Exploring the Intersection of Fine-Tuned and General-Purpose Language Models for Financial Sentiment Analysis

Joshua Chen
jpc005@ucsd.edu

1 Introduction

The task of sentiment analysis in financial text presents unique challenges due to specialized and nuanced language of the financial domain. Accurate sentiment analysis is critical and can be used to aid investors, analysts and financial institutions in interpreting market trends and the overall economy. Traditional pre-trained language models, such as FinBERT (Araci, 2019), have demonstrated the effectiveness of domain-specific adaptation. However, the emergence of chat-capable large language models (LLMs) such as ChatGPT, offers the opportunity to explore new methodologies like in context learning for specialized tasks like financial sentiment analysis.

This project aims to evaluate the performance of zero-shot and fewshot prompting LLMs and data augmentation against fine-tuned FinBERT in financial sentiment analysis. The Financial PhraseBank dataset (Malo et al, 2014) serves as the benchmark for evaluation.

The following schedule guided the direction of this project

- Collected and preprocessed dataset: DONE.
- Build and train FinBERT on collected dataset and examine its performance: DONE
- In context learning (zero shot and few shot) with Llama2 and GPT-4o: DONE
- Utilize data augmentation to try and improve performance: DONE

Initial findings reveal that default FinBERT struggles on the training dataset (7% accuracy), while fine-tuned FinBERT achieves strong results (85% accuracy). The performance of LLMs in zero-shot and few-shot settings further highlights the trade-offs between computational efficiency and task-specific accuracy.

2 Related Work

The field of sentiment analysis has seen significant advancements with the introduction of pre-trained language models. Traditional approaches often relied on feature engineering and domain-specific lexicons, but these methods have been largely replaced by the invention of models like BERT. For financial sentiment analysis, FinBERT was introduced as a BERT-based model fine-tuned on financial texts (Araci, 2019). This work demonstrated that domain-specific fine-tuning significantly improves model performance, especially in understanding nuanced financial language. Similarly, (Gururangan et al., 2020) emphasized the importance of domain-adaptive pretraining for specialized NLP tasks, highlighting its effectiveness across multiple domains, including finance.

Large language models, such as GPT-3, have transformed natural language processing by enabling zero-shot and few-shot learning through in-context examples (Brown et al., 2020). These models eliminate the need for task-specific fine-tuning, offering flexibility in addressing diverse tasks. Recent advancements like LLaMA 2 (Touvron et al., 2023) have further refined this approach, achieving high performance on various benchmarks. However, the effectiveness of these models in domain-specific applications, such as financial sentiment analysis, remains an area of active exploration.

Data augmentation has also emerged as a critical technique for improving model robustness in low-resource settings. Techniques such as paraphrasing and synonym replacement have shown success in enhancing the generalization capabilities of NLP models (Wei and Zou, 2019). These methods are particularly useful when labeled data is scarce.

Finally, comparative studies have provided insights into the trade-offs between task-specific tuning and general-purpose models. Peters et al.(Peters et al., 2019) examined these trade-offs, showing that while fine-tuned models excel in specialized tasks, general-purpose models often offer greater adaptability. This contrast is especially relevant in financial sentiment analysis, where both approaches have distinct advantages and limitations. By comparing fine-tuned models like FinBERT with chat-capable large language models, this project builds on these foundational studies to explore the potential of domain-specific and general-purpose NLP techniques in finance.

3 Our Dataset

The dataset used in this project is the Financial PhraseBank[1], a collection of 4,846 financial sentences labeled with sentiment categories (positive, neutral, or negative). This dataset is well-suited for sentiment analysis in the financial domain, providing a benchmark to evaluate the performance of both fine-tuned models and large language models.

3.1 Dataset Statistics

The dataset has the following characteristics:

- **Number of sentences:** 4,846
- **Total number of words:** 103,247
- **Vocabulary size:** 10,114 unique tokens
- **Most common words:** "EUR" (euro), "company," "said," "Finnish," and "MN" (million)

Examples of input-output pairs include:

- **Input:** "According to Gran , the company has no plans to move all production to Russia , although that is where the company is growing."
  Output: Neutral
- **Input:** "In the third quarter of 2010 , net sales increased by 5.2% to EUR 205.5 mn , and operating profit by 34.9% to EUR 23.5 mn "
  Output: Positive

Several aspects of the dataset make the task challenging. Financial texts include jargon, acronyms, and numerical data, which require specialized understanding to infer sentiment.

3.2 Data Preprocessing

Preprocessing was conducted using a pretrained BERT tokenizer, as provided by Yi Yang et al.[2]

This tokenizer efficiently converts sentences into tokens suitable for FinBERT, preserving domain-specific terminology and handling numerical values effectively.

4 Baselines

The baseline model for this project is a fine-tuned version of FinBERT [3], a domain-specific BERT-based model pre-trained on financial texts. This baseline evaluates the effectiveness of fine-tuning FinBERT on the Financial PhraseBank dataset for financial sentiment classification.

4.1 Baseline Model

The fine-tuned FinBERT serves as the foundation for this project. FinBERT was specifically designed for tasks in the financial domain, making it an ideal starting point for sentiment analysis of financial texts. By fine-tuning the pre-trained model on the Financial PhraseBank dataset, the model was further specialized to classify sentences as positive, neutral, or negative.

4.2 Hyperparameters and Training Setup

The fine-tuning process for FinBERT used the following hyperparameters:

- **Learning rate:** 2×10^{-5}
- **Batch size:** 16 for both training and evaluation
- **Number of epochs:** 4
- **Weight decay:** 0.01
- **Evaluation strategy:** Validation was conducted at the end of each epoch, and the best model was selected based on validation accuracy.
- **Logging:** Training progress was logged every 10 steps.

The training process was managed using the Hugging Face Trainer API. Validation accuracy

[1] https://www.researchgate.net/publication/251231364_FinancialPhraseBank-v10

[2] https://huggingface.co/yiyanghkust/finbert-tone

[3] https://huggingface.co/yiyanghkust/finbert-tone?text=growth+is+strong+and+we+have+plenty+of+liquidity

was used as the key metric for selecting the best model, with the model automatically saved at the end of the best epoch.

4.3 Train/Validation/Test Split

The Financial PhraseBank dataset was divided into three subsets:

- **Training set:** 70% of the dataset (3,392 sentences)
- **Validation set:** 15% of the dataset (727 sentences)
- **Test set:** 15% of the dataset (727 sentences)

The training set was used to optimize the model parameters, while the validation set was used to evaluate model performance at the end of each epoch and select the best-performing version. The test set was reserved exclusively for final performance evaluation.

5 Our Approaches

- **Conceptual Approach:** Our approach involves testing multiple methods and models to classify sentiment from a financial text dataset. Initially, we evaluated a default pretrained FinBERT model to establish a baseline. Subsequently, we fine-tuned FinBERT to improve its performance. We also explored in-context learning using LLaMA 2 and GPT-4 for both zero-shot and few-shot classifications. Finally, we enhanced the dataset by augmenting neutral sentences with positive sentiment paraphrases to address class imbalance and retrained FinBERT.

- **Working Implementation:**

  1. **Default FinBERT:** We tested the default FinBERT model without any fine-tuning. This resulted in a test accuracy of 0.0742 and a confusion matrix:

  $$\begin{bmatrix} 24 & 2 & 65 \\ 398 & 23 & 11 \\ 94 & 103 & 7 \end{bmatrix}$$

  2. **Fine-tuning FinBERT (Baseline):** Fine-tuning FinBERT improved the test accuracy to 0.8514 with the following confusion matrix:

  $$\begin{bmatrix} 81 & 7 & 3 \\ 18 & 385 & 29 \\ 7 & 44 & 153 \end{bmatrix}$$

  The training time was 2 minutes and 20 seconds.

  3. **Zero-shot with LLaMA 2:** Using LLaMA 2 for zero-shot classification, where the model was asked to classify each sentence in the test set with the prompt: ”Classify the sentiment of the following text as Positive, Negative, or Neutral: [text]”, resulted in a test accuracy of 0.69 and a confusion matrix:

  $$\begin{bmatrix} 12 & 79 & 0 \\ 1 & 399 & 32 \\ 0 & 110 & 94 \end{bmatrix}$$

  It appears that most of the predictions were neutral, which was the dominant class in the dataset. This process took about 1 hour.

  4. **Few-shot with LLaMA 2:** Using LLaMA 2 for Few-shot classification, where one example for each sentiment class was randomly provided from training set and model was asked to classify each sentence in test set with the prompt: ”Classify the sentiment of the following financial text as Positive, Negative, or Neutral given the below examples:[examples] and the text:[text]”, resulted in a test accuracy of 0.70 and a confusion matrix:

  $$\begin{bmatrix} 84 & 7 & 0 \\ 17 & 266 & 149 \\ 3 & 45 & 156 \end{bmatrix}$$

  Although the accuracy only improved by 1%, the model did not mostly predict neutral. This process took about 1.5 hours.

  5. **Zero-shot with GPT-4:** Zero-shot classification (same prompt as the one used in LLaMA 2) using GPT-4o achieved a test accuracy of 0.78 with the following confusion matrix:

  $$\begin{bmatrix} 91 & 0 & 0 \\ 30 & 312 & 90 \\ 4 & 34 & 166 \end{bmatrix}$$

  This process took about 15 minutes.

  6. **Few-shot with GPT-4 (Single Example per Class):** Using GPT-4 in few-shot (same prompt as the one used in

LLaMA 2) yielded a test accuracy of 0.83 with the following confusion matrix:

$$\begin{bmatrix} 89 & 2 & 0 \\ 19 & 387 & 26 \\ 1 & 74 & 129 \end{bmatrix}$$

This process took approximately 20 minutes.

7. **Few-shot with GPT-4 (Two Examples per Class):** When two examples per sentiment class were used for few-shot learning, the test accuracy was 0.81 with a confusion matrix:

   $$\begin{bmatrix} 87 & 4 & 0 \\ 19 & 387 & 26 \\ 2 & 84 & 118 \end{bmatrix}$$

   This process took about 20 minutes.

8. **Data Augmentation and Retraining FinBERT:** Noticing difficulty in distinguishing neutral and positive sentiments, we used GPT-4o to paraphrase 1000 neutral sentences into positive ones as the training data contained 2015 neutral sentences while only having 954 positive sentences. The augmented positive data was then added to the training data and was used to retrain the FinBERT model. After retraining FinBERT, the test accuracy improved to 0.862 with the following confusion matrix:

   $$\begin{bmatrix} 83 & 6 & 2 \\ 16 & 375 & 41 \\ 3 & 32 & 169 \end{bmatrix}$$

   Training took 3 minutes and 5 seconds while data augmentation took approximately 30 minutes.

- **Compute:** All experiments were conducted locally using a 3070 GPU.
- **Runtime:** Training times varied by method, as detailed above. Only the training time for FinBERT were exact, the other processing time were approximates.
- **Other Details:**
  - Code for all experiments listed above is consolidated in a single Jupyter notebook file
  - OpenAI API calls incurred costs

6 Error Analysis

Since the model struggles to distinguish between positive and neutral sentences, for error analysis, we decided to look at sentences that were labeled positive but classified as neutral and sentences that were labeled neutral but classified as positive to find commonalities.

6.1 Neutral Sentences Misclassified as Positive

> ”The contract covers the supply of temporary healthcare personnel in Finnish municipalities.”

This sentence is factual and neutral in tone. However, the mention of ”healthcare personnel” and ”supply” could have been interpreted by the models as indicating progress or success, which led to a positive sentiment classification.

6.2 Positive Sentences Misclassified as Neutral

> ”Kaido Kaare, general director for Atria Eesti, says the company’s investments in the upgrade of the pig farms surpass EEK 150mn EUR 9.59 mn USD 14.19 mn in the past years.”

This sentence describes significant financial investments, which are typically viewed as a positive action. However, the models might have struggled due to the highly factual and technical presentation of the information. The lack of explicit positive sentiment markers, such as adjectives like ”successful” or ”groundbreaking,” might have led the models to classify it as neutral.

6.3 Takeaways and Hypothesis

Overlap Vocabulary: Denmark, Finland, Finnish, June, Korea, Oyj, a, acquired, added, also, an, and, are, be, company, countries, design, for, has, he, heat, in, is, it, its, market, most, next, of, on, other, said, share, solution, terms, than, the, this, to, up, was, with.

The overlap vocabulary between misclassified positive and misclassified neutral suggests that both models have biases toward interpreting certain business and financial terms as indicators of sentiment, without fully grasping the context. For neutral sentences, terms such as ”solution” or

”market” might have triggered positive classifications, while for positive sentences, the factual presentation without emotive language could lead to sentences being misclassified as neutral.

7 Conclusion

In this project, we explored the performance of FinBERT and large language models (LLMs) such as LLaMA 2 and GPT-4o for sentiment classification in financial text. Our results show that FinBERT, fine-tuned on the dataset, achieved the best accuracy of 86.2%, outperforming the LLMs. However, LLMs, particularly GPT-4o, demonstrated competitive performance in a few-shot prompting setup, with an accuracy of 83%. This highlights the potential of LLMs for tasks where domain-specific models like FinBERT are unavailable.

One surprising finding was the poor performance of the default pre-trained FinBERT model, which achieved only 7% accuracy on the dataset. This underscores the importance of fine-tuning pre-trained models to adapt them to specific tasks and domains—even when the task is within the model's presumed domain. While FinBERT is designed for financial text, our results show that fine-tuning is still critical for aligning the model to the specific characteristics and nuances of a given dataset.

One significant observation is the accessibility of LLMs. With OpenAI's API call for GPT-4o functioning similarly to using GPT4-0 model on ChatGPT, anyone with access can replicate this study's few-shot prompting approach and achieve results comparable to FinBERT. This accessibility opens up practical use cases, such as enabling smaller firms or independent analysts to perform sentiment analysis on financial texts without needing specialized models or large computational resources.

Furthermore, we found that data augmentation can effectively improve model performance. By paraphrasing neutral examples into positive ones, we addressed class imbalances and provided the model with more diverse training data, which likely helped it better generalize sentiment patterns.

Looking ahead, future work could explore fine-tuning LLMs like GPT-4o or LLaMA 2 specifically on financial text datasets and retrying few-shot prompting to evaluate whether such domain-adapted models achieve better performance. Combining the strengths of fine-tuning and in-context learning may help bridge the gap between general-purpose LLMs and specialized models like FinBERT. Additionally, further research could leverage advanced financial-specific LLMs, such as BloombergGPT, to better understand their effectiveness for tasks like sentiment analysis.

8 Acknowledgements

ChatGPT was used to generate the title, the formatting of the Our Approach section, the footnotes in the Dataset section, and the citations. It was also used to proofread the work.

References

Araci, D. (2019). Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:1908.10063*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Gururangan, S., Marasovic, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2019). To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, M., Bashlykov, D., Batra, P., Bhargava, S., Bhosale, Y., et al. (2023). Llama 2: Open and efficient foundation language models. *arXiv preprint arXiv:2307.09288*.

Wei, J. and Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.