Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model

Md. Arif Istiake Sunny[1], Mirza Mohd Shahriar Maswood[1], Abdullah G. Alharbi[2]

[1]Department of ECE, Khulna University of Engineering & Technology, Khulna-9203, Bangladesh
[2]Department of Electrical Engineering, Faculty of Engineering, Jouf University, Sakaka 72388, Saudi Arabia
Email: sunny1509006@gmail.com, mmnt7@mail.umkc.edu, a.g.alharbi@ieee.org

***Abstract*—In the financial world, the forecasting of stock price gains significant attraction. For the growth of shareholders in a company's stock, stock price prediction has a great consideration to increase the interest of speculators for investing money to the company. The successful prediction of a stock's future cost could return noteworthy benefit. Different types of approaches are taken in forecasting stock trend in the previous years. In this research, a new stock price prediction framework is proposed utilizing two popular models; Recurrent Neural Network (RNN) model i.e. Long Short Term Memory (LSTM) model, and Bi-Directional Long Short Term Memory (BI-LSTM) model. From the simulation results, it can be noted that using these RNN models i.e. LSTM, and BI-LSTM with proper hyper-parameter tuning, our proposed scheme can forecast future stock trend with high accuracy. The RMSE for both LSTM and BI-LSTM model was measured by varying the number of epochs, hidden layers, dense layers, and different units used in hidden layers to find a better model that can be used to forecast future stock prices precisely. The assessments are conducted by utilizing a freely accessible dataset for stock markets having open, high, low, and closing prices.**

***Index Terms*—RNN, LSTM, BI-LSTM, Stock Market Prediction, Deep Learning**

I. INTRODUCTION

The total market capitalization of the whole stock exchanges in the world started from $2.5 trillion in 1980. At the end of 2018, it became $68.65 trillion. The value of total market capitalization of all stocks in the world reached approximately US $70.75 trillion as the record of December 31, 2019. At present, there are 60 stock exchanges existent in the world [1]. So, retail financial specialists need to invest a lot of time to discover investment openings. Wealthier speculators look for proficient budgetary for stock price prediction. For that, retail financial specialists need to make sense of the market themselves and settle on educated choices all alone. This makes ventures exceptionally unsettled in present day social orders. Motivated by the expanding ubiquity of deep learning algorithms to forecast future patterns in different time series applications, hidden structures can be discovered to forecast stock prices. This can be valuable to give additional knowledge to retail speculators when settling on venture choices.

These days, time series forecasting is a very demanding area of research due to its enormous potential in different applications such as stock price forecasting, business planning, weather forecasting, resources allocation, and numerous others. Despite the fact that forecasting can be considered as a subset of supervised regression problems, some particular tools are vital because of the worldly idea of perceptions. Some common time series forecasting models are Autoregressive (AR), Autoregressive Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA) [2] [3] [4]. Different analyses in stock markets are often discussed and considered as a unique kind of time series. Some significant patterns cannot be precisely caught through traditional methods which rely upon the linear regression methods because of the complex nature of time series models. A large portion of the time series shows nonlinearity if these models are used [5]. Additionally, it is hard to foresee or estimate the financial exchange conducted without more robust, and exceptionally nonlinear demonstrating techniques [6] [7].

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 [8]. 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.

A deep learning method known as Gated Recurrent Unit (GRU) [9] consists of same type of structure like LSTM model, except that the design of the memory cell is simplified in GRU. To reduce the cell structure of memory, GRU contains only two gates, i.e., the reset gate, and update gate. Reset gate controls the data measurements to overlook when the new data is taken while update gate maintains updating the extent of the state of memory cell. GRU is remarkable as it takes less time to train, and performs well enough on small amount of data [10]. However, LSTM is appeared as more efficient compared to

978-1-7281-8226-1/20/$31.00 ©2020 IEEE

GRU especially to handle non-linearity in larger datasets [11]. Therefore, a stock price forecasting scheme is developed using LSTM, since this type of forecasting often involves analyzing large non-linear datasets. Stock price forecasting is a classical, and significant problem. An insight of the market behavior can be gained over time through using a successful model for stock forecasting. In this research, a novel architecture is proposed through which stock forecasting is integrated with the current Deep Neural Network algorithms such as LSTM, and BI-LSTM using the public historical data to promote our analysis. The performance of both LSTM, and BI-LSTM methods were compared by varying different parameters. Finally, through a systematic study, the best performance was achieved utilizing the BI-LSTM. In this process, the suitable values of the parameters were figured out to predict the stock price that produced the lowest RMSE.

The rest of the paper is sorted as follows. Different related researches are presented in Section II. Section III briefly discusses the proposed methodology. The system architecture of our experiment is shown, and results are analyzed in section IV. Finally, the paper is concluded in Section V.

II. Related Works

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. [12]. 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 [13]. 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 [14]. 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 (Multi-Layer Perceptron), RBF (Radial Basis Function), and SVM (Support Vector Machine) are also used. The MLP algorithm performed best when contrasted with different methods [14].

LSTM model was also used in different time series forecasting applications. Shao et al. [15] introduced a framework that can forecast available parking spaces in multi-steps ahead using LSTM model. Seong et al. [16] used encoder decoder LSTM model that utilized current vehicle trajectory to forecast the future trajectory of surrounding vehicles. Rui et al. [17] predicted traffic flow using LSTM and GRU neural network methods. Salman et al. built [18] 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 multi layers LSTM. An architecture combining LSTM and GRU together is proposed to predict the future load of the Virtual Machines (VMs) of cloud precisely in [19].

For financial time series forecasting, Persio et al. [20] investigated the adequacy and proficiency of introducing LSTM. Akita et al. [21] 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 [22] for energy load prediction. The performance was compared using BI-LSTM, and multi-layer LSTM. The better execution was accomplished using BI-LSTM design. Moreover, BI-LSTM was likewise used to forecast traffic arrival rate in [23]. This investigation involved the methods such as unidirectional Stacked LSTM (SLSTM), and BI-LSTM neural system. Sreelekshmy et al. [24] applied LSTM, and CNN-sliding window methods for predicting stock price. Kai et al. [25] showed the improvement in accuracy of LSTM model compared to other regression models through their research. Also LSTM was applied by Murtaza et al. [26] and Bidirectional LSTM was applied by Khaled A. Althelaya [27] 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. Methodology

LSTM was first proposed by Hochreiter and Schmidhuber in 1997, and later became very popular especially to address time series prediction problems [29]. Being a modified RNN method, LSTM works well on a large variety of problems, and is widely used now. LSTM handles the issue of figuring out how to recollect data over a period of time, by presenting gate units and memory cells in the neural network design. The memory cells have cell states that store recently experienced data. Each moment the information reaches to a memory cell, the outcome is controlled through the combination of cell state, and then, the cell state is refreshed. Now, if any other information is received by the memory cell, the output is processed utilizing both this new information, and the refreshed cell state. LSTM is intended to maintain a problem having long term dependency. Their default conduct is to remember information for long period of time, not something they learn through struggle. In all recurrent neural network,

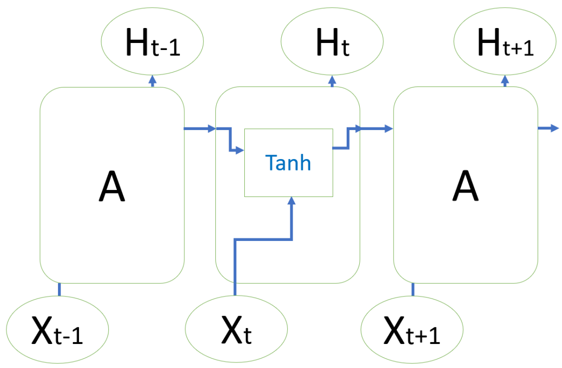

Fig. 1: The repeating module in RNN [28]

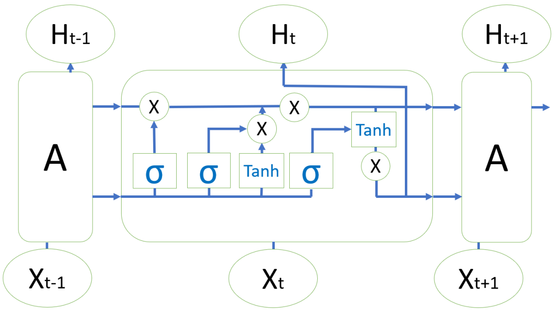

Fig. 2: The repeating module in LSTM [28]

the rehashing modules of neural system are attached like a chain. The rehashing module of standard RNNs has a basic structure that includes a single tanh layer as depicted in Fig. 1. Similar to standard RNN, LSTM contains this structure of chain as depicted in Fig. 2, however, the rehashing module follows a different structure.

The LSTM can include or exclude data to, or from the cell state. These data are deliberately managed by the structures termed as gates. Gate is an approach to control whether the data can enter into the cell state, or not. Gate is a combination of a sigmoid function, and a point-wise multiplication process. The sigmoid function can generate any number from zero to one. This value controls the passage of data in a way that an estimated zero signifies ”do not pass anything” while an estimated one signifies ”pass everything”. For LSTM model, different gates are used to pass our recently experienced data from one cell to another cell. These gates are known as update gate, forget gate, and output gate as shown in Eqn. 1, 2, and 3 [30]. These cells are used to control the memory of LSTM model. Here in LSTM, both the activation values and candidate values were used. Thus, LSTM generates two outputs from the cell, one is the activation, and another is the candidate value (Eqn. 4) [30]. The data is passed through the level line which is the highest point of Fig. 2. This level line is termed as cell state. Therefore, cell state is somewhat similar to a transport line. It runs straight down the whole chain, with just some minor linear cooperations. It can simply pass the data without any alteration.

$$\text{Update gate, } \Gamma_u = \sigma(W_u[h^{<t-1>}, x^t] + b_u) \tag{1}$$

$$\text{Forgate gate, } \Gamma_f = \sigma(W_f[h^{<t-1>}, x^t] + b_f) \tag{2}$$

$$\text{Output gate, } \Gamma_o = \sigma(W_o[h^{<t-1>}, x^t] + b_o) \tag{3}$$

$$Outputs \begin{cases} c^{<t>} = \Gamma_u * c^{N<t>} + \Gamma_f * c^{<t-1>} \\ a^{<t>} = \Gamma_o * c^{<t>} \end{cases} \tag{4}$$

BI-LSTM presented in Fig. 3 is a modified augmentation of the LSTM model. BI-LSTM improves the execution of the model for sequence classification types of problems. BI-LSTM incorporates two LSTMs in the training process of the sequence of inputs rather than using one LSTM. The notion behind Bidirectional Recurrent Neural Networks (BRNNs) can be easily understood. This overcomes the constraints of a conventional RNN. The deep learning model known as BRNN can be utilized to access all the previous information, and predicted future information simultaneously. The state neurons of regular RNN are spitted into two types, where one can act as the forward states (positive time heading), and other as the backward states (negative time heading) [31].

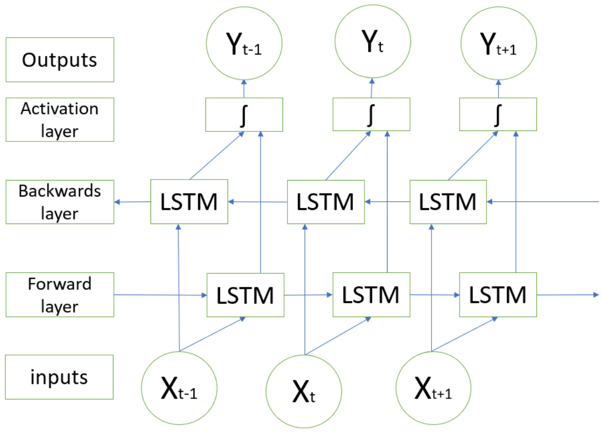

Fig. 3: Bidirectional LSTM layer [31]

IV. Simulation Study Setup and Result Analysis

A. *Datasets and system Architecture*

The raw data is collected from the yahoo finance which is publicly available. To conduct this experiment, Google stock market data for the period from 19/08/2004 to 04/10/2019 is used [32]. The format of the input data is numeric. The data consist of everyday's opening value, high value, low value, and closing value of the stock to predict future data. Total data of 4170 days are used. Google colaboratory is used with GPU, Ubuntu 18.04.3 LTS OS, and 12 GB RAM as a simulation environment in our research [33]. Tensor flow is used as our deep learning framework. The data was first preprocessed using minmax feature scaling. Then, the processed dataset is divided into two segments known as training, and testing dataset. Among the dataset, 88% of the data is utilized as training dataset, and the remaining 12% data is used as testing dataset. The training dataset was run through both LSTM, and

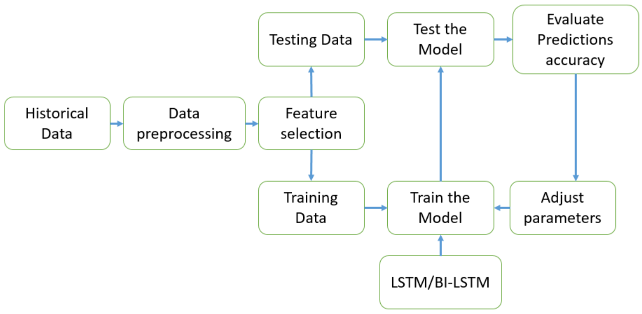

Fig. 4: The propesed architecture of stock price prediction

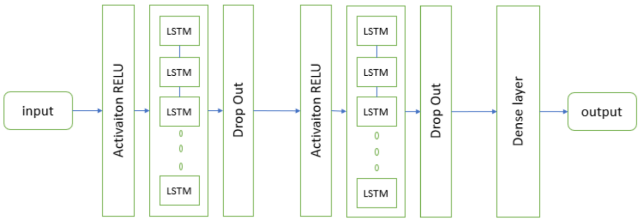

Fig. 5: Hidden layer architecture of our framework

BI-LSTM model with different tuning parameters to produce the predicted stock prices. Then, this predicted dataset was compared with the testing datasets, and the prediction accuracy was evaluated. The system architecture, and the hidden layer architecture of our scheme are shown in Fig. 4, and 5, respectively.

B. Evaluation Metrics

Lots of evaluation metrics can be used to estimate the accuracy of a prediction model. Root Mean Squared Error (RMSE) is one of them. It is a performance metric that is widely used for accuracy measurement. Here, the original value, and predicted value are denoted by y_i and $\hat{y}_i$, respectively, and n represents the total amount of data. This error is characterized as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \tag{5}$$

C. Result analysis

After proper scaling, training, and testing between the real data, and predicted data, we observed different types of RMSE for using different layers, different units in the hidden layers and dense layers, and also for different epochs for the output prediction. After analyzing different epochs for LSTM model, and BI-LSTM model, it was came to realize that the training epochs should be chosen in the best way to train the model. Here, it can be seen that 100 epochs training model has lower RMSE error which gives us the best prediction accuracy than others as presented in TABLE I. Also, it can be seen that BI-LSTM model has lower RMSE than LSTM model (TABLE I). Here, 128 units was used for each hidden layer and RELU was the activation function used by us.

TABLE I: Comparison of the RMSE between LSTM and BI-LSTM model for different epochs

NO. of Epochs	LSTM RMSE	Time (min)	BI-LSTM RMSE	Time (min)
10	0.0011000	3	0.0007167	8
20	0.0007250	6	0.0006459	15
50	0.0004933	15	0.0004219	40
100	0.0004928	30	0.0004127	70
250	0.0031980	75	0.0003568	200

From TABLE I, it can be noted that, as we continue to increase the number of epochs, at some point such as 250 epochs, the training model faced under fitting problem for LSTM. This is also depicted in Fig. 6. However, while using BI-LSTM model, it worked quite well even for 250 epochs, and generated prediction with lower RMSE. This can be seen from Fig. 7. Thereafter, from this observation, it can be stated that BI-LSTM model performs better than the LSTM model as BI-LSTM model has a backward propagation in each training time to predict the data. It can also be seen that for both the LSTM, and BI-LSTM model the efficiency of the testing accuracy reduces with the increasing amount of hidden layers. Furthermore, it increases the training time. These findings are summarized in TABLE II, and TABLE III, respectively. These can be further observed from Fig. 8, and 10 for 50 epochs using LSTM model with 2, and 4 hidden layers, respectively, and from Fig. 9, and 11 for 50 epochs by BI-LSTM model with 2, and 4 hidden layers, respectively.

TABLE II: Comparison of RMSE between different hidden layers used in the LSTM model

NO. of Epochs	LSTM RMSE(2 hidden layers)	Time (min)	LSTM RMSE(4 hidden layers)	Time (min)
20	0.0007250	6	0.0009750	12
50	0.0004933	15	0.0005832	35
100	0.0004928	30	0.0005441	70

TABLE III: Comparison of RMSE between different hidden layers used in the BI-LSTM model

NO. of Epochs	BI-LSTM RMSE(2 hidden layers)	Time (min)	BI-LSTM RMSE(4 hidden layers)	Time (min)
10	0.0007167	8	0.0010000	12
50	0.0004219	40	0.0018000	60

TABLE IV: Comparison of RMSE between different dense layers used in the BI-LSTM model

NO. of Epochs	BI-LSTM RMSE (2 hidden layer with 128 unit) and (1 dense layer 1 unit)	Time (min)	BI-LSTM RMSE(2 hidden layer with 64 unit) and (2 dense layers with 16 unit and 1 unit)	Time (min)
100	0.0004127	70	0.0002421	50

Through further investigation, it can also be noticed that for the BI-LSTM model, as increasing the amount of dense layers and decreasing the amount of neurons in the hidden layers, the efficiency of the testing accuracy increases, and the

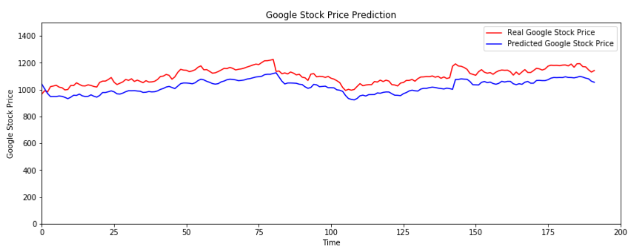

Fig. 6: Output prediction for 250 epochs with 2 hidden layers by LSTM model

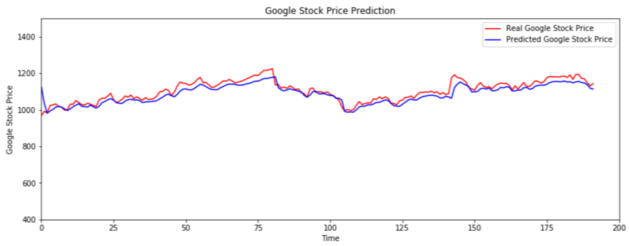

Fig. 7: Output prediction for 250 epochs with 2 hidden layers by BI-LSTM model

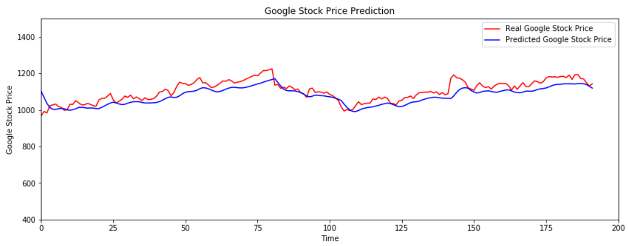

Fig. 8: Output prediction for 50 epochs with 2 hidden layers by LSTM model

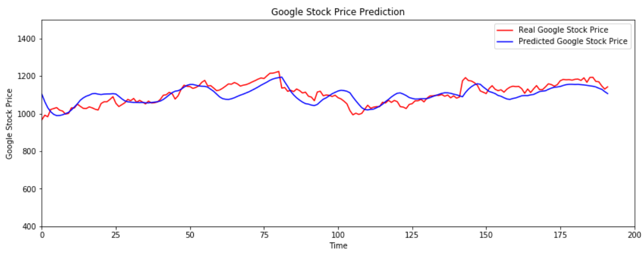

Fig. 9: Output prediction for 50 epochs with 2 hidden layers by BI-LSTM model

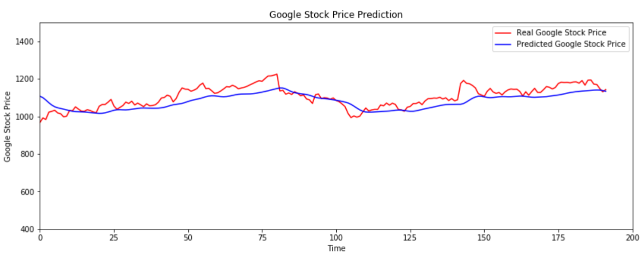

Fig. 10: Output prediction for 50 epochs with 4 hidden layers by LSTM model

training time decreases. This is shown in TABLE IV. However, to achieve a good accuracy, the amount of neuron in a hidden layer should be at least equal to the amount of input taken in per epochs for training the model. It is also known that the end result is passed through a non-linear activation function in the dense layer. Thus, choosing the amount of dense layer is an important factor for training a model. Proper tuning of the parameters in the hidden layers is also very important for designing a proper framework which is seen from TABLE IV.

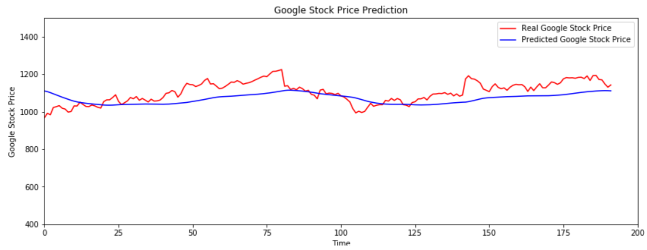

Fig. 11: Output prediction for 50 epochs with 4 hidden layers by BI-LSTM model

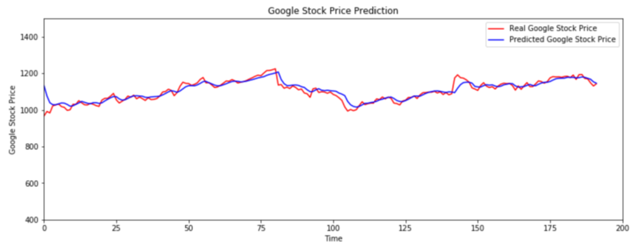

Fig. 12: Output prediction for 100 epochs with 1 dense layer by LSTM model

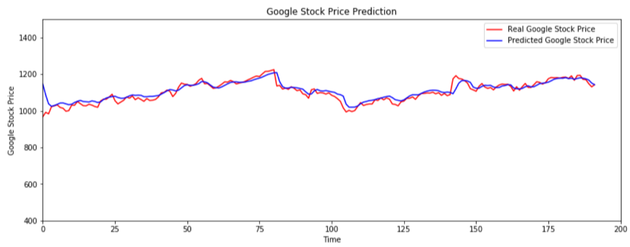

Fig. 13: Output prediction for 100 epochs with 2 dense layers by BI-LSTM modell

After performing a lot of training analyses in the LSTM model, we achieved the best result for 100 epochs with 2 hidden layers, and 1 dense layer. Here, the RMSE is much less, and accuracy is higher. This can be seen from the Fig. 12 and TABLE II. For BI-LSTM model, we also conducted a comprehensive analysis by varying the number epochs, hidden layer, dense layer, unit in the hidden layer, and other hyper-parameters. After going through this systematic study, the best results was found with 2 dense layers, and 100 epochs. This produces the lowest RMSE that can be seen from Fig. 13, and TABLE IV.

From the Fig. 12, and 13, it can be seen that BI-LSTM model has higher accuracy than LSTM model. However, LSTM model takes less amount of time to predict the data for future stock price forecasting. LSTM model takes 30 minutes to predict the result (TABLE I) where BI-LSTM model takes 50 minutes (TABLE IV). Even though, slightly more time is required for BI-LSTM, it comes up with lower RMSE compared to LSTM. Furthermore, this slight increase in prediction time does not have much impact on purchasing stocks. Therefore, BI-LSTM model with 2 hidden layers including 64 units, and 2 dense layers has the lowest RMSE, and thus can be utilized for stock price prediction.

V. Conclusion and Future Work

Through this study, it can be seen that Deep Learning algorithms have significant influence on modern technologies especially to develop different time series based prediction models. For stock price prediction, they can generate the highest level of accuracy compared to any other regression models. Among different Deep Learning models, both LSTM, and BI-LSTM can be used for stock price prediction with proper adjustment of different parameters. To develop any kind of prediction model, adjustment of these parameters is very important as the accuracy in prediction depends significantly upon these parameters. Therefore, LSTM, and BI-LSTM models also require this proper tuning of parameters. Using the same parameters between these two models, BI-LSTM model generates lower RMSE compared to LSTM model. Therefore, our proposed prediction model using BI-LSTM can be used by individuals and ventures for stock market forecasting. This can help the investors to gain much financial benefit while retaining a sustainable environment in stock market. In future, we plan to analyze the data from more stock markets of different categories to investigate the performance of our approach.

References

[1] "Market capitalization of listed domestic companies-world," accessed: 2020-06-17. [Online]. Available: https://data.worldbank.org/indicator/CM.MKT.LCAP.CD?locations=1W
[2] D. K. Kılıç and Ö. Uğur, "Multiresolution analysis of s&p500 time series," *Annals of Operations Research*, vol. 260, no. 1-2, pp. 197–216, 2018.
[3] P. Li, C. Jing, T. Liang, M. Liu, Z. Chen, and L. Guo, "Autoregressive moving average modeling in the financial sector," in *2015 2nd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*. IEEE, 2015, pp. 68–71.
[4] G. Zhang, X. Zhang, and H. Feng, "Forecasting financial time series using a methodology based on autoregressive integrated moving average and taylor expansion," *Expert Systems*, vol. 33, no. 5, pp. 501–516, 2016.
[5] M. Bildirici, Ö. Ö. Ersin *et al.*, "Nonlinearity, volatility and fractional integration in daily oil prices: Smooth transition autoregressive st-fi (ap) garch models," *Romanian Journal of Economic Forecasting*, vol. 3, pp. 108–135, 2014.
[6] I. Kaastra and M. Boyd, "Designing a neural network for forecasting financial," *Neurocomputing*, vol. 10, pp. 215–236, 1996.
[7] A. Lendasse, E. de Bodt, V. Wertz, and M. Verleysen, "Non-linear financial time series forecasting-application to the bel 20 stock market index," *European Journal of Economic and Social Systems*, vol. 14, no. 1, pp. 81–91, 2000.
[8] D. P. Mandic and J. Chambers, *Recurrent neural networks for prediction: learning algorithms, architectures and stability*. John Wiley & Sons, Inc., 2001.
[9] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[10] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.
[11] G. Weiss, Y. Goldberg, and E. Yahav, "On the practical computational power of finite precision rnns for language recognition," *arXiv preprint arXiv:1805.04908*, 2018.
[12] S. A. Verma, G. Thampi, and M. Rao, "Inter-comparison of artificial neural network algorithms for time series forecasting: Predicting indian financial markets," *International Journal of Computer Applications*, vol. 162, no. 2, 2017.
[13] M. S. Babu, N. Geethanjali, and B. Satyanarayana, "Clustering approach to stock market prediction," *International Journal of Advanced Networking and Applications*, vol. 3, no. 4, p. 1281, 2012.
[14] M. Usmani, S. H. Adil, K. Raza, and S. S. A. Ali, "Stock market prediction using machine learning techniques," in *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*. IEEE, 2016, pp. 322–327.
[15] W. Shao, Y. Zhang, B. Guo, K. Qin, J. Chan, and F. D. Salim, "Parking availability prediction with long short term memory model," in *International Conference on Green, Pervasive, and Cloud Computing*. Springer, 2018, pp. 124–137.
[16] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1672–1678.
[17] R. Fu, Z. Zhang, and L. Li, "Using lstm and gru neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, 2016, pp. 324–328.
[18] A. G. Salman, Y. Heryadi, E. Abdurahman, and W. Suparta, "Single layer & multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting," *Procedia Computer Science*, vol. 135, pp. 89–98, 2018.
[19] M. N. H. Shuvo, M. M. S. Maswood, and A. G. Alharbi, "Lsru: A novel deep learning based hybrid method to predict the workload of virtual machines in cloud data center," in *2020 IEEE Region 10 Symposium (TENSYMP), 5-7 June 2020*. IEEE, 2020, pp. 1604–1607, (in press).
[20] L. Di Persio and O. Honchar, "Artificial neural networks architectures for stock price prediction: Comparisons and applications," *International journal of circuits, systems and signal processing*, vol. 10, no. 2016, pp. 403–413, 2016.
[21] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, "Deep learning for stock prediction using numerical and textual information," in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*. IEEE, 2016, pp. 1–6.
[22] L. Di Persio and O. Honchar, "Analysis of recurrent neural networks for short-term energy load forecasting," in *AIP Conference Proceedings*, vol. 1906, no. 1. AIP Publishing LLC, 2017, p. 190006.
[23] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction," *arXiv preprint arXiv:1801.02143*, 2018.
[24] S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon, and K. Soman, "Stock price prediction using lstm, rnn and cnn-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*. IEEE, 2017, pp. 1643–1647.
[25] K. Chen, Y. Zhou, and F. Dai, "A lstm-based method for stock returns prediction: A case study of china stock market," in *2015 IEEE international conference on big data (big data)*. IEEE, 2015, pp. 2823–2824.
[26] M. Roondiwala, H. Patel, and S. Varma, "Predicting stock prices using lstm," *International Journal of Science and Research (IJSR)*, vol. 6, no. 4, pp. 1754–1756, 2017.
[27] K. A. Althelaya, E.-S. M. El-Alfy, and S. Mohammed, "Evaluation of bidirectional lstm for short-and long-term stock market prediction," in *2018 9th international conference on information and communication systems (ICICS)*. IEEE, 2018, pp. 151–156.
[28] C. Olah, "Understanding lstm," accessed: 2020-03-12. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs
[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[30] "Multilayer bidirectional lstm/gru," accessed: 2020-05-15. [Online]. Available: https://hackernoon.com/multilayer-bidirectional-lstm-gru-for-text-summarization-made-easy-tutorial-4-a63db108b44f
[31] "Deep dive into bidirectional lstm," 2019. [Online]. Available: https://www.i2tutorials.com/deep-dive-into-bidirectional-lstm/
[32] "Yahoo finance," accessed: 2020-03-12. [Online]. Available: https://finance.yahoo.com/quote/GOOG/history?p=GOOG
[33] T. Carneiro, R. V. M. Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. Reboucas Filho, "Performance analysis of google colaboratory as a tool for accelerating deep learning applications," *IEEE Access*, vol. 6, pp. 61 677–61 685, 2018.