

A Review on using Long-Short Term Memory for Prediction of Stock Price

Sanket Chaudhari

Student, Department of Computer Engineering,
Pimpri Chinchwad College of Engineering,
Pune, India

K. Rajeswari

HOD, Department of Computer Engineering,
Pimpri Chinchwad College of Engineering,
Pune, India

Sushma Vispute

Asst. Professor, Department of Computer Engineering,
Pimpri Chinchwad College of Engineering,
Pune, India

Abstract— Stock Market has received the attention of investors and computer programmers. Stock exchange price estimates are an excellent challenge because they are very complex, and have chaotic and dynamic environment. Many researches have been carried around the world to effectively predict the future stock prices. Machine learning is widely used in stock price forecasts. This problem falls under time series forecasting which can be solved by analysing time series data of the stock prices. Long-Short Term Memory (LSTM) works better and has significant effect on time series problem. This paper focuses on different LSTM models that can be used to forecast stock prices. LSTM originates from Recurrent neural Network (RNN) and can store long-term dependencies. The paper will cover the challenges and advantages of the studied models. LSTM long with sentiment analysis can be adopted to have an eye on the current sentiments among people about the chosen stock.

Keywords—Component; formatting; style; styling; insert (key words) Stock Market, investors, Machine learning, LSTM, time series, RNN, sentiments.

I. INTRODUCTION

The actual owners of an establishment sell their stocks to others to get financial strength through investment and in turn to make company grow. The buyer of these stocks owns a portion of the company. Stock prices change as a result of supply and demand. Suppose, when more people want to shop for stock, the fee is going up as there is a greater need. When more people are willing to let go stock, the fee is going down as there may be greater delivery than demand. Or understanding the offer and demand is simple, to determining the contribution of factors that cause to the increase in demand or supply is difficult. These factors will often depend on social factors such as market behaviour, price rises, trends and most importantly, what is good about the corporation in the news and what is harmful. Investing in stocks is a complex task and requires deep knowledge and study of the market to obtain high returns.

With the advancement in computer techniques like ML and AI predicting the stock prices has fascinated the programmers. Stock price prediction isn't best accurate with

linear models, it's a complex process depending on the dynamic nature of stocks as well the public sentiments. With proper in-depth study humans have managed to predict the prices. But it's not everyone's cup of tea to devote so much extensive time and energy. The question arises can we build something that can think like humans. This brings Neural Networks in picture. The Traditional NN cannot store memory which is required for the better stock price prediction. Then RNN can be used but they too suffer from Vanishing Gradient and fail to store long term dependencies [1]. Recurrent networks have a memory state that can store background information and can store details about past inputs, but the amount of period of time that it can remember this is not fixed. This time depends on its weights, context and on the provided input data. LSTM tackled the problem of long-term dependencies that RNN lacked by which RNN could not use information passed to it in early stages. As a result, RNN model could not always provide accurate results for long term, but can provide more accurate outcome based on recent data. LSTM is advance RNN. LSTM can by default retain the information for long period of time to serve a better network. With LSTM the more meaningful and context related information can be stored for long term whereas a less meaningful information can be removed. This removal and addition of information is controlled by the gates of LSTM.

Prediction of prices of stocks is a time series problem in which the data is a series of data from the sporadic behaviour of several domains such as social science, finance, engineering, physics, and economics. Finance, engineering, physics, and economics are all areas of study. Price patterns are extremely difficult to anticipate when such complexity exists. Predicting a series of time series is mostly used to create future value simulation models based on their previous values. The relationship between past and future recognition is often ambiguous, which amounts to revealing the distribution of conditional opportunities as a function of foresight. Almost every research focuses on these parameters of the stock data the Open Price, High Price, Low Price, and Close Price[2], [3] are the four basic price elements of a general stock (OHLC). With the study of these factors, numerous systems to estimate the future price of a stock [4], [5].

This paper studies different LSTM models developed and used for Stock Price prediction according to existing published papers. The papers will be analysed based on the datasets used, the model implemented, results and any future scope if applicable.

II. LITERATURE SURVEY

LSTM works better than Gated Recurrent Unit on Non-Linear Data[6] and also stock involves large amount of Non-Linear data. Istiake Sunny et al. [6] introduced a new framework for stock price prediction in which he employed two prominent RNN models, LSTM and BI-LSTM. By altering the amount of epochs, dense layers, hidden layers, and many hidden layers elements utilised to discover a forecasting fit that may be used to forecast future occurrences, both models can yield high accuracy with low RMSE on a freely available dataset from yahoo finance. The BI-LSTM model provided less RMSE criteria as compared to LSTM but required more time to compute[6].

The problem with stock market forecasting is that it is based on enormous numbers[2], a large amount of data and are largely reliant on the long-term past. As a result, LSTM [2], [6] regulates error by assisting RNNs by storing information for later use. The forecast becomes increasingly accurate as the stage progresses[2]. Thus, demonstrating that it is far more dependable than other methods. Ishita Parmar et al. [2] proposed a LSTM model where two LSTM layers were stacked on each other with the output value of 256 and dropout factor as 0.3 to increase the speed of training and avoid over fitting. The test score of 0.00875 MSE was obtained for the stacked LSTM model [2].

Sirimevan [7] has used the correlation between stock price and sentiment to improve the prediction. Twitter sentiments, web news, historical stock data were used along with and live stock prices from Yahoo Finance API. The sentiment scores from twitter and web and the google trend were input to the univariate and multivariate LSTM [7]. The models were integrated through Weighted Average Ensemble [7]. The accuracies decreased with the days. For first day the best prediction score accuracy was 0.99 , for seven days 0.92 and reduced to 0.62 till thirtieth day.

LSTMs were used by Nelson et al.[3] to forecast future stock price patterns utilising stock price and technical analysis data. The suggested LSTM model has shown greater precision than existing models of machine learning such as random forests RF, multilayers perceptron MLP and pseudo-random models. in experiments [3]. This model was created with a rolling window fashion in mind. At the end of each trading day, a fresh neural network was constructed, which means a fresh set of weights was originated using a new set of model training and justification data. The model is trained using data from the previous 10 months of trade preceding the present day, and the model's working is authenticated using data from the past week. The model was built with Google's

TensorFlow, and it comprises of a sigmoid output layer that is fed by an LSTM input layer that takes both technical pointers and stock valuing data as input [3].

Investors value historical data as a foundation for making investing decisions[4]. In the attention layer, the model selects and learns the input data via computing the weighting of the input data. The attention layer weighted the feature vector [4].

For short-term and long-term prediction of stock market, K. A. Althelaya et al. [8] developed Stacked LSTM and Bidirectional LSTM. The writers used historical data from the Standard and Poor 500 Index (S&P500) [8]. The data was pre-processed and normalised. As a baseline, shallow NN and the basic form of LSTM were utilised. For the sake of analysing the system's performance, the closing price was used. The authors presented two systems, one for short-term forecast (one day) and the other for long-term forecast (30 days), both having a 10-day window size. Both models were assessed and differentiated against various models, namely MLP-ANN and LSTM, in order to assess their performance. In assessment to the other models, the BLSTM [8] had a lower RMSE and MAE score. For both short- and long-term forecasts, the BLSTM outperformed current models and achieved greater convergence [8].

M. Nikou suggested an LSTM model and compared it to the ANN, Support Vector Regression (SVR), and the RF models. The outcomes showed that the LSTM model accomplished better than the other models in the study in predicting the closing day stock prices of iShares MSCI United Kingdom [9]. Sumeet Sarode [10] and colleagues employed LSTM with the most recent trading data and analysis indicators as inputs. News is gathered from a huge set of business news for news analysis. When the price rises, the results are combined to make a suggestion.

LSTM and its variants had been studied by Klaus Greff et al. [11] in their work. They studied 8 variants of LSTM. The gates play a major part in LSTM [11].

TABLE I. Summary of Literature Review

Title	Author	Year	Model	Dataset	Performance criteria	Result	Comments
Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model [6]	Istiake Sunny Md Arif, Maswood Mirza Mohd Shahriar, Alharbi A.	2020	LSTM, BI-LSTM	Yahoo Finance	RMSE.	BI-LSTM gave low RMSE	Hyper parameters were tuned at different values.
Stock Market Prediction Using Machine Learning [2]	Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan.	2018	Regression, LSTM.	Yahoo Finance	RMSE, MSE.	LSTM showed better accuracy than regression model with RMSE of 0.09	Comparison showed LSTM to be better.
Stock Market Prediction Using Machine Learning Techniques[7]	Naadun Sirimevan, I.G. U. H. Mamalgaha, Chandira Jayasekara, Y. S. Mayuran, Chandimal Jayawardena.	2019	Ensemble model formed by integrating Univariate LSTM, Multivariate LSTM.	Yahoo finance API, Web news, Google trends, twitter sentiments	Accuracy.	Ensemble model proved to be highly Accurate	In order to create a single ideal predictive model, numerous basic models were coupled. Accuracy decreased with time. Effect of sentiments diminished with time
Stock Market's Price Movement Prediction With LSTM Neural Networks[3]	David M. Q. Nelson, Adriano C. M. Pereira, Renato A. de Oliveira.	2017	LSTM from TensorFlow.	Brazil Stock Exchange data (Candlestick database.)	Accuracy, Precision, Recall, F1-score, % return, Maximum drawdown	LSTM more accurate than RF, MLP and Pseudo-random	The model was used on a number of Brazilian stocks and was successful in identifying high differences. Had a bit high variance.
Prediction of Stock Price Based on LSTM Neural Network [4]	Dou Wei	2019	Attention layered LSTM	Chinese stocks index	lag	For small time window output had hysteresis	Time delay for small time span. With increase in time span time delay decreases.
Evaluation of Bidirectional LSTM for Short- and Long-Term Stock Market Prediction [8]	Khaled A. Althelaya, El-Sayed M. El-Alfy, Salahadin Mohammed	2018	Stacked LSTM, BI-LSTM	(S&P500) from Yahoo Finance	MAE, RMSE, and coefficient of determination R ²	BI-LSTM: 0.0289 and 0.0746 RMSE for short and long term. SLSTM: 0.0382 RMSE and 0.090	For both the short and long term, the BLSTM network achieved superior results and excellent convergence. Models performed better for short term than long term.

III. METHODOLOGIES

The Open, Close, High, Low, Adjusted Closing price, and Volume are all included in several data sets used for price prediction. The maximum and minimum prices of a certain stock on a given day are referred to as high and low, respectively. Adjusted Closing refers to the closing price after any corporate actions have been taken into account, as opposed to the raw closing price. Finally, volume refers to the number of stocks that are traded and bought each day. Earnings per share (EPS) is a key metric that measures a company's profitability. The Price to Earnings Ratio (P/E) is

the ratio of a concern's present stock price to its earnings per share (EPS) [5].

Deep learning is a subset of machine learning algorithms that are based on learning data representations. Deep learning models employ a network of multi-layered non-linear processing units known as neurons, which can automatically extract and transform features. An Artificial Neural Network is a network of such neurons. ANN layers are made up of clusters of nodes that are coupled to produce a view of the human brain. The yield of one layer is fed into the next layer as input. These system models have a proclivity for learning from their training data.

A. Recurrent Neural Network (RNN)

A Recurrent Neural Network (RNN) is a kind of Artificial Neural Network (ANN). RNNs are a kind of neural network that is designed to deal with sequential input [5], and are made up of ordinary recurrent cells, in which the networks between the neurons form a directed graph, or, to put it another way, the hidden layers have a self-loop. This allows RNNs to learn current state by using the hidden neurons' previous state, as well as RNNs use the information they've learned before in time and the current input [6]. This allows them to learn a wide range of skills. Handwriting recognition, speech recognition, and other activities are among them.

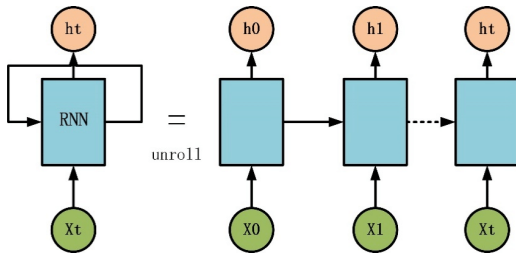


Fig. 1. An unrolled recurrent neural network

Initially (at time step t), the RNN produces an output of h_t for some input X_t . The RNN uses two inputs, X_{t+1} and h_t , to generate the output h_{t+1} in the next time step ($t+1$). Information can be transmitted from one network step to the next via a loop [12].

All input vector units in a typical neural network are presumed to be not reliant on other. As a result, in a typical neural network, sequential information cannot be used. In the RNN model, sequential data from a time series produces a hidden state, which is then combined with the network output that is reliant on the hidden state. Because stock trends are a type of time series numbers, RNNs are the ideal fit for this application. Although training an RNN is difficult due to the RNN's structure's reverse reliance over time [13]. As a result, as the learning period lengthens, RNNs get more complex. As demonstrated in a study addressed by Yashoua et al [14], RNNs lack the ability to learn long-term dependencies. While the fundamental objective of employing an RNN is to understand long-term addictions but it fails at it due to Vanishing Gradient problem that occurs when weights are getting updated during back propagation using chain rule, the weights become too small [15].

As a result, Hochreiter and Schmidhuber presented a method called Long Short-Term Memory (LSTM) [16] in 1997 to deal with these long-term dependencies [16].

B. Long-Short Term Memory (LSTM)

Since the change from state to state of RNN is indicated, the learning model just has the same dimensionality of the data. At each time step, the architecture also employs the same changeover function with the no different new arguments [17].

The RNN has a structure of recurring modules of NN, as indicated above. In typical RNNs this module that is repetitive has a single layer of simple tanh.

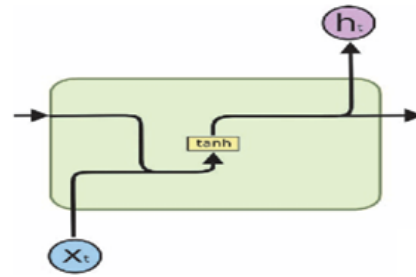


Fig. 2. RNN's single layer

Since LSTM is an advance version of RNN, it too follows chain-like structure of RNN, but has four neural network layers instead of a single, interacting in a special way [6], [12], [15].

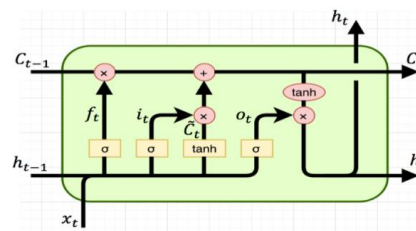


Fig. 3. LSTM's four layers

LSTM consists of memory cells that are responsible to store memory, and gates. These gates control the flow of data and act as controlling knobs which are responsible for the mixing of data. The data in LSTM is manipulated by these gates [6].

The memory unit can keep track of the certain amount of training data. It is the memory unit consisting of cell states that makes LSTM capable of remembering long term dependencies. The three gates are forget gate, input gate and output gate [3], [5], [15], [16], [18].

Memory Cells: They store both short- and long-term memory. It is rather analogous to a conveyer belt [6]. It runs through the entire chain with some trivial interactions with the gates in form of pointwise operations [6], [15], [17].

Hidden State: Hidden state is the output of a LSTM layer that is fed an input to next layer [5].

The sigmoid layer generates values ranging from zero to one, indicating how much of each element should be allowed to pass. The Tanh layer creates a new vector that further makes an addition to the state [17].

Forget Gate: The forget gate is responsible to forget some information from the memory cell. If there is change in context or something then the forget gate generates zeroes which are pointwise multiplied with the memory cell and the corresponding information is forgotten. It generates a vector with values between 0 and 1 with the sigmoid layer [2], [17].

Input Gate: The input gate examines the conditions under which any information should be added (or updated) in the state cell based on the input (e.g., previous output o_{t-1} , input x_t , and the prior state of cell c_{t-1}). This is then pointwise added to memory cell [5], [16], [17].

Output Gate: The filtered form of the memory state will be the output. The Output Gate controls the quantity of data that passes from the present cell state to the concealed state. First, we route a sigmoid layer to determine which aspects of the cell state will be output. The cell state is then passed through tanh-tanh (to make the values to be in range of -1 to 1) and is multiplied by the output of the sigmoid gate, resulting in only the portions we choose to output [5], [12].

The updation of the memory cell is defined by the following equations [11], [15], [17], [18]:

$$ft = \sigma(Wf.[ht-1, xt] + bf) \tag{1}$$

$$it = \sigma(Wi.[ht-1, xt] + bi) \tag{2}$$

$$ct = \tanh(Wc.[ht-1, xt] + bc) \tag{3}$$

$$ot = \sigma(Wo[ht-1, xt] + bo) \tag{4}$$

$$ht = ot * \tanh(ct) \tag{5}$$

where,

ft = forget gate

ht-1 = previous hidden state output.

xt = current input

Wx = weight for respective gate

it = input gate

ot = output gate

σ = sigmoid function

ct = memory state.

ht = current hidden state output.

bx = biases for respective gates

The following univariate LSTM's are used frequently for [5]

TABLE II. Frequently used types of LSTM

Vanilla LSTM	Stacked LSTM	Bidirectional LSTM
Classic LSTM with 4 NN layers. It has memory cell and the gates – input, forget, output. Has single hidden layer.	Multiple LSTM models. output of one layer is passed as input to next. Has multiple hidden layers.	The Bidirectional LSTM trains two input sequences instead of one, thus the first is the original and the second is the reversed version.

CONCLUSION

The goal of this work was to investigate LSTM models for stock price prediction. Despite the fact that after examining the primary influencing elements and combining social reviews connected to stock, reliable stock prediction is still difficult to achieve. It is not feasible to set a price. However, certain strategies have been successful in obtaining a near approximation.

For time series prediction of stock prices majorly LSTMs are used for their long term dependencies and to remove the drawback of Vanishing Gradient Descent of RNNs. Vanilla LSTM, Stacked LSTM, and Bidirectional LSTM are the most commonly utilised LSTM models. Models can be trained to improve with the right adjustment of hyper-parameters. Attention layer can be utilised to eliminate time lag, and tiny step iteration from a small batch gradient descent optimizer can be used to speed up the process. BI-LSTM appeared as having greater accuracy and low error when the models were created with the identical parameters.

As the dataset size grows, the model's accuracy improves as it is trained on additional data. Because stock prices are affected by sentiments, sentiment analysis with LSTM models will produce improved accuracy. However, the timeframe during which feelings will have an effect must be recognised. For the same, different LSTM models can be utilised and then ensembled. LSTM models have a low risk profile when compared to other techniques. Short-term forecasted prices are more reliable than long-term forecasted prices.

There are still a lot of models that can be used and improved upon. Based on their assessing variables and the datasets utilised for their research, each model has its own benefit and disadvantage over the others. Some models are more effective with historical data, while others are more effective with sentiment data. According to the literature review, the BI-LSTM model predicted findings with greater precision than the other models reviewed.

REFERENCES

- [1] Y. Bengio, P. Simard, and P. Frasconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," in *IEEE Transactions On Neural Networks*, 1994, vol. 5, no. 2, pp. 157–166. doi: 10.1109/72.279181.
- [2] I. Parmar *et al.*, "Stock Market Prediction Using Machine Learning," in *First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, 2018, vol. ICSCCC 2018, pp. 574–576. doi: 10.1109/ICSCCC.2018.8703332.
- [3] David M. Q. Nelson, Adriano C. M. Pereira, and Renato A. de Oliveira, "Stock Market's Price Movement Prediction With LSTM Neural Networks," in *International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 1419–1426. doi: 10.1109/IJCNN.2017.7966019.
- [4] D. Wei, "Prediction of Stock Price Based on LSTM Neural Network," in *Proceedings - 2019 International Conference on Artificial Intelligence and Advanced Manufacturing, AIAM*, Oct. 2019, pp. 544–547. doi: 10.1109/AIAM48774.2019.00113.
- [5] S. Gavriel, "Stock Market Prediction using Long Short-Term Memory," *Bachelor's thesis, University of Twente*. 2021.
- [6] M. A. I. Sunny, M. M. S. Maswood, and A. G. Alharbi, "Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model," in *2nd Novel Intelligent and Leading Emerging Sciences Conference, Niles, Giza, Egypt*, Oct. 2020, pp. 87–92. doi: 10.1109/NILES50944.2020.9257950.
- [7] N. Sirimevan, I. G. U. H. Mamalghaha, C. Jayasekara, Y. S. Mayuran, and C. Jayawardena, "Stock Market Prediction Using Machine Learning Techniques," in *International Conference on Advancements in Computing (ICAC), IEEE*, vol. 8, 2019, pp. 192–197. doi: 10.1109/ICAC49085.2019.9103381.
- [8] K. A. Althelaya, E. 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)*, 2018, pp. 151–156. doi: 10.1109/IACS.2018.8355458.
- [9] M. Nikou, G. Mansourfar, and J. Bagherzadeh, "Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms," *Intelligent Systems in Accounting, Finance and Management*, vol. 26, no. 4, pp. 164–174, Oct. 2019, doi: 10.1002/isaf.1459.
- [10] S. Sarode, H. G. Tolani, P. Kak, and L. CS, "Stock Price Prediction Using Machine Learning Techniques," in *International Conference on Intelligent Sustainable Systems (ICISS 2019)*, 2019, pp. 177–181. doi: 10.1109/ISS1.2019.8907958.
- [11] K. G. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017, doi: 10.1109/TNNLS.2016.2582924.
- [12] A. Ghosh, S. Bose, G. Maji, N. C. Debnath, and S. Sen, "Stock price prediction using lstm on indian share market," in *EPiC Series in Computing*, 2019, vol. 63, pp. 101–110. doi: 10.29007/qgcz.
- [13] Y. Wang, Y. Liu, M. Wang, and R. Liu, "LSTM Model Optimization on Stock Price Forecasting," in *17th International Symposium on*

- Distributed Computing and Applications for Business Engineering and Science, DCABES 2018*, Dec. 2018, pp. 173–177. doi: 10.1109/DCABES.2018.00052.
- [14] Y. Bengio, P. Frasconi, and P. Simard, "The Problem of Learning Long-Term Dependencies in Recurrent Networks," in *invited paper at the IEEE International Conference on Neural Networks 1993, San Francisco, IEEE Press*, 1993, pp. 1183–1188. doi: 10.1109/ICNN.1993.298725.
- [15] C. Olah, "Understanding LSTM Networks -- colah's blog," 2015. <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (accessed Sep. 09, 2021).
- [16] S Hochreiter and J Schmidhuber, "Long Short-Term Memory," in *Neural Computation*, vol. 9, no. 8, , 1997, pp. 1735–1780. doi: 10.1162/neco.1997.9.8.1735.
- [17] IEEE Communications Society and Institute of Electrical and Electronics Engineers, *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI) : 13-16 Sept. 2017*. 2017, pp. 1643–1647. doi: 10.1109/ICACCI.2017.8126078.
- [18] Mohammad Asiful Hossain, Rezaul Karim, Rupa Thulasiram, Neil D. B. Bruce, and Yang Wang, "Hybrid Deep Learning Model for Stock Price Prediction," in *Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI 2018) : 18 - 21 November 2018, Bengaluru, 2018*, pp. 1837–1844. doi: 10.1109/SSCI.2018.8628641.