CS5100 - Foundations of Artificial Intelligence
Sentiment Driven Market Analysis

Kishore Sampath
Master of Science in AI
Northeastern University
sampath.ki@northeastern.edu

Satya Venkata Anudeep Ragata
Master of Science in AI
Northeastern University
ragata.s@northeastern.edu

Kumar Prakhar
Master of Science in AI
Northeastern University
prakhar.k@northeastern.edu

Deepansh Gandhi
Master of Science in AI
Northeastern University
gandhi.dee@northeastern.edu

1 Abstract

The efficient-market hypothesis in financial economics states that asset prices reflect all available information. With the advent of large language models, it is possible to leverage information sources such as news headlines and social media feeds to get new metrics like prevailing public sentiment scores. Along with historic market data, this project aims to use these new metrics to build prescriptive models which will be able to outperform existing stock investment strategies. Models such as these can be a valuable tool for making informed and proactive investment decisions.

Keywords: Sentiment Analysis, Predictive Modelling, Time Series Forecasting, Stock Market Prediction

2 Introduction

In the world of the stock market, figuring out when to buy and sell is a significant challenge. Every day, people make decisions that impact economies worldwide. Investors and analysts have been attempting to solve this problem for a while by using traditional time series modeling of historical data, but haven't found much success yet. Recently, people have tried to move beyond conventional methods by integrating machine learning with the discipline of finance.

Now, let's shift our focus to news - a game changer in understanding the stock market. We feel that news can help us capture the movement of the stock market, something which traditional methods haven't been able to. By blending machine learning and Natural Language Processing (NLP), we have tried to understand this hidden parameter of news, which can help us navigate the inherent volatility of the stock market.

We've used NLP as our language decoder of the stock market. We use it to understand the emotions hidden in news stories, turning words into measurable and insightful data. Alongside NLP, we've got powerful tools like LSTM and GRUs - they are types of recurrent neural networks, which are

essentially algorithms that process sequences of data. They help us analyze the ups and downs of the stock market by recognizing patterns in data over time.

Autoregressive models are widely used to predict stock prices. They rely on past data to predict trends in the future. By understanding the historical performance of a given stock or market index, these models aim to identify trends and potential shifts in the financial landscape. Despite their historical significance, autoregressive models inherently assume that future behavior is solely determined by past observations, often overlooking real-time influences and evolving market sentiments. We have tried to integrate them with news sentiments and make these reliable models even more accurate.

In summary, we've used NLP and sequential models to discern patterns in stock market data over time, ultimately enhancing our ability to navigate market fluctuations.

3 Related Work

[1] investigated whether the collective mood states of public (positive, negative or neutral) predicted from twitter are correlated to the Dow Jones Industrial Index. They used a Fuzzy neural network for their prediction. Their results show that public mood states in twitter are strongly correlated with Dow Jones Industrial Index. [2] investigated the use of news articles for sentiment analysis. They used Bayesian Networks, Artificial Neural Networks and Support Vector Machines to develop a dictionary based sentiment analysis model on stocks from the pharmaceutical market. [3] investigated the use of twitter chatter to predict box-office revenues. They used the DynamicLMClassifier for sentiment analysis and linear regression model on twitter chatter rates to show correlation of both sentiments and chatter rates with box-office revenues.

Research in this field has predominantly been classification based sentiment analysis. But after looking at their results, we noticed there is no quantization of the sentiments. So our project is developing a regression based sentiment scoring model which can quantize the sentiments.

4 Methodologies

4.1 Data Collection and Preprocessing

We have sourced the historical news data for the sentiment analysis model from Kaggle and the financial dataset from Yahoo Finance. The news dataset for the predictive model is scrapped using the FinnHub API. The data is for 10 companies with news headlines from 26^{th} November 2022 to 15^{th} November 2023. It contains news headline about Google, Tesla, Microsoft, Apple, Amazon, AirBnB, Coinbase, Duolingo, Meta, Moderna, Nvidia, Palantir, Wework and Zoom. These companies are from diverse backgrounds like the e-commerce, software, hardware, automotive, finance and pharma industry. We chose these companies to train a general model which can predict any stock's trends. The financial indicators for these stocks are sourced from Yahoo Finance.

For the sentiment analysis model, we have generated our own class labels. Instead of a classification model, we have trained a regression model to predict sentiment scores. The class labels is the lag difference of current index price and the next day index price. In the dataset, the magnitude of highest dip was higher than the magnitude of the highest rise. Due to this the standard normalizer functions were skewed towards the negative sentiments. To overcome this problem, we normalised the negative sentiment scores and the positive sentiment scores separately.

For the textual data, we have followed the standard practices for preprocessing.

- Stopword removal - The word 'the' is the most frequently used word. It is used 7 percent of the time but does not bring any contextual value to the sentence. Along with 'the', there are other stopwords like 'a', 'an', 'is', 'are', 'etc'. We have removed these words from our text data.
- Numerical texts to words - Financial news headlines have a lot of numerical context in them. The data is completely converted into words. Non-alphabetical characters are then removed.
- Lemmatization - The same word has a lot of variations depending on the tense. All words are converted into their root form so that the dataset has less variance.

After data collection and preprocessing, each document is then transformed into a token sequence. There are 22350 words in our dataset. The wordcounts are sparse so the tokenizer only uses the 100 most frequently occurring words.

4.2 Sentiment Analysis

We have trained a sentiment analysis model using LSTMs for predicting the sentiment scores of a news article. The model takes in the tokenized news headlines as the input and predicts the sentiment scores which ranges from -1 to 1, where values from 0 to -1 indicates a negative score while the values ranging from 0 to 1 indicates that the given news headlines are positive. A regression model is used as different news have variable impact on the stock market and we want to quantize it. The current news is used to predict the stock price of the next day. In our model, the first layer is an embedding layer which turns the tokenized headlines to dense vectors of fixed size. Then there is an LSTM layer followed by a dropout layer. Dropout is used to prevent the LSTM from learning specific patterns and reduce over fitting. After dropout there is another LSTM layer followed by a dense layer and then another dropout. In the end, there is another dense layer with linear activation to output the sentiment scores. The model was trained using the Adam optimizer with a learning rate of 0.001. The architecture is shown below:

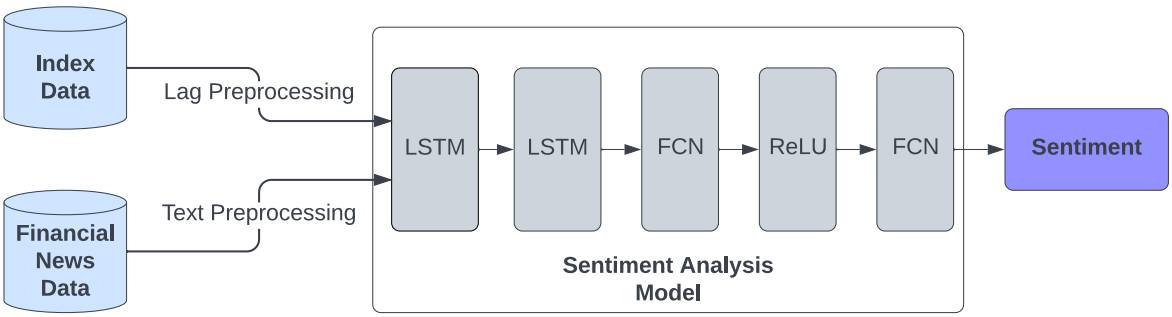

Figure 1: Sentiment Analysis Model Architecture

4.3 Predictive Model

We have trained sequential autoregressive models using both LSTMs and GRUs. The autoregressive model is trained on the historical stock market indicator data and the stock news data. The regression model uses sentiment scores from the sentiment analysis model of the news headlines and market

indicators as inputs. The sequential model has two LSTM/GRU layers followed by a dropout layer witha dropout rate of 0.2. Followed by a dense layer for stock price forecasting. The model is optimized using the Adam optimizer with a learning rate of 0.001. The model architecture is shown below.

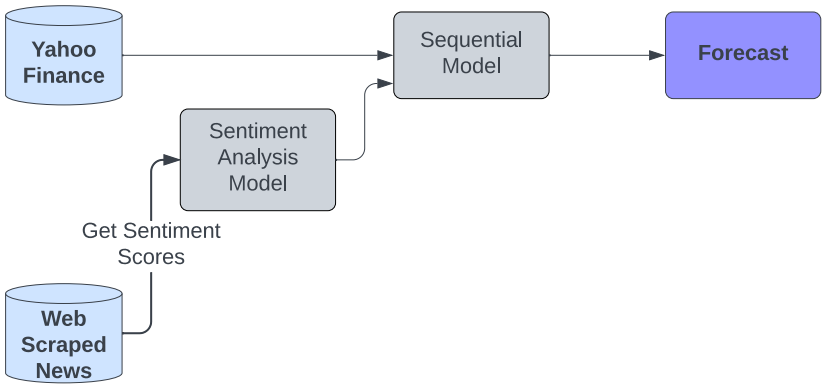

Figure 2: Predictive Model Architecture

5 Results

This section provides a look at the results we achieved in our project

5.1 Sentiment Analysis Model

Pre-trained models like FinBERT and TextBlob were used along with a custom LSTM sentiment analyser model trained on our index data to get the sentiment scores of the news data. In the case of the pre-trained models, since they are binary classifiers (positive and negative), confidence scores for each class was used as the sentiment score. To test the accuracy of each model and find out how close on average was each model able to predict the sentiment, RMSE (Root Mean Squared Error) was used. The below graph captures the RMSE scores of each of the three models.

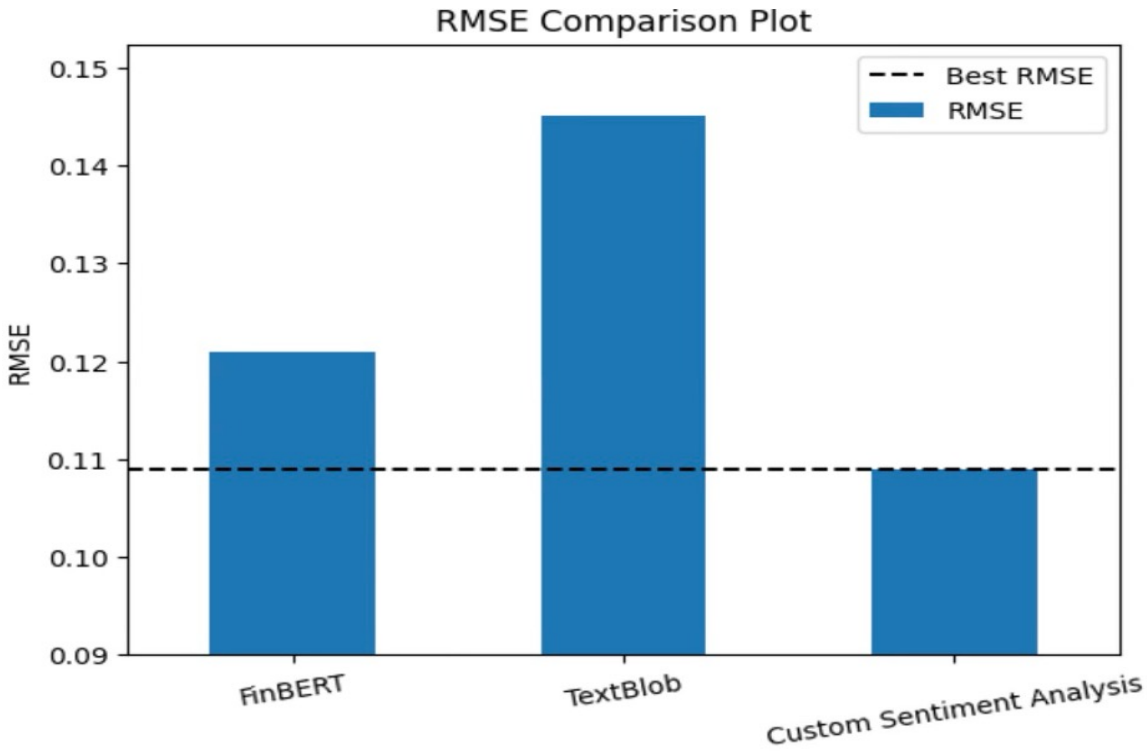

Figure 3: RMSE of Sentiment Models

The custom sentiment model performed the best and achieved the least RMSE score while the TextBlob model performed the worst. FinBERT was slightly worse than our model but performed significantly better than TextBlob. These results imply that to get accurate sentiment scores on financial news headlines, training a model on financial text data is necessary as the general TextBlob model which was not trained on financial text performed the worst.

5.2 Predictive Model

Two different predictive models with GRU and LSTM were used to compare the usefulness of sentiment scores generated by each sentiment model. A predictive model without sentiment data was also trained as a baseline. The below graphs show the predicted prices generated by the GRU model which performed the best both with and without sentiment.

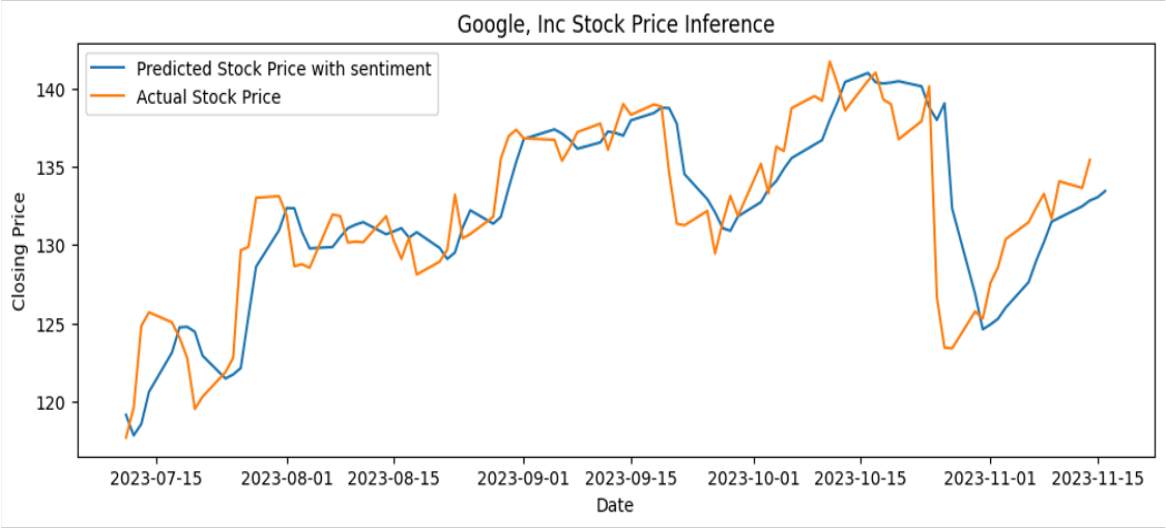

Figure 4: With Sentiment Data

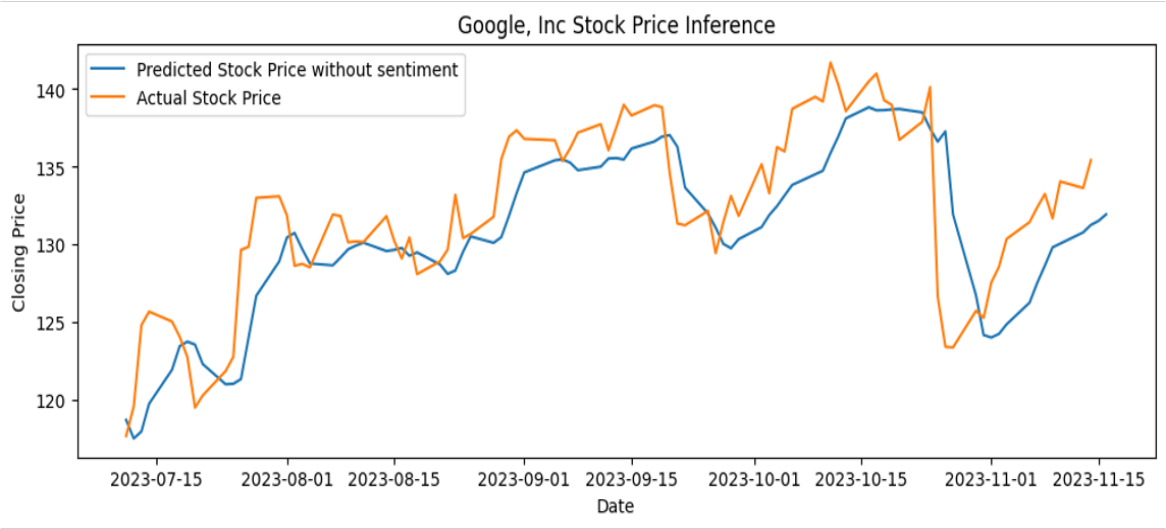

Figure 5: Without Sentiment Data

The predicted prices with sentiment scores are much more tightly fit around the actual prices than the ones without sentiment data. To look at the results in more depth, the below graph shows MSE (Mean Square Error) of each model. MSE was used to magnify the difference between each model to more accurately see each models performance.

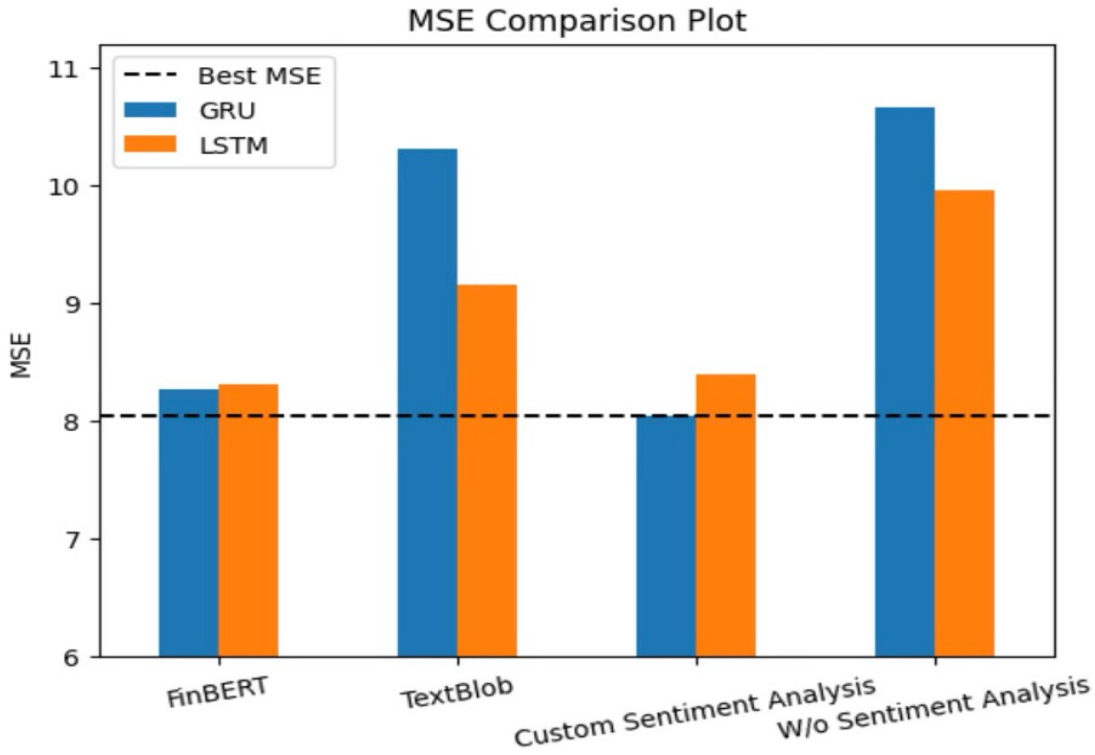

Figure 6: MSE of Predictive Models

The GRU models performed better on FinBERT sentiment scores and our custom sentiment scores while LSTM performed better without sentiment data and on TextBlob model scores. The GRU model which was trained on our custom model was the best model with the least MSE score of 8.04. The TextBlob model as expected performed the worst among sentiment models. FinBERT was comparable to the custom model in terms of performance, performing only slightly worse. Comparing these models to a model which was not trained on sentiment data, we find that all the models with sentiment data constantly outperformed the ones without sentiment data. The model without sentiment had MSE 10.66 on GRU and 9.95 on LSTM model.

6 Conclusion

In conclusion, based on our results it is clear that adding sentiment scores data to the predictive model increases model accuracy and therefore is a valid indicator of a stocks performance. To get sentiment scores, the custom trained model works best as existing pretrained models are binary classifiers and do not capture the impact news headline have on a stocks performance. One thing to note is that the performance improvement achieved by adding sentiment scores is only incremental and some other indicators along with sentiment scores can be used to increase model performance.

7 Future Scope of Work

Looking ahead, we can make several improvements in this project. Firstly, we can use better data by not just relying on news headlines but the entire news articles. This would help the model better understand the patterns. Currently, we can see some improvement over existing models, but we believe

there is some room for better performance. By performing more hyperparameter tuning, we believe that we can make the model even more accurate.

Right now, the model only predicts the stock prices for a given date. We aim to enhance this so it can give specific buy or sell recommendations. This would provide more insights for investors and analysts. Additionally, we plan to include more data features. This includes bringing in traditional stock market signs such as moving averages and Sharpe ratio. This way our model will get a broader perspective on how the market works.

8 Link to Code Repository

The project source code can be found at this GitHub repository. (Link)

9 Team Contributions

1. Kishore Sampath - Collected & Collated Historical and News Data, Data preprocessing and performed Exploratory Data Analysis.
2. Anudeep Ragata - Trained the Autoregressive Predictive Model and implemented the inference pipeline.
3. Kumar Prakhar - Trained the Sentiment Analysis Model and implemented the inference pipeline.
4. Deepansh Gandhi - Analyzed the model's predictions, generated insightful graphs to provide a comprehensive understanding of the project outcomes.

References

[1] Brian Dickinson, Wei Hu (2015), Sentiment Analysis of Investor Opinions on Twitter.

[2] Dev Shah, Haruna Isah, Farhana Zulkernine (2018) Sentiment Analysis of Investor Opinions on Twitter.

[3] Sitaram Asur, Bernardo Huberman (2010), Predicting the Future With Social Media.

[4] Bishan Yang, Claire Cardie (2014), Context-aware Learning for Sentence-level Sentiment Analysis with Posterior Regularization

[5] Matheus Gomes Sousa, Et al, (2019), BERT for Stock Market Sentiment Analysis.

[6] Thien Hai Nguyen, Kiyoaki Shirai (2015), Topic Modeling based Sentiment Analysis on Social Media for Stock Market Prediction.

[7] Aditya Bhardwaj, Et al, (2015), Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty.

[8] Hiroshi Ishijima, Et al (2015), Sentiment analysis for the Japanese stock market.

[9] Masoud Makrehchi, Et al (2013), Stock Prediction Using Event-Based Sentiment Analysis.

[10] Xiaodong Li, Et al (2014), News impact on stock price return via sentiment analysis.