Deep Learning for Stock Price Forecasting: Combining Financial News, SEC 8-K Filings, and Twitter Data

(Team 16) Yashwanth Reddy Pothireddy | Chamanthi Aki | Aravindh Gowtham Bommisetty

1. INTRODUCTION:

Stock price forecasting is crucial for investors as it provides them with valuable information to make informed decisions about buying or selling stocks. By utilizing data from various sources such as Twitter, financial news, and SEC filings, investors can gain insights into a company's performance, public opinion, and other factors that may affect its stock price. This can help individuals and organizations make more informed financial decisions, leading to greater financial stability and success. The datasets will be joined based on the date and company tickers. A sentiment analysis model will be developed to calculate sentiment scores for each company using Twitter data, financial news headlines, and SEC 8-K filings. Finally, deep learning models such as LSTM will be built and trained on sentiment scores and historical stock prices to forecast future stock prices. While there have been some studies [1][2][3] that used Twitter data or SEC 8-K filings separately, there is minimal research that has used all three sources of information. The studies that used Twitter data, financial news, or SEC 8-K filings separately have shown improvements over using only past stock data. This prompted us to use all three sources of information.

2. SIGNIFICANCE OF THE SOLUTION:

The solution we worked on is important in the application space because it addresses one of the most critical financial challenges: predicting stock prices. By combining various sources of data, including financial news, SEC 8-K filings, and Twitter data, the project aims to improve the accuracy of stock price forecasting using deep learning techniques. This approach is highly relevant in today's digital age, where a massive amount of financial data is generated every second. Traditional statistical models are often insufficient to handle this vast amount of data, and deep learning models have shown promising results in capturing complex relationships and patterns in the data. In today's age, when tweets of influential figures like Elon Musk could influence the stock movement, this solution is a step in the right direction, which considers the influence (i.e., the sentiment) of relevant tweets and news.

Overall, the project's solution has significant implications for the application space. It can potentially revolutionize the way stock prices are forecasted, providing better insights and opportunities for investors and traders.

3. DATASETS:

Our project will utilize several datasets to gather information on financial news, stock data, and tweets about top companies.

1. The first dataset from Kaggle contains over 800,000 financial news headlines with accompanying URLs, author names, publication dates, and corresponding stock tickers.
2. The second dataset, also from Kaggle, includes over 3 million tweets about top companies from 2015 to 2020, with data on tweet IDs, authors, posting dates, and text content, as well as the number of comments on each tweet.
3. We used the Yahoo Finance API to retrieve past stock data for our selected companies.
4. Lastly, we used Python libraries such as Beautiful Soup to scrape SEC filings from the SEC website for our project analysis.

4. METHODOLOGY:

4.1 EDA – Low Risk:

From Fig – 1a and Fig – 1b, it can be observed that Trade Volume is co-related with Tweets volume and News Volume, which signifies that tweets and news can help in stock forecasting.

Fig – 1a

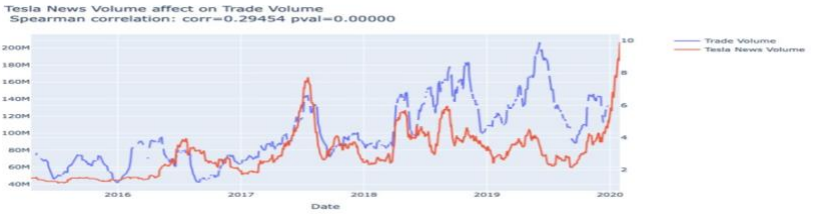

Fig – 1b

From Fig – 2a and Fig – 2b , it can be observed that Share Price is co-related with Tweets Sentiment and News Sentiment, which signifies that we can use this sentiment score in stock forecasting.

Fig – 2a

Fig – 2b

4.2 SENTIMENT ANALYSIS – Medium Risk:

To calculate the sentiment for our tweets, news, and filings data, we will primarily use LSTMs and a pre-trained model called finBERT.

- **LSTM**, or Long Short-Term Memory, is a Recurrent Neural Network (RNN) designed to address the vanishing gradient problem that arises in traditional RNNs. LSTMs utilize memory cells for retaining information over longer periods of time, as well as gates to regulate information flow and hidden units to generate output. Due to their ability to effectively handle long-term dependencies in data, LSTMs have been widely utilized in various natural language tasks, speech recognition, and other sequential data problems.
- **finBERT**, on the other hand, is a pre-trained language model specifically developed for financial and economic text data. Based on the BERT architecture, finBERT has been fine-tuned on a large corpus of financial and economic text to produce an improved performance on various NLP tasks in the financial domain, such as sentiment analysis, named entity recognition, and question answering. As a result, finBERT represents a significant advancement in applying NLP techniques to financial data, as it has been specifically trained and optimized for financial language.

The output from the sentiment analysis models will be probabilities of that text being positive, negative, or neutral. Example output for a sample text “Stocks rallied and british pound soared.”

```
inputs = tokenizer('Stocks rallied and the British pound soared.', padding=True, truncation=True,
                   return_tensors='pt').to(device)  # tokenize text to be sent to model
outputs = model(**inputs)
predictions = torch.nn.functional.softmax(outputs.logits, dim=-1)
predictions
```

✓ 1.4s

```
tensor([[0.8772, 0.0385, 0.0843]], device='mps:0', grad_fn=<SoftmaxBackward0>)
```

Here, the probabilities of the sentence being positive, negative, and neutral is 0.8772, 0.0385, and 0.0843, respectively.

4.3 MERGING DATASETS:

The project's final goal is to build a stock price forecasting model that utilizes the sentiment from Twitter Data, Financial News, and SEC 8-k filings. Before that, we need to merge all three sources of data with past stock data. FinBERT outputs three logit scores for each data point corresponding to three classes: positive, neutral, and negative. We calculate the sentiment score by subtracting the negative score from the positive score. Now, for each dataset, we have multiple sentiment scores for a single date because there are multiple tweets and multiple news for a single date. But we need a single sentiment score for a single record. We have performed several aggregations of the sentiment scores on the date column for all the datasets to achieve this. The aggregations we have used on the date column are:

1. % of positive documents
2. % of negative documents
3. % of neutral documents
4. Mean of the sentiment scores
5. Maximum of the sentiment scores
6. Minimum of the sentiment scores
7. Range of the sentiment scores

After this, we merged all the aggregation features from all the datasets with stock data on the date column to get the final data. The two pictures in the appendix give a glimpse of the final data, in which the column 'Close' will be the y-column.

4.4 BUILDING STOCK FORECASTING MODEL – High Risk:

We have built several stock price forecasting models using deep learning-based LSTM models on different datasets for comparative analysis to check if there is any positive influence on the forecasting performance because of using additional datasets. We have taken the number of epochs to be 75, batch size to be 256, and look back (number of days of past stock data to look at) to be 3, and the test size to be 20% of the complete data. We trained several models on different combinations of data (Separately on Apple and Tesla):

1. Only Stocks
2. Stocks and Tweets
3. Stocks and News
4. Stocks and SEC 8-k Filings
5. Stocks, News, Tweets, and SEC 8-k Filings

5. RESULTS and CONCLUSION:

Model performances of all the models are summarized in the two tables below, illustrating that the model trained on all the datasets outperformed the models trained only on the stock data. We can also clearly see from Fig. 1a, and Fig. 1b, the LSTM model trained on all the datasets has better-forecasted prices compared to the model trained only on the past stock data.

	Test MSE	Test RMSE	Test MAE	Test MAPE
Only Stocks	1.69	1.3	0.69	4%
Stocks, Tweets	0.77	0.87	0.55	3.50%
Stocks, News	0.89	0.94	0.53	3%
Stocks, SEC Filings	1.44	1.2	0.65	4%
Stocks, Tweets, News, SEC Filings	0.59	0.76	0.49	2.70%

Table 1a. LSTM Model Performance on two years Tesla Data.

	Test MSE	Test RMSE	Test MAE	Test MAPE
Only Stocks	5.37	2.32	2.48	5%
Stocks, Tweets	4.4	2.1	1.56	3.60%
Stocks, News	3.89	1.97	1.48	3%
Stocks, SEC Filings	5	2.23	2	4%
Stocks, Tweets, News, SEC Filings	2.75	1.65	1.35	2.90%

Table 1b. LSTM Model Performance on two years Apple Data

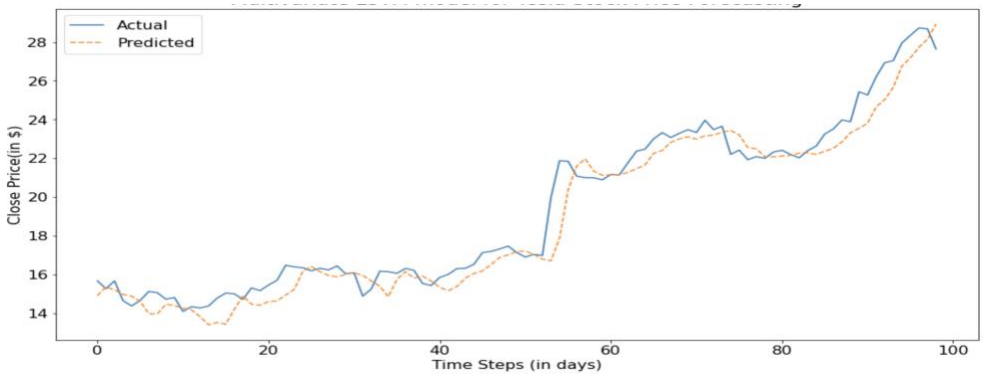

Fig. 1a Stock Forecast on only past stock data (Tesla)

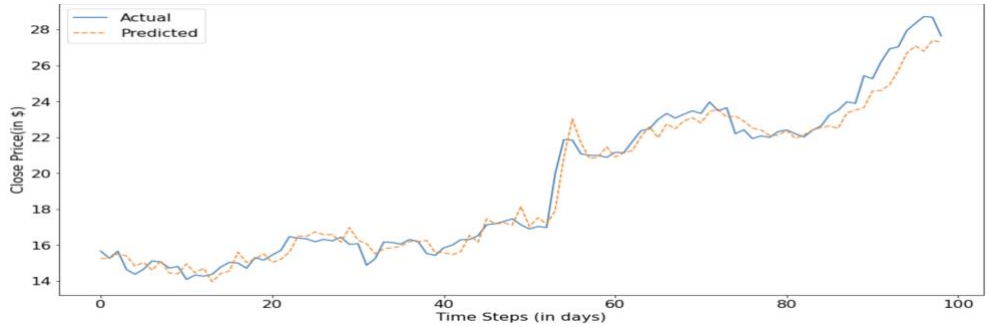

Fig. 1b Stock Forecast on past stock data, Tweets, News, SEC Filings (Tesla)

6. FUTURE WORK

We have made good progress in our project, successfully built a price forecasting model, and shown improvement over the normal model. But there are few things that can be built on the progress we have made so far. Firstly, we can build a stock movement forecasting model that predicts the up or down movement of the stock. We can also make use of 10-K and 10-Q reports. We can build interactive dashboards that different stakeholders can use to track the influence of different data sources on the stock price movement. We can experiment with different models other than LSTM for stock price forecasting.

APPENDIX:

	Date	Open	High	Low	Close	Volume	% of positive documents_x	% of negative documents_x	% of neutral documents_x	mean sentiment score_x	max sentiment score_x	min sentiment score_x	range of sentiment score_x	% of positive documents_y
0	2018-01-02	20.799999	21.474001	20.733334	21.368668	65283000	17.573222	10.460251	71.966527	0.090429	0.935310	-0.957432	1.892742	0.0
1	2018-01-03	21.400000	21.683332	21.036667	21.150000	67822500	14.423077	23.557692	62.019231	-0.074146	0.934394	-0.963219	1.897613	25.0

% of negative documents_y	% of neutral documents_y	mean sentiment score_y	max sentiment score_y	min sentiment score_y	range of sentiment score_y
50.0	50.0	-0.495679	-0.329094	-0.662264	0.333170
25.0	50.0	0.084148	0.864063	-0.944721	1.808784

Fig 1. A sample of the final merged data

REFERENCES

1. https://www.semanticscholar.org/paper/Neural-based-event-driven-stock-rally-prediction-Saleh-Nair/8733ca7fed320bb88b628e32d4769597c53b6c6c
2. https://aclanthology.org/L14-1048/
3. https://link.springer.com/chapter/10.1007/978-3-030-96634-8_4