Predicting Stock Market Movement with NLP

Amogh Sawant Zexian Li Ju Hyun Kim Chin-Yu Lin
Oregon State University
{sawantam, lizexi, kimjuhyu, linchiny}@oregonstate.edu

Abstract

Accurate prediction of stock market trends is crucial in financial decision-making. This study explores the application of distilled BERT and RoBERTa models, using the FNSPID dataset including NASDAQ stocks prices and summarized news articles generated through various text summarizing algorithms. By utilizing percentage changes in stock prices as a metric rather than raw values, the research aims to provide a nuanced understanding of price fluctuations and their impact. The integration of transformer based natural language processing models with diverse data modalities is investigated, demonstrating the effectiveness of these distilled models in predicting stock price movements. The findings focuses on the potential of distilled BERT and RoBERTa in capturing sector-specific trends and highlight the significance of enriched data features in enhancing prediction accuracy by using the price change percent rather than the raw currency values.

1. Introduction

Predicting stock prices accurately is crucial for making informed financial decisions, a task that has historically relied on a variety of artificial intelligence techniques. Among these, Long Short-Term Memory networks (LSTMs) [9] capture complex time-series patterns essential for understanding the cyclic nature of markets, while Support Vector Machines (SVMs) [5] adeptly handle nonlinear data, reflecting the multifaceted factors influencing market behavior. On the other hand, Natural Language Processing (NLP) [8] models analyze sentiment from news articles, offering insights into market moods that might affect stock prices. This study seeks to enhance these traditional approaches by integrating advanced NLP techniques with quantitative data analyses to improve predictive accuracy and market understanding.

While conventional stock market prediction models typically utilize raw currency values, this project conceived by Talyer and Ng [10] to use the percentage change in stock value as training data. This approach acknowledges that the impact of price fluctuations is relative to the stock's value, thus offering a more nuanced perspective on market dynamics. For instance, a $10 decrease in a $100 stock is more impactful than in a $1,000 stock. Using the Financial News and Stock Price Integration Dataset(FNSPID) [2], our models are trained not just on numerical data but also on the qualitative insights derived from financial news, thereby providing a comprehensive view of potential market shifts.

This paper extends existing research [10] of using stock price change percentages by utilizing pre-trained and distilled versions of BERT (also known as BERT-Tiny) [1] and RoBERTa [6] (optimized for analyzing financial news) [7]. Using these advanced NLP models, we preprocess financial news articles using text summarization techniques like TextRank to extract essential information [2]. Our study investigates two ranges of price change percentages to provide deeper insights into immediate market reactions, a less explored area in stock market prediction research [10]. By focusing on the contextual impact of price changes, this research aims to improve the predictive capabilities of models and contribute valuable insights to the field of financial forecasting.

2. Related Work

2.1. Traditional NLP integration for stock market analysis

Historically, methods like SVMs and LSTMs have been popular in predicting stock movements based on quantitative data and basic text analysis. For instance, SVMs have been favored for their ability to manage high-dimensional spaces, which is characteristic of financial data sets. In contrast, LSTMs have been extensively used for their efficacy in handling sequences, such as time series data from stock prices, by capturing temporal dependencies.

On the other hand, introducing transformer-based models like BERT and RoBERTa has shifted the landscape. These models makes use of deep bidirectional contexts, offering a significant improvement over unidirectional predecessors by effectively understanding the nuanced implications of textual information in financial reports and news

articles [1] [6]. However, despite their effectiveness, the size and computational demands limit their practical deployment, especially in resource-constrained environments typical of real-time financial analysis.

This challenge has led to the development of model distillation techniques. TinyBERT [3] can aim to compress the knowledge of a large model into a smaller, more efficient one without significant loss of accuracy (see Figure 1). This version of BERT [3] is designed with a compact architecture featuring 4 transformer layers and a hidden size of 312 dimensions, aimed at maintaining BERT's capabilities while significantly reducing the model's size and computational demands. This efficient architecture enables the deployment of state-of-the-art NLP features in resource-constrained environments, using the distilled knowledge from a larger BERT model while optimizing speed and resource usage.

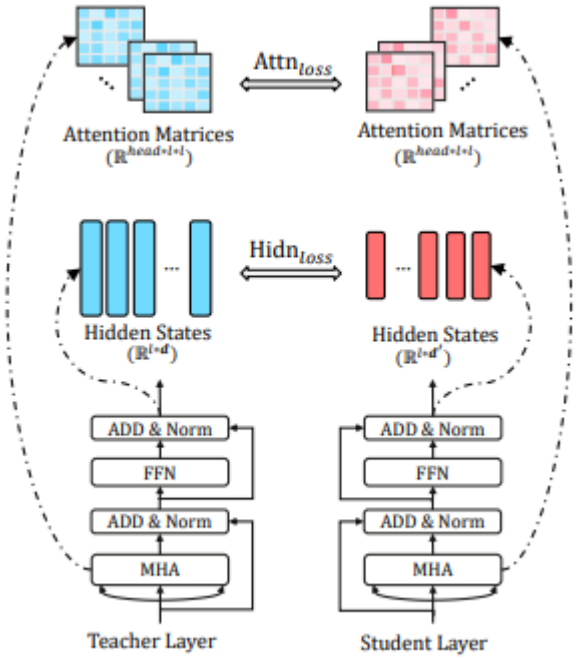

Figure 1. TinyBERT Transformer-layer [3]

On the other hand, RoBERTa is optimized for performance with fewer layers, retains core RoBERTa [6] elements while ensuring efficiency. It features fewer layers than the standard 24-layer RoBERTa, significantly reducing computational complexity. Despite its streamlined design, it undergoes extensive pre-training on large datasets and additional fine-tuning on sector-specific data like financial news. The model employs dynamic masking during training to enhance generalization across different contexts. Advanced optimization techniques such as large mini-batches and a tailored learning rate schedule are used to ensure high performance. Task-specific adaptations further refine its ability to analyze and interpret nuanced language in economic contexts, conserving computational resources while maintaining high accuracy in specialized applications.

2.2. Dataset

Before the development of comprehensive datasets, most financial datasets were either purely numerical or lacked adequate sentiment analysis integration. Previous datasets such as those provided by Bloomberg and Reuters were extensive but did not combine these elements effectively, limiting the depth of analysis possible. This gap meant that while models could predict based on historical price data, they often missed the nuances that current market sentiment could provide.

FNSPID [2] represents a substantial advancement in this area. This dataset not only includes a vast amount of stock prices and financial news records but also integrates sentiment analysis directly tied to specific stock movements. Covering a broad time span and multiple companies, FNSPID provides a robust foundation for developing predictive models that consider both numerical trends and sentiment-driven influences.

This paper leverages the FNSPID dataset to enhance the predictive accuracy of distilled BERT and RoBERTa models. By utilizing this dataset, we are able to train our models on a rich mix of quantitative data and qualitative sentiment analysis, offering a more holistic view of potential market movements. This approach addresses previous limitations by allowing for a dynamic analysis of how public sentiment reflected in financial news impacts stock prices.

2.3. Use of percent change as a metric

Most existing literature in stock market prediction predominantly relies on using raw currency prices as the primary metric for modeling and forecasting. However, the student [10] represents a departure from this conventional practice. Instead of raw currency values, the study emphasizes the utilization of percentage changes in stock prices as a more insightful metric. By adopting transformer-based models such as BERT-Tiny, the research explores how these models can effectively analyze financial news articles and predict stock price movements based on these percentage fluctuations. This innovative methodology challenges the traditional focus on raw currency prices, suggesting that incorporating percentage changes offers a more nuanced understanding of market dynamics and potentially enhances predictive accuracy in financial forecasting.

3. Methodology

3.1. Data preprocessing

Our study uses the FNSPID [2] dataset to predict stock market trends using NLP models. The FNSPID dataset initially contains 25 million data points, with approximately 1 million of these points having associated news articles. Each row in the dataset includes fields such as Date, Article_title, Stock_symbol, Url, Publisher, Author, Article, Lsa_summary, Luhn_summary, Textrank_summary, and Lexrank_summary. However, not every row contains all these fields, and a significant portion of the articles are deemed superfluous for our task.

To create a more refined dataset, we introduce a new column that indicates the percentage change between the opening and closing prices of each stock. This percentage change serves as the primary metric for predicting stock price trends based on the articles. We focus on articles from NASDAQ as provided in the FNSPID dataset, a credible source for stock-related news, to ensure the reliability of our data.

We furthur preprocess the dataset by calculating the price change percent based on the articles. Specifically, we divide the dataset into three major classes based on price change percent for the model to predict:

- $price \geq +x\%$
- $price \leq -x\%$
- $-x\% \leq price \leq +x\%$

Here x indicates the the percent that we choose. We choose to create two different datasets from these 1 million dataset points for the reasons which will be apparent later. While this filtering process we ensure that we have a approximately equal number of dataset points in each set.

For the model training, we consider two scenarios:

1. The price change percent within the same day of the article's publication, focusing on changes greater than ±5%. We have around 100k examples after using $x = 5$ as percent cap.

2. The price change percent of the next day following the article's publication, focusing on changes greater than ±6%. We have around 60k examples after using $x = 6$ as percent cap.

We choose 5 percent threshold because a 5% increase or decrease in stock price typically signifies significant news affecting the stock profile. Such substantial changes are more likely to be linked to impactful articles, making them more indicative for accurate stock price predictions. We selected a 6% threshold for next-day price changes because it better reflects the impact on stock prices that will manifest the following day. For example, an article published after market hours will influence stock prices when the market reopens the next day. This threshold helps capture these delayed reactions more accurately.

We hereby refer to the the dataset focusing on changes greater than ±5% as Dataset I and the dataset focusing on changes greater than ±6% as Dataset II. During inference, we use the ensembled output of models trained on these two scenarios. This approach helps mitigate the impact of articles that speculate on stock growth rather than providing concrete information about immediate stock performance. We opt to divided each of our dataset in the 90:10 split, meaning 90% of the dataset is used for training while the rest 10% is used for testing.

3.2. Trainer API

In our study, we utilized the HuggingFace [11] Trainer architecture to streamline and enhance the training and evaluation process of our models. The Trainer's high-level API significantly reduced the complexity of implementing various training strategies, such as gradient accumulation and mixed precision training. By using this architecture, we efficiently managed distributed training and hyperparameter optimization, ensuring our models were trained effectively on large-scale datasets.

The seamless integration with the HuggingFace Transformers library allowed us to quickly implement state-of-the-art NLP models like BERT and RoBERTa. The Trainer provided built-in functionalities for data preprocessing, model checkpointing, and logging, which were invaluable in maintaining organized and reproducible experiments. This powerful tool enabled us to focus on fine-tuning our models and exploring different configurations to achieve the best predictive performance, rather than getting bogged down by the intricacies of training implementation. We converted our dataset to a Pandas DataFrame before transforming it into the Datasets format provided by the Hugging Face library. This conversion facilitated easier manipulation and preprocessing of the data, ensuring compatibility with the Hugging Face Trainer architecture.

For training, we used the default arguments of the Hugging Face Trainer, which include essential configurations for model optimization and loss computation. Specifically, we employed a learning rate of 2e-5 and used the AdamW optimizer, which is well-suited for fine-tuning transformer models. We utilized BCEWithLogitsLoss as the loss function, which is the default loss for multi-label classification in the Hugging Face Trainer API. Additionally, we employed the default learning rate scheduler provided by the Hugging Face library, which dynamically adjusts the learning rate during training to improve model convergence and performance. We also used the BERT tokenizer to convert

all the inputs into tokens, ensuring that the text data was appropriately preprocessed for the BERT and RoBERTa models.

3.3. Training

Using the NVIDIA RTX 3060 GPU wiht 6GB of memory, which incorporates tensor cores, we resort to using the bfloat16 format to accelerate the training process [4]. For training the distilled BERT architecture, we set the batch size to 128. In contrast, for the distilled RoBERTa models, we employ a batch size of 64. To optimize memory usage and training efficiency, we implement gradient accumulation over 4 steps with an initial batch size of 16 for the RoBERTa model, thus effectively giving us batch size of 64. This configuration allows us to maintain a balance between computational resource utilization and training performance.

We train a total of 8 models, 4 with each architecture (distilled BERT and distilled RoBERTa), using various configurations to enhance predictive accuracy. These configurations include:

1. Article title with stock symbol.
2. TextRank summary.
3. TextRank summary with stock symbol.
4. Article title, TextRank summary, and Stock symbol.

We choose not to work with the entire article because, as highlighted in study [10], using the full article can lead to worse performance. This is due to the model attempting to find relationships among an increasing number of variables, which can introduce noise and reduce predictive accuracy. Instead, we focus on key components such as the article title, TextRank summary, and stock symbol to streamline the input and improve model performance.

Before detailing our training flow, we highlight several key observations. Through our experiments, we found that training distilled RoBERTa models takes approximately 3.5 times longer than training distilled BERT models, given identical parameters and hyperparameters. Despite this increased training time, distilled RoBERTa only provides a marginal performance improvement of about 2%. Consequently, we opt to use distilled RoBERTa solely for training the model that predicts the price change percent on the next day following the article's publication, focusing on changes greater than ±6%. This decision is motivated by two key reasons:

1. The training dataset with a ±6% threshold is comparatively smaller, making the longer training time more manageable.
2. The robust nature of RoBERTa is more beneficial in capturing the significant price changes indicated by the higher percent threshold, thereby maximizing the model's performance gains where it is most impactful.

For our experiments, each configuration of the distilled BERT model was trained for 6 epochs and required approximately 1.5 hours of training, resulting in a combined training time of around 6 hours for all configurations. In contrast, training a single configuration of the distilled RoBERTa model took about 5 hours, trained for 7 epochs, leading to a total of roughly 20 hours for all configurations using RoBERTa architecture. In total, more than 60 hours of training time was invested in tuning and optimizing hyper-parameters to identify the most impactful configurations. However, for the purposes of our study, we primarily consider the training times that yielded the most significant insights and improvements in predictive accuracy. This focused approach ensures that our findings are both practical and relevant to real-world applications.

4. Results

We define accuracy based on the model's ability to predict significant changes in stock prices. Specifically, if the model correctly forecasts a class indicating change greater than $x\%$ in the specified stock after analyzing the input sequence, we consider that a success. Conversely, if the model predicts that the stock will not exceed the $x\%$ threshold and the stock indeed remains within this limit, we also regard that as a success. This approach ensures that both correct predictions of substantial changes and correct rejections of insignificant changes are accounted for in our accuracy metric.

To establish a baseline for our models, we compare the performance of the distilled BERT and distilled RoBERTa architectures trained solely on the article title with stock symbol, as performed in this study [10]. Specifically, we use Dataset I for the BERT model and Dataset II for the RoBERTa model. During initial training, we observed that the distilled BERT model's performance on Dataset II was unsatisfactory, steering us to use the distilled RoBERTa model, which demonstrated acceptable performance on Dataset II. This adjustment allowed us to continue our analysis with a more robust model, ensuring that our results remained reliable and significant.

As evident from Table 1, using the TextRank summary of articles significantly improves prediction accuracy. Furthermore, including the stock symbol along with the article title and TextRank summary yields even better results. This improvement can be attributed to BERT's ability to better understand the context when provided with the stock symbol as an input, enhancing its performance in predicting stock price trends. These findings highlight the importance of in-

Dataset Configuration	Accuracy
Article Title and Stock Symbol	71%
Text Rank Summary	73%
Text Rank Summary and Stock Symbol	74%
Article Title, Text Rank Summary and Stock Symbol	**76%**

Table 1. Accuracy on Dataset I trained on Distilled BERT with different configurations

corporating contextual information such as Stock symbol, to improve model accuracy when using encoder architecture.

Dataset Configuration	Accuracy
Article Title and Stock Symbol	60%
Text Rank Summary	61%
Text Rank Summary and Stock Symbol	64%
Article Title, Text Rank Summary and Stock Symbol	**66%**

Table 2. Accuracy on Dataset II trained on Distilled RoBERTa with different configurations

We observed similar results when performing experiments with Dataset II using distilled RoBERTa, as shown in the Table 2. The model achieved better accuracy when incorporating the article title, TextRank summary, and stock symbol together. This consistency across different datasets and models underscores the effectiveness of combining these elements to enhance the predictive performance of our models. The inclusion of the stock symbol alongside the Article title and Text Rank summary data helps the model better understand the context, leading to more accurate predictions of stock price trends. During inference, after obtaining predictions from both models, we adopt an ensemble approach by averaging their outputs. This method combines the predictions from multiple models to mitigate individual model biases and improve overall prediction accuracy. By averaging the outputs of these models, we aim to use their complementary strengths and enhance the reliability of our predictions for stock market trends.

5. Conclusion

In this study, we explored the effectiveness of using distilled BERT and RoBERTa models to predict stock market trends based on financial news articles. By using the FNSPID dataset, which includes comprehensive NASDAQ stock data and news article summaries, we extended upon the a novel approach of using percentage change in stock prices as a training metric, rather than raw currency values, by using the article title, stock symbol and textrank summary together.

Our methodology involved preprocessing the FNSPID dataset to calculate price change percentages and categorizing the data into three major groups based on these changes. We trained 8 different models, four for each architecture, using various configurations such as Article title, TextRank summary, TextRank summary and Stock symbol and TextRank summary with article title and stock symbol. Despite the longer training times, distilled RoBERTa models showed only a marginal 2% performance improvement over distilled BERT models. Therefore, we used distilled RoBERTa exclusively for scenarios where it was most impactful, such as predicting next-day price changes exceeding ±6%.

Baseline comparisons revealed that using the article title with stock symbols alone was insufficient for achieving high accuracy. Incorporating the TextRank summary alongside the article title improved the results significantly. Including the stock symbol further enhanced the model's performance, highlighting the importance of comprehensive contextual information.

Our findings were consistent across both Dataset I and Dataset II, demonstrating that the combined use of article titles, TextRank summaries, and stock symbols provides a robust framework for predicting stock price trends. This approach underscores the value of integrating diverse data sources and advanced NLP techniques to improve the accuracy and reliability of financial forecasting models.

One of the primary limitations of this study is the inherent complexity of financial news analysis. Unlike typical sentiment analysis, predicting stock market trends based on financial news is challenging due to the volatile and often unpredictable nature of the stock market. Financial news articles can vary widely in their impact, and the nuanced language used in such reports can be difficult for models to interpret accurately. This volatility makes it harder to achieve consistently reliable predictions, and further research is needed to refine these models and improve their robustness in the face of fluctuating market conditions.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 1, 2

[2] Zihan Dong, Xinyu Fan, and Zhiyuan Peng. Fnspid: A comprehensive financial news dataset in time series, 2024. 1, 2, 3

[3] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding, 2020. 2

[4] Dhiraj Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu

Huang, Hector Yuen, Jiyan Yang, Jongsoo Park, Alexander Heinecke, Evangelos Georganas, Sudarshan Srinivasan, Abhisek Kundu, Misha Smelyanskiy, Bharat Kaul, and Pradeep Dubey. A study of bfloat16 for deep learning training, 2019. 4

[5] Yuling Lin, Haixiang Guo, and Jinglu Hu. An svm-based approach for stock market trend prediction, 2013. 1

[6] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. 1, 2

[7] Hugging Face mrm8488. Distilroberta-financial-sentiment. https://huggingface.co/mrm8488/distilroberta-finetuned-financial-news-sentiment-analysis. 1

[8] Karlo Puh and Marina Bagić Babac. Predicting stock market using natural language processing, April 2023. 1

[9] Md. Arif Istiake Sunny, Mirza Mohd Shahriar Maswood, and Abdullah G. Alharbi. Deep learning-based stock price prediction using lstm and bi-directional lstm model, 2020. 1

[10] Kevin Taylor and Jerry Ng. Natural language processing and multimodal stock price prediction, 2024. 1, 2, 4

[11] Victor Sanh Julien Chaumond Clement Delangue Anthony Moi Pierric Cistac Tim Rault Rémi Louf Morgan Funtowicz Joe Davison Sam Shleifer Patrick von Platen Clara Ma Yacine Jernite Julien Plu Canwen Xu Teven Le Scao Sylvain Gugger Mariama Drame Quentin Lhoest Alexander M. Rush Thomas Wolf, Lysandre Debut. Huggingface's transformers: State-of-the-art natural language processing, 2014. 3