A Hybrid Approach for Stock Price Prediction using Sentiment and Technical Analysis

VIVEK SINHA, VEDANG KULKARNI, CALVIN SUARES, POONGUNDRAN SELVAPRABHU

Vellore Institute of Technology, Vellore, Tamil Nadu
Email: cviveksinha@gmail.com, vedangk11@gmail.com, calvinjsuares@gmail.com, poongundran.selvaprabhu@vit.ac.in

Abstract: Recent years have seen a significant amount of study into the application of machine learning techniques for stock price prediction. However, most existing work in this field focuses on the examination of historical stock price data for forecasting stock prices, ignoring the role of public sentiments and mood on the stock market. In this study, we introduce a new approach that takes into account the sentiment factor along with the historical stock price in order to increase the precision of stock price forecasting. Our model includes a deep neural network that takes both historical stock price information and sentiment analysis of news headlines as input characteristics. We evaluate the performance of our framework using a dataset of news headlines and stock prices of six different stocks across various industries. There is currently limited literature in the field of Stock Price prediction which incorporates both historical stock price data and sentimental data, hence the proposed approach represents a significant contribution to the said field.

Index terms: Machine Learning; Stock Market Prediction; Sentiment Analysis; Long Short-Term Memory (LSTM); Bidirectional Encoder Representations from Transformers (BERT); Time Series Data Prediction.

I. INTRODUCTION

Stock price prediction has always been a prevalent topic for academics from both financial and technical backgrounds since it is one of the primary fields that investors are focused on. Artificial Intelligence has gained major traction in recent years in the Fintech sector given the advances in deep learning, which has resulted in significantly higher accuracy in market predictions compared to standard machine learning models [1]. Predictive models are continuously being developed to estimate future stock prices. Recurrent neural network (RNN) enables effective prediction from a temporal sequence, transforming it into a natural and promising deep learning tool for stock price prediction that outperforms traditional machine learning approaches [2].

In terms of parameters, stock market data is complex. There are multiple variables to consider, including opening and closing prices, product sales, political climate, and so on. Financial analysis can be classified into two categories based on many parameters: technical analysis, and sentiment analysis. Technical analysis is based on the movement of stock prices. The market is thought to move in trends, with price changes following predictable patterns. Sentiment analysis is a method of predicting stock prices that takes into account the impact of human emotions on stock prices. This entails analyzing news headlines, tweets, and so on, and then utilizing Natural Language Processing (NLP) techniques to gauge attitudes about said materials and see how they affect the stock market.

In this study, we propose a hybrid deep learning framework for predicting stock prices, which takes into consideration public sentiment along with the stock price. We conduct sentiment analysis on a large dataset of news headlines to obtain sentiment scores and incorporate that with traditional time series stock market data indicators, i.e., Open, Close, High, Low, and Volume to create a dataset fit for a combined system of technical and sentiment analysis for stock market forecasting in order to analyze the impact of sentiment factors when it comes to stock market trends and movements.

II. RELATED WORK

A significant amount of work has been done on stock price prediction over the recent years, with many different methods and models being implemented for the same. In this section not only existing work on stock price prediction is discussed but also the methods of sentiment analysis.

G. Ding et al. [3] offer an LSTM network model for predicting the current open price as well as the current lowest and highest prices. It has three gates: an input, an output, and a forget gate. These gates allow the LSTM to more selectively process data. The forget gate makes sure that pertinent information is passed through, which is helpful in a setting like the Indian stock market where trends change quickly. Dropouts are used to accommodate overfitting, which generally fits the erratic behavior of some stocks.

Cho et al. [4] introduced the Gated Recurrent Unit (GRU), a novel neural network based on a dual recurrent neural network (RNN) mechanism consisting of an encoder and a decoder. Similar to LSTMs, this deep learning model was shown to be able to overcome the gradient vanishing problem when learning long-term dependencies. Hence it has also been used in a number of time series data forecasting applications, such as Stock Price prediction and Bitcoin Price Prediction [5].

Tej Bahadur Shahi et al. [6] did a comparative study on GRUs and LSTMs for stock price forecasting. Experiments showed better results when adding financial news sentiment to the data and that LSTMs are better suited for larger data.

Jacob Delvin et al. [7] BERT (Bidirectional Encoder Representations from Transformers) forms relationships on both sides to the left and right to gain a better understanding of the context. The self-attention mechanism in BERT makes it possible to train the bidirectional depth representation of the input text simultaneously. As it is a pre-trained model it can be easily made task specific just by adding a fine-tuning layer to increase efficiency and reduce training time.

Rui Ren et al. [8] analyzed the role of investor sentiments in stock price movements by performing sentiment analysis to construct sentiment indices and combining the said indices along with stock price data and feeding the same to a Support Vector Machine model to forecast Stock Close Prices. Their results suggested a significant improvement in predictions when considering sentiment compared to when it's not.

Jiawei et al. [9] studied the impact of different factors playing a role in affecting stock market movements apart from stock prices and developed a sentiment analyzer to calculate the impact of said factors which was identified to be the public sentiment. Their results showed that incorporating said factors improved accuracy in stock market predictions but they didn't compare their findings with the scenarios not considering their sentiment analysis data.

III. DATASETS USED

In this research, two types of datasets are required. Historical stock price data and news articles from mainstream media. For the financial market data, the dataset was downloaded from Yahoo! Finance using the yfinance API. It contains 'Date', 'Open', 'High', 'Low', 'Close', 'Volume', and 'Adj Closed' (Adjusted Closed) - six columns in total.

For the sentimental analysis, we're making use of news headlines since these tend to be more accurate and authentic in their information over social media platforms like Twitter. Hence, for this reason, we're making use of a news headlines dataset from Kaggle [10] which is a large dataset of news headlines published from US publications with the top 25 news headlines selected for each date on which the stock market was functioning. The time interval taken was from 08/08/2008 to 29/04/2014. The dates have been taken so as to correspond with the news headlines dataset's time period.

IV. CONTRIBUTION

A stock price prediction project that combines deep learning for historical stock prices with sentimental analysis for human sentiment has significant potential in financial markets. By incorporating both technical and emotional factors this framework can provide a more comprehensive understanding of market patterns and likely improve the precision of stock price predictions. The novelty of this project lies in the work done to combine both technical and sentimental factors to improve on the existing approaches to the stock price prediction problem as there is currently limited literature that involves analyzing both, hence the proposed approach represents an important contribution to this field.

V. METHODOLOGY

A) Performance Analysis

i)Performance Analysis of Sentiment Analysis Techniques

We compared two different models for determining which would work better for stock price prediction using sentiment data. We compared LSTM (Long Short-Term Memory) to BERT (Bidirectional Encoder Representations from Transformers). Recent developments in the NLP field have demonstrated how transfer learning can be used to tune pre-trained models rather than beginning from scratch to produce cutting-edge outcomes for new jobs. For numerous NLP tasks, including but not limited to text categorization, text generation, and sequence labeling, Transformers have made substantial progress in producing new state-of-the-art results. These success stories were primarily based on sizable datasets.

BERT is able to capture context and meaning more accurately. Unlike LSTM, which processes text sequentially and can sometimes struggle with long-term dependencies, BERT is able to understand the context and meaning of words by processing the entire sentence at once using attention mechanisms. Furthermore, BERT's ability to capture both the left and right contexts of a word makes it more effective in capturing complex sentence structures and understanding the relationships between words. This makes BERT more suitable for sentiment analysis tasks where understanding the nuances of language is crucial to accurately determining sentiment.

Sentiment Analysis	LSTM	BERT
Accuracy Score	0.87	0.90

Table 1: LSTM vs BERT Accuracy score

As can be seen in Table 1, BERT is shown to have a better accuracy score than LSTM, and although not by much, taking into account BERT's ability to capture context and meaning more accurately, coupled with its pre-training and attention mechanisms, makes it a better option than LSTM for sentiment analysis tasks.

ii)Performance Analysis of Technical Analysis Techniques

Similarly, for technical analysis as well we compared two different models for determining which would work better for stock price prediction using historical stock price data. We compared LSTM (Long Short-Term Memory) to GRU (Gated Recurrent Unit).

Once the stock price data is fetched from yfinance API, the close prices are taken as the primary feature for the data preprocessing. This data is then scaled down to the range between (0,1) since LSTMs and GRUs are sensitive to the scale of the data. Hence for this purpose, MinMaxScaler is used.

Then the data is split into training and testing datasets, with the stock prices ranging from 2012 till 2021 as the dataset for the training and the stock prices from 2021 till 2023 as the dataset for the testing. The training for both the models is similar and is done as follows:
The number of timesteps is taken as 100, basically the number of features in each iteration. For the first iteration, the first 100 data samples (day's stock price) are taken as the 100 features, and the 101st sample becomes the output for that sample. Then in the next iteration, the samples 2nd to 101st are taken as the 100 data samples, and the 102nd sample is taken as the output for that iteration. And so on, the training is done for as many iterations till the training dataset is completely exhausted. The same method is followed for the testing data set and at the end the outputs for the training and testing datasets are compared to gauge the forecasting of the model.

The stock prices of the stock IBM were taken for this performance analysis and the results are shown in Figures 1 and 2.

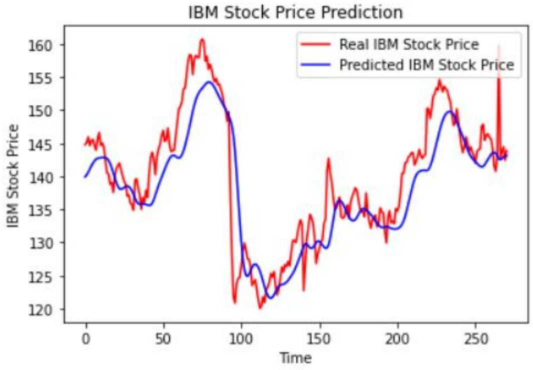

Figure 1 - IBM Stock Price Prediction using LSTM

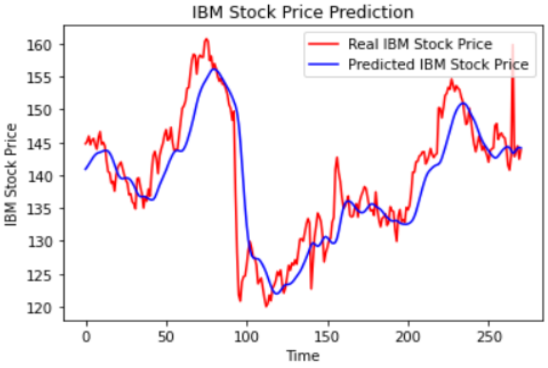

Figure 2 - IBM Stock Price Prediction using GRU

As can be seen from Figures 1 and 2, the results for both the models are similar in accuracy when comparing the predicted prices to the real prices. When looking at the root mean square error, the following were the results.

Time Series Analysis	LSTM	GRU
Root Mean Square Error (RMSE) Score	4.93	4.90

Table 2: LSTM vs GRU RMSE score

Looking at Table 2 we can see that GRUs have a lower RMSE score than LSTM, but not by a lot. And since LSTMs tend to perform better for larger datasets [6], like the kind that we'll be using for our purposes, hence we have taken LSTM as the model for the technical analysis part of our study.

B) Flowchart & System Design

From the performance analysis that has been conducted, we were able to determine the models which proved to be suitable for the requirements of the system design of our proposed approach, i.e., BERT for Sentiment Analysis and LSTM for the final Stock Price prediction.

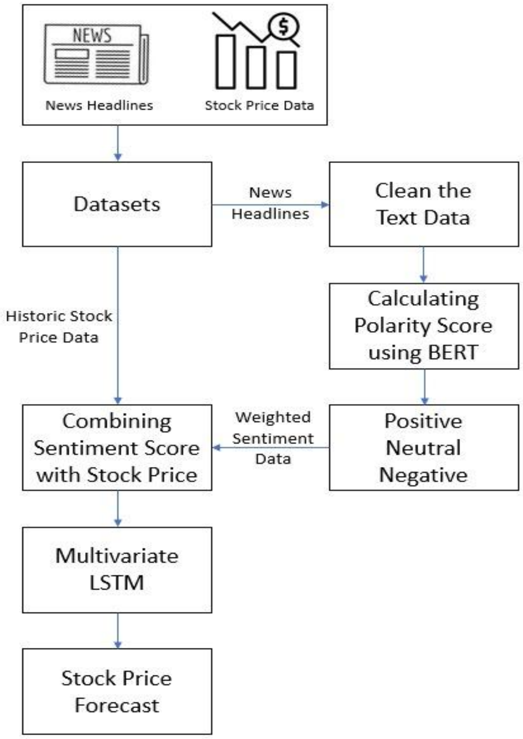

Figure 3: Flowchart of proposed system

The flowchart of our proposed approach has been shown in Figure 3, highlighting the two parts of our system. The news headlines dataset first undergoes data cleaning which consists of null values and miscellaneous characters. The text data is then fed to the BERT model where it is processed and the sentiment score of each news headline for each day in the dataset is calculated and subsequently combined with the corresponding dates in the stock price dataset consisting of various parameters such as Open, Close, High, Low, and Volume. This gives us our final dataset consisting of both historical stock price data and sentimental data. Once the combined dataset of historical stock price data and sentiment scores have been obtained, this is then fed to the multivariate LSTM model which takes as input all the various parameters present in the combined dataset as its features, i.e., Open, Close, Low, High, Volume and Sentiment Score and uses it for training the model and predicting stock prices.

C) Sentiment Analysis

As concluded earlier from the performance analysis of the sentiment analysis techniques, BERT gave better results and hence we have gone ahead with that for the purpose of sentiment analysis.
We are making use of a dataset of news headlines from news publications in the US covering the time period from 2008 August to 2014 April where for each day we have 25 news headlines. The total number of days across our time period is 1440 days and the total number of news headlines is 36,000.

The first step after taking input texts is to perform preprocessing to remove punctuations and other irrelevant information to clean the text. The next step is tokenization, where the sentence is broken down into smaller understandable parts which are easier to analyze. Once the text is tokenized, a corpus is created of positive, neutral, and negative words. The large size of the dataset that would be required for training, makes it unfeasible for us to train the BERT model using our hardware which lacks the computational power to carry out the said process. Hence, we have made use of a pre-trained variant of BERT which is available in the transformers library offered by HuggingFace, referred to as FinBERT [11] which is a pre-trained BERT model trained on a large financial corpus dataset. In the final step, another layer is added to finetune this pre-trained classification model, optimizing it to assign the ideal sentiment scores catered to our test data.

First sentiment analysis is performed on each news headline to get an individual Sentiment Score, which is divided into positive, neutral, and negative and represented by 1, 0, and -1 respectively. Then in order to get the Weighted Sentiment Score of a given day, the mean of all the individual Sentiment Scores for that given day is taken since we have 25 news headlines for every given day. The values of the Weighted Sentiment Score will range between -1 and 1. This process is repeated throughout all the days in the dataset giving us the Weighted Sentiment Score for every day.

D) Multivariate LSTM

For the final phase of our system, the combined dataset consisting of historical stock price data and sentiment score is fed to a multivariate LSTM model. Once the data has been loaded it is then preprocessed and scaled. This involves selecting the features and scaling the data to a standard value range. We select the features for our neural network based on the ones already present in our dataset, i.e., Open, Close, High, Low, Volume, and Sentiment Scores. The data under these parameters are further scaled to a range between 0 and 1 to increase the training time of the model and improve model performance and for this, we use the MinMaxScaler.

As discussed earlier in the performance analysis, the timestep for the data is taken as 100 days, hence our input sequences fed to the LSTM will each be of 100 steps and 6 features. The dataset is then split into training and testing datasets where the training dataset is 80% and the testing is 20%. Since the data has been prepared, we now train the model.

The architecture of our LSTM model is constructed as follows:
-An LSTM layer, which is the input layer that takes the input sequences as input and returns the whole sequence.
-Another LSTM layer that takes the sequence from the preceding layer as input and returns the output for the selected features.
-A dense layer
-A final dense layer for the final output, i.e., the predicted value.
The optimizer selected for this model is Adam and the number of epochs has been taken as 100 for the training process.

VI. RESULTS

A) Performance Evaluation Metrics

In order to compare our model's performance for predicting stock prices with and without sentiment score we have used the following performance evaluation metrics

i) Root Mean Square Error (RMSE)

Root mean square error is one of the most commonly used metrics for evaluating the quality of predictions. It shows how far predictions fall off from the true measured values using Euclidean distance.

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}|y_i - x_i|^2}$$

ii) Mean Absolute Percentage Error (MAPE)

It is a commonly used metric for evaluating the accuracy of predictions. It shows the average absolute deviation of all the actual values from the predicted values as a percentage of the actual values.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| * 100$$

iii)Median Absolute Percentage Error (MDAPE)

Median Absolute Percentage Error is an error metric used to measure the performance of machine learning models. It shows the middle value of all absolute percentage errors calculated between the predictions and their corresponding true values.

$$MDAPE = median\left(\left|\frac{A_t - F_t}{A_t}\right|\right) * 100$$

B) Performance Analysis Results

To evaluate the performance of our model we applied stock price data along with our sentimental data of six different stocks across various industries in the market in order to conduct a thorough analysis of our model. The model was tested with and without the sentiment score of six different stocks in industries such as tech (Apple), retail (Walmart), pharmaceutical (Johnson & Johnson), manufacturing (Procter & Gamble), and e-commerce (Amazon). The results of the same have been shown in Tables 3,4,5.

RMSE	Apple	Walmart	Johnson & Johnson	Procter & Gamble	Amazon
With Sentiment Score	0.40	0.63	1.02	0.82	0.37
Without Sentiment Score	0.47	0.85	1.20	0.89	0.39

Table 3: Comparison of RMSE values

MAPE(%)	Apple	Walmart	Johnson & Johnson	Procter & Gamble	Amazon
With Sentiment Score	1.77	0.74	0.89	0.75	1.6
Without Sentiment Score	2.20	0.95	1.14	0.83	1.72

Table 4: Comparison of MAPE values

MDAPE (%)	Apple	Walmart	Johnson & Johnson	Procter & Gamble	Amazon
With Sentiment Score	1.51	0.61	0.69 %	0.56	1.20
Without Sentiment Score	1.93	0.83	0.99 %	0.66	1.22

Table 5: Comparison of MDAPE values

From observing the results of the performance analysis of our system it was found that when the sentiment score is taken into account the model outperforms the cases for when the sentiment score isn't considered. The graphs of the predicted close prices when using Sentiment Score for all the stocks used for performance evaluation have been shown in Figures 2,3,4,5 and 6. The plot in orange is of y_test which denotes the actual prices while the plot in dark blue is of y_pred which denotes the predicted prices.

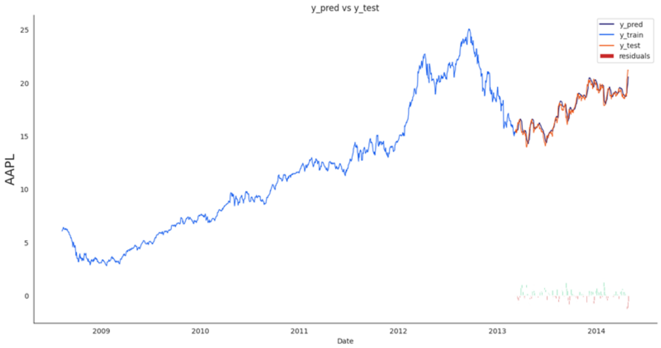

Figure 2: Apple stock price predictions with Sentiment Score

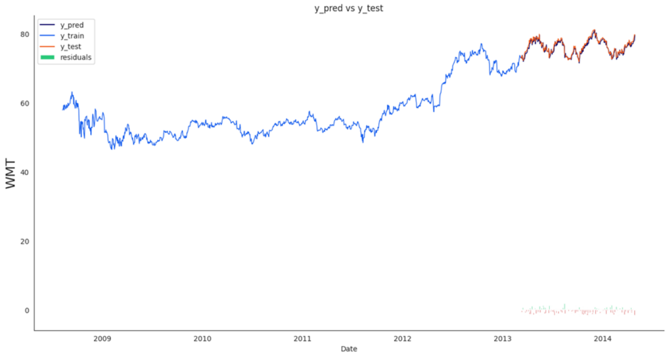

Figure 3: Walmart stock price predictions with Sentiment Score

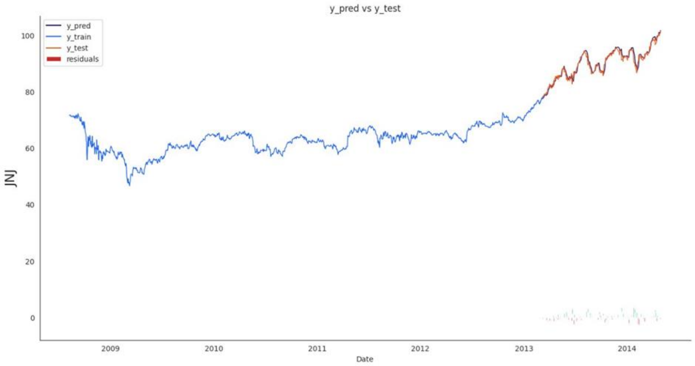

Figure 4: Johnson & Johnson stock price predictions with Sentiment Score

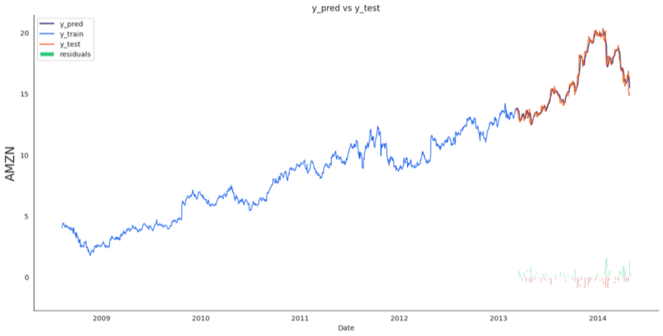

Figure 5: Amazon stock price predictions with Sentiment Score

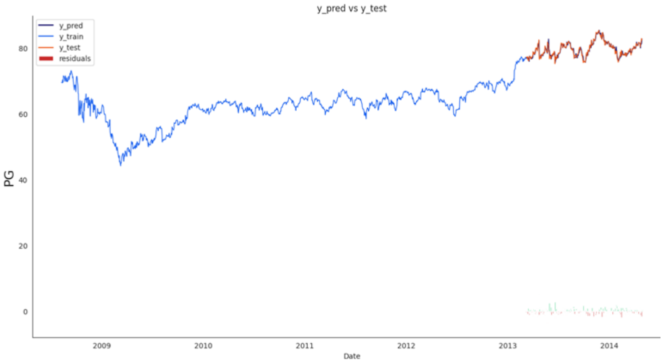

Figure 6: Procter & Gamble stock price predictions with Sentiment Score

VII. CONCLUSION

We have proposed a framework that incorporates the market sentiment with historic time series stock data by applying a hybrid model consisting of BERT and multivariate LSTM to make stock price predictions. The results of our experiments have illustrated significant improvement in stock price predictions when taking sentiment scores into account along with traditional stock price data like Open, Close, High, Low, and Volume. The findings of our study have shown that public sentiments play a major role in the Stock Market and can be a useful tool for investors when assessing their investment strategies. Future work can involve additional work on the sentimental analysis by conducting a more industry and company-specific data collection depending on the stock which is to be used for predicting prices. This could result in more relevant and focused information in regards to the public sentiment towards a certain company, stock, or sector leading to better results in stock price predictions.

VIII. REFERENCES

[1] A. Site, D. Birant, and Z. Isik, “Stock market forecasting using machine learning models,” in 2019 Innovations in Intelligent Systems and Applications Conference (ASYU), pp. 1–6, 2019.

[2] Samarawickrama, A.; Fernando, T. A recurrent neural network approach in predicting daily stock prices an application to the Sri Lankan stock market. In Proceedings of the 2017 IEEE International Conference on Industrial and Information Systems (ICIIS), Peradeniya, Sri Lanka, 15–16 December 2017; pp. 1–6.

[3] G. Ding and L. Qin, “Study on the prediction of stock price based on the associated network model of LSTM,” International Journal of Machine Learning and Cybernetics, vol. 11, no. 6, pp. 1307–1317, 2020

[4] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation.” In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

[5] A. Dutta, S. Kumar, and M. Basu, “A Gated Recurrent Unit Approach to Bitcoin Price Prediction,” Journal of Risk and Financial Management, vol. 13, no. 2, p. 23, Feb. 2020, doi: 10.3390/jrfm13020023.

[6] Shahi, Tej Bahadur, Ashish Shrestha, Arjun Neupane, and William Guo. 2020. "Stock Price Forecasting with Deep Learning: A Comparative Study" Mathematics 8, no. 9: 1441. https://doi.org/10.3390/math8091441

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

[8] Ren, Rui, Desheng Dash Wu, and Tianxiang Liu. "Forecasting stock market movement direction using sentiment analysis and support vector machine." IEEE Systems Journal 13.1 (2018): 760-770.

[9] Jiawei, Xu, and Tomohiro Murata. "Stock market trend prediction with sentiment analysis based on LSTM neural network." In International multiconference of engineers and computer scientists, pp. 475-9. 2019.

[10] Sun, J.: Daily news for stock market prediction, version 1, August 2016 retrieved from https://www.kaggle.com/aaron7sun/stocknews

[11] Huang, Allen and Wang, Hui and Yang, Yi, “FinBERT - A Large Language Model for Extracting Information from Financial Text” (July 28, 2020). Contemporary Accounting Research, Advance Access published 29 September, 2022 10.1111/1911-3846.12832.