LSTM Based Stock Price Predictor

[1]CALVIN SUARES (19BEC0433), [1]VIVEK SINHA (19BEC0482), [1]VEDANG KULKARNI (19BEC0582)

[1]Vellore Institute of Technology, Vellore, Tamil Nadu1
Email: [1] calvinjerome.suares2019@vitstudent.ac.in, [1]vivekchandra.sinha2019@vitstudent.ac.in, [1]vedang.kulkarni2019@vitstudent.ac.in,
Contact: [1]+91-9880719963, [1]+91-9167046425, [1]+91-9167046425

Abstract:
The price of the stocks is an important indicator for a company and many factors can affect their values. Different events may affect public sentiments and emotions differently, which may have an effect on the trend of stock market prices. Because of dependency on various factors, the stock prices are not static, but are instead dynamic, highly noisy and nonlinear time series data. Due to its great learning capability for solving the nonlinear time series prediction problems, machine learning has been applied to this research area. Learning-based methods for stock price prediction are very popular and a lot of enhanced strategies have been used to improve the performance of the learning-based predictors. However, performing successful stock market prediction is still a challenge. News articles and social media data are also very useful and important in financial prediction, but currently no good method exists that can take these social media into consideration to provide better analysis of the financial market. This paper aims to successfully predict stock price through analyzing the relationship between the stock price and the news sentiments. A novel enhanced learning-based method for stock price prediction is proposed that considers the effect of news sentiments. Compared with existing learning-based methods, the effectiveness of this new enhanced learning-based method is demonstrated by using the real stock price data set with an improvement of performance in terms of reducing the Mean Square Error (MSE). The research work and findings of this paper not only demonstrate the merits of the proposed method, but also points out the correct direction for future work in this area.

Index terms: Machine learning; stock market prediction; sentiment analysis; enhanced learning-based method; time series data prediction.

I. INTRODUCTION

Stock market is one of the major fields that investors are dedicated to, thus stock market price trend prediction is always a hot topic for researchers from both financial and technical domains. AI has gained significant momentum in adoption in the area of Fintech thanks to the advancement of deep learning, which allows simple end-to-end model training with much higher accuracy than traditional machine learning models. Predictive models are being actively built to forecast the stock price in the future. Recurrent neural network (RNN) supports effective prediction from a temporal sequence and becomes a natural and promising deep learning tool for stock price prediction that can beat conventional machine learning approaches.

The fluctuation of stock market is chaotic and there are many complicated financial indicators. However, the advancement in technology, provides an opportunity to gain steady fortune from stock market and also can help experts to find out the most informative indicators to make better prediction. The prediction of the market value is of paramount importance to help in maximizing the profit of stock option purchase while keeping the risk low.

After examining the historical price curves of various stocks, we find that such similarity is prevalent and thus valuable in boosting the prediction accuracy.

Financial data is complex in term of its parameters. There are a lot of parameters to be considered, such as opening and closing prices, product sales, political issues and so on. Based on different parameters, the financial analysis can be divided into two types of methods - fundamental analysis and technical analysis. Fundamental analysis analyzes the stock price based on the physical nature of the company by considering product sales, infrastructure of the company, etc. Technical analysis is based on the stock price movements. It is assumed that the market moves in trends and the price movements follow certain pattern.

Using deep neural methods, we can foresee the future value of stock price with exceptionally nonlinear demonstrating data. A neural network endeavors to map, and highlight the information that is required to be familiar with a function and thus, achieve a better forecasted output. It is comprised of a network of neurons with a weighted sum of inputs. Activation functions are used to fit the output from neurons which acquaint non-linearity to the network, and afterward this nonlinearity feature is passed to some different neurons. Neural network's streamlining is typically conducted through back propagation using the gradient descent. Errors are propagated from the output layer to the input layer through this back propagation system. The performance of Deep Learning Method is superior than any other models to forecast nonlinear data in different time series prediction problems.

II. RELATED WORK

Stock price forecasting attracts researchers for a long time due to its significant financial benefits. Among different types of techniques applied thus far, most ordinarily utilized system is Artificial Neural Network (ANN) proposed by Verma et al. ANNs are mainly affected by over-fitting problem. Additionally, Support Vector Machines (SVMs) can be utilized as an option to avoid such an over fitting issue. Usmani et al. foresee the trend of Karachi Stock Exchange (KSE) by proposing the primary target of this examination on day closing utilizing diverse machine learning algorithms. They utilized the old statistical models including ARIMA, and SMA to predict stock prices. Furthermore, other machine learning models such as SLP (Single Layer Perceptron), MLP (MultiLayer Perceptron), RBF (Radial Basis Function), and SVM (Support Vector Machine) are also used. The MLP algorithm performed best when contrasted with different methods. LSTM model was also used in different time series forecasting applications. Shao et al. introduced a framework that can forecast available parking spaces in multi-steps ahead using LSTM model. Seong et al. used encoder decoder LSTM model that utilized current vehicle trajectory to forecast the future trajectory of surrounding vehicles. Rui et al. predicted traffic flow using LSTM and GRU neural network methods. Salman et al. built a flexible but robust statistical model to forecast weather conditions in the surrounding area of airport in Indonesia. They also explored the effect of weather on flight departure and takeoff using single and multilayers LSTM. An architecture combining LSTM and GRU together is proposed to predict the future load of the Virtual Machines (VMs) of cloud precisely in. For financial time series forecasting, Persio et al. investigated the adequacy and proficiency of introducing LSTM. Akita et al. consolidated data by the information of paper articles to display the effect of previous incidents on the opening price of stock market. Their presented formula took care of numerical and printed information to LSTM system to execute precise forecasting. Be that as it may, BI-LSTM was utilized by for energy load prediction. The performance was compared using BI-LSTM, and multilayer LSTM. The better execution was accomplished using BI-LSTM design. Moreover, BI-LSTM was likewise used to forecast traffic arrival rate in. This investigation involved the methods such as unidirectional Stacked LSTM (SLSTM), and BI-LSTM neural system. Sreelekshmy et al. applied LSTM, and CNN-sliding window methods for predicting stock price. Kai et al. showed the improvement in accuracy of LSTM model compared to other regression models through their research. Also LSTM was applied by Murtaza et al. and Bidirectional LSTM was applied by Khaled A. Althelaya for stock price prediction. However, to our knowledge, none of the works presented thus far showed the comparison between LSTM and BI-LSTM in terms of the performance improvement of stock price prediction. We further proposed a novel approach by utilizing the BI-LSTM method to achieve the best performance compared to the state-of-the-art works in stock price forecasting.

III. DATASETS USED

Nowadays, there are various ways to obtain datasets: for example, using application programing interface (API) provided by related companies, buying data from data companies, and also obtaining them from some open source communities. In this research, two types of datasets are necessary. The first one is the stock market price data. The second one is the news articles from mainstream media. For financial market value dataset, data was downloaded from Yahoo! Finance. It contains 'Date', 'Open', 'High', 'Low', 'Close', 'Volume', and 'Adj Closed' (Adjusted Closed) - six columns in total. The time interval taken was from 04/01/2016 to 12/04/2022 - seven years in total.

IV. FLOWCHART

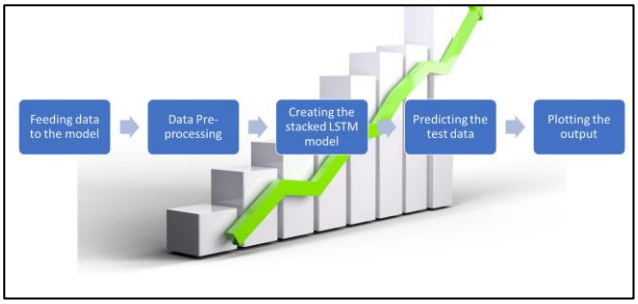

Figure 1: Flowchart – LSTM Based Stock Price Predictor

In Figure 1, we can see the Flowchart of the model, where we first retrieve the historic prices using Yahoo Finance API. We then perform data pre-processing using Min-Max Scaler. Once this is done, we stack the LSTM model one above the other on order to create an RNN-based LSTM model. The data is then divided into training and testing data with a ratio of 65:35. Once we train the model using our algorithm, we predicted the stock prices for the next 20 days. These predicted values are then plotted using 'matplotlib'.

V. METHODOLOGY

LSTMs are widely used for sequence prediction problems and have proven to be extremely effective. The reason they work so well is because LSTM is able to store past information that is important, and forget the information that is not. LSTM has three gates: 1. The input gate: The input gate adds information to the cell state 2. The forget gate: It removes the information that is no longer required by the model 3. The output gate: Output Gate at LSTM selects the information to be shown as output It can not only process single data points (such as images), but also entire sequences of data (such as speech or video). LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications.

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 are sensitive to scale of the data. Hence for this purpose, MinMaxScaler is used. Then the data is split into training and testing datasets, with 65% of the data for the training and the rest for the testing. The training is done as follows:

The number of timesteps is taken as 100 which is basically the number of features in each iteration. For the first iteration, the first 100 data samples (day's stock price) is 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 logic 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. Figure 2 explains the aforementioned logic where the number of timesteps is taken as 3 to understand the logic better.

```
Timeseries data---> Train- 120,130,125,140,134,150  Test---160,190,154

120,130,125,140,134,150                              160,190,154,160,170

Timesteps=3
    X_train   y_train                        y_train f1   f2   f3  o/p(y_test)
f1   f2   f3   o/p
120  130  125  140                                    160  190  154  160
130  125  140  134                                    190  154  160  170

f1 ----------------------------------f100
130 136                                  131   140
```

Figure 2: Algorithm for training the model

VI. RESULT AND DISCUSSION

In order to understand the results, first, the close price data of the stocks of the chosen company, here, Microsoft, are plotted which can be seen in figure 3.

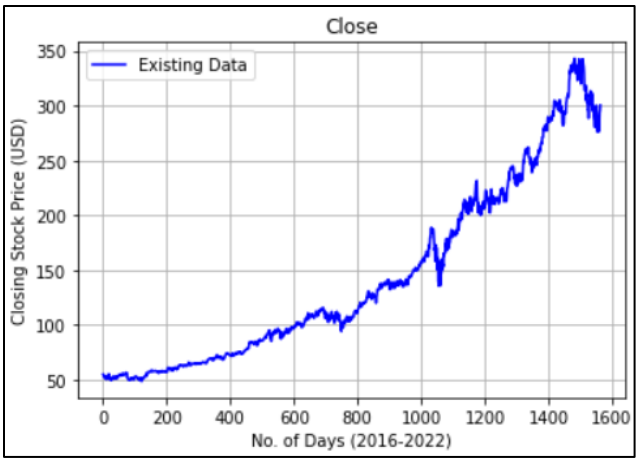

Figure 3: Existing Close Price – Microsoft

Then once the training and testing is done, we have plotted over this existing close price data in order to compare and illustrate how well the model is able to predict the stock price data, and as can be seen in figure 4, the training data is coinciding over the existing close price data but the testing dataset output is also almost coinciding very closely to the existing close price data, with hardly any error.

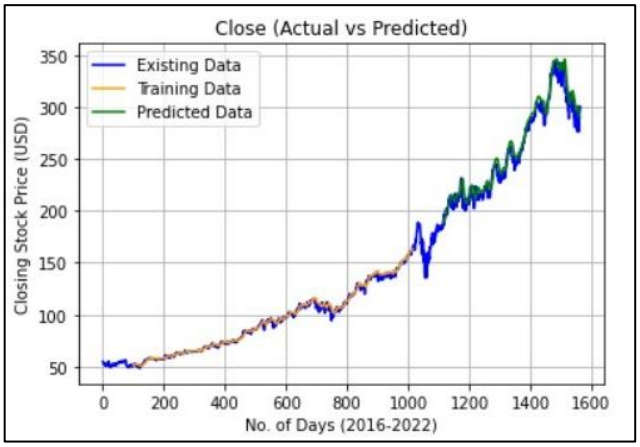

Figure 4: Output of the Model after training – Microsoft

Predicted Data	Real Time Data
292.9491316	300.429999
295.2012256	299.160004
297.4981086	304.059998
299.7327613	299.48999
301.886302	304.100006
293.9745387	303.679993
286.0206562	310.700012
308.0436064	315.410004
310.0551047	313.859985
312.0604379	308.309998
314.0606248	309.420013
316.0542075	314.970001
318.0388149	310.880005
320.0119703	299.5
321.9716364	301.369995
323.9163376	296.970001
325.8452662	285.26001
327.7580355	282.059998
329.6547687	287.619995
331.5359048	279.829987

Table 1: Success Rate Analysis for Microsoft

Here, we are checking for the hike and dip in prices on an everyday basis, where the transition for every two days acts as a trial run and this is then compared with the next day's value. This is being done in order to calculate the success rate of the model.

$$Success\ Rate = \frac{No.of\ positive\ trials}{No.of\ total\ trials}$$

The no. of positive trials is 8 and total trials is 20, hence the success rate comes out to be 0.4

The same analysis was then done for 6 other companies like Google, Apple, Tesla and others to see how well the model works for different stock price datasets. The figures 5-10 display the outputs respectively.

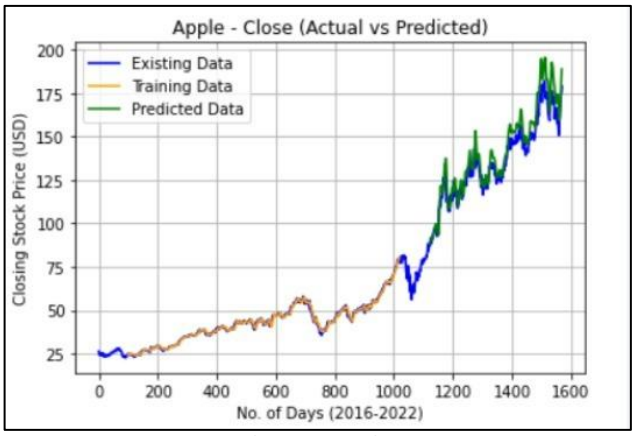

Figure 5: Output of the Model after training – Apple

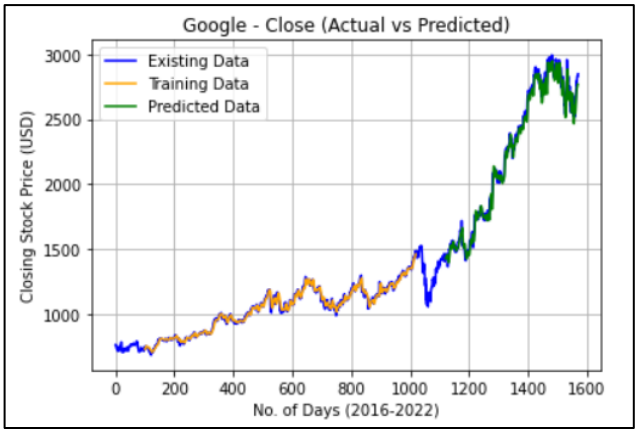

Figure 6: Output of the Model after training – Google

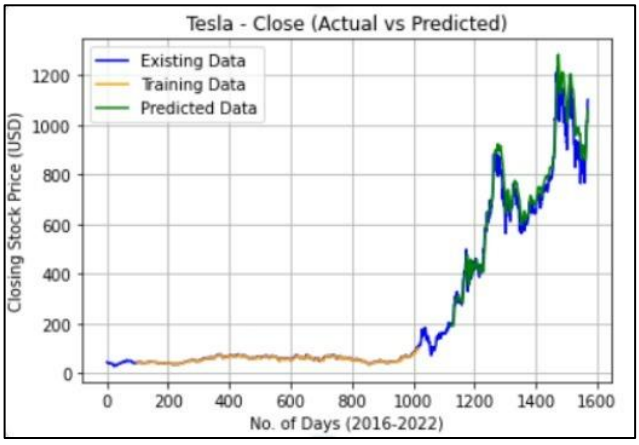

Figure 7: Output of the Model after training – Tesla

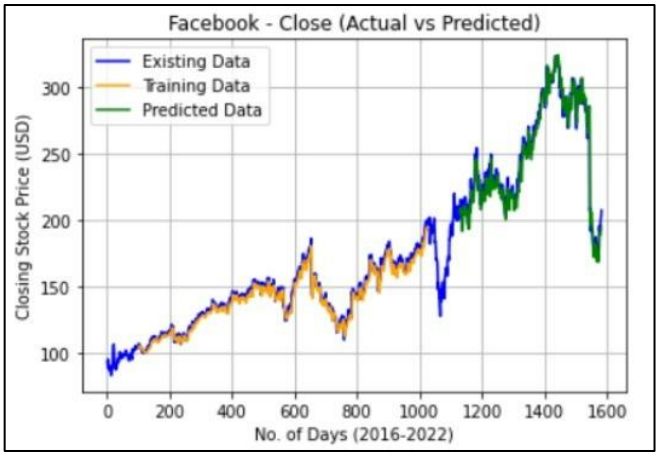

Figure 8: Output of the Model after training – Facebook

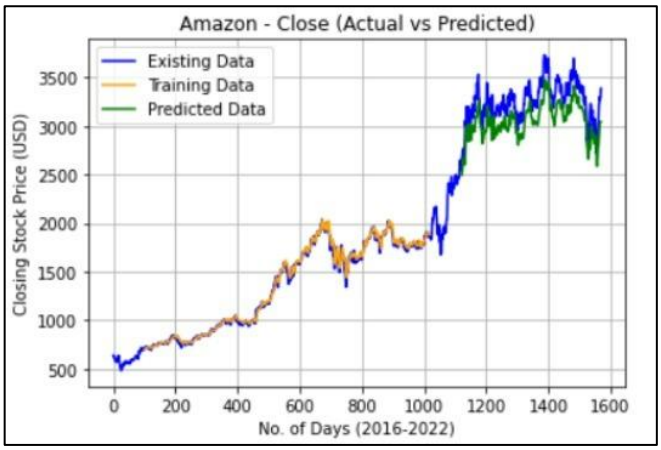

Figure 9: Output of the Model after training – Amazon

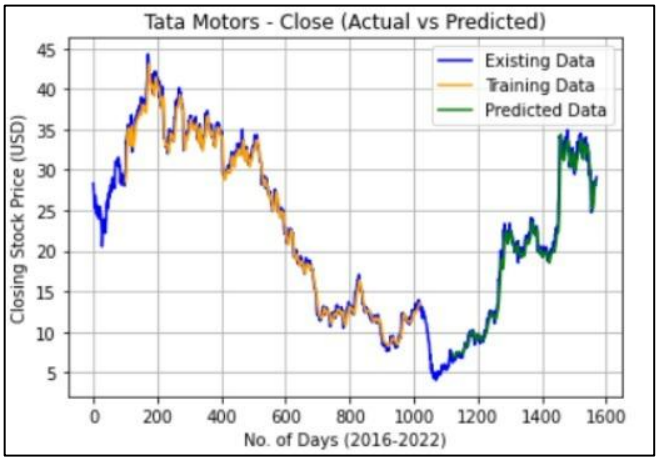

Figure 10: Output of the Model after training – Tata Motors

VII. CONCLUSION

The stock market is known for its extreme complexity and volatility, and people are always looking for an accurate and effective way to guide stock trading. This project establishes a forecasting framework to predict the prices of various stocks. We leveraged the combinations of price, volumes and corporate statistics as input data. Long short-term memory

(LSTM) neural networks are developed by recurrent neural networks (RNN) and have significant application value in many fields. In addition, LSTM avoids long-term dependence issues due to its unique storage unit structure, and it helps predict financial time series. LSTM has more predictive outcomes for price prediction than other methods such as moving averages, linear regression, K-Nearest Neighbors, ARIMA and Prophet. One direction of future work will be dealing with the volatility of stock time series. One difficulty of predicting the stock market arises from its non-stationary behavior. Our future work would include how well the LSTM model performs on denoised data.

REFERENCES

[1] W. Bao, J. Yue, and Y. Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS one, 12(7):e0180944, 2017.

[2] S. Borovkova and I. Tsiamas. An ensemble of lstm neural networks for high-frequency stock market classification. Journal of Forecasting, 2019.

[3] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. Spectral networks and locally connected networks on graphs. In ICLR, 2014.

[4] D. Chen, Y. Lin, W. Li, P. Li, J. Zhou, and X. Sun. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In AAAI, pages 3438–3445, 2020.

[5] J. Chen, T. Ma, and C. Xiao. Fastgcn: Fast learning with graph convolutional networks via importance sampling. In ICLR. OpenReview.net, 2018.

[6] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In SSST@EMNLP, pages 103–111, 2014.

[7] M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In NIPS, pages 3837–3845, 2016.

[8] Y. Dong, D. Yan, A. Almudaifer, S. Yan, Z. Jiang, and Y. Zhou. Belt: A pipeline for stock price prediction using news. In BigData. IEEE, 2020.

[9] B. Ellickson, M. Sun, D. Whang, and S. Yan. Estimating a local heston model. SSRN 3108822, 2018.

[10] M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In IJCNN, volume 2, pages 729–734. IEEE, 2005.

[11] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In SIGKDD, pages 855–864. ACM, 2016.

[12] Y. Gu, D. Yan, S. Yan, and Z. Jiang. Price forecast with high-frequency finance data: An autoregressive recurrent neural network model with technical indicators. In CIKM, 2020.

★★★