Exploring FinBERT for Stock News Sentiment Analysis: A Comparative Project with LSTM

Yoona Lee
Master's in Data Science
University of Michigan
Ann Arbor, Michigan, USA
yoonalee@umich.edu

***Abstract*—Sentiment analysis is critical for understanding stock market trends and guiding investment decisions. While traditional methods rely on technical indicators, recent advancements leverage domain-specific models like FinBERT and deep learning architectures such as LSTMs. This project develops a hybrid approach, integrating FinBERT with LSTM networks to enhance sentiment classification accuracy. The proposed method is evaluated on financial news datasets, addressing class imbalance through SMOTE and incorporating sentiment scores for richer feature representation. Results demonstrate the hybrid model's potential for achieving improved precision, recall, and F1-scores compared to individual models.**

***Index Terms*—Sentiment analysis, FinBERT, LSTM, Machine learning**

I. INTRODUCTION

Accurate sentiment analysis of stock-related news is essential for informed financial decision-making. Positive sentiments often indicate market optimism, while negative ones can signal potential downturns. Automating this analysis reduces the labor-intensive process of manually filtering financial news. Advances in deep learning, such as FinBERT and LSTM models, offer opportunities to streamline this process and enhance prediction accuracy. This project aims to analyze financial(stock) news sentiment using FinBERT, a pre-trained model tailored for financial text. And compare FinBERT's performance with Long Short-Term Memory (LSTM) networks using BERT tokenizer. Then explore a hybrid approach by integrating FinBERT and LSTM architecture for potential performance improvements.

Recent advancements in sentiment analysis have focused on leveraging domain-specific models like FinBERT, which achieves state-of-the-art performance in financial sentiment analysis. FinBERT excels in handling the unique linguistic challenges of financial text, as demonstrated on datasets like the Financial PhraseBank[1]. Traditional stock market prediction relied on technical indicators and lexicon-based sentiment analysis, but these approaches struggled with the complexity of financial language. The introduction of deep learning models, particularly LSTMs, revolutionized sentiment analysis by effectively capturing sequential data. However, LSTMs and RNNs face limitations in contextual understanding. Advancements such as ELMo[3] improved upon traditional embeddings by capturing context dependent word meanings through bidirectional language models. Building on these ideas, transformers like BERT introduced self-attention mechanisms[4] that provide fully bidirectional contextual representations, overcoming RNN and ELMo limitations. FinBERT, a fine-tuned version of BERT for financial text, incorporates these innovations, enabling precise sentiment classification tailored to the financial domain.

II. METHOD

The problem is framed as a multi-class sentiment classification task, where the model predicts the sentiment label (negative(0), neutral(1), or positive(2)) of an input text. The input comprises tokenized text representations, including `input_ids` and `attention_mask` processed by a pre-trained BERT model to extract contextual embeddings, and an additional numerical feature (sentimentScore). The model is trained using CrossEntropyLoss, and the architecture leverages both contextual text understanding from BERT and auxiliary numerical information for enhanced predictions.

A. Stock news sentiment dataset

The dataset for this project consisted of stock news with fields like text, sentiment, and sentimentScore, which were used for analysis. SentimentScore (ranging from 0 for Neutral, 0 to 1 for Positive, and -1 to 0 for Negative) was added as an additional feature to enhance model performance. The original dataset of 142,000 rows was reduced to 4,000 rows to optimize computation, like the size of dataset presented in FinBERT paper. To address class imbalance, SMOTE was applied, and the train-validation-test split (6:2:2) was stratified to ensure balanced sentiment distribution. This approach maintained strong model performance while reducing computational costs.

B. FinBert

This model leverages FinBERT, a pre-trained transformer model specialized for financial text, enhancing performance for sentiment classification tasks in this domain. The input consists of tokenized text sequences using FinBERT's tokenizer and a numerical feature, sentimentScore. FinBERT generates contextual embeddings for the text, which are passed through a classification head to output sentiment logits. A custom weighted loss function incorporates sentimentScore to adjust the importance of samples during training, ensuring that examples with higher sentiment intensity contribute more to the learning process as follows:

$$\text{adjusted\_loss} = \text{mean}(\text{CrossEntropyLoss}(logits, labels) \times \text{sentimentScore})$$

To handle class imbalance, SMOTE (Synthetic Minority Oversampling Technique) is applied to oversample minority classes in the training data. Evaluation metrics include accuracy and F1-score to account for class imbalances, with predictions evaluated through a detailed classification report on the test set. By combining FinBERT's domain-specific knowledge with auxiliary features and robust handling of imbalanced data, this model effectively captures sentiment nuances in financial text.

C. LSTM with BERT

This model integrates BERT[2] with an LSTM architecture to perform multi-class sentiment classification(negative, neutral, positive). The input consists of tokenized text sequences using the BERT tokenizer (bert-base-uncased) and sentimentScore. BERT generates contextual embeddings for the input text, which are processed by an LSTM to capture temporal dependencies across the token sequence. The hidden state from the LSTM is concatenated with the sentimentScore to form a hybrid feature representation, which is passed to a fully connected layer for classification.

The model is fine-tuned end-to-end, including the pre-trained BERT parameters. A CrossEntropyLoss is used as the loss function, and gradient clipping is applied to stabilize training. The AdamW optimizer with a learning rate of 2e-5 and a learning rate scheduler (StepLR) ensure efficient training. During evaluation, metrics such as validation loss and accuracy are computed to monitor performance. The inclusion of both BERT's contextual embeddings and sentimentScore makes this model capable of effectively capturing complex sentiment patterns.

D. FinBert + LSTM

This model integrates FinBERT, a pre-trained transformer model specialized for financial sentiment analysis, with an LSTM architecture[5]. So FinBERT is fine-tuned during training to adapt its embeddings to the dataset. The input consists of tokenized text sequences, processed using FinBERT's tokenizer, and an additional numerical feature, sentimentScore. FinBERT generates contextual embeddings for the text, which are then passed to an LSTM layer to capture sequential dependencies across tokens[6]. The output of the LSTM is combined with the sentimentScore to form a hybrid feature representation. This combined representation is passed to a fully connected layer for sentiment classification, predicting classes such as positive, neutral, or negative.

The model uses a weighted cross-entropy loss function that incorporates the sentimentScore to prioritize samples with higher sentiment intensity during training as follows:

$$\text{weighted\_loss} = \frac{1}{N}\sum_{i=1}^{N} \text{CrossEntropy}(y_i, \hat{y}_i) \times (1 + \text{sentimentScore}_i)$$

FinBERT is fine-tuned alongside the LSTM and fully connected layers, allowing the model to adapt its pre-trained embeddings to the specific dataset. By leveraging FinBERT's domain-specific knowledge and the sentiment score, the model achieves a robust approach to financial sentiment analysis, combining textual understanding with numerical sentiment insights.

III. RESULTS

The data pipeline and model setups involve preparing and processing financial text data for sentiment analysis using three different model architectures: FinBERT, LSTM with BERT, and FinBERT + LSTM. The data is tokenized using pre-trained tokenizers (e.g., BERT tokenizer or FinBERT tokenizer), and additional features such as sentimentScore are incorporated into the model training process. Techniques like SMOTE are applied to address class imbalance, and evaluation metrics such as loss, accuracy, and F1-score are used to monitor model performance. The training process includes fine-tuning the pre-trained models with custom optimizers and schedulers.

Fig. 1 summarizes the key performance metrics for the three models.

Model	Validation Loss	Validation Accuracy	Test Loss	Test Accuracy	F1-Score
FinBERT (Trainer)	0.4913	0.899	0.4900	0.91	0.88
LSTM with BERT	0.3644	0.8925	0.3687	0.8912	0.88
FinBERT + LSTM	0.1223	0.89	0.3672	0.8812	0.87

Fig 1. Model Performance Comparison

The table presents a comparative analysis of three sentiment analysis models: FinBERT, LSTM with BERT, and FinBERT + LSTM. Key performance metrics, including validation loss, validation accuracy, test loss, test accuracy, and F1-score, are reported to evaluate the effectiveness of each model.

FinBERT achieves the highest test accuracy (0.91) and a strong F1-score (0.88), showcasing its superior ability to generalize to unseen data, though with a slightly higher validation and test loss compared to other models.

LSTM with BERT offers a balance between validation loss (0.3644) and test accuracy (0.8912), closely trailing FinBERT in performance.

FinBERT + LSTM exhibits the lowest validation loss (0.1223) and competitive test accuracy (0.8812). While its test loss (0.3672) and F1-score (0.87) are slightly lower, the integration of FinBERT and LSTM demonstrates strong potential for capturing sequential dependencies.

Overall, the results indicate that FinBERT is the most effective model for this task, achieving the highest overall accuracy and F1-score. The results underscore the effectiveness of incorporating domain-specific pre-trained models (e.g., FinBERT) and highlight the trade-offs between generalization and overfitting across the different architectures.

IV. CONCLUSION

This project demonstrated FinBERT's effectiveness for financial sentiment analysis and the potential of integrating it with LSTM for improved performance. The results highlight the value of combining transformer-based models with LSTM for capturing both semantic and sequential patterns in financial text, contributing to advancements in financial sentiment classification. Future work could further optimize this integration and explore its application to larger, real-world datasets.

REFERENCES

[1] Dogu Tan Araci, “FinBERT: Financial Sentiment Analysis with Pre-trained Language Models ”, August 2019.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, Octber 2018

[3] Matthew E. Peters, Mark Neumann, Mohit lyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, “Deep contextualized word representations”, Feburary 2018

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Kukasz Kaiser, Illia Polosukhin, “Attention is All You Need”, June 2017

[5] Shayan Halder, “FinBERT-LSTM: Deep Learning based stock price prediction News Sentiment Analysis”, November 2022

[6] Wenjun Gu, Yihao Zhong, Shizun Li, Changsong Wei, Liting Dong, Zhuoyue Wang, Chao Yan, “Predicting Stock Prices with FinBERT-LSTM: Integrating News Sentiment Analysis”, July 2024