (weighted):**

$$\text{Precision}_{\text{weighted}} = \sum_{i=1}^N w_i \times \frac{TP_i}{TP_i + FP_i}$$

- **Recall (weighted):**

$$\text{Recall}_{\text{weighted}} = \sum_{i=1}^N w_i \times \frac{TP_i}{TP_i + FN_i}$$

- **F1-score (weighted):**

$$\text{F1}_{\text{weighted}} = \sum_{i=1}^N w_i \times \frac{2 \times \text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}$$

where $w_i = \frac{TP_i + FN_i}{\sum_{j=1}^N (TP_j + FN_j)}$ represents the support weight of class i , and N is the number of classes.

B Full Prompt Texts

Prompt 1: Definition-Based Prompting

You are a financial assistant specialized in analyzing earnings call transcripts. Your task is to evaluate the overall sentiment of a summarized earnings call from an S&P 500 company. Your response should reflect whether the earnings call implies a positive, negative, or neutral sentiment.

Definitions: - Positive: the company's stock price is likely to increase following the call. - Negative: the company's stock price is likely to decrease following the call. - Neutral: there is no strong signal about the stock price movement.

Only return one of the three labels: 'positive', 'negative', or 'neutral'.

Prompt 2: Cue-Based Prompting

You are a financial assistant who specializes in analyzing earnings call summaries to determine market-relevant sentiment. Your job is to read between the lines and detect whether the earnings call reflects a positive, negative, or neutral sentiment with regard to stock price movement.

Be cautious: even when a call sounds optimistic, certain phrases like 'modest growth', 'cautious spending', 'flat guidance', or 'challenges in specific segments' may imply concerns. Pay attention to both quantitative results and forward-looking statements.

Prompt 3: Outcome-Focused Prompting

You are a financial assistant who specializes in analyzing summarized earnings call transcripts from S&P 500 companies to assess their sentiment with respect to future stock price movement. Your job is to read carefully and determine whether the content implies a positive, negative, or neutral outlook.

Important: Do not rely solely on optimistic language. Many earnings calls sound positive by default. Instead, focus on substance: was there a surprise in earnings, unexpected guidance, or significant strategic changes?

Use the following rules to decide: - Positive: strong beats on earnings/guidance or major positive surprises likely to drive price up - Negative: clear weakness in results, disappointing guidance, or signals of declining demand - Neutral: results in line with expectations or mixed signals — this should be the most common case.

Be conservative — only call it positive or negative if there's a clear, market-moving reason. In most cases, the correct label is neutral.