Finetune FinBERT for Financial Sentiment Analysis

Chen Zhijing A0281338B

I. Introduction

Sentiment analysis plays a crucial role in modern finance, where market movements are often influenced by public perception and investor confidence. The daily influx of unstructured data from financial news, social media, and corporate reports contains valuable signals that can inform investment strategies and risk management. Manually processing this vast amount of text is unfeasible, making automated sentiment analysis an indispensable tool. By accurately classifying the sentiment of financial texts, stakeholders can gain real-time insights into market trends, predict stock price volatility, and make more informed, data-driven decisions.

The advent of pre-trained Large Language Models (LLMs) has revolutionized natural language processing, but their generalist nature often falls short in specialized domains like finance. These models are trained on broad internet text and may not grasp the specific jargon, context, and nuances of financial discourse. Fine-tuning is the critical process of adapting a pre-trained model to a specific task and domain by continuing its training on a smaller, specialized dataset. This adaptation allows the model to learn the unique vocabulary and semantic relationships of the target domain, significantly boosting its performance and reliability for tasks such as financial sentiment classification.

This project compares two fine-tuning strategies for financial sentiment classification on the FinancialPhraseBank dataset using a FinBERT backbone. We evaluate (i) full fine-tuning, which updates all model parameters, and (ii) parameter-efficient fine-tuning (LoRA), which keeps the base model frozen and learns a small set of low-rank adapter weights (plus the classification head). We measure the accuracy/F1 trade-off against compute efficiency (trainable parameters, GPU memory, wall-clock).

II. Methodology

a. Data collection

For this project, the FinancialPhraseBank dataset, developed by Malo et al. (2014), was selected. This well-established corpus consists of 4,848 English sentences sourced from financial news. The primary justification for its selection is the dataset's rigorous and high-quality annotation process. Each sentence was manually labelled by a panel of 16 professionals with backgrounds in finance and business. Crucially, the annotators were instructed to assign sentiment based on the potential impact the information would have on a company's stock price. The resulting classifications fall into four categories: Positive, Negative, Neutral, and Uncertain. This domain-specific and expert-driven methodology makes the Financial PhraseBank a reliable gold-standard benchmark for training and evaluating financial sentiment analysis models.

b. Model selection: FinBERT

For this task, we selected FinBERT, a BERT-based model pre-trained specifically on a large corpus of financial texts, including corporate reports and news articles. Unlike general-purpose models such as DistilBERT, FinBERT has already learned the specific vocabulary, syntax, and context of the financial domain. This specialized pre-training provides a powerful foundation, as the model already possesses a nuanced understanding of financial jargon (e.g., "bullish," "headwinds," "volatility"), making it an ideal candidate for achieving high performance on financial sentiment classification.

c. Fine-tuning strategies

The core of this project is a direct comparison of two fine-tuning approaches to adapt the pre-trained FinBERT model to our specific dataset:

- Full-Parameter Fine-Tuning (FT): This is the traditional and most comprehensive method, where all the weights of the pre-trained FinBERT model are updated during training. While this approach allows the model to adapt fully to the new data, it is computationally expensive, requiring significant memory, processing power, and training time. It serves as our high-performance baseline.
- Low-Rank Adaptation (LoRA): This is a Parameter-Efficient Fine-Tuning (PEFT) technique. Instead of updating all the model's parameters, LoRA freezes the original weights and injects small, trainable "adapter" layers into the model. Only these lightweight adapters are trained. The motivation for this comparison is to determine if the resource efficiency of LoRA—which offers faster training and much smaller model storage—can be achieved without a significant sacrifice in performance compared to the full fine-tuning baseline.

d. Evaluation metrics

To rigorously assess the performance of each strategy, we used a set of standard classification metrics that provide a holistic view of their effectiveness:

a) Accuracy: This gives a straightforward measure of the overall percentage of headlines that were correctly classified.
b) Precision, Recall, and F1-Score: These metrics offer a more detailed performance breakdown.
    a. Precision measures the reliability of the model's positive classifications.

b. Recall measures the model's ability to identify all relevant instances of a class.

c. The F1-Score, which is the harmonic mean of precision and recall, provides a single, balanced score that is especially useful. We specifically use the macro-averaged F1-score, which calculates the F1 for each sentiment class independently and then averages them. This ensures that the performance on less frequent classes (e.g., negative) is weighted equally with more frequent ones (e.g., neutral), providing a fair assessment across all categories.

III. Experiment Setup

a. Data preprocessing

We use the FinancialPhraseBank corpus and keep only the headline text and its gold label. Text is read as UTF-8/ISO-8859-1, stripped, and mapped to three classes {negative, neutral, positive}. To control class balance and make comparisons fair, we stratify by label: for each class we sample 300 items for train, 300 for test, and draw 50 items per class for a small validation split (used for model selection). We tokenize with the FinBERT tokenizer using WordPiece, max_length=256, truncation=True, and padding="max_length". The HuggingFace Dataset objects expose input_ids, attention_mask, and integer labels.

b. Model training

FinBERT (ProsusAI/finbert) serves as the base encoder with a 3-way classification head. We train using HuggingFace Trainer with AdamW, a linear LR schedule, seed 42, and mixed precision when CUDA is available. Full fine-tuning uses batch sizes 8/8, LR 2e-5, for 5 epochs; LoRA uses batch sizes 32/64, LR 1e-3, for 5 epochs with adapters on Q/V projections (r=16, alpha=32, dropout=0.05). We keep identical splits, tokenization, and label mapping across runs.

Validation performance: LoRA achieved F1 = 0.91185, Accuracy = 0.91333, while full fine-tuning achieved F1 = 0.89767, Accuracy = 0.90000. These results indicate LoRA matched—and slightly surpassed—full fine-tuning on this task while training fewer parameters, showing its efficiency advantage.

c. Fine-tuning strategies

We compare two strategies on identical splits and tokenization.

- **Full fine-tuning:** all transformer weights plus the classifier head are updated. This maximizes capacity at higher compute/memory cost.
- **LoRA (parameter-efficient):** the backbone is frozen and low-rank adapters are inserted on

attention projections (query/value) with r=16, lora_alpha=32, and lora_dropout=0.05. Only these adapters and the classifier head are optimized. This cuts trainable parameters and optimizer state by orders of magnitude while retaining strong task adaptation.

For both strategies we use the same label mapping and evaluation protocol, reporting accuracy and weighted/macro F1 to compare effectiveness and efficiency side-by-side.

d. Result Analysis

Table 1 — Core test metrics (overall)

Method	Accuracy	Macro-F1	Weighted-F1
LoRA	0.9000	0.8997	0.8997
Full FT	0.8633	0.8630	0.8630

Table 2 — Per-class precision / recall / F1

Class	Strategy	Metrics		
		Precision	Recall	F1
Negative	Full FT	0.9262	0.9200	0.9231
	LoRA	0.9412	0.9600	0.9505
Neutral	Full FT	0.8694	0.7767	0.8204
	LoRA	0.8912	0.8467	0.8684
Positive	Full FT	0.8024	0.8933	0.8454
	LoRA	0.8673	0.8933	0.8801

e. Comparative Analysis

The results presented in the preceding tables offer a clear and compelling narrative: the LoRA-tuned model not only matched but significantly outperformed the fully fine-tuned model. This section provides a detailed analysis of these findings, interpreting the overall and per-class metrics to understand the underlying reasons for this outcome.

i. Overall Performance Insights

A review of the aggregate metrics in Table 1 immediately reveals the superior performance of the LoRA method. The LoRA model achieved a Macro-F1 score of 0.8997, substantially higher than the 0.8630 score from the Full Fine-Tuning (FT) model. This result is significant as it counters the common expectation that full fine-tuning represents the upper limit of performance. The higher accuracy and F1-scores suggest that the parameter-efficient approach led to a model that generalizes better to unseen data.

ii. Per-Class Performance Breakdown

The detailed breakdown in Table 2 confirms that LoRA's advantage is not an anomaly but a consistent trend across all sentiment categories. The most notable improvements were observed in the Neutral and Positive classes. For the Neutral class, the LoRA model's F1-score of 0.8684 far surpassed the Full FT's 0.8204, primarily driven by a dramatic increase in recall (from 0.7767 to 0.8467). This indicates that the fully fine-tuned model struggled to correctly identify neutral statements, a weakness LoRA effectively rectified.

Similarly, for the Positive class, the LoRA model's F1-score improved to 0.8801 from 0.8454. This gain was fuelled by a major enhancement in precision (from 0.8024 to 0.8673), meaning the LoRA model was far more reliable when predicting positive sentiment and made fewer false positive errors. Even for the Negative class, where the Full FT model was already performing well, LoRA provided a consistent boost to all metrics.

iii. The Role of LoRA as a Regularizer

The most plausible explanation for these results is that LoRA acted as an effective regularizer, mitigating the overfitting that likely occurred during full fine-tuning. With over 66 million trainable parameters, the Full FT model may have started to memorize noise and artifacts specific to the training set. In contrast, by freezing most of the model's weights and only updating a small set of parameters, LoRA constrains the model's adaptation. This forces it to learn more robust and generalizable features, leading to superior performance on the unseen test data.

IV. Conclusion

This project aimed to investigate the effectiveness of parameter-efficient fine-tuning (PEFT) through the LoRA approach, compared to traditional full fine-tuning, using a headline-level sentiment classification dataset. The experimental setup fine-tuned both models under identical training conditions and evaluated them with accuracy and macro-F1 as the primary metrics. Results showed that LoRA achieved comparable or even superior performance while using significantly fewer trainable parameters and computational resources. These findings highlight that PEFT methods can deliver strong predictive capability without the extensive cost typically associated with full model adaptation.

Nonetheless, the observed advantage should be interpreted considering the task characteristics. The dataset in this study consists of short, syntactically simple, and contextually constrained headlines, where LoRA’s low-rank adaptation is sufficient to capture key semantic patterns. However, for more complex tasks—such as those involving longer documents, richer contextual dependencies, or nuanced language understanding—full fine-tuning may outperform LoRA, as it updates all parameters and thus offers greater representational flexibility. Further empirical studies across varied datasets and task complexities are needed to comprehensively assess how LoRA’s efficiency and performance trade-offs generalize beyond the present setting.

V. Appendix

https://github.com/zhijing31/Finetune-FinBERT-for-Financial-Sentiment-Analysis