ICIC Express Letters
Volume **16**, Number **2**, February **2022**

ICIC International ©2022 ISSN 1881-803X
pp. 187–194

STOCK PRICE MOVEMENT PREDICTION BASED ON RE-EXTRACT FEATURE LSTM

Yongli Zhang, Sin Hong Tan, Jaekyung Yang, Taejin Kim and Joonsoo Bae*

Department of Industrial and Information Systems Engineering
Jeonbuk National University
No. 567, Baekje-daero, Deokjin-gu, Jeonju-si, Jeollabuk-do 54896, Korea
{ 202055385; sinhong96; jkyang; tjkim }@jbnu.ac.kr
*Corresponding author: jsbae@jbnu.ac.kr

Received June 2021; accepted August 2021

Abstract. *As one of the most important stock prediction methods, the Long Short-Term Memory (LSTM) does produce good results. However, the pursuit of higher accuracy has always been the goal of scholars. In order to obtain a more precise result than LSTM, this paper presents a Re-Extract Feature LSTM (RE-LSTM) for the stock price movement prediction based on LSTM, convolution and max-pooling operation. Firstly, the LSTM layer takes a group of stock data as training input and produces the cell state and hidden state. Then, the convolutional layer and the max-pooling layer are employed to re-extract the features of the hidden state. Ultimately, an LSTM layer is executed again to achieve the stock price movement prediction. The results of experiments on stock index price of China Securities 300 Index (CSI300) and Korean Composite Stock Price Index (KOSPI) show that RE-LSTM outperforms naive LSTM and achieves excellent result.*
Keywords: Stock price prediction, Time series, LSTM, Convolution, Max-pooling

1. **Introduction.** The modern stock market is rife with instability, chaos, and complexity and dynamic, so the stock price movement prediction has emerged as a critical technique for ensuring the lucrative return investment. The investor received lots of help from candlestick chart [1-3]; however, with the increasingly advanced research of Deep Learning (DL) technique, the DL is commonly applied in the natural language processing [4,5], image recognition [6], medical predictions [7], time-series data analysis and prediction [8], and more. In [9], Hu et al. indicated the trend of using deep-learning-based methods for financial modeling is rising exponentially. Furthermore, the available and efficient deep learning algorithms [10-20] have been proven to yield better accuracy and prediction results.

Different from the traditional linear and statistical [21] models, the DL methods can handle a dynamic, nonlinear, non-stable, noisy financial time series. [10] used a deep convolutional neural network to model the combined influence of long-term events and short-term events on stock. Singh and Srivastava [11] adopted the $(2D)^2$PCA+ Deep Neural Network (DNN) method to demonstrate that deep learning can improve stock market forecasting accuracy. [12] proposed a method of systematic analysis by using deep learning networks to extract features from large amounts of raw data without relying on the prior knowledge of the forecaster. In [13], Rout et al. put forward a low complexity recurrent functional link artificial neural network. In order to address the overfitting limitation, [14] introduced a data augmentation approach for stock market index forecasting through their ModAugNet framework. In [15], Cao et al. combined the Empirical Mode Decomposition (EMD) with the LSTM to get a hybrid forecasting model which results in

DOI: 10.24507/icicel.16.02.187

the improvement of the prediction accuracy. [16] used three different DL architectures to predict the stock prices and compared their performance. In [18], the authors proposed a multivariate time series prediction method. [19] expounded a model based on ensemble EMD and LSTM to overcome the disadvantage. Rikukawa et al. used the stock price of the predicted stock brand to predict the stock price based on recurrent neural network in [20]. Thus, we find stock prediction methods have received more and more attention from research institutions and scholars. With the deepening of research, the prediction has been gradually improved.

According to the above-mentioned studies, whether it is LSTM or other combination algorithms, the input is fed into LSTM layer that contains cell state, input gate, output gate and forget gate to produce the new hidden state and all of these methods get some ideal results. However, algorithms that are combined or improved based on LSTM did not consider the extraction of the features from hidden state which is produced by LSTM layer. As we all know, the convolutional layer can extract a high-level feature, the max-pooling layer can reduce the overfitting and improve the fault tolerance of the model.

In order to solve problem that the feature of hidden state is not being extracted, the convolutional layer and the max-pooling layer are introduced into the LSTM. Therefore, we herein propose a method which employs the convolutional layer and the max-pooling layer to extract the features from hidden state which is produced by LSTM layer and max-pooling layer to reduce the overfitting. Furthermore, this paper stacks LSTM layer, convolutional layer and the max-pooling layer to construct a new network structure to enhance the prediction accuracy.

In this study, we conduct two sets of experiments to demonstrate the effectiveness of the proposed method. The results of experiments on stock index price of CSI300 and KOSPI show that RE-LSTM outperforms naive LSTM and achieves the excellent result in NMSE.

The rest of this paper is organized as follows. In Section 2, we introduce preliminary; the model is presented at this section. Section 3 is methodology; we describe RE-LSTM in detail. In order to demonstrate our method, Section 4 presents the results of experiment using the data from CSI300 and KOSPI. Finally, we conclude this paper in Section 5.

2. **Preliminary.** In fact, the stock price movement prediction is a time series prediction problem. Inspired by [18], we suppose that the stock information of some company on former T trading day is training series $X = (X_1, X_2, \dots, X_T)$, $X_t = \left(x_t^1, x_t^2, \dots, x_t^L\right) \subset X$ and the time-step is defined as L. Meanwhile, the corresponding target values are given $y = (y_1, y_2, \dots, y_{T-1})$. In most of the researches, the aim of the researches is to predict the target value of the T-th day. Furthermore, a non-linear mapping function will be constructed through former T trading day, we supposed X is training series and y is corresponding target values, namely: $\tilde{y}_T = f(X, y)$ is constructed, where the $\tilde{y}_T$ is the prediction value of the T-th day, and the $f(\cdot)$ is a mapping which will be learned.

3. **Approach.** In this section, we will introduce our method. Based on the powerful ability, in which convolutional layer extracts a high-level feature and max-pooling layer can reduce the overfitting and improve the fault tolerance of the model, the convolutional layer and the max-pooling layer are employed to re-extract the features of the hidden state.

3.1. **Long short-term memory.** The LSTM is a special Recurrent Neural Network (RNN), which is being commonly used in many applications. Compared to the RNN, LSTM increased cell state, input gate, output gate and forget gate; thus, the problems that effective historical information cannot be preserved for a long time, can be well solved.

The data with sequential nature can be dealt with the LSTM layer by using the cell state C_t in addition to the hidden state h_t, input gate, output gate and forget gate.

The $X_t = \left(x_t^1, x_t^2, \ldots, x_t^L\right)$, h_{t-1} and C_{t-1} are propagated to the LSTM layer, conduct

$$f_t = \sigma(W_f[h_{t-1}, X_t] + b_f),$$

then, in the input gate, we calculate

$$\xi_t = than(W_\xi[h_{t-1}, X_t] + b_\xi),$$
$$j_t = \sigma(W_j[h_{t-1}, X_t] + b_j),$$

next, in the cell state, we compute

$$C_t = f_t C_{t-1} + j_t \xi_t,$$

at last, with the input, cell state, and previous hidden state, in the output gate, the network will execute

$$o_t = \sigma(W_o[h_{t-1}, X_t] + b_o),$$
$$h_t = o_t than(C_t),$$

where W, b are the corresponding parameters to be learnt, j_t represents the input gate state, f_t is forget gate state, C_t is the cell state, o_t is the output gate and h_t is the hidden layer output in current time-step.

3.2. **Re-extract feature long short-term memory.** In order to obtain a more precise result than LSTM, a stacked convolutional layer and maxpool layer operation is carried out due to the fact that convolutional layer extracts a high-level feature and max-pooling layer can reduce the overfitting and improve the fault tolerance of the model.

For the hidden state h_t, we execute the convolution and maxpool operation

$$h_t' = (h_t * K')_i = \varphi'\left(\sum K_m' \cdot (h_t) + b'\right),$$
$$h_t' = \max \mathrm{pool}(h_t').$$

Next, we conduct the $LSTM(X_t', h_t', C_t)$ and update

$$\tilde{y}_t = h_t'.$$

So, the final output vector is $\tilde{y}_T = (\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_t)$.

The architecture of the RE-LSTM is shown in detail as Figure 1. Firstly, the LSTM layer takes a group of stock data as training input and produces the cell state and hidden state. Then, the convolutional layer and the max-pooling layer are employed to re-extract the features of hidden state. Ultimately, an LSTM layer is executed again to achieve the stock price movement prediction. The B is the mini-batch size, I is the input size, O is the output size and H is the hidden size in Figure 1.

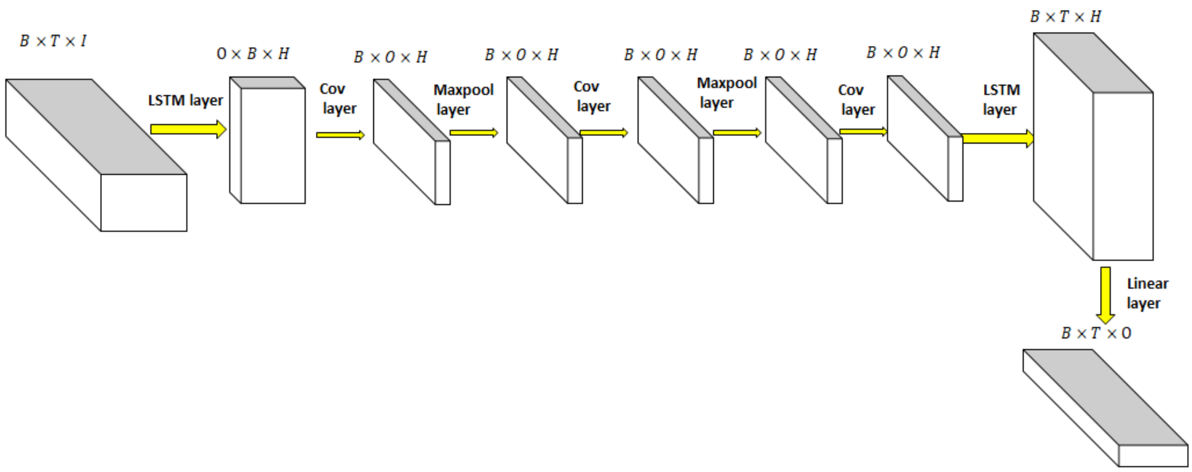

Figure 1. The network structure of RE-LSTM

4. **Experiment.** In this section, we present our experiments by using the CSI300 and KOSPI stock data. To evaluate the performance of the model, we randomly select six listed companies from the CSI300 and KOSPI respectively, the companies of AKAM, CERN, AIG, ZBH, KMB and EQT come from CSI300, and the companies of HD, SK, POSCO, JW, LOT and JEJUB come from KOSPI, and the names of the companies in this paper are pseudonyms in order to prevent the disclosure of any business information. The details of the data are shown in Table 1. It is worth noting that the KOSPI data are obtained in the form of web crawlers.

Table 1. Prediction results of RE-LSTM and LSTM on CSI300

Company name	*The number of trading days*	*The attributes contained in the data*
AKAM	1258	OPEN, HIGH, LOW, CLOSE, VOL
CERN	1258	OPEN, HIGH, LOW, CLOSE, VOL
AIG	1257	OPEN, HIGH, LOW, CLOSE, VOL
ZBH	1257	OPEN, HIGH, LOW, CLOSE, VOL
KMB	1258	OPEN, HIGH, LOW, CLOSE, VOL
EQT	1257	OPEN, HIGH, LOW, CLOSE, VOL
HD	2924	OPEN, HIGH, LOW, CLOSE, DIFF, VOL
SK	3200	OPEN, HIGH, LOW, CLOSE, DIFF, VOL
POSCO	2924	OPEN, HIGH, LOW, CLOSE, DIFF, VOL
JW	2898	OPEN, HIGH, LOW, CLOSE, DIFF, VOL
LOT	2949	OPEN, HIGH, LOW, CLOSE, DIFF, VOL
JEJUB	2859	OPEN, HIGH, LOW, CLOSE, DIFF, VOL

In Table 1, we can intuitively notice that 12 companies come from CSI300 and KOSPI respectively, containing varying quantities trading day information. The OPEN is the opening price of the stock; HIGH is the stock highest price of the day; LOW is the stock lowest price of the day; CLOSE is the closing price of the stock; VOL is total number of transactions and DIFF is the difference between highest price and lowest price of the day.

The Normalization Mean Squared Error (NMSE) [12,22] is used as the evaluation indicator. The formula of NMSE is indicated as the following:

$$NMSE = \frac{1}{N}\sum_{i=1}^{N}\left(\tilde{y}_T^i - y_T^i\right)^2 \Big/ \operatorname{var}\left(y_T^i\right),$$

where $\tilde{y}_T^i$ is prediction, y_T^i is real value, N is the number of testing samples, and the $\operatorname{var}(\cdot)$ is the variance.

In the designed experiment, we will predict the movements of recent 40 days HIGH (the stock highest price of the day) and LOW (stock lowest price of the day) by using the remaining attributes (OPEN, CLOSE, DIFF, VOL). Furthermore, the proposed model is compared with naive LSTM for the evaluation of the performance. For consistency of comparison, all the used parameters are the same in both RE-LSTM and LSTM. In the LSTM layer, the hidden size is 128, dropout rate is 0.2 and time step is 40, in the Conv layer, the kernel size is 5, the input and output size follow LSTM layer, padding is 2 and bias is Ture, max pooling kernel and padding are respectively 5 and 2, the training batch size is 64, learning rate is 0.001 and epoch is 40. In this designed experiment, the last 40 days' data are used for test dataset, while the remaining data will be treated as the training dataset. The validation process is important as it can prevent the overfitting; therefore, 15% of training data are used for validation dataset in this experiment.

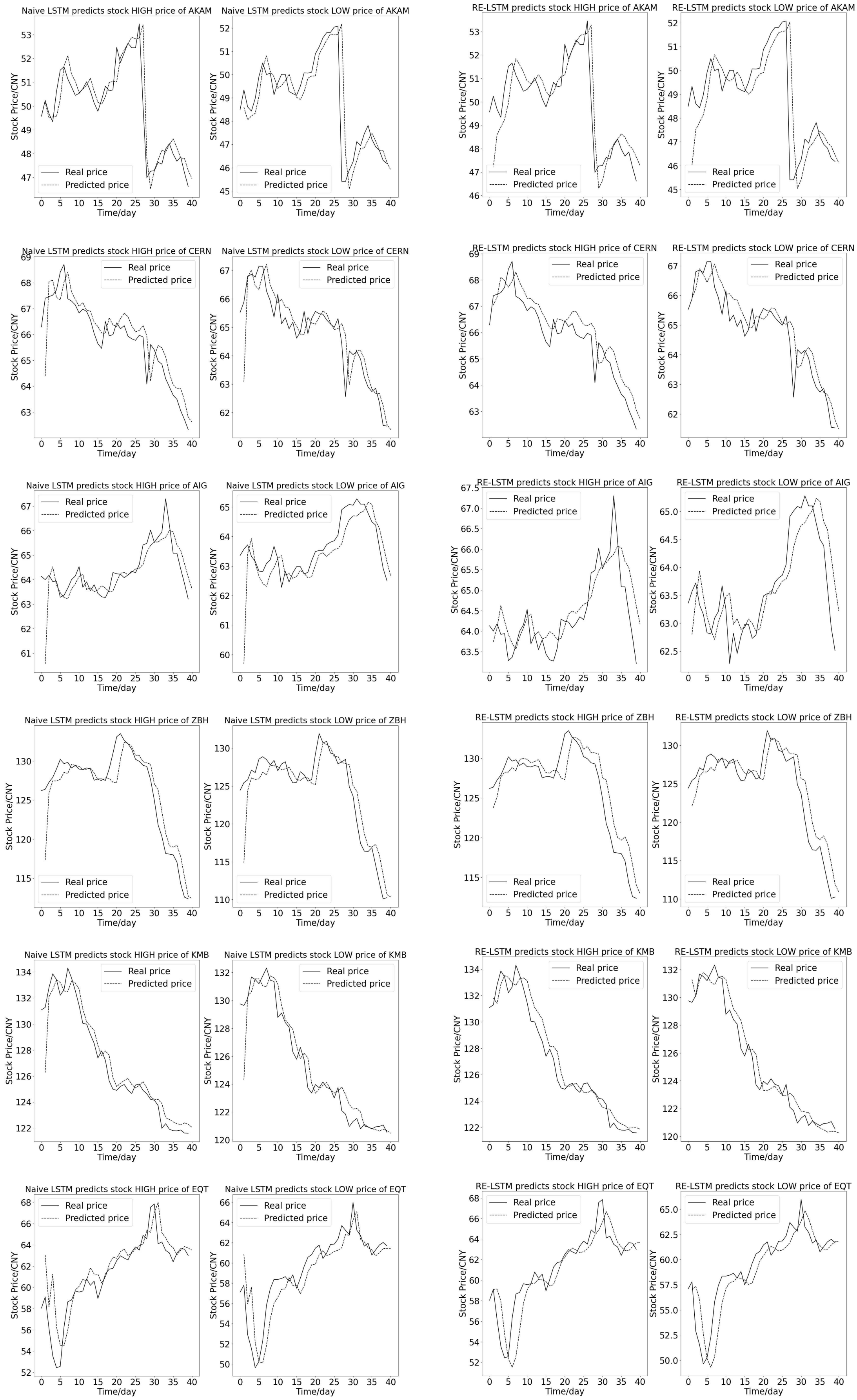

Figure 2. Predicted HIGH and LOW movements by RE-LSTM and LSTM on CSI300

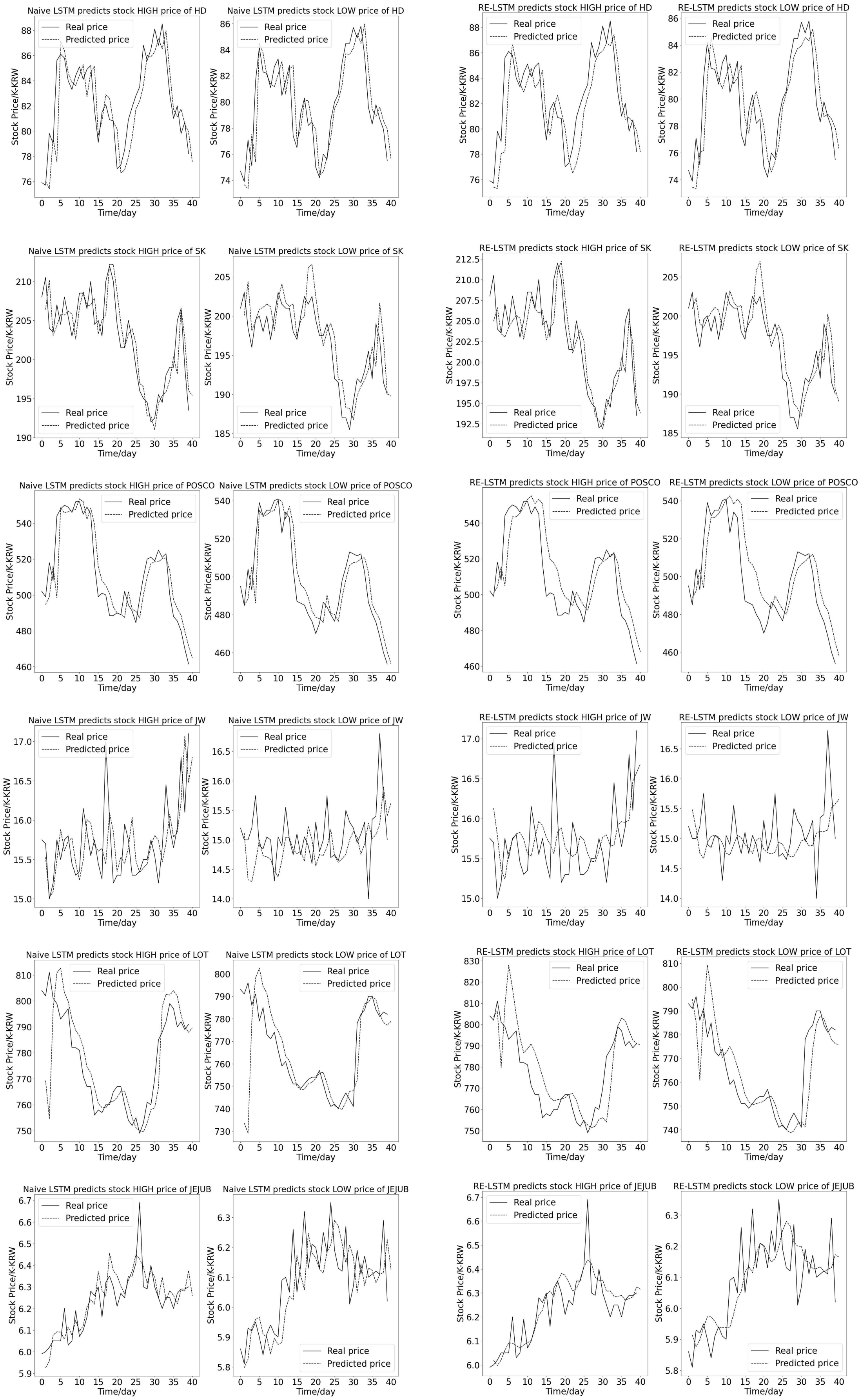

Figure 3. Predicted HIGH and LOW movements by RE-LSTM and LSTM on KOSPI

Table 2 illustrates the NMSE values of RE-LSTM and LSTM on CSI300. We can see that the RE-LSTM again outperforms the naive LSTM in the experiments of CERN, AIG, ZBH, KMB and EQT. Meanwhile, Figure 2 demonstrates recent 40 days stock HIGH and LOW price movements of AKAM, CERN, AIG, ZBH, KMB and EQT from CSI300, which are predicted by naive LSTM and RE-LSTM. Based on the results, in most cases, the RE-LSTM is more superior compared to naive LSTM.

Table 2. The NMSE values of RE-LSTM and LSTM on CSI300

Company name	*RE-LSTM*		*LSTM*	
	HIGH	*LOW*	*HIGH*	*LOW*
AKAM	0.0080	0.0152	0.0048	0.0120
CERN	0.0055	0.0048	0.0079	0.0069
AIG	0.0041	0.0035	0.0069	0.0080
ZBH	0.0148	0.0150	0.0174	0.0194
KMB	0.0045	0.0045	0.0065	0.0078
EQT	0.0172	0.0179	0.0174	0.0159

The RE-LSTM and naive LSTM prediction results for KOSPI are presented in Table 3. Although in most of the cases, RE-LSTM is more superior compared to naive LSTM, there are minor cases where RE-LSTM shows inferior result. It suggested that RE-LSTM is still limited to a small part of the data set, which will be the direction of our future efforts. Meanwhile, Figure 3 shows that recent 40 days stock HIGH and LOW price movements of HD, SK, POSCO, JW, LOT and JEJUB from KOSPI.

Table 3. The NMSE values of RE-LSTM and LSTM on KOSPI

Company name	*RE-LSTM*		*LSTM*	
	HIGH	*LOW*	*HIGH*	*LOW*
HD	0.0064	0.0059	0.0059	0.0055
SK	0.0031	0.0041	0.0034	0.0046
POSCO	0.0197	0.0201	0.0126	0.0111
JW	0.0011	0.0018	0.0011	0.0020
JEJUB	0.0014	0.0026	0.0015	0.0036
LOT	0.0003	0.0003	0.0004	0.0007

5. Conclusions.

In this work we proposed a model named RE-LSTM for the stock price prediction, which is based on the convolution, max-pooling and LSTM. The training data are propagated into LSTM layer and produce cell state and hidden state, the employed convolutional layer and max-pooling layer re-extract the feature for hidden state, at last, they are fed into the LSTM to finish the prediction. Ultimately, the results of experiments indicate RE-LSTM is superior compared to naive LSTM in most of the cases. Meanwhile, investigation of the reason behind getting poor result in minor cases is the next research's direction. Furthermore, the proposed RE-LSTM may be applied in the other time series prediction problem and we plan to apply it in the Remaining Useful Life (RUL) prediction of rolling bearings in mechanical engineering.

Acknowledgment. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1F1A1064460).

REFERENCES

[1] C. F. Tsai and Z. Y. Quan, Stock prediction by searching for similarities in candlestick charts, *ACM Transactions on Management Information Systems*, vol.5, no.2, pp.1-21, 2014.

[2] M. Ananthi and K. Vijayakumar, Stock market analysis using candlestick regression and market trend prediction (CKRM), *Journal of Ambient Intelligence and Humanized Computing*, vol.12, no.5, pp.4819-4826, 2021.

[3] H. Wang, W. Huang and S. Wang, Forecasting open-high-low-close data contained in candlestick chart, *arXiv Preprint*, arXiv: 2104.00581, 2021.

[4] R. Collobert and J. Weston, A unified architecture for natural language processing: Deep neural networks with multitask learning, *Proc. of the 25th International Conference on Machine Learning*, pp.160-167, 2008.

[5] Y. Goldberg, Neural network methods for natural language processing, *Synthesis Lectures on Human Language Technologies*, vol.10, no.1, 2017.

[6] G. Koch, R. Zemel and R. Salakhutdinov, Siamese neural networks for one-shot image recognition, *Proc. of the 32nd International Conference on Machine Learning*, Lille, France, 2015.

[7] F. Milletari, N. Navab and S. A. Ahmadi, V-Net: Fully convolutional neural networks for volumetric medical image segmentation, *2016 4th International Conference on 3D Vision (3DV)*, pp.565-571, 2016.

[8] L. Wang, Y. Zeng and T. Chen, Back propagation neural network with adaptive differential evolution algorithm for time series forecasting, *Expert Systems with Applications*, vol.42, no.2, pp.855-863, 2015.

[9] Z. Hu, Y. Zhao and M. Khushi, A survey of forex and stock price prediction using deep learning, *Applied System Innovation*, vol.4, no.1, pp.1-30, 2021.

[10] X. Ding, Y. Zhang, T. Liu et al., Deep learning for event-driven stock prediction, *The 24th International Joint Conference on Artificial Intelligence*, Buenos Aires, Argentina, pp.2327-2333, 2015.

[11] R. Singh and S. Srivastava, Stock prediction using deep learning, *Multimedia Tools and Applications*, vol.76, no.18, pp.18569-18584, 2017.

[12] E. Chong, C. Han and F. C. Park, Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies, *Expert Systems with Applications*, vol.83, pp.187-205, 2017.

[13] A. K. Rout, P. K. Dash, R. Dash et al., Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach, *Journal of King Saud University – Computer and Information Sciences*, vol.29, no.4, pp.536-552, 2017.

[14] Y. Baek and H. Y. Kim, ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module, *Expert Systems with Applications*, vol.113, pp.457-480, 2018.

[15] J. Cao, Z. Li and J. Li, Financial time series forecasting model based on CEEMDAN and LSTM, *Physica A: Statistical Mechanics and Its Applications*, vol.519, pp.127-139, 2019.

[16] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan et al., Stock price prediction using LSTM, RNN and CNN-sliding window model, *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp.1643-1647, 2017.

[17] T. Fischer and C. Krauss, Deep learning with long short-term memory networks for financial market predictions, *European Journal of Operational Research*, vol.270, no.2, pp.654-669, 2018.

[18] Y. Li, Z. Zhu, D. Kong et al., EA-LSTM: Evolutionary attention-based LSTM for time series prediction, *Knowledge-Based Systems*, vol.181, 2019.

[19] Y. X. Wu, Q. B. Wu and J. Q. Zhu, Improved EEMD-based crude oil price forecasting using LSTM networks, *Physica A: Statistical Mechanics and Its Applications*, vol.516, pp.114-124, 2019.

[20] S. Rikukawa, H. Mori and T. Harada, Recurrent neural network based stock price prediction using multiple stock brands, *International Journal of Innovative Computing, Information and Control*, vol.16, no.3, pp.1093-1099, 2020.

[21] Y. Lin, Y. Xiao and F. Li, Forecasting crude oil price volatility via a HM-EGARCH model, *Energy Economics*, vol.87, 2020.

[22] M. Plutowski, G. Cottrell and H. White, Experience with selecting exemplars from clean data, *Neural Networks*, vol.9, no.2, pp.273-294, 1996.