International Journal of Advanced Smart Convergence Vol.8 No.2 132-139 (2019)
http://dx.doi.org/10.7236/IJASC.2019.8.2.132

IJASC 18-2-16

A Novel Parameter Initialization Technique for the Stock Price Movement Prediction Model

Thu Nguyen-Thi, Seokhoon Yoon*

Department of Electrical and Computer Engineering, University of Ulsan, Korea
*hathu.bk56@gmail.com, seokhoonyoon@ulsan.ac.kr**

Abstract

We address the problem about forecasting the direction of stock price movement in the Korea market. Recently, the deep neural network is popularly applied in this area of research. In deep neural network systems, proper parameter initialization reduces training time and improves the performance of the model. Therefore, in our study, we propose a novel parameter initialization technique and apply this technique for the stock price movement prediction model. Specifically, we design a framework which consists of two models: a base model and a main prediction model. The base model constructed with LSTM is trained by using the large data which is generated by a large amount of the stock data to achieve optimal parameters. The main prediction model with the same architecture as the base model uses the optimal parameter initialization. Thus, the main prediction model is trained by only using the data of the given stock. Moreover, the stock price movements can be affected by other related information in the stock market. For this reason, we conducted our research with two types of inputs. The first type is the stock features, and the second type is a combination of the stock features and the Korea Composite Stock Price Index (KOSPI) features. Empirical results conducted on the top five stocks in the KOSPI list in terms of market capitalization indicate that our approaches achieve better predictive accuracy and F1-score comparing to other baseline models.

Keywords: *Stock price movements, Long short-term memory, Parameter initialization*

1. Introduction

Forecasting the future of stock price movements is essential for investors. It can help them make the decision to buy and sell stock and reduce risk effectively. However, stock price movements prediction is usually considered as one of the most challenging and important problems in time series prediction due to its noisy and non-stationary nature [1]. Therefore, this task attracts many researchers and needs to be addressed effectively in the future.

During the past decades, along with the development of artificial intelligence (AI), several articles using machine learning models such as artificial neural networks (ANNs) [2, 3] and the support vector machine (SVM) [4, 5] have been used to predict the stock price and gained high predictive accuracy. Unlike

Manuscript Received: Apr. 30, 2019 / Revised: May. 5, 2019 / Accepted: May. 11, 2019
Corresponding Author: seokhoonyoon@ulsan.ac.kr
Tel: +82-52-259-1403, Fax: +82-52-259-1687
Department of Electrical and Computer Engineering, University of Ulsan, Korea

traditional linear model, machine learning methods were modeling the non-linear relationship between historical signal and the future direction of the stock price, and those methods have better performance when comparing with generalized autoregressive conditional heteroskedasticity (GARCH) model and autoregressive integrated moving average (ARIMA) [6, 7].

Recently, recurrent neural network (RNN) is used popularly in time series prediction because this network can handle sequence data effectively by feedback connections inside network that helps past information persist. In this study, we consider long short-term memory (LSTM) which is a type of RNN as our model. It overcomes the problem of vanishing (or exploding) gradients [8] and it can effectively learn long-term dependencies through memory cells and gates. There are several existing studies which addressed the future stock price movements by using LSTM [9-12]. For example, in [10], the authors used a combination of LSTM, stack autoencoder (SAE) and wavelet transforms (WT) as new framework to predict the stock price in the next day. This study is conducted in six stock indices such as CSI 300 index, Nifty 50 index, Hang Seng index, Nikkei 225 index and S&P 500 index. For each stock, three types of variables are used as model inputs. The first set is the historical stock trading data, such as the open, high, low and close price, the second set is the technical indicators and the third set is the macroeconomic variables (i.e., exchange rate and interest rate). In [12], the authors et al. used the LSTM in predicting S&P 500 index close price in the next day and found that this network can achieve better performance when only using daily basic information of the stock (i.e., high price, low price, close price, volume, and adjust price). Indeed, most of the publications only focus on choosing suitable input features and reducing the noisy data to improve the performance. However, they need to a large amount of training samples to achieve optimal parameters when they use existing parameter initialization methods as Xavier initialization, He initialization and zero initialization, etc. In fact, collecting the large training samples for training the stock prediction model for each stock is difficult, it leads to the overfitting problem and worst performance prediction in case of insufficient data.

Therefore, in our study, we introduce a novel parameter initialization technique for the stock price movement prediction model. Specifically, we design a framework which has two phases, the first phase is building the base model by using the large data generated by the top 50 stocks in terms of market capitalization to achieve optimal parameters, and then the second phase will build the main prediction model with the dataset of the given stock to collect results. The main idea of this approach is initializing parameters not randomly, but rather with values that may be close to the optimal values. In this study, we select the top five stocks of the KOSPI list in terms of the highest market capitalization to test the prediction ability of the proposed approach. Each stock is downloaded within 4 years from July 31, 2014 to August 31, 2018 and two types of variables are used as model inputs. Instead of using daily basic information of the stock, we use a return value which is the change of two consecutive days as a feature. Thus, the first input set is a sequence including 20 features as the information of the most recent 20-day period for each stock. In addition, we believe that other information in the stock market can influence the direction of the stock price movements. Therefore, in this paper, we consider another information which is the KOSPI index information and hope that this information can improve the performance. Thus, the second input set is a combination of the sequence including 20 stock features and the sequence including 20 KOSPI index features. Experiment results show that our approach gains higher the performance in terms of the predictive accuracy and F1-score than other baseline models using the existing parameter initialization methods. Besides, the results prove that the related information is useful in forecasting the stock price movements.

The remainder of this paper is organized as follows: Section 2 describes the KOSPI dataset and the stock dataset that we use in this paper, as well as how the useful data is extracted from them. Section 3 demonstrates our approach in more detail. In this section, we explain how the novel parameter initialization technique is applied in our study. In Section 4, the experiment results and analysis are presented. Finally, we conclude our work in Section 5.

2. Dataset

The section briefly presents how the useful data is extracted and how we formulated them to stock features and how the output of the model is defined. For our experiments, we exploit historical closing price data from the Korea Composite Stock Price Index (KOSPI) market, the major stock market in South Korea. First, we choose the KOSPI index stock and the 50 largest stocks in terms of market capitalization from this index from July 31, 2014 to August 31, 2018. The data are all obtainable from financial.yahoo.com. Each stock contains open price, high price, low price, close price, volume and adj close price.

Let P_t^s denote the close price values of stock s on the day t and stock returns over m days are calculated using the formula $r_{t,m}^s = (P_t^s - P_{t-m}^s)/P_{t-m}^s$ [13], where the stock returns determine the price difference between day t and day $t-m$.

To construct the raw level input for the LSTM model, we adopt a sequence length of 20, thus including the information of the most recent 20-day period for each stock s, specially, $R_t^{s_n} = \{r_{t-19,1}^{s_n}, r_{t-18,1}^{s_n}, \cdots, r_{t,1}^{s_n}\}$ and each $t \geq 20$. Following the way, in total, each stock contains 718 sequences for training (from July 31, 2014 to July 31, 2017), 117 sequences for validation (from August 01, 2017 to February 28, 2018) and 104 sequences for prediction (from March 01, 2018 to August 31, 2018). Meanwhile, y_t^s is used to the output of the estimation model which shows the direction of the price movement of the stock s on the day $t+1$. As a result, $y_t^s = 1$ implies that increasing the stock price on the day $t+1$ as compared to the previous day, and $y_t^s = 0$ indicates the opposite direction of the stock price.

3. The Framework for Training the Prediction Model

In this section, we describe about the novel parameter initialization technique in more detail. Our framework consists of two phases, which are building the base model and training the main prediction model. Firstly, we use LSTM [8] as the base model and obtain model parameters through the large data generated by a large amount of stock data. Then, the main prediction model uses optimal parameters as initialized parameters to achieve final performance.

3.1 Building the Base Model

The first phase of our framework is to find optimal parameters of the base model, which are obtained by training the large data generated from a large amount of stocks. To build the base model, we have two steps which are generating the training data from a lot of the available stock data and training the LSTM model.

3.1.1. Generation Training Data for the Base Model

In this subsection, we have two different methods to generate the training data for the base model, one is using only the stock features, and another is using the stock features integrating the KOSPI index features. Commonly, in both methods, we choose the 50 largest stocks in terms of market capitalization and stack them in one large data which has $(n_{seq} \times 718)$ samples in which $n_{seq} = 718$ is a number sequence of each stock. However, the difference between two methods is the information we are considering as input features of the base model. The goal of this task is to evaluate the influence of the addition information in forecasting the stock price movements.

To easily understand these methods, some notations are denoted as $R_t^{s_n} = \{r_{t-19,1}^{s_n}, r_{t-18,1}^{s_n}, \dots, r_{t,1}^{s_n}\}$, $R_t^{index} = \{r_{t-19,1}^{index}, r_{t-18,1}^{index}, \dots, r_{t,1}^{index}\}$, which are the information of the stock s_n and the KOSPI index on the day t, respectively. In our context, $s_n \in \{s_1, s_2, \dots, s_{50}\}$ corresponds to the location of the stock in descending order of market capitalization. To generate inputs of the base model, we design a 3-D matrix $X_{(k)}$ with dimensions of $n_{seq} \times 20 \times k$, where k is a number of input features, in our study, we consider $k \in \{1,2\}$.

In the first method, only the information of the stock s_n is considered as the input of the base model, which means $k = 1$. Therefore, $X_{(1)}$ can be formed as follows:

$$X_{(1)} = [R_t^{s_1}, R_{t+1}^{s_1}, \dots, R_{t+718-1}^{s_1}, R_t^{s_2}, R_{t+1}^{s_2}, \dots, R_{t+718-1}^{s_2}, \dots, R_t^{s_{50}}, R_{t+1}^{s_{50}}, \dots, R_{t+718-1}^{s_{50}}] \tag{1}$$

In the second method, we believe that the stock price movements can be affected by other related information in the stock market. Therefore, we use the KOSPI index features as another input feature, along with the stock features to construct the input of the base model. Thus, $k = 2$ and the input of the base model can be formed as follows:

$$X_{(2)} = [\{R_t^{s_1}, R_t^{index}\}, \dots, \{R_{t+718-1}^{s_1}, R_{t+718-1}^{index}\}, \dots, \{R_t^{s_{50}}, R_t^{index}\}, \dots, \{R_{t+718-1}^{s_{50}}, R_{t+718-1}^{index}\}] \tag{2}$$

Recall that y_t^s is used to the output of the estimation model which shows the direction of the price movement of the stock s on the day t+1.

Thus, $Y = [y_t^{s_1}, y_{t+1}^{s_1}, \dots, y_{t+718-1}^{s_1}, y_t^{s_2}, \dots, y_{t+718-1}^{s_2}, \dots, y_t^{s_{50}}, \dots, y_{t+718-1}^{s_{50}}]^T$ which is 1-D matrix including the output of all stocks s_n. The objective of the base model is to predict the up/down of each stock price s_n on the day $t + 1$ given the recent 20-day information, or to maximize the following probability:

$$P(Y^{(i)} | X_{(k)}^{(i)}, W, b) \tag{3}$$

where W and b are model parameters.

Therefore, the total number of training samples available becomes at least 35900 samples, 5700 samples for validation set and 5200 samples for test set. The goal is that $\{W, b\}$ are optimized by training the largest 50 stock data instead of by training one stock.

3.1.2. Training the Base Model for Obtaining Parameters

For the training of the base model, a variety of LSTM architectures which have a different number of layers, hidden units, and activation function examined. Then, among the considered setting, the most appropriate LSTM architecture was selected by using the validation set. The specified topology of our trained LSTM model is hence as follows:

- Input layer has a number of units based on the information we used and 20 timesteps.
- There are 2 LSTM layers, which are followed by 2 full connection layers. Each layer has 16 units and a dropout value of 0.5.
- Output layer with one neuron and sigmoid activation function – a standard configuration.

Moreover, weight and bias values are initialized with normal distribution. The learning rate in RMSprop optimization, batch size and the maximum training epochs are set at 0.001, 512, and 5000, respectively.

3.2. Training the Main Prediction Model

After obtaining the parameters of the base model in the first phase of the framework, we use these parameters as initialization values and train main prediction model using a small amount of each stock data to obtain the final performance. In this case, inputs of this model can be represented as $X_{(1)}^{S_n} = [R_t^{S_n}, R_{t+1}^{S_n}, \dots, R_{t+718-1}^{S_n}]^T$ or $X_{(2)}^{S_n} = [\{R_t^{S_n}, R_t^{index}\}, \dots, \{R_{t+718-1}^{S_n}, R_{t+718-1}^{index}\}]^T$ which corresponds two types of inputs we used in predicting each stock price movement. Each of them is fed into the topology of our main prediction model correspondingly.

For training the main prediction model, the learning rate and the maximum training epochs are set at 0.0005, 500. The small learning rate are chosen because we do not want to distort the optimal parameters of the base trained model too quickly.

4. Evaluation Results and Analysis

In our experiments, we evaluated the performance of our proposal by replacing it with three baseline classifiers with the different parameter initialization methods, one is support vector machine (SVM) with zero initialization, other models are LSTM with Xavier initialization and LSTM with He initialization. We use the SVM model as one of benchmark models for compelling reasons. One of the most important reasons is because it is state-of-the-art machine learning model that requires virtually no tuning and delivers good results [14]. Especially, SVM is good at dealing with small data samples; it is suitable for our training data. To design input features for the SVM baseline model, with each stockss_n, we use cumulative returns$R'^{S_n}_t = \{r_{t,20}^{S_n}, r_{t,10}^{S_n}, r_{t,5}^{S_n}, r_{t,2}^{S_n}, r_{t,1}^{S_n}\}$, representing information of the most recent 20-day period. Besides, the output is the same as our model defined in Subsection 2.2.

Next, detailed configurations of three baseline models are described. SVM with two popular kernels such as Radial Basis Functions (RBF) and Poly-nominal are considered. In addition, hyper-parameters of SVM (i.e.,γ of RBF kernel, degree, d of Poly-nominal and regularization for both kernels, C) are chosen by grid search for each branch. Specially, the searching interval for γ and d are [0.5, 10.5] with a step size of 0.5 and [1, 4] with a step size of 1, respectively, and C is chosen in {0.5, 1.5, 10, 100}. For two LSTM baseline models, the configuration of them has the similar with our base model.

In this experiment, we choose predictive accuracy and F_1-score to evaluate the performance of our proposed approach. Similar to [15], computation of these evaluation measures requires estimating Precision and Recall which are evaluated from True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). These parameters are defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

The predictive accuracy and F-measure are estimated using Equation. (6) and (7) respectively.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (6)$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (7)$$

We give the result in the form of a table. As shown in Table. 1, we compare the performance of our method with two types of inputs, one is using only the stock features (Our model + Input $X_{(1)}^{S_n}$) and another is using the combination of the stock features and the KOSPI index features (Our model + Input $X_{(2)}^{S_n}$). In addition, our models are compared with three baseline models (i.e., SVM + Zero Initialization + input $R'^{S_n}_t$, LSTM + Xavier Initialization + Input $X_{(2)}^{S_n}$, and LSTM + He Initialization + Input $X_{(2)}^{S_n}$).

As we can see in Table 1, with the same input which is the combination of the stock features and the KOSPI features, our model outperforms than other LSTM models in all data that we use. For example, this approach yields 58.8% accuracy for the prediction of the stock price movements in the next day, as compared with 53.2% accuracy of the LSTM model using Xavier initialization and 52% accuracy of the LSTM model using He initialization. Moreover, our model with the novel parameter initialization technique achieves better performance than SVM with zero initialization.

Stock ID	Prediction Models									
	Our model + Input $X_{(1)}^{S_n}$		Our model + Input $X_{(2)}^{S_n}$		SVM + Zero Initialization + Input $R'^{S_n}_t$		LSTM + Xavier Initialization + Input $X_{(2)}^{S_n}$		LSTM + He Initialization + Input $X_{(2)}^{S_n}$	
	Acc	F_1-score	Acc	F_1-score	Acc	F_1-score	Acc	F_1-score	Acc	F1-score
S_1	0.51	0.28	0.58	0.65	0.49	0.45	0.47	0.34	0.48	0.4
S_2	0.52	0.64	0.56	0.57	0.51	0.58	0.49	0.51	0.55	0.53
S_3	0.59	0.68	0.61	0.69	0.6	0.68	0.59	0.71	0.51	0.6
S_4	0.59	0.71	0.62	0.71	0.61	0.66	0.6	0.74	0.6	0.72
S_5	0.5	0.61	0.57	0.65	0.51	0.67	0.51	0.63	0.46	0.53
Average	0.542	0.584	**0.588**	**0.654**	0.544	0.608	0.532	0.586	0.52	0.556

* Note: Samsung Electronics Co. (S_1), SK Hynix (S_2), Celltrion (S_3), Hyundai (S_4), LG Chem (S_5)

Table 1. Comparison of our model and baseline models

In addition, as shown in Table.1, our model using input $X_{(2)}^{S_n}$ obtains higher the predictive accuracy than our model using input $X_{(1)}^{S_n}$. Results prove that our assumption is feasible, and we can improve the performance of the prediction by using other related information. Especially, in predicting of some stocks as Samsung Electronics Co., SK Hynix, and LG Chem, the KOSPI index features are more useful. As can be observed, those features help improve the predictive accuracy of three stocks so much as 7%, 4% and 7%, respectively. In case of F1-score, our model with the second input has the highest performance, especially, it is very effectively when we use to predict the Samsung Electronics Co. stock price movement. Because Samsung Electronics company is the largest stock in the Korea market, the stock price movements of this company can be affected by the KOSPI index price more than other companies.

5. Conclusion

In this paper, we proposed the novel parameter initialization technique for the stock price movement prediction model. We built the framework which has two different models as the base model and the main prediction model. Building the base model helps us find the optimal parameters which are used to initialization parameters of the main prediction model. Therefore, the main prediction model is trained more effectively. Moreover, we conducted a new input that is the integration of the stock features and the KOSPI index features to improve the predictive accuracy. Experimental results suggest that the novel parameter initialization technique achieved better the performance than other parameter initialization methods and the stock price movements are affected by other related information.

About a future work, we will consider the correlation between stocks in forecasting the stock price movements (i.e., the highest cosine similarity, the similar field, and the highest market capitalization). Then, we will conduct other input which includes more features as other related stock features. We believe that this input can improve the performance of the prediction model.

Acknowledgement

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF2016R1D1A3B03934617) and MSIT (2019R1F1A1058147)

References

[1] Y. S.Abu-Mostafa and A. F.Atiya, "Introduction to financial forecasting," *Applied Intelligence,* vol. 6, no. 3, pp. 205--213, 1996.
DOI: https://doi.org/10.1007/BF00126626

[2] E. Guresen, G. Kayakutlu and T. U.Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications,* vol. 38, no. 8, pp. 10389--10397, 2011.
DOI: https://doi.org/10.1016/j.eswa.2011.02.068

[3] M. Qiu and Y. Song, "Predicting the direction of stock market index movement using an optimized artificial neural network model," *PloS one,* vol. 11, no. 5, p. e0155133, 2016.
DOI: https://doi.org/10.1371/journal.pone.0155133

[4] K. Kyoung-jae, "Financial time series forecasting using support vector machines," *Neurocomputing,* vol. 55, no. 1-2, pp. 307--319, 2003.
DOI: https://doi.org/10.1016/S0925-2312(03)00372-2

[5] S. Madge and S. Bhatt, "Predicting stock price direction using support vector machines," *Independent Work Report Spring,* 2015.

[6] L. D. Persio and O. Honchar, "Artificial neural networks architectures for stock price prediction: Comparisons and applications," *International Journal of Circuits, Systems and Signal Processing,* vol. 10, no. 2016, pp. 403--413, 2016.

[7] A. A. Adebiyi, A. O. Adewumi and C. K. Ayo, "Comparison of ARIMA and artificial neural networks models for stock price prediction," *Journal of Applied Mathematics,* 2014.
DOI: http://dx.doi.org/10.1155/2014/614342

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

DOI: https://doi.org/10.1162/neco.1997.9.8.1735

[9] S. Liu, G. Liao and Y. Ding, "Stock transaction prediction modeling and analysis based on LSTM," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2018.
DOI: 10.1109/ICIEA.2018.8398183

[10] 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.
DOI: https://doi.org/10.1371/journal.pone.0180944

[11] 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.
DOI: https://doi.org/10.1016/j.eswa.2018.07.019

[12] T. Gao, Y. Chai and Y. Liu, "Applying long short term momory neural networks for predicting stock closing price," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 2017.
DOI: 10.1109/ICSESS.2017.8342981

[13] 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.
DOI: https://doi.org/10.1016/j.ejor.2017.11.054

[14] J. Patel, S. Shah, P. Thakkar and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," *Expert Systems with Applications,* p. 2015, 259--268.
DOI: https://doi.org/10.1016/j.eswa.2014.07.040

[15] A. Mohammed, G. Biegert, J. Adamec and T. Helikar, "Identification of potential tissue-specific cancer biomarkers and development of cancer versus normal genomic classifiers," *Oncotarget,* vol. 8, no. 49, p. 85692, 2017.
DOI: 10.18632/oncotarget.21127