AUT Jouranl of Modeling and Simulation

AUT J. Model. Simul., 52(2) (2020) 179-188
DOI: 10.22060/miscj.2021.17712.5191

Long-term prediction in Tehran stock market using a new architecture of Deep neural networks

Mahsa Rajabi, Hamid Khaloozadeh*

Department of Systems and Control, K.N. Toosi University of Technology, Tehran, Iran

ABSTRACT: Financial markets play an important role in the economy of modern societies. Therefore, many researchers have investigated to forecast these markets using various statistical and soft computing methods. Financial time series are essentially complex, dynamic, nonlinear, noisy, nonparametric, and chaotic, so they cannot be described by analytical equations, because their dynamics are too complex or unknown. In recent years, deep learning methods have attracted lots of attention, due to their exceptional performance compared to other existing approaches in many learning problems. The objective of this paper is long-term prediction of price time series in Tehran Stock Exchange. For this purpose, a new architecture of two deep learning methods, Long-Short Term Memory (LSTM) and Recurrent Neural Network (RNN), for ten-step ahead simultaneous prediction, are proposed. That is a multivariable structure with multi outputs. By using the output error feedbacks as internal inputs, the network can learn error dynamics during the training phase. Experimental results show the high capability of the proposed structure for both methods in multi-step ahead stock price forecasting and the superiority of the LSTM network compared to RNN for long-term predictions.

Review History:

Received: Jan. 12, 2020
Revised: Jun. 05, 2020
Accepted: Nov. 19, 2020
Available Online: Dec. 01, 2020

Keywords:

Deep Learning
LSTM
RNN
Long Term Forecasting
Tehran Stock Exchange

1- INTRODUCTION

One of the important problems in economics is portfolio management that focuses on the scientific management of a combination of assets that meets ultimate investment objectives: maximizing return and minimizing the risk of the portfolio [1]. Obviously, achieving these goals requires an accurate prediction of stock price changes, especially for a long-time. Forecasting financial markets is one of the most important issues in the field of forecasting and time series studies. In recent years, different methods have been suggested for time series forecasting. Typical methods with few nonlinear operations cannot accurately model financial time series data. With the increasing availability of different types of financial data as well as recent advances in data science and computer processing, it is now possible to utilize large amounts of data for stock market analysis and forecasting. Using deep learning methods, researchers have found that for almost all sequence prediction problems such as time series, LSTM[1] networks have been suggested as the most effective solution, and this is due to the feature of memorizing selective features for a long time in them.

Two main methods for time series prediction are statistical methods and machine learning approaches [2]. Statistical methods with few parameters, which often assumes linearity, stationarity, and normality, cannot model financial time series. The Auto-Regressive Moving Average (ARMA), the Auto-Regressive Integrated Moving Average (ARIMA), the Generalized Autoregressive Conditional Heteroskedastic (GARCH) volatility, and the Smooth Transition Autoregressive (STAR) model are some statistical approaches that widely used for stock market analysis [3]. Machine learning methods, on the other hand, are capable of dealing with highly noisy and nonlinear data with uncertainty, so they are powerful tools for this field. Among these methods, ANNs[2] have received more attention due to their ability to deal with uncertain, noisy and irregular data. According to [4] most of the studies that choose ANNs for stock market forecasting, used MLP[3] trained by the back-propagation algorithm with relative success. Various types of ANNs have been proposed so far for forecasting in this research area. Despite the widespread use of ANNs, these approaches still have limitations in learning financial data features. SVMs[4] are other machine learning methods for pattern recognition in financial time series. They use a risk function consisting of the empirical error and a regularized term, which is derived from the structural risk minimization principle [5]. Due to the high capability of this method in the nonlinear estimation, it has been used in both classification and regression problems. Decision trees like ANNs and SVMs,

1 Long-Short Term Memory

*Corresponding author's email: h_khaloozadeh@kntu.ac.ir

2 Artificial Neural Networks
3 Multi-Layer Perceptron
4 Support Vector Machines

has been widely used in the machine learning literature, as reviewed by [6]. Combination of decision trees in the RF[1] method for regressions or classifications can be lead to good results for stock market forecasting [7]. However, it should be noted that these approaches have disadvantages, such as local optimization, over-fitting, and difficulty in selecting many parameters, which directly affects the prediction accuracy [8]. To solve these problems, some researchers have suggested using a hybrid approach that combines separate intelligent approaches. The combination of different approaches can lead to reduction in the randomness of the training process and inaccuracy in parameters [9]. In the forecasting history of financial markets, different approaches have applied the hybrid idea to improve forecasting accuracy. For example, the combination of parametric and nonparametric methods, linear and nonlinear methods [10], SVM and KNN[2] [11], GA with different ANN techniques, genetic fuzzy systems and artificial neural networks, Adaptive Neuro-Fuzzy Inference systems and ANFIS with indirect Approach TKS-fuzzy based [12]. Recently, the focus of machine learning studies is on applying different approaches for extracting high-level features through the huge amounts of input data [13]. The main purpose of these methods is to extract useful information among complex data using deep nonlinear networks. Deep learning methods proposed by Hinton and Salakhutdinov provide new approaches for training deep neural networks [14]. Although these are new methods but have received much attention in the last few years, and due to their great capability for nonlinear learning, dimensionality reduction, and high-level feature extraction from large amounts of data, they have led to unexpected advances in conventional neural networks [15] and [16]. CNN[3] [17], RBM[4] [18] and AE[5] [19] are some of the deep learning approaches. Deep learning approaches have been successfully used in a wide variety of problems such as classification, time series forecasting, dimensionality reduction and speech recognition. A comprehensive overview of the history of the development of deep learning methods that have led to their current position and their useful applications is provided by [20].

Researchers have found that long-short term memory (LSTM) networks have been suggested as the most effective solution for almost all sequence prediction problems such as time series. LSTM network is a recurrent neural network with the capability of learning long-term dependencies. Hackerter and Schmeidbar [21] introduce this structure and after that many researches have improved and refined it in the following works. This method is superior to many forward and recursive neural networks because of the feature of selectively memorizing patterns for a "long time". These networks, which perform well for a wide variety of problems, are actually designed to solve the problem of long-term dependencies. Recursive structures have a chain iterative form. One of the limitations of the RNN is that it only works well when there are short-term dependencies, while the information required may be far from the current point of view through invalid data. However, the LSTM can maintain important information for a long time, because of its special structure.

According to all superiority of deep learning methods in time series forecasting field, it will be very helpful to investigate the advantages of their application in Tehran Stock Exchange, which unfortunately is not done so far. This study was undertaken to investigate the efficiency of using deep learning methods for price time series forecasting in TSE; especially the advantage of the LSTM network over the RNN. In addition, since long-term prediction plays a significant role in increasing return and decreasing the risk of investment, a new architecture of multi steps ahead simultaneous prediction, has been proposed [22]. In this model, which is based on learning the output error dynamics, the prediction errors have been used as the internal inputs of the forecasting network. Therefor the network tends to learn error dynamics during the training phase and as a result, and the prediction accuracy will be increased and the error values become close to zero

The rest of the paper is organized as follows: Section 2 introduces related works. Section 3 describes the LSTM structure and its comparison with the RNN method. Section 4 presents the multivariable structure for long-term prediction, simulation details, and obtained results. Finally, Section 5 presents our conclusion and future works.

2- RELATED WORKS

[23] examined how deep learning algorithms such as LSTM performs better than conventional algorithms such as ARIMA to predict time series. The results show that the LSTM method improved the prediction accuracy on average by 85% compared to ARIMA. [24] investigated the performance of LSTM networks to predict future trends in stock prices compared to other machine learning techniques such as multilayer perceptron, random forest and pseudo-random, and demonstrates the superiority of the LSTM method. [25] compared the performances of the bidirectional and stacked LSTM models with shallow neural networks for predicting short- and long-term prices. The results showed that both BLSTM and SLSTM networks produced better performance for predicting short-term prices. The results also showed superiority of deep learning methodology over MLP for three performance measures including: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and R-squared $\left(R^2\right)$. BLSTM had the best performance generally. [26] demonstrated that deep learning can improve the accuracy of stock market prediction. 2-Dimensional Principal Component Analysis $(2D)^2$ PCA + Deep Neural Network (DNN) method is compared with state of the art $(2D)^2$ PCA + Radial Basis Function Neural Network (RBFNN) and $(2D)^2$ PCA + RNN. The results show that proposed model for deep learning method with four layers improved the accuracy of predicted return and Hit Rate compared with conventional neural network method and also shallow RNN. [27] proposed

1 Random forest
2 K Nearest Neighborhood
3 Convolutional Neural Networks
4 Restricted Boltzmann Machines
5 Stacked Auto Encoders

a new method for predicting exchange rates by combining DBN[1] and conjugate gradient. In that study, a comparison was made between the feed forward neural network and the proposed method and the results showed that the deep learning approach performed better. [28] has developed a DBN involving two Boltzmann machines. This approach has been evaluated by chaotic time series modeling and the results have shown that the proposed method has higher prediction accuracy than MLP and ARIMA methods. [29] examined whether feature extraction from stock returns is possible by deep learning method. This research used a stacked auto encoder consist of RBMs. The results of this study show that the stacked auto encoder can extract features even from time series with low signal-to-noise ratio such as stock prices. In other applications, various studies have been performed. [30] evaluated the efficiency of Extended LSTM network for air pollutant concentration predictions. In the proposed model, first LSTM layers are used to extract features from historical data as the main input and then month and hour data encoded using one-hot encoding method. Next, fully connected layers are used to obtain the output from merged features. Simulation results show that deep learning models exhibited better prediction performance than traditional shallow models, such as the SVR, ARMA models. In addition, compared with RNN model, the extended LSTME model and the simple LSTM exhibited better prediction performance, as indicated by the RMSE, MAE and MAPE values. Moreover, although the prediction performance of long-term prediction is reduced, the performance of the proposed model was suitable for long-term prediction. [31] proposed a fault diagnosis framework based on LSTM model to learn features from multivariate time series data of two wind turbine datasets and capture long-term dependencies. In the proposed model, the data-fusion strategy via the input preparation process is used. The simulation results show that proposed LSTM model is superior to the state of the art approaches including CNN and RNN for fault classification. Therefore, the ability of LSTM for modeling long-term dependencies is verified. [32] presented a probabilistic long-term load forecasting model based on stacked LSTM network. Experimental results show that for load data of a power plant, the proposed stacked LSTM model performs better than other models including SVM and ANN (Back Propagation). [33] used a method for short-term load forecasting with multi-source data using GRU[2] neural networks. The GRU units introduced into the network for its simpler structure and faster convergence compared to LSTM units. The GRU method outperformed the other methods such as BPNNs, SAEs, RNNs and LSTM. In addition, LSTM network performed better than RNN, SAE and BPNN. Iranian studies of time series forecasting in the stock market have been conducted using various methods, but despite the high efficiency of deep learning methods, especially in dealing with big data problems, these methods have not been used for time series forecasting in the stock market to the best of our knowledge. [34] used deep neural networks for spatial-temporal prediction of monthly rainfall in north-west Iran. Results showed that deep belief network is capable of handling very large spatial-temporal data sets and is able to solve the complexities of forecasting precipitation, too.

1 Deep Belief Network
2 Gated Recurrent Unit

3- TIME SERIES FORECASTING USING DEEP LEAR-NING METHODS

Forecasting time series requires a recursive structure that can maintain the dependency between the information. It should be noted that in theory, standard RNNs can preserve dependencies, but unfortunately, due to the gradient vanishing problem, they do not seem to be able to learn “long-term” dependencies in practice. This problem was explored in depth in [35] and [36]. LSTM is a recurrent neural network that prevents back propagated errors from vanishing or exploding. Simple RNN repeating module consists of a single $tanh$ layer but LSTM network contains four layers that interact in a specific way to memorize selective features for a long time; therefore, it can preserve long-term dependencies. This method has been widely used to predict sequence problems. The structure of LSTM will be described below.

3-1- RNN and LSTM structure

RNN structure is consists of an input layer, one hidden layer, and an output layer. The states of the hidden layer are transmitted to the next one. Let x_t , h_t and y_t represent the input, hidden and output vectors at time t, respectively. The hidden and output vectors at sampling time t are calculated as follows:

$$X_t = \sigma(W_x h_{t-1} + U_x x_t + b_x) \tag{1}$$

$$y_t = \sigma(W_y h_{t-1} + b_y) \tag{2}$$

where

$x_t \in R^d$: input vector, $h_t \in R^h$: hidden state, $y_t \in R^h$: output vector, $\sigma \in R^h$: activation function, $W \in R^{h\times h}$ and $U \in R^{h\times d}$ weight matrices and $b \in R^h$: bias vectors.

LSTM repeating module is shown in Fig.1.

The key element in LSTM network is the cell state. In every step, new information can be added to it or some old information can be removed from it. In the first step, the information we want to delete is removed from the cell state by a sigmoid layer called the “forget gate”. (Given x_t and h_{t-1} , in the ultimate situation, removing all information is equal to 0 and preserving all information is equal to 1 for f_t) [38].

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f) \tag{3}$$

The second one is updating the cell state, which is a combination of the two steps including updating old information and adding new information.

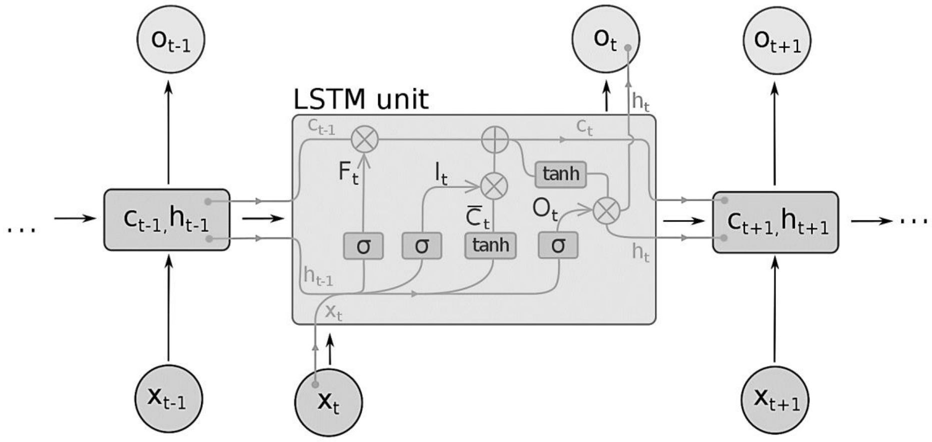

Fig. 1. LSTM unit [37].

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i) \tag{4}$$

$$\overline{C}_t = \tanh(W_c h_{t-1} + U_c x_t + b_c) \tag{5}$$

The new cell state is obtained by the Eq. (6):

$$C_t = f_t * C_{t-1} + i_t * \overline{C}_t \tag{6}$$

In the last step the output information is calculated by the following equations:

$$O_t = \sigma(W_o h_{t-1} + U_o x_t + b_o) \tag{7}$$

$$h_t = O_t * \tanh(C_t) \tag{8}$$

where

$x_t \in R^d$: input vector, $f_t \in R^h$: forget gate, $i_t \in R^h$: input gate, $o_t \in R^h$: output gate, $h_t \in R^h$: hidden state and output vector of the LSTM unit, $\bar{c}_t \in R^h$: cell input activation vector, $c_t \in R^h$: cell state, $W \in R^{h \times h}$ and $U \in R^{h \times d}$ weight matrices and $b \in R^h$: bias vectors.

Various types of LSTM structures have been introduced and implemented so far. Given that LSTMs operate on sequence data, by adding the number of layers the level of abstraction of input data over time increases.

4- SIMULATION

4-1- Data Normalization

Input data is extracted from tsetmc[1] website for 2000 daily closed prices and trading volumes of "Steel Mobarakeh" (one of 50 active companies) in the period of (2009/02/13 to 2020/05/25). The data normalization process helps to normalize the minimum and maximum values of the time series to a boundary. There are different methods for data normalization, the most appropriate and the most common for time series preprocessing is the log-return. The statistical properties of log-return are better behaved than the simple return [39]. Using this method, the price changes at each step compared to the previous step are calculated to make the network input. In addition, the network input will be in the proper range around zero.

$$X_{normalized} = \log\left(\frac{X(t)}{X(t-1)}\right) \tag{9}$$

In Eq. (9) $X(t)$ is the price in the day t.

4-2- Network Topology

In this paper, the performance of deep neural networks for long-term stock price prediction in Tehran Stock Exchange is investigated. For this purpose, two methods including LSTM network and deep RNN for simultaneous forecasting of h-step ahead (where 'h' is the prediction horizon and it is considered 10 in this study, arbitrarily) are compared. Indeed, the networks should have 10 outputs (**Fig.2.**). In problems such as stock price forecasting, several factors affect the prediction accuracy, some of which, despite being known, are ignored to reduce the complexity of the problem and make it solvable. Therefore, the output error contains useful information that can be used to help increase the accuracy of the prediction. According to this point, the output errors has been used as the input of the proposed model.

Our proposed architecture of multivariable neural network [22] is shown in **Fig.3.** for 10-step ahead simultaneous prediction of daily stock prices. That is, a deep neural network with multi outputs that each of the outputs represents the daily stock price of 1 to 10-step ahead predictions. The input vector consists of 4 external inputs including closed prices and trading volumes of two days ago ($X_{t-1}.X_t$ and $V_{t-1}.V_t$ respectively) plus 10 internal inputs ($E_{t+1}.E_{t+2}.\cdots.E_{t+10}$), which are defined as Eq. (10):

$$E_{t+i} = X_{t+i} - \hat{X}_{t+i} \quad , \quad i=1,\ldots,10 \tag{10}$$

1 http://www.tsetmc.com

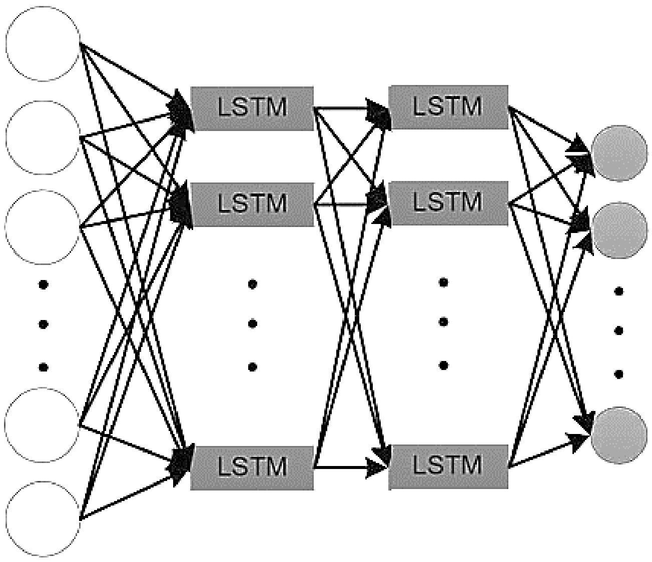

Fig. 2. 3-layer LSTM network

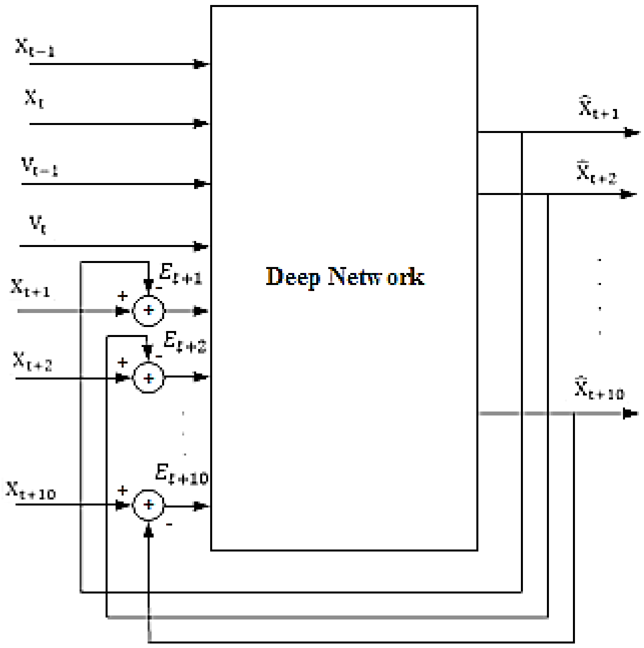

Fig. 3. Proposed model for multi-step ahead forecasting

In the above structure, $[X\ V\ E]^T$ is the input vector where X and V are normalized closed price and trading volumes of two days ago, and E is output errors of the next 10 closed price predictions of the previous step.

In order to investigate the effect of output error feedbacks on the network weights, can consider the simple structure of the two steps ahead simultaneous prediction using the prices and trading volumes of two days ago for a network with two layers, for example. Therefore, the formula for updating w_{11} and b_1, the weight and the bias of the x_1 in the first layer, will be as follows:

$$\Delta w_{11} = \eta_w \times \frac{x_1}{1+(w_{51}v_{11}+w_{61}v_{12})\times f_1^{'}\times f_2^{'}} \times f_1^{'} \times f_2^{'}(v_{11}e_1+v_{12}e_2) \quad (11)$$

$$\Delta b_1 = \eta_b \times \frac{1}{1+(w_{51}v_{11}+w_{61}v_{12})\times f_1^{'}\times f_2^{'}} \times f_1^{'} \times f_2^{'}(v_{11}e_1+v_{12}e_2) \quad (12)$$

Table. 1. Optimal Parameters and Structure

Method	RNN	LSTM
Num. of Layers	4	4
Num. of Neurons	400-100-50-10	400-100-50-10
Num. of epoch	150	200
Optimization algorithm	Adam	Adam
Activation Function	ELU-ELU-ELU-Linear	ELU-ELU-ELU-Linear
Loss Function	MSE	MSE

Where w_{51} and w_{61} are the weights of the first layer for the output error feedbacks (e_1 and e_2, respectively) and v_{11} and v_{12} are the weights of the second layer. It is observed that the output errors and their related weights affect the value of the weight and the bias of the external input, x_1. Therefore, the network tends to learn error dynamics during training phase and as a result, the efficiency of learning enhances and the value of error becomes close to zero.

After the training phase (when the network weights and biases were determined and fixed), output error feedbacks are considered zero in validation and test procedures, because these output errors are not available. However, due to the bias, the outputs related to these inputs will be non-zero. Eliminating feedback errors, on the other hand, removes their effect on the output, while if these errors are considered zero at the input, they still affect the output because the bias is non-zero.

After normalizing the extracted data, 65% of all data is considered as the training set and 35% as the test set. RNN and LSTM structures have been implemented using Python programming language. Finding a good fit is one of the central problems in machine learning. Some of the most common issues in machine learning are over-fitting and under-fitting. In this research, RNN and LSTM structures have three hidden layers with ELU[1] activation function (Eq. 13). After several running and obtaining the least error, the optimal number of neurons are obtained. Adam is an adaptive learning rate optimization algorithm that's been designed specifically for training deep neural networks and first published in 2014 []. Adam can be looked at as a combination of RMSprop[2] and SGD[3] with momentum. It uses the squared gradients to scale the learning rate like RMSprop and it takes advantage of momentum by using moving average of the gradient instead of the gradient itself like SGD with momentum. Adam is used in both LSTM and RNN as the optimization algorithm. $tanh$ is used as the activation function, since we have normalized the dataset using log-return. Table 1 shows the optimal structures and parameters of the above networks.

1 Exponential Linear Unit is a function that tend to converge cost to zero faster and produce more accurate results. ELU is very similar to RELU except negative inputs.

2 Root Mean Square Propagation. ELU is very similar to RELU except negative inputs.

3 Stochastic Gradient Descent

$$\text{ELU: } \begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases} \tag{13}$$

4-3- Simulation Results

To demonstrate the capability of deep learning methods for stock price forecasting in Tehran Stock Exchange, the proposed method is used to forecast daily closed prices of "Steel Mobarakeh" as one of the 50 active companies in this market. Simulation results of forecasting the next 10 closed prices are shown in the following figures. It should be noted that to avoid showing too many figures, only the simulation results for one and ten days ahead forecasting are given below. **Fig.4.** shows 2000 daily closed prices and Figs. 5-8 show one and ten steps ahead price time series predictions.

Fig. 5. and 6 show 1-step ahead price forecasts using LSTM and RNN methods. The train and test data are plotted after training the network. The error at the beginning of the train chart is due to a large jump in the real prices. Comparing the predicted price time series with the real prices, it can be seen that both methods achieved very good results for short-term predictions.

According to Figs. 7 and 8 for 10 days ahead stock price forecasting using the proposed method, it can be found that both methods have a good efficiency for long-term predictions, but it is obvious by comparing these diagrams that the LSTM network has better prediction performance than the RNN method in both train and test procedures.

To demonstrate the advantage of LSTM method over RNN, Table 2 illustrates the performance of both methods, in terms of MAE for train and test procedures. Indeed, normalized data using log-return method are given to the proposed model as input and real prices are considered as targets. Then the proposed model with output error feedback for LSTM and RNN are employed to forecast the stock prices for 10-step ahead, simultaneously. For performance comparison, MAE values have been computed for all predicted days using Eq. 14.

$$MAE = \frac{\sum_{i=1}^{N} | y_i - \overline{y}_i |}{N} \tag{14}$$

where

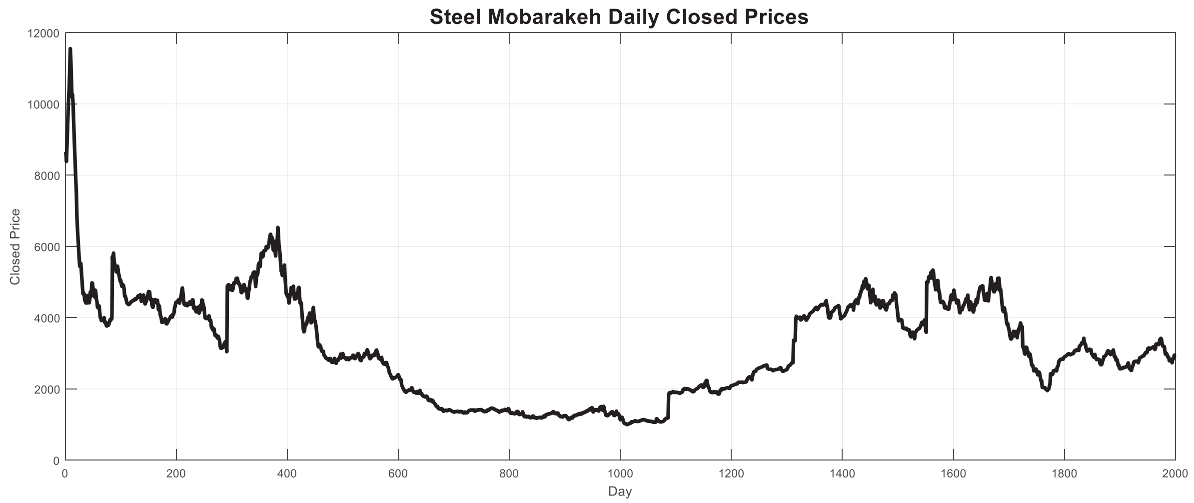

Fig. 4. “Steel Mobarakeh” daily closed prices

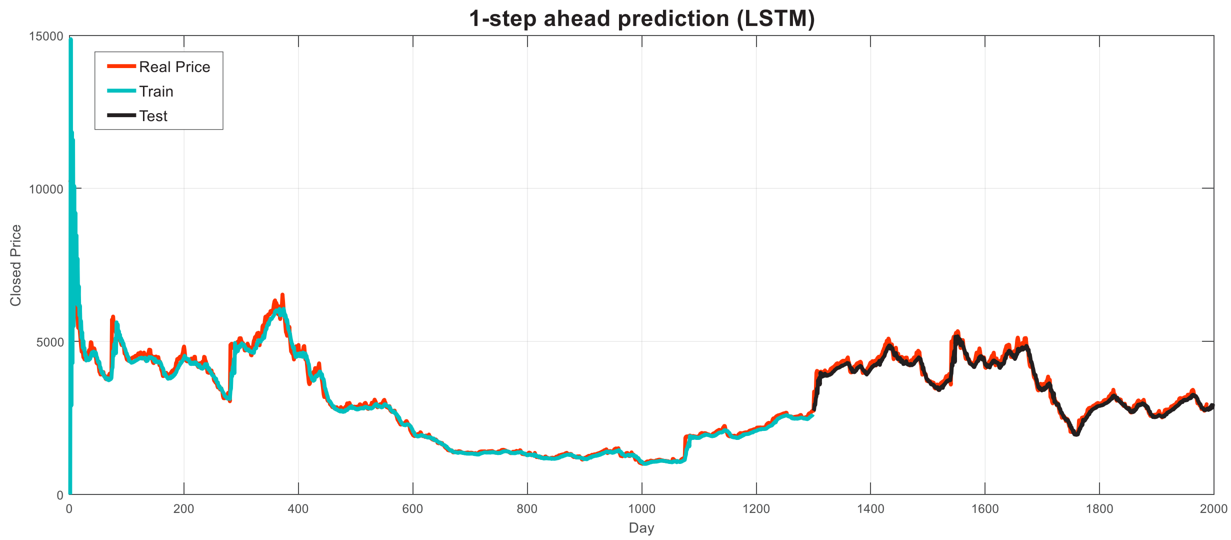

Fig. 5. 1-step ahead prediction of daily closed price - LSTM method

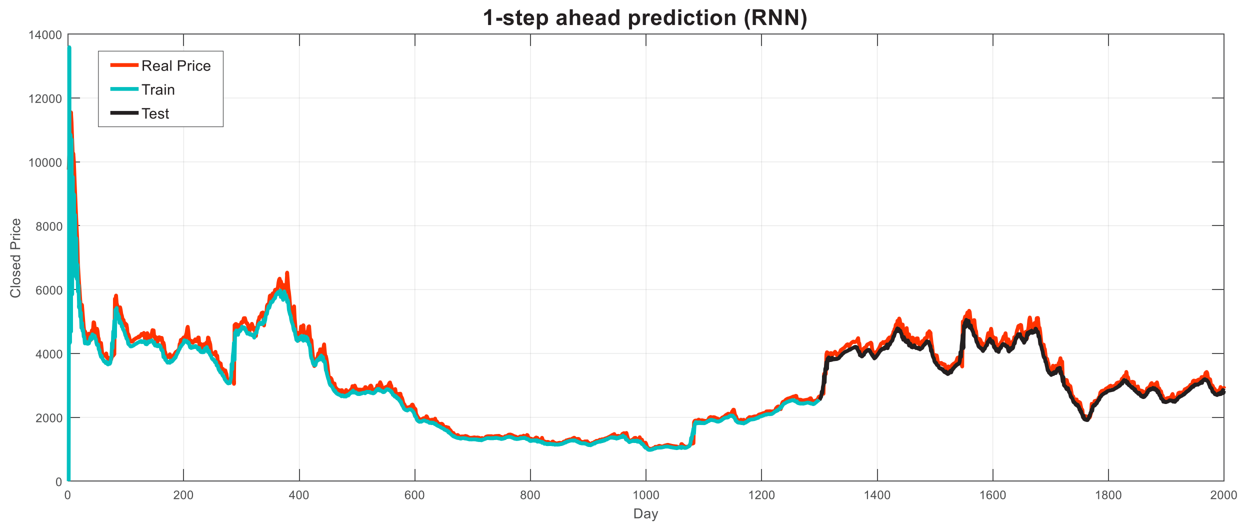

Fig. 6. 1-step ahead prediction of daily closed price - RNN method

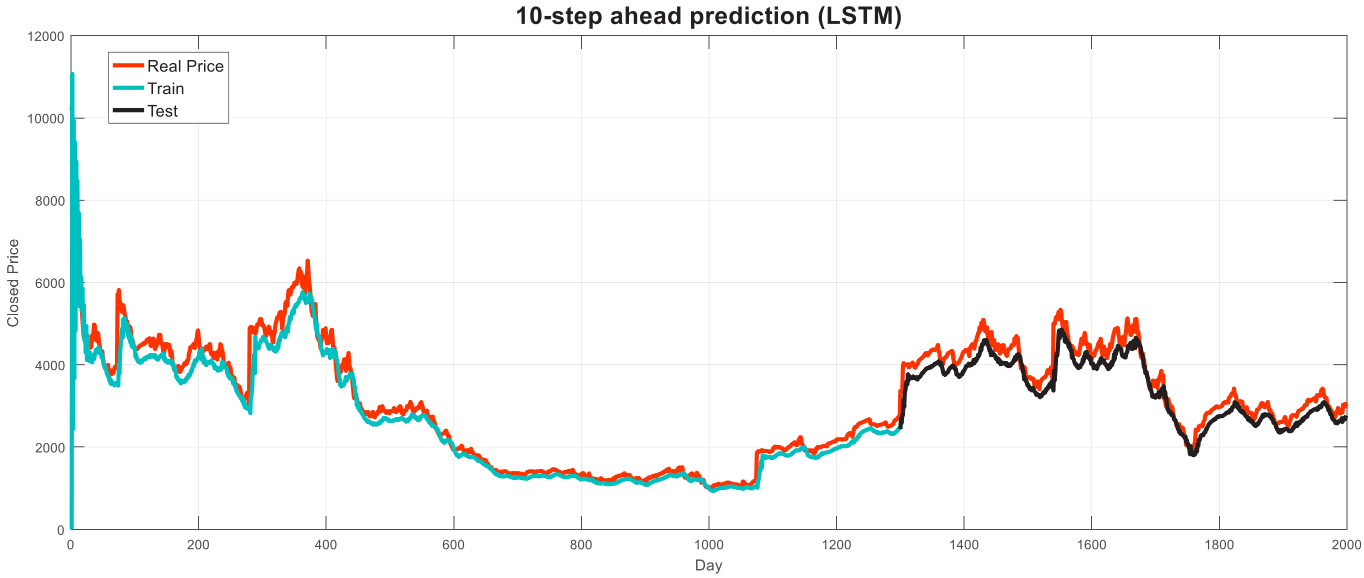

Fig. 7. 10-step ahead prediction of daily closed price - LSTM method

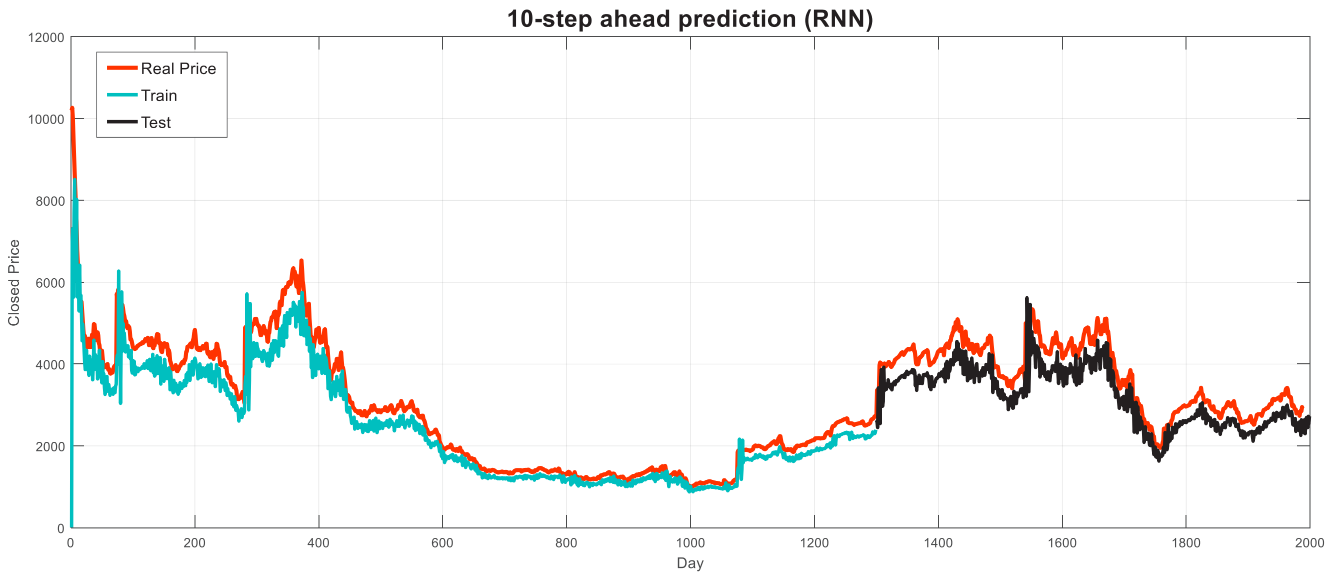

Fig. 8. 10-step ahead prediction of daily closed price RNN method

y_i is the real price and $\overline{y}_i$ is the predicted price and N is the total number of data.

We can see in the Table 2 that the accuracy of LSTM method is better than the RNN in general for MAE performance measure. For example, comparing the results of the tenth column show that the values of MAE for LSTM method are less than RNN for 10-step ahead forecast.

5- CONCLUSION AND FUTURE WORKS

This study was undertaken to investigate multi steps ahead prediction of price time series in Tehran Stock Exchange. To predict the future behavior of a process, we need to model its dynamics. Therefore, a multivariable structure with multi outputs was proposed for deep recurrent neural networks. The simulation results confirm that both methods have a very good performance for short-term predictions. In addition, with the feature added to the standard forecasting method, which is based on learning the dynamics of prediction error, the results show that for long-term forecasting, both methods have good performance and also the LSTM method has better prediction accuracy compared to the RNN generally. It is due to the ability of the LSTM network in memorizing long-term dependencies to predict time series and sequence problems. Based on the results, it is highly recommended to use deep recurrent methods and especially LSTM as well as its variant structures when there is long-term dependence in the dataset, to predict time series. Moreover, the proposed method can

Table. 2. Prediction errors for next 10 days closed prices using proposed model

Method		Day	1	2	3	4	5	6	7	8	9	10
RNN	MAE	train	1.49702e+02	2.14125e+02	2.89362e+02	2.98186e+02	3.96672e+02	5.84313e+02	7.75981e+02	8.84313e+02	1.03014e+03	1.18485e+03
		test	1.62239e+02	2.64686e+02	3.78899e+02	3.46396e+02	4.88010e+02	7.57495e+02	1.03769e+03	1.13975e+03	1.29878e+03	1.51252e+03
LSTM	MAE	train	1.56031e+02	1.71331e+02	2.07731e+02	2.43644e+02	3.63975e+02	4.71709e+02	5.87291e+02	7.144012e+02	9.26824e+02	1.00275e+03
		test	1.60947e+02	1.823845e+02	2.41883e+02	3.02717e+02	4.25423e+02	5.48373e+02	7.53933e+02	9.01927e+02	1.17495e+03	1.28485e+03

be exploited in any fields where the long-term prediction of time series is needed. Also because of the capability of deep learning methods in dealing with big data and extracting high-level features, other kinds of financial data such as news texts can be used as input; to increase the prediction accuracy of financial markets.

REFERENCE

[1] Rajabi, M., & Khaloozadeh, H. (2018, May). Investigation and Comparison of the Performance of Multi-Objective Evolutionary Algorithms Based on Decomposition and Dominance in Portfolio Optimization. In *Electrical Engineering (ICEE), Iranian Conference on* (pp. 923-929). IEEE.

[2] Wang, J. Z., Wang, J. J., Zhang, Z. G., & Guo, S. P. (2011). Forecasting stock indices with back propagation neural network. *Expert Systems with Applications*, *38*(11), 14346-14355.

[3] Shah, D., Isah, H., & Zulkernine, F. (2019). Stock Market Analysis: A Review and Taxonomy of Prediction Techniques. *International Journal of Financial Studies*, *7*(2), 26.

[4] Martinez, L. C., da Hora, D. N., Palotti, J. R. D. M., Meira, W., & Pappa, G. L. (2009, June). From an artificial neural network to a stock market day-trading system: A case study on the bm&f bovespa. In *2009 International Joint Conference on Neural Networks* (pp. 2006-2013). IEEE.

[5] Chen, J. (2010, October). SVM application of financial time series forecasting using empirical technical indicators. In *2010 International Conference on Information, Networking and Automation (ICINA)* (Vol. 1, pp. V1-77). IEEE.

[6] Barak, S., Arjmand, A., & Ortobelli, S. (2017). Fusion of multiple diverse predictors in stock market. *Information Fusion*, *36*, 90-102.

[7] Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, *259*(2), 689-702.

[8] Gheyas, I. A., & Smith, L. S. (2011). A novel neural network ensemble architecture for time series forecasting. *Neurocomputing*, *74*(18), 3855-3864.

[9] Kourentzes, N., Barrow, D. K., & Crone, S. F. (2014). Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, *41*(9), 4235-4244.

[10] Zhu, B., & Chevallier, J. (2017). Carbon price forecasting with a hybrid Arima and least squares support vector machines methodology. In *Pricing and Forecasting Carbon Markets* (pp. 87-107). Springer, Cham.

[11] Nayak, R. K., Mishra, D., & Rath, A. K. (2015). A Naïve SVM-KNN based stock market trend reversal analysis for Indian benchmark indices. *Applied Soft Computing*, *35*, 670-680.

[12] Iqbal, Z., Ilyas, R., Shahzad, W., Mahmood, Z., & Anjum, J. (2013). Efficient machine learning techniques for stock market prediction. *International Journal of Engineering Research and Applications*, *3*(6), 855-867.

[13] Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1798-1828.

[14] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, *313*(5786), 504-507.

[15] Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep machine learning-a new frontier in artificial intelligence research. *IEEE computational intelligence magazine*, *5*(4), 13-18.

[16] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

[17] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[18] Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, *18*(7), 1527-1554.

[19] Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning* (pp. 1096-1103). ACM.

[20] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

[21] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

[22] Khaloozadeh, H., & Sedigh, A. K. (2001, July). Long term prediction of Tehran price index (TEPIX) using neural networks. In *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)* (Vol. 1, pp. 563-567). IEEE.

[23] Siami-Namini, S., & Namin, A. S. (2018). Forecasting economics and financial time series: Arima vs. lstm. *arXiv preprint arXiv:1803.06386*.

[24] Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017, May). Stock market's price movement prediction with LSTM neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)* (pp. 1419-1426). IEEE.

[25] Althelaya, K. A., El-Alfy, E. S. M., & Mohammed, S. (2018, April). Evaluation of bidirectional lstm for short-and long-term stock market prediction. In *2018 9th international conference on information and communication systems (ICICS)* (pp. 151-156). IEEE.

[26] Singh, R., & Srivastava, S. (2017). Stock prediction using deep learning. *Multimedia Tools and Applications*, *76*(18), 18569-18584.

[27] Shen, F., Chao, J., & Zhao, J. (2015). Forecasting exchange rate using deep belief networks and conjugate gradient method. *Neurocomputing*, *167*, 243-253.

[28] Kuremoto, T., Kimura, S., Kobayashi, K., & Obayashi, M. (2014). Time series forecasting using a deep belief network with restricted Boltzmann machines. *Neurocomputing*, *137*, 47-56.

[29] Takeuchi, L., & Lee, Y. Y. A. (2013). Applying deep learning to enhance momentum trading strategies in stocks. In *Technical Report*. Stanford University.

[30] Li, X., Peng, L., Yao, X., Cui, S., Hu, Y., You, C., & Chi, T. (2017). Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environmental pollution*, *231*, 997-1004.

[31] Lei, J., Liu, C., & Jiang, D. (2019). Fault diagnosis of wind turbine based on Long Short-term memory networks. *Renewable energy*, *133*, 422-432.

[32] Yuan, C., Xiu, T., & Lou, T. (2019, April). Probabilistic Long-term Load Forecasting Based on Stacked LSTM. In *Proceedings of the 2019 4th International Conference on Mathematics and Artificial Intelligence* (pp. 80-84).

[33] Wang, Y., Liu, M., Bao, Z., & Zhang, S. (2018). Short-term load forecasting with multi-source data using gated recurrent unit neural networks. *Energies*, *11*(5), 1138.

[34] Farnaghi, M & Rahimi, H. (2017). Spatio-Temporal Prediction of Monthly Rainfall using Deep Neural Network: A Case Study in North-west Iran. *JGST*. *6(4)*, 123-142. *(in Persian)*

[35] Hochreiter, S. (1991). Untersuchungen zu dynamischen neuronalen Netzen. *Diploma, Technische Universität München*, *91*(1).

[36] Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), 157-166.

[37] Luu, Q., Nguyen, S., & Pham, U. (2020). Time series prediction: A combination of Long Short-Term Memory and structural time series models. *Science & Technology Development Journal-Economics-Law and Management*, *4*(1), 500-515.

[38] Yuan, X., Li, L., & Wang, Y. (2019). Nonlinear dynamic soft sensor modeling with supervised long short-term memory network. *IEEE Transactions on Industrial Informatics*.

[39] Quigley, L., & Ramsey, D. (2008). Statistical analysis of the log returns of financial assets. *Financial mathematic, University of Limerick*, *32*.

[40] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

HOW TO CITE THIS ARTICLE

M. Rajabi, H. Khaloozadeh, Long-term prediction in Tehran stock market using a new architecture of Deep neural networks, AUT J. Model. Simul., 52(2) (2020) 179-188.

DOI: 10.22060/miscj.2021.17712.5191