2018 9th International Conference on Information and Communication Systems (ICICS)

# Evaluation of Bidirectional LSTM for Short- and Long-Term Stock Market Prediction

Khaled A. Althelaya, El-Sayed M. El-Alfy, Salahadin Mohammed

Department of Information and Computer Science, College of Computer Sciences and Engineering
King Fahd University of Petroleum and Minerals, Dhahran 31261, Kingdom of Saudi Arabia
Email: {g201532610, alfy, adam}@kfupm.edu.sa

***Abstract*—Recently, there has been a rapidly growing interest in deep learning research and their applications to real-world problems. In this paper, we aim at evaluating and comparing LSTM deep learning architectures for short- and long-term prediction of financial time series. This problem is often considered as one of the most challenging real-world applications for time-series prediction. Unlike traditional recurrent neural networks, LSTM supports time steps of arbitrary sizes and without the vanishing gradient problem. We consider both bidirectional and stacked LSTM predictive models in our experiments and also benchmark them with shallow neural networks and simple forms of LSTM networks. The evaluations are conducted using a publicly available dataset for stock market closing prices.**

## I. Introduction

Applications of time series are vast in many fields such as electrical signal analysis, language processing and speech recognition, traffic analysis, weather forecasting, unemployment rate analysis, inflation dynamics analysis, and many others [1]–[3]. At abstract level, a time series is any sequence of measurements or observations at regular intervals in chronological order over a specific time period. The analysis of time series mainly comprises statistical techniques to characterize and model data patterns. Common techniques for modeling sequential data involve estimating some parameters for fitting a presumed time series model, such as Autoregressive (AR), Autoregressive Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA) [4]–[6].

Various activities in financial markets are typically represented and analyzed as a special type of time series. Due to the noisy and complex nature of such time series, important patterns cannot be accurately captured by traditional methods which depend only on linear regression and parameter estimation. Most of the financial time series tend to exhibit nonlinear patterns in their constructs [7], [8]. As a result, it is difficult to predict or forecast the stock market behavior without more robust and highly nonlinear modeling techniques [9], [10].

Deep learning is a recent methodology that can be employed to perform prediction and classification operations based on highly complex training data. Due to the unprecedented advances in deep learning, many scientific fields exploit its high accurate performance to build efficient solutions to different kinds of problems. Deep neural networks (DNNs) showed a superior performance in many other areas of applications such as signal processing, speech recognition, and image classification. Therefore, exploring DNNs techniques in financial time series prediction is a highly recommended solution. Moreover, financial time series can be considered a good candidate and suitable subject for deep learning techniques.

A number of studies have applied different deep learning techniques to time series prediction. Deep recurrent neural network (RNN) is one of the techniques employed by many prediction approaches. These techniques can remember preceding data inputs while using current data to learn network weights. Different variations of deep RNN, such as Long-Short Term Memory (LSTM), have been developed to improve RNN network ability to preserve previous network states and capture long term dependencies as well. The original LSTM was developed to extend the RNN memory state to enable it to deal with longer input sequences [11], [12]. Another form of RNN is Bidirectional LSTM (BLSTM) where the preceding and succeeding input sequences can be used to exploit all input data to satisfy best learning process performance. A third form of RNN is to stack several LSTM layers to build Stacked LSTM (SLSTM) network to perform deep learning and is usually used to capture more complex patterns in the time series at different scales.

In this work, we mainly focus on the deep learning methodology and explore its advantages in financial time series prediction. We contribute to the literature by conducting several experiments to investigate the potential of integrating deep learning approaches into financial time series prediction. We study, evaluate and compare both bidirectional and stacked LSTM architectures for Short- and Long-Term stock market prediction. Moreover, we compare their performance with shallow neural networks and simple forms of LSTM. Stock market data of S&P500 for period from 01/01/2010 to 30/11/2017 from yahoo finance is utilized to conduct the experiments.

The organization of this paper is as follows. Section II describes related background and preliminaries. Section III briefly discusses and reviews related work. In section IV, we describe the proposed methodology. Experiments and results are discussed in Section V. Finally, conclusions are presented in Section VI.

978-1-5386-4366-2/18/$31.00 ©2018 IEEE


Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.

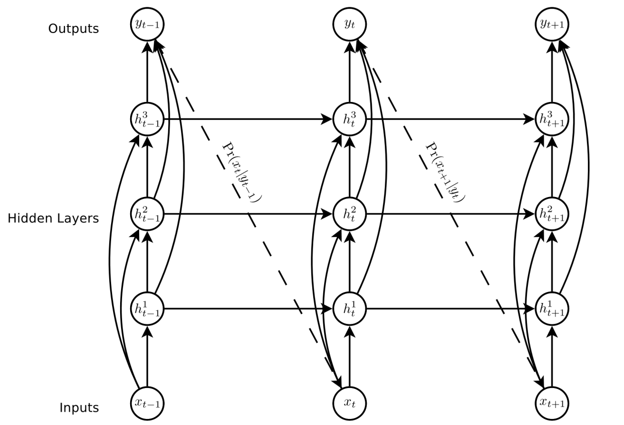


Fig. 1. A typical deep recurrent neural network architecture [13].

## II. Preliminaries

Deep learning methodology utilizes a set of computational layers designed to learn patterns from input data. Each layer is employed to extract a specific type of information. The output of a certain layer is the input to the succeeding layer. The input data is fed into the input layer and the target output is ultimately generated by the output layer. One of the techniques used to construct deep learning architectures is deep RNN. It uses the current input data in addition to data from previous processing steps to learn network weights. While the network is cascading forward through the sequential input data, information is preserved in the hidden states of the network. The training algorithm is a variation of the so called backpropagation through time (BPTT). Calculations are performed according to time sequence and series order by linking every time step to the preceding steps. Figure 1 illustrates a typical architecture of deep RNN and the prediction process.

### A. Long-Short Term Memory (LSTM)

Due to the vanishing gradient problem, many variations of RNN have been developed to solve this problem such as LSTM, which was proposed by Hochreiter and Schmidhuber [11]. It defines gated cells that can act on the received input by blocking or passing information based on the importance of the data element. The learning process through backpropagation estimates the weights that allow the data in the cells either to be stored or deleted. The LSTM transition equations are as follows:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1}) \quad (1)$$
$$f_t = \sigma(W_f x_t + U_f h_{t-1} + V_f c_{t-1}) \quad (2)$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_t) \quad (3)$$
$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1}) \quad (4)$$
$$c_t = f_t^i \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (5)$$
$$h_t = o_t \odot \tanh(c_t) \quad (6)$$

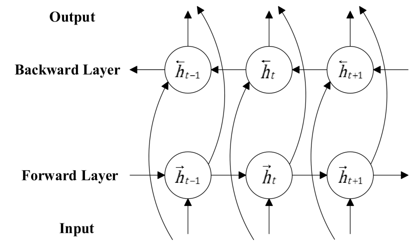


Fig. 2. Bidirectional LSTM (BLSTM) [15].

where $i_t$ denotes the input gate and $o_t$ denotes the output gate. The forget gate, memory cell, and hidden state are denoted by $f_t$, $c_t$, and $h_t$, respectively [13].

### B. Bidirectional LSTM

Another variation of RNN is Bidirectional RNN, which is developed by Schuster and Paliwal [14] to train the network using input data sequences in the past and future. Two connected layers are used to process the input data. Each layer performs the operations using reversed time step direction. The results can then be combined using different types of merging methods. Similarly, Bidirectional LSTM uses two layers such that one layer performs the operations following the same direction of the data sequence and the other layer applies its operations on in reverse direction of the data sequence as shown in Figure 2. BLSTM have been found more effective than unidirectional LSTM in some applications such as phoneme classification [13].

## III. Related Work

Deep learning has been applied for prediction and classification in several domains. It has been used in speech recognition, text classification, language modeling, and many others. It has also been applied in time series prediction. Many studies developed different prediction approaches using LSTM networks. Persio et al. [16] explored the effectiveness and efficiency of applying LSTM network to financial time series prediction. The study included several experiments to compare the prediction performance of Convolutional Neural Network (CNN) and LSTM. Other types of DNNs such as Stacked Autoencoders (SAEs) have been used by Bao et al. [17] to propose a prediction approach using LSTM. The study concluded that when combining LSTM network with SAEs, higher prediction performance can be achieved.

Some studies made use of textual information on news articles and web forum posts as input to deep LSTM network to perform stock market prediction. Verma et al. [18] studied the impact of the presence of various types of information from news articles on stock market prediction. They categorized the information by means of a proposed factoring framework denoted by PESTEL (Political, Economic, Social, Technological, Environmental and Legal). The prediction process accomplished by the integration of LSTM network using information components. The study reported the superiority of LSTM network over linear SVM based on experiments conducted



Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.

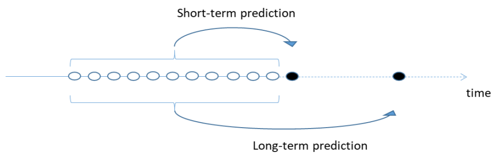


Fig. 3. Illustration of short-term and long-term prediction

on data from multiple indices of stock markets. Li et al. [19] attempted to improve the prediction model built using Long Short-Term Memory (LSTM) neural network by integrating a naïve Bayes modeling technique to include and extract investor sentiment and market factors from forum posts. Akita et al. [20] combined both textual and numerical information by representing data from newspaper articles using Paragraph Vector to model the impact of past events on opening prices. The proposed approach fed textual and numerical input vectors to LSTM network to perform prediction.

Deep neural networks built using bidirectional LSTM have been widely applied in speech recognition and text classification but rarely applied in time series prediction in general, and stock markets in particular. To the best of our knowledge no previous study attempted to investigate the potential of using BLSTM for stock market prediction. However, BLSTM was used by [21] for energy load forecasting. The study compared the performance of multiple architectures including BLSTM, multilayer LSTM, and decoder-encoder architecture. The best performance was achieved by the BLSTM architecture. Furthermore, BLSTM was also applied by [22] for network-wide traffic speed prediction. The study compared both unidirectional SLSTM and BLSTM neural network. It investigated different structures of LSTM and concluded that two-layers of stacked BLSTM outperforms other LSTM-based structures. Our work investigates and evaluates the integration of a proposed methodology using BLSTM into financial time series prediction by performing several experiments to compare BLSTM and SLSTM.

## IV. Methodology

The structure of a unidirectional SLSTM depends on learning from data inputs on which its hidden states have passed through. It only sees information from the past. It has no mechanism to enable it to consider information in the future while predicting the present. The stacked architecture can model more sophisticated data patterns. It can perform deeper analysis on the training data. On the other hand, BLSTM has the ability to process and learn from data in both directions; from future to past and from past to future. BLSTM while going through data combines forward and backward contextual information and uses it to make prediction or classification.

To take advantage of the merits of both architectures in stock market prediction, we applied this methodology to build effective prediction models for the stock market closing prices. Depending on the adopted time lag, the network is fed with data within a sliding window and estimate its weights. In our experiments, we use a window of size 10 (working days of two weeks) to predict one-day ahead (short term) and 30-day ahead (long term). Figure 3 illustrates the input values (predictors) and output (target) for short-term and long-term predictions.

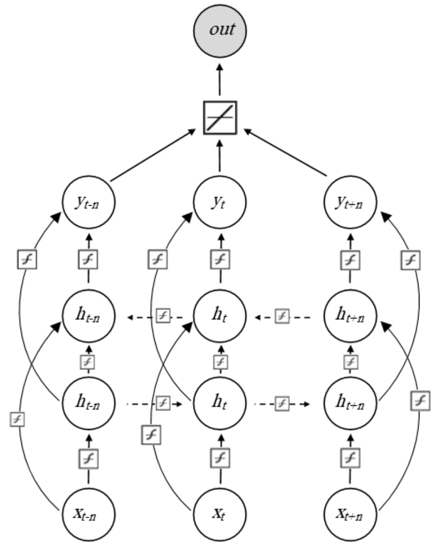


Fig. 4. Methodology used to train the BLSTM network

Figure 4 illustrates the methodology followed to build the BLSTM network. $y_t$ denotes the returned value of cell $t$ from the learning process for each training sample, $x_t$ denotes the input data, and $h_t$ denotes the hidden layers. LSTM layers use the hard sigmoid transfer function for the recurrent activation and use tanh transfer function for the output layer. The final output is computed from a dense layer using a linear activation function. We tune models by conducting a set of experiments on different BLSTM and SLSTM structures by changing the number of memory cells and the number of epochs to control the learning process. The same operations are performed on the different structures to measure the performance of each designed model.

## V. Experiments

We download and used the historical data of the Standard & Poor 500 Index (S&P500) for the period from 01/01/2010 to 30/11/2017 from Yahoo finance to perform our experiments. S&P500 is one of the important benchmark datasets and a leading indicator for 500 top-traded companies in the US market. We have implemented the proposed stock prediction methodology in Python using Keras open-source package for deep learning with TensorFlow backend [23]. The closing price at the end of every trading day is used for the purpose of evaluating prediction performance. Data is preprocessed to be more stationary by focusing on daily changes, i.e. taking the difference between each two consecutive days. Then, the data is normalized between 0 and 1 using min-max normalization. We split the data into two parts: training and testing. The first 80% duration of the whole data is allocated for training and the rest 20% duration allocated for testing. The same



Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.

TABLE I
MAE, RMSE, AND $R^2$ OF BLSTM NETWORK RESULTS FOR SHORT-TERM PREDICTION

| Networks | MAE | | | | RMSE | | | | $R^2$ |
|---|---|---|---|---|---|---|---|---|---|
| | Testing | Training | Normalized Testing | Normalized Training | Normalized Testing | Normalized Training | Testing | Training | |
| 4 Neurons Network | 54.651 | 20.065 | 0.034062 | 0.012506 | 0.044202 | 0.015934 | 70.920 | 25.566 | 0.993 |
| 8 Neurons Network | 34.768 | 19.352 | 0.021669 | 0.012061 | 0.027032 | 0.015135 | 43.372 | 24.284 | 0.994 |
| 16 Neurons Network | 28.664 | 18.883 | 0.017865 | 0.011769 | 0.022718 | 0.015340 | 36.451 | 24.613 | 0.994 |
| 32 Neurons Network | 29.677 | 18.496 | 0.018496 | 0.011527 | 0.021747 | 0.014549 | 34.893 | 23.344 | 0.994 |
| Total average | 36.940 | 19.199 | 0.023023 | 0.011966 | 0.028925 | 0.015240 | 46.409 | 24.452 | 0.994 |

TABLE II
MAE, RMSE, AND $R^2$ OF STACKED LSTM NETWORK RESULTS FOR SHORT-TERM PREDICTION

| Networks | MAE | | | | RMSE | | | | $R^2$ |
|---|---|---|---|---|---|---|---|---|---|
| | Testing | Training | Normalized Testing | Normalized Training | Normalized Testing | Normalized Training | Testing | Training | |
| 4 Neurons Network | 65.463 | 23.162 | 0.040801 | 0.014436 | 0.052363 | 0.018228 | 84.014 | 29.247 | 0.991 |
| 8 Neurons Network | 54.990 | 22.902 | 0.034273 | 0.014274 | 0.042244 | 0.017729 | 67.779 | 28.445 | 0.992 |
| 16 Neurons Network | 44.120 | 19.278 | 0.027498 | 0.012015 | 0.033103 | 0.015353 | 53.113 | 24.633 | 0.994 |
| 32 Neurons Network | 34.950 | 18.384 | 0.021783 | 0.011458 | 0.025862 | 0.014663 | 41.495 | 23.526 | 0.994 |
| Total average | 49.740 | 20.891 | 0.031001 | 0.013020 | 0.038264 | 0.016465 | 61.393 | 26.418 | 0.993 |

TABLE III
MAE, RMSE, AND $R^2$ OF BLSTM NETWORK RESULTS FOR LONG-TERM PREDICTION

| Networks | MAE | | | | RMSE | | | | $R^2$ |
|---|---|---|---|---|---|---|---|---|---|
| | Testing | Training | Normalized Testing | Normalized Training | Normalized Testing | Normalized Training | Testing | Training | |
| 4 Neurons Network | 120.988 | 46.813 | 0.075407 | 0.029176 | 0.089146 | 0.036763 | 143.031 | 58.985 | 0.968 |
| 8 Neurons Network | 110.066 | 48.459 | 0.068600 | 0.030203 | 0.079403 | 0.037441 | 127.399 | 60.072 | 0.966 |
| 16 Neurons Network | 91.857 | 47.883 | 0.057251 | 0.029843 | 0.067557 | 0.037199 | 108.393 | 59.684 | 0.967 |
| 32 Neurons Network | 83.687 | 48.576 | 0.052159 | 0.030276 | 0.062648 | 0.038041 | 100.517 | 61.035 | 0.965 |
| Total average | 101.650 | 47.933 | 0.063354 | 0.029875 | 0.074689 | 0.037361 | 119.835 | 59.944 | 0.966 |

TABLE IV
MAE, RMSE, AND $R^2$ OF STACKED LSTM NETWORK RESULTS FOR LONG-TERM PREDICTION

| Networks | MAE | | | | RMSE | | | | $R^2$ |
|---|---|---|---|---|---|---|---|---|---|
| | Testing | Training | Normalized Testing | Normalized Training | Normalized Testing | Normalized Training | Testing | Training | |
| 4 Neurons Network | 135.944 | 47.340 | 0.084729 | 0.029505 | 0.101257 | 0.036984 | 162.463 | 59.340 | 0.967 |
| 8 Neurons Network | 124.323 | 47.807 | 0.077486 | 0.029796 | 0.091482 | 0.036919 | 146.780 | 59.235 | 0.967 |
| 16 Neurons Network | 119.455 | 51.293 | 0.074452 | 0.031969 | 0.087484 | 0.039332 | 140.365 | 63.107 | 0.963 |
| 32 Neurons Network | 108.131 | 47.780 | 0.067394 | 0.029779 | 0.080033 | 0.037301 | 128.411 | 59.849 | 0.967 |
| Total average | 121.963 | 48.555 | 0.076015 | 0.030262 | 0.090064 | 0.037634 | 144.504 | 60.382 | 0.966 |

data is used to train many networks based on variations of number of epochs and neurons for every applied technique and time scale. The data framed into ten days window setup with overlapping as illustrated in section IV. To measure how well the model predicted values represent the actual data, we used three performance measures in the experiments: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination which is denoted as $R^2$. For better prediction, MAE and RMSE should be close to zero and $R^2$ should be close to one. These measures are defined mathematically as follows:

$$MAE = \frac{1}{n}\sum_{t=0}^{n-1}|y_t - \hat{y}_t| \tag{1}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=0}^{n-1}(y_t - \hat{y}_t)^2} \tag{2}$$

$$R^2 = 1 - \frac{\sum_{t=0}^{n-1}(y_t - \hat{y}_t)^2}{\sum_{t=0}^{n-1}(y_t - \bar{y})^2} \tag{3}$$

where $y_t$ and $\hat{y}_t$ represent the actual and predicted value at step $t$ for $0 \leq t < n$, respectively, and $\bar{y} = \sum_{t=0}^{n-1} y_t/n$. We ran several experiments and built several models to predict both short- and long-term scales using BLSTM, SLSTM, LSTM and Multi-Layer Perceptron Artificial Neural Network (MLP-ANN). The short-term scale networks are trained to predict one day price in the future, while the long-term scale networks are trained to predict the stock price for 30 days in the future.

Initially, we conducted four sets of experiments by applying both BLSTM and SLSTM networks for short- and long-term



Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.

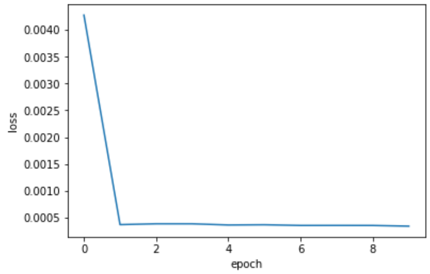


Fig. 5. The learning curve of the BLSTM short-term prediction model.

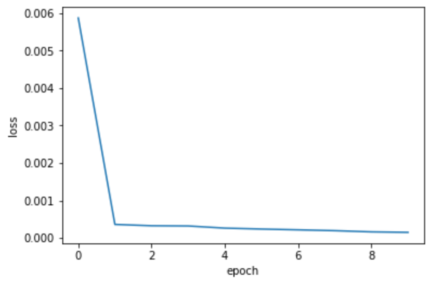


Fig. 6. The learning curve of the SLSTM short-term prediction model.

prediction. Four network structures were designed for each applied technique by varying the number of units to be 4, 8, 16 and 32 neurons (memory cells). The networks are trained using one to ten epochs for several runs and based on random initials for every epoch. The final results are dependent on averages of the performance of many different executions of the learning operations for every structure. The averages of the results per network structure and the total average of all networks structures for each technique are shown in Tables I, II, III, and IV. For short-term prediction, the MAE and RMSE averages using normalized data of the BLSTM network are 0.023 and 0.0289, respectively. In contrast, the MAE and RMSE averages of the short-term SLSTM networks are 0.031 and 0.0382, respectively. For long-term prediction, the MAE and RMSE averages using normalized data of BLSTM networks are 0.0633 and 0.0746, respectively, and the MAE and RMSE averages of SLSTM network are 0.076 and 0.090, respectively. We can see that both LSTM networks show high performance for predicting short-term and long-term prices. BLSTM networks produce higher performance and better convergence for short-term prediction and the difference gets much higher for long-term prediction.

For more analysis and investigation, we selected the best produced models and compared them based on normalized RMSE, MAE, and $R^2$. The comparisons are designed to investigate the performance of BLSTM and SLSTM network structures for short- and long-term predictions. The selected models were compared with another two models developed using LSTM and MLP-ANN. The short-term prediction comparison is shown in Table V. The long-term prediction comparison is shown in Table VI. The learning curves of the short-term

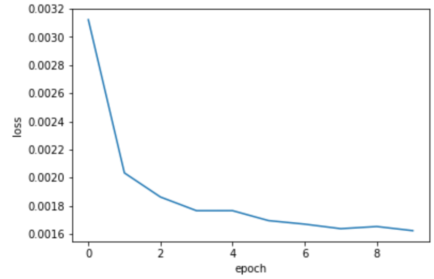


Fig. 7. The learning curve of the BLSTM long-term prediction model.

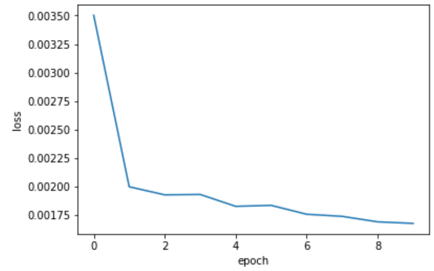


Fig. 8. The learning curve of the SLSTM long-term prediction model.

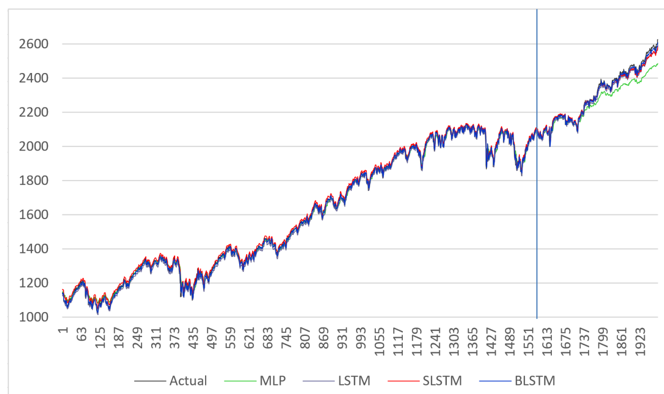


Fig. 9. The short-term prediction using BLSTM, SLSTM, LSTM, and MLP.

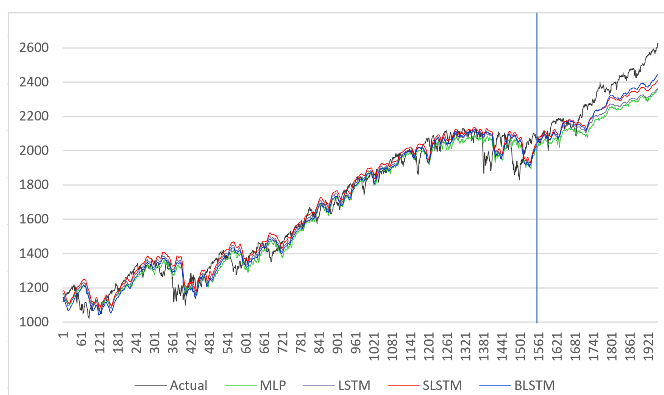


Fig. 10. The long-term predictions using BLSTM, SLSTM, LSTM, and MLP.



Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.

TABLE V
COMPARING BEST SELECTED MODELS FOR SHORT-TERM PREDICTION

| | Training | | | Testing | | |
|---|---|---|---|---|---|---|
| | **MAE** | **RMSE** | $R^2$ | **MAE** | **RMSE** | $R^2$ |
| **MLP-ANN** | 0.01025 | 0.01376 | 0.995 | 0.03202 | 0.03875 | 0.995 |
| **LSTM** | 0.01146 | 0.01351 | 0.996 | 0.01398 | 0.01582 | 0.996 |
| **SLSTM** | 0.01031 | 0.01329 | 0.996 | 0.00987 | 0.01248 | 0.996 |
| **BLSTM** | 0.00737 | 0.00982 | 0.997 | 0.00736 | 0.00947 | 0.997 |

TABLE VI
COMPARING BEST SELECTED MODELS FOR LONG-TERM PREDICTION

| | Training | | | Testing | | |
|---|---|---|---|---|---|---|
| | **MAE** | **RMSE** | $R^2$ | **MAE** | **RMSE** | $R^2$ |
| **MLP-ANN** | 0.03278 | 0.04070 | 0.96 | 0.08285 | 0.09369 | 0.96 |
| **LSTM** | 0.03063 | 0.04111 | 0.96 | 0.07123 | 0.08371 | 0.96 |
| **SLSTM** | 0.03244 | 0.04478 | 0.95 | 0.05668 | 0.06637 | 0.95 |
| **BLSTM** | 0.03247 | 0.04211 | 0.96 | 0.05242 | 0.06055 | 0.96 |

and long-term prediction for all four models are illustrated in Figures 5, 6, 7, and 8. The short- and long-term prediction versus target curve for all compared models are illustrated in Figure 9 and 10, respectively. The prediction curves plotted to clarify the difference between the four compared models. The results show that BLSTM network attained higher performance and better convergence for short- and long-term predictions.

## VI. CONCLUSIONS

In this study, we evaluated the performance of bidirectional and stacked LSTM deep learning methodology for stock market prediction. The performance is evaluated on a benchmark dataset for short- and long-term prediction using three performance measures. The performances of the tuned BLSTM and SLSTM models are also compared with shallow neural networks and unidirectional LSTM. The results showed that both BLSTM and stacked LSTM networks produced better performance for predicting short-term prices as opposed to the long-term prediction results. The results also showed superiority of deep learning methodology over shallow neural networks. Overall, BLSTM networks demonstrated better performance and convergence for both short- and long-term predictions.

## ACKNOWLEDGMENT

The authors would like to thank King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia, for the support during this work.

## REFERENCES


[1] A. Altunkaynak and T. A. Nigussie, "Monthly water consumption prediction using season algorithm and wavelet transform–based models," *Journal of Water Resources Planning and Management*, vol. 143, no. 6, p. 04017011, 2017.

[2] P. Liang, H.-D. Yang, W.-S. Chen, S.-Y. Xiao, and Z.-Z. Lan, "Transfer learning for aluminium extrusion electricity consumption anomaly detection via deep neural networks," *International Journal of Computer Integrated Manufacturing*, pp. 1–10, 2017.

[3] H. Wang, G. Wang, G. Li, J. Peng, and Y. Liu, "Deep belief network based deterministic and probabilistic wind speed forecasting approach," *Applied Energy*, vol. 182, pp. 80–93, 2016.

[4] D. K. Kılıç and Ö. Uğur, "Multiresolution analysis of S&P500 time series," *Annals of Operations Research*, pp. 1–20, 2016.

[5] P. Li, C. Jing, T. Liang, M. Liu, Z. Chen, and L. Guo, "Autoregressive moving average modeling in the financial sector," in *2nd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 2015, pp. 68–71.

[6] 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.

[7] M. Bildirici, 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.

[8] M. Khashei, M. Bijari, and G. A. R. Ardali, "Hybridization of autoregressive integrated moving average (ARIMA) with probabilistic neural networks (PNNs)," *Computers & Industrial Engineering*, vol. 63, no. 1, pp. 37–45, 2012.

[9] I. Kaastra and M. Boyd, "Designing a neural network for forecasting financial and economic time series," *Neurocomputing*, vol. 10, no. 3, pp. 215–236, 1996.

[10] 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.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[13] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5, pp. 602–610, 2005.

[14] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[15] Y. Fan, Y. Qian, F.-L. Xie, and F. K. Soong, "Tts synthesis with bidirectional lstm based recurrent neural networks," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[16] 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, pp. 403–413, 2016.

[17] 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*, vol. 12, no. 7, p. e0180944, 2017.

[18] I. Verma, L. Dey, and H. Meisheri, "Detecting, quantifying and accessing impact of news events on indian stock indices," in *Proceedings of the International Conference on Web Intelligence*, 2017, pp. 550–557.

[19] J. Li, H. Bu, and J. Wu, "Sentiment-aware stock market prediction: A deep learning method," in *International Conference on Service Systems and Service Management (ICSSSM)*, 2017, pp. 1–6.

[20] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, "Deep learning for stock prediction using numerical and textual information," in *IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, 2016, pp. 1–6.

[21] 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, 2017, p. 190006.

[22] Z. Cui, R. Ke, and Y. Wang, "Deep stacked bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," in *6th International Workshop on Urban Computing (UrbComp 2017)*.

[23] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.




Authorized licensed use limited to: Universitatsbibliothek Erlangen Nurnberg. Downloaded on July 01,2021 at 11:04:04 UTC from IEEE Xplore. Restrictions apply.