Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 227 (2023) 939–947

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Computer Science and Computational Intelligence (ICCSCI 2023)

Stock Price Prediction Model Using Deep Learning Optimization Based on Technical Analysis Indicators

Timothy Julian[a], Theodorus Devrison[a], Varian Anora[a], Kristien Margi Suryaningrum[a]*

[a]*Computer Science Departement, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480*

Abstract

Stock price prediction is one of the processes of analyzing and determining stock prices in the future. With technical analysis, future stock price predictions can be predicted through the pattern of fluctuations in the stock price in the past. In this study, the researcher predicts the stock price for the next week using the Deep Learning method, namely the Multilayer Perceptron, and combined with the day-shifting method. To expect the results of this stock, the author also observes the model's usefulness and proposes a Mean Error to Mean Price Ratio (MEMPR) to increase the insights processed by the model. Then to find out the accuracy of stock price predictions for each algorithm, testing is carried out using stock data which consists of new data which is then carried out by a training process to get an absolute error value. The experimental results show that the model can predict stock prices with an R2 metric of 0.995.

Keywords: stock price; prediction, deep learning; multilayer perceptron; technical analysis; technical indicator

1. Introduction

The significant development of the Indonesian stock market has made it a great subject for scientific research [1]. Stocks are financial instrument that can be used as a long-term investment or short-term profit by buying and

* Corresponding author. Tel.: +62-21- 5345830
E-mail address : kristien.s@binus.edu

10.1016/j.procs.2023.10.601

selling stocks [2]. On the stock market, issuers of stocks can obtain funds to run their businesses [3] and shareholders can benefit from buying stocks through capital gains and dividends that the company pays to shareholders.

Stock prices movements are difficult to predict because stock prices do not have a fixed pattern [4], [5], [6]. There are two approaches to analyze the movement of stock prices, technical analysis and fundamental analysis [4], [7]. Technical analysis is performed by looking at the historical price of the stock; Fundamental analysis, on the other hand, is done through the business approach, where the company's financial data and market sentiment is analyzed.

In recent years, deep learning has been studied extensively in order to be able to predict stock prices with good precision [8], [9]. This paper focused on using multilayer perceptron, one of deep learning architecture, to predict stock prices for the next 7 market days. Multilayer perceptron is chosen because of its architectural structure. MLP is considered a deep neural network because it consists of more than 1 hidden layer. Therefore, it can solve problem more sophisticatedly and with better approximation. Hopefully, this paper will be able to contribute to the

scientific knowledge by providing information about the capability of multilayer perceptron to predict stock prices for time window of 7 market days, providing day shifting method to predict stock prices for longer time window, and providing Mean Error to Mean Price Ratio (MEMPR) metric to show the error tolerance area of stock price prediction.

2. Literature Review

The movement of stock prices is influenced by many macro and micro factors [10] which is why the stock market is very chaotic, complex, and highly dynamic [11]. Fundamental analysis approach analysed stock prices based on external factors such as politics, economic circumstances, company business principles [12], interest rates, company performance, and media sentiment [9], [13], [14]. While technical analysis relies on historical data from stock such as open price, close price, lowest price, highest price, and transactions volume [15]. There are many technical indicators used to do technical analysis, including Simple Moving Average [16], Weighted Moving Average [17], and Relative Strength Index. Technical indicator can also be used to improve the precision of the deep learning model [6].

There have been many studies that have used deep learning to do classifications and predictions in various industries. For example, deep learning is used to classify medical data [18]; predict short-term traffic conditions [19]; and predict the likelihood of a power outage caused by lightning [20]. Deep learning is a derivative of machine learning based on the human's brain cells [6]. There are several architectures of deep learning, including Multilayer Perceptron (MLP), Long-Short Term Memory (LSTM) [21], [22], and Convolutional Neural Network (CNN) [5], [21]. Results produced by deep learning model must be measured statistically to understand the generalizability and the precision of the constructed model [16]. Several error metrics can be used, such as Mean Absolute Deviation (MAD) [16], Mean Square Error (MSE) [16], and Root Mean Squared Error (RMSE) [5].

Many deep learning related study has been done to predict stock prices. Deep learning trained by inputting historical data from stocks [10], [11], financial news, and investor opinions in open forums [23], [24]. Predicting stock prices is challenging because it is highly dynamic and complex [13], It is observed that data normalization [20], optimization methods [26], model architecture, and hyperparameters greatly affects the outcome of deep learning.

3. Proposed Method

From the papers observed by the authors, most of the papers only attempted to predict stock prices in a short time window i.e., on the same day [9], [10], [15] or the next day [2], [11], [14]. Predicting stock prices in a short time window can reduce the complexity and unpredictability. But in a real-world scenario, predicting stock prices in a short time window would only give traders or investors a little information.

The authors came up with the idea of conducting experiment to predict stock prices in a longer time window, i.e., 7 days. The purpose of the experiment is to measure the capability of multilayer perceptron of doing so and to give traders or investors more information about the stock prices by lengthening the time window of prediction.

A. Research Methodology

Research methodology used in this paper is explained by the flowchart shown in Fig. 1. Stock price data needed for this research gathered and undergoes data pre-processing stage which consists of data cleaning, applying technical indicators, applying day shifting method, and data normalization. Afterwards, multilayer perceptron model with the lowest error metrics will be constructed based on a few hyperparameter options. Then the prediction will be visualized side by side with the actual closing price of the stock.

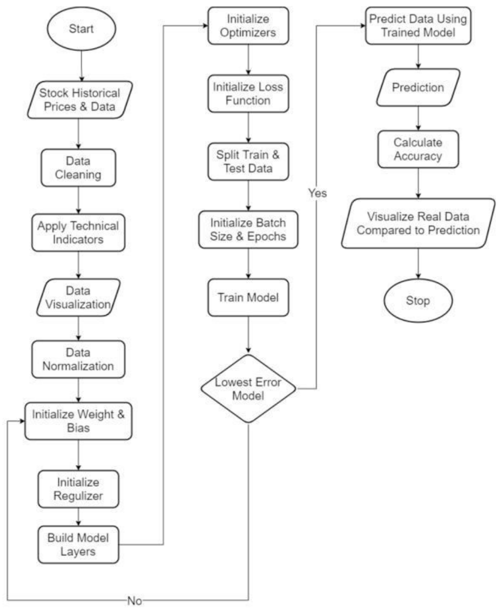

Fig. 1. Research methodology's flowchart

B. Day shifting method.

For the model to be able to predict stock price on the next 7 market days, the authors proposed a method called day shifting. In the training phase, the model is forced to predict the closing price (y label) on h+7 based on stock price data (x label) from h+0 as shown in Fig. 2. resulting data as shown in Fig. 3.

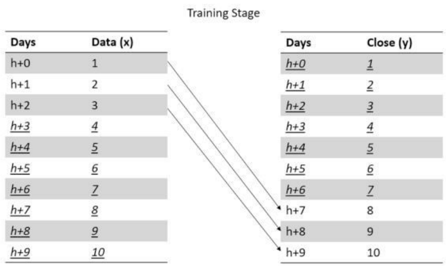

Fig. 2. Visualization of day shifting method

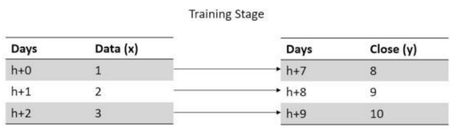

Fig. 3. Data after processed by day shifting method

C. Multilayer perceptron

An example of deep learning architecture is multilayer perceptron (MLP). MLP is one type of feedforward artificial neural network with more than one hidden layer. As shown in Fig. 4, MLP for this study consists of the input layer x with node numbers corresponding to the dimension of the data, hidden layers y with each layer consisting of neurons connected to each input node and each neuron in the next hidden layer, and output layer z consisting of only one neuron with linear activation function

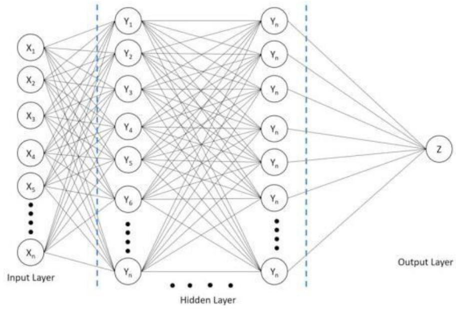

Fig. 4. Multilayer Perceptron Architecture. (x) Nodes in the input layer. (y) Neurons in the hidden layers. (z) A single neuron with linear activation in the output layer

D. Technical Analysis

In technical analysis, historical data of stocks is used as a base to perform a set of analysis using wide variety of technical indicators. This study used 4 types of moving average as the technical indicators.

- SMA is simply the average of stock price in a certain window of time as seen in (1) where x represents the stock price, n represents the window of time and i represents the day

$$SMA_n = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

- Exponentially Moving Average (EMA) is a moving average that gives more weight to the most recent price. As seen in (2), k represents the weight, x represents the stock price, t represents today, and n represents the window of time. The calculation for k can be seen in (3).

$$EMA_t = x_t k + EMA_{t-1}(1-k) \tag{2}$$

$$k_n = \frac{2}{(n+1)} \tag{3}$$

- Smoothed Moving Average (SMMA) has the basic idea of SMA, but the averaged value is smoothed as seen in (4) where x represents the stock price, t represents today, n represents the window of time, and i represents the day

$$SMMA_{n,t} = \frac{\sum_{i=1}^{n} x_i - SMMA_{t-1}x_t}{n} \tag{4}$$

- Linear Weighted Moving Average (LWMA) has the same background ideas as EMA, but LWMA gives weight to the stock prices linearly as seen in (5) where x represents the stock price, t represents today, n represents the window of time, and i represents the day.

$$LWMA_n = \frac{\sum_{i=1}^{n} x_i i}{\sum_{i=1}^{n} i} \tag{5}$$

4. Experiment & Result

Data for this experiment is gathered from https://finance.yahoo.com/ through a Python package called pandasdatareader. One stock price data from the Indonesian stock market (IDX) is chosen. In total, there are 4,249 data points consisting of 6 dimensions. The description of the dataset and its statistical characteristics are explained in Fig. 5 and Fig. 6. Environment used for this experiment is listed as follow: (1) Intel Core i7-9750H @2.60GHz, (2) NVIDIA GeForce GTX 1650, (3) Tensor Flow Version: 2.3.0, (4) Keras Version: 2.4.0, (5) Python 3.7.2, (6) Windows 10 Home Single Language Version - 10.0.19042 Build 19042.

Date	High	Low	Open	Close	Volume	Adj Close
2004-06-08	900.0	875.0	875.0	887.5	99830000.0	553.706055
2004-06-09	912.5	875.0	887.5	900.0	58858000.0	561.505005
2004-06-10	900.0	887.5	900.0	900.0	33118000.0	561.505005
2004-06-11	900.0	887.5	887.5	900.0	27166000.0	561.505005
2004-06-14	900.0	875.0	900.0	887.5	31708000.0	553.706055
...	...	...	...	...	...	...
2021-07-13	30875.0	30075.0	30850.0	30225.0	13189300.0	30225.000000
2021-07-14	30150.0	29900.0	30100.0	29950.0	11864200.0	29950.000000
2021-07-15	30575.0	29975.0	30000.0	30575.0	9944800.0	30575.000000
2021-07-16	30850.0	30275.0	30850.0	30550.0	9599700.0	30550.000000
2021-07-19	30550.0	30025.0	30550.0	30025.0	9423800.0	30025.000000

4249 rows × 6 columns

Fig. 5. Data description

Statistic	High	Low	Open	Close	Volume	Adj Close
Maximum	36900.000000	35800.000000	36725.000000	36725.000000	3.899920e+08	36217.312500
Minimum	887.500000	875.000000	875.000000	887.500000	0.000000e+00	553.706055
Mean	12047.570016	11801.453283	11931.074959	11926.376795	2.229304e+07	11099.327298
Median	9200.000000	8900.000000	9100.000000	9100.000000	1.460580e+07	8118.734375

Fig. 6. Data's statistical characteristics

A. Data Preprocessing

First step in data preprocessing is data cleaning. Uncomplete data found in the raw data is dropped. The column 'Adj Close' is also dropped. Cleaned data then applied with technical indicators SMA, EMA, SMMA, LWMA, with window of time chosen of 7 and 30. The visualization of technical indicators applied to the data is presented in Fig. 7 for window of time 7 and Fig. 8 for window of time 30. For each visualization, the last 100 data points is chosen.

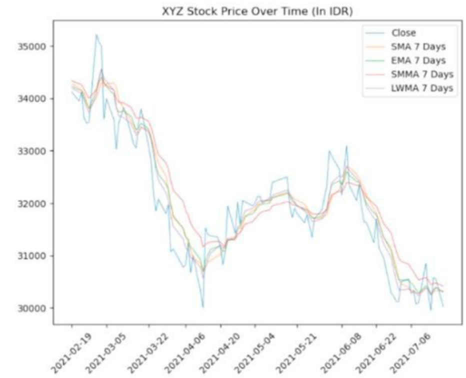

Fig. 7. Moving average with window of time 7

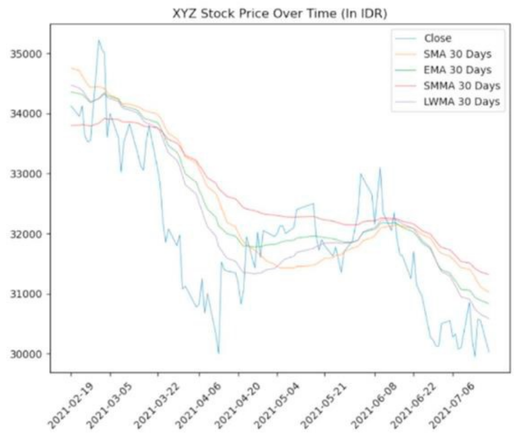

Fig. 8. Moving average with window of time 30

The day shifting method also included in this stage. Normalization method used is Min-Max Normalization. Min-Max Normalization scale used is [0, 1] for the nMin and nMax respectively [18]. After data preprocessing stage, there will be 4,213 normalized data points with 13 dimensions which is open price, lowest price, highest price, close price, transaction volume, SMA_7, SMA_30, SMMA_7, SMMA_30, EMA_7, EMA_30, LWMA_7, LWMA_30.

B. Architecture & hyperparameters

For this study, hyperparameters configuration for the model are shown in Table 1:

Table 1. Hyperparameter options

Hyperparameter	**Value**	**Constant/ Variable**
No. of neurons / hidden layer	{10, 100, 1000}	Variable
No. of hidden layer	{1, 2, 3, 4, 5}	Variable
Learning rate	{0.0001, 0.001, 0.01}	Variable
No. of epoch	{50, 100, 200}	Variable
Activation function for hidden layers	ReLU	Constant
Activation function for output layers	Linear	Constant
Weight Initializer	He Uniform	Constant
Bias Initializer	He Uniform	Constant
Optimizer	Stochastic Gradient Descent with Momentum (SGDM)	Constant
Momentum	0.8	Constant
Loss Function	Root Mean Square Error (RMSE)	Constant

Hyperparameter optimization is done using an exhaustive search called Grid Search and the lowest error model found with a configuration of 1000 neurons/ hidden layer, 2 hidden layers, 0.0001 learning rate, and 100 epochs.

C. Training & result

The lowest error model then used to predict the stock price for the next 7 market days. The graphical visualization of the prediction is shown in Fig. 9 for the last 200 data points. In Fig. 9, the prediction line is colored orange, and the actual price line is colored blue. Error metrics value from the predictions are shown in Table 2.

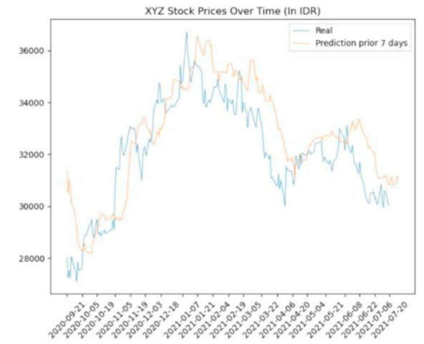

Fig. 9. Prediction Visualization for Last 200 Data Points

Table 2. Unnormalized Error Metric Values

Error metrics	Value
RMSE	657.1796077435799
R^2	0.9955108642032026
MAE	350.7431643377594
MAPE	0.033498239976073485

As can be seen in Fig. 9, the prediction line appeared to be behind the actual price. The delay occurred because the model was trying to predict stock price for longer time window. This result is similar to the study that was based on technical analysis [9], [15].

D. Further experiments

An error tolerance must also be included to the graph to give more insight about the range of error the model might have made. This paper will propose a new metric to cover the error ratio called Mean Error to Mean Price Ratio (MEMPR) shown in (6) where x is the actual price, x' is the prediction price, i is the particular data points, and n is the total data

$$MEMPR = \frac{\frac{1}{n}\sum_{i=1}^{n}|x_i' - x_i|}{\frac{1}{n}\sum_{i=1}^{n}x_i} \quad (6)$$

The idea is to take the MAE of the model then divide it by the mean from the actual stock price. The MEMPR of the model studied is 0.0292. The obtained MEMPR then applied to obtain the upper and lower area of error tolerance using equation shown in (7) where x' is the prediction price and i is the particular data points.

$$Bound = x_i'(1 \pm MEMPR) \quad (7)$$

By applying the bounds to the graph, an error tolerance area can be made and shown in Fig. 10. The error tolerance is the area of orange with the prediction line in the middle. The last 200 data points is chosen same as before.

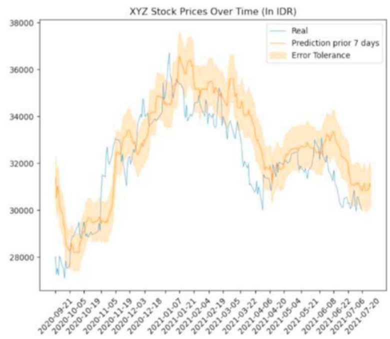

Fig. 10. Prediction Visualization for Last 200 Data Points with Error Tolerance

E. Implementation

From real case perspective, if there is someone wanting to buy the chosen stock, by looking at Fig. 10, that person can tell that for the next 7 days, the prediction shown by the orange line predicts that the price will be above the current price shown by the blue line. Therefore, that person can acknowledge this as a buy signal because the prediction price is higher than the current price.

5. Conclusions

This paper tried to predict stock price using multilayer perceptron. This paper proposed the day shifting method so the model is able to predict price for the next 7 market days and the Mean Error to Mean Price Ratio (MEMPR) as a measurement to calculate the upper and lower bound of the error tolerance area. The model used in this study was able to achieve R^2 of 0.9955

There are still a lot of things to be improved from this research. For examples applying a more advanced technical indicators like Relative Strength Index (RSI) or Stochastic Oscillator; trying different configuration of activation function, optimizer, and batch size; and using more sophisticated optimization like Dropout, Regularization, or Batch Normalization.

References

[1] Li, J., Bu, H., & Wu, J. (2017, June). Sentiment-aware stock market prediction: A deep learning method. 2017 International Conference on Service Systems and Service Management, 1–6. https://doi.org/10.1109/icsssm.2017.7996306

[2] Vargas, M. R., de Lima, B. S. L. P., & Evsukoff, A. G. (2017, June). Deep learning for stock market prediction from financial news articles. 2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 60–65. https://doi.org/10.1109/civemsa.2017.7995302

[3] Wei, D. (2019, October). Prediction of Stock Price Based on LSTM Neural Network. 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), 544–547. https://doi.org/10.1109/aiam48774.2019.00113

[4] Suryani, I., & Buani, D. C. P. (2020). Stock price prediction using artificial neural network integrated moving average. Journal of Physics: Conference Series, 1641, 012028. https://doi.org/10.1088/1742-6596/1641/1/012028

[5] Jayanth Balaji, A., Harish Ram, D., & Nair, B. B. (2018). Applicability of Deep Learning Models for Stock Price Forecasting An Empirical Study on BANKEX Data. Procedia Computer Science, 143, 947–953. https://doi.org/10.1016/j.procs.2018.10.340

[6] Agrawal, M., Ullah Khan, A., & Kumar Shukla, P. (2019). Stock Price Prediction using Technical Indicators: A Predictive Model using Optimal Deep Learning. International Journal of Recent Technology and Engineering, 8(2), 2297–2305. https://doi.org/10.35940/ijrte.b3048.078219

[7] Lien Minh, D., Sadeghi-Niaraki, A., Huy, H. D., Min, K., & Moon, H. (2018). Deep Learning Approach for Short-Term Stock Trends Prediction Based on Two-Stream Gated Recurrent Unit Network. IEEE Access, 6, 55392–55404. https://doi.org/10.1109/access.2018.2868970

[8] Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83, 187–205. https://doi.org/10.1016/j.eswa.2017.04.030

[9] Persio, L. D., & Honchar, O. (2016). Artificial Neural Networks Architectures for Stock Price Prediction: Comparisons and Applications. International Journal of Circuits, Systems, and Signal Processing, 10, 403–413. https://core.ac.uk/download/pdf/217561207.pdf

[10] Yu, P., & Yan, X. (2019). Stock price prediction based on deep neural networks. Neural Computing and Applications, 32(6), 1609–1628. https://doi.org/10.1007/s00521-019-04212-x

[11] Singh, R., & Srivastava, S. (2016). Stock prediction using deep learning. Multimedia Tools and Applications, 76(18), 18569–18584. https://doi.org/10.1007/s11042-016-4159-7

[12] Hu, Z., Liu, W., Bian, J., Liu, X., & Liu, T. Y. (2018, February). Listening to Chaotic Whispers: A Deep Learning Framework for News-oriented Stock Trend Prediction. Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, 261– 269. https://doi.org/10.1145/3159652.3159690

[13] Huynh, H. D., Dang, L. M., & Duong, D. (2017, December). A New Model for Stock Price Movements Prediction Using Deep Neural Network. Proceedings of the Eighth International Symposium on Information and Communication Technology, 57–62. https://doi.org/10.1145/3155133.3155202

[14] Lee, S. I., & Yoo, S. J. (2020). Multimodal deep learning for finance: Integrating and forecasting international stock markets. The Journal of Supercomputing, 76, 8294–8312. https://doi.org/10.1007/s11227-019-03101-3.

[15] Roondiwala, M., Patel, H., & Varma, S. (2017). Predicting Stock Prices Using LSTM. International Journal of Science and Research (IJSR), 6(4), 1754–1756. https://www.ijsr.net/search_index_results_paperid.php?id=ART20172 755

[16] Maricar, M. A., Widiadnyana, P., & Arta Wijaya, I. W. (2017). Analysis of Data Mining for Forecasting Total Goods Delivery with Moving Average Method. International Journal of Engineering and Emerging Technology, 2(1), 7. https://doi.org/10.24843/ijeet.2017.v02.i01.p02

[17] Bakar, N. A., & Rosbi, S. (2018). Weighted Moving Average of Forecasting Method for Predicting Bitcoin Share Price using High Frequency Data: A Statistical Method in Financial Cryptocurrency Technology. International Journal of Advanced Engineering Research and Science, 5(1), 64–69. https://doi.org/10.22161/ijaers.5.1.11

[18] Singh, D., & Singh, B. (2019). Investigating the Impact of Data Normalization on Classification Performance. Applied Soft Computing Journal, 97, 105524., in press. https://doi.org/10.1016/j.asoc.2019.105524

[19] Zhao, Z., Chen, W., Wu, X., Chen, P. C. Y., & Liu, J. (2017). LSTM network: A deep learning approach for short-term traffic forecast. IET Intelligent Transport Systems, 11(2), 68 − 75. https://doi.org/10.1049/iet-its.2016.0208

[20] Xie, Y., Li, C., Lv, Y., & Yu, C. (2019). Predicting lightning outages of transmission lines using generalized regression neural network. Applied Soft Computing, 78, 438–446. https://doi.org/10.1016/j.asoc.2018.09.042

[21] M, H., E.A., G., Menon, V. K., & K.P., S. (2018). NSE Stock Market Prediction Using Deep-Learning Models. Procedia Computer Science, 132, 1351–1362. https://doi.org/10.1016/j.procs.2018.05.050

[22] Padmaja, D., Anandan, R., & Kumar, N. (2020). Stock price prediction using artificial neural networks. Journal of Critical Reviews, 7(11), 846–850. https://doi.org/10.31838/jcr.07.11.152

[23] Akita, R., Yoshihara, A., Matsubara, T., & Uehara, K. (2016, June). Deep learning for stock prediction using numerical and textual information. 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). https://doi.org/10.1109/icis.2016.7550882

[24] Pang, X. W., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2018). An innovative neural network approach for stock market prediction. The Journal of Supercomputing, 76(3), 2098–2118. https://doi.org/10.1007/s11227-017-2228-y

[25] Moghaddam, A. H., Moghaddam, M. H., & Esfandyari, M. (2016). Stock market index prediction using artificial neural network. Journal of Economics, Finance and Administrative Science, 21(41), 89–93. https://doi.org/10.1016/j.jefas.2016.07.002

[26] Keskar, NS., & Socher, R. (2017). Improving Generalization Performance by Switching from Adam to SGD. arXiv:1712.07628 [cs.LG]