Research on Efficient Stock Prediction Method Based on LSTM Network

Dong Li[1,*], Iokieng Liao[2]

[1]Hong Kong Uniwise International Education, Hong Kong, China
[2]KaoYip Middle School, Macau, China

*donglee19880610@gmail.com, 20210419lzy@gmail.com

Abstract. Stock prediction is an important application in the field of data mining, which can help investors mitigate risks, estimate returns, and also anticipate company development. By providing historical stock prices of a certain company, machine learning/deep learning methods can be employed to estimate the future stock performance of the company. Due to the ability of deep networks to extract discriminative features and possess good time modeling capabilities, using deep networks for stock price estimation is an effective stock prediction method. In this paper, we utilize Long Short-Term Memory (LSTM) networks to predict the stock prices of the S&P500. By providing the historical stock data for the previous 9 days, the closing price on the 10th day is taken as the prediction indicator. Mean Squared Error (MSE) is used as the evaluation metric for this method. Experimental results demonstrate that our approach can effectively estimate future stock prices and achieve the MSE indicator for the prediction accuracy of the stock price prediction model.

Keywords: Stock Prediction, Long Short-Term Memory Network, Neural networks.

1 Introduction

With the development of deep learning [1-4, 14-17, 20], using deep learning for stock prediction is a highly effective data processing approach. Due to the powerful recognition capability, excellent learning ability, good prediction accuracy, and strong time-series modeling capability of deep neural networks, stock price prediction based on deep learning has become one of the mainstream methods. Currently, there are various machine learning-based stock prediction methods emerging continuously. Pahwa et al. [5] and Soni, Srivastava [9] organized machine learning-based stock price prediction methods into regression and classification methods and summarized them, including linear regression, logistic regression, and other methods, demonstrating the effectiveness of machine learning-based stock price prediction methods. Shen et al. [6] proposed a time-series correlation-based stock price prediction method by exploring the temporal relationships between global stock markets and different financial products and using

L. Liu et al. (eds.), *Proceedings of the 3rd International Conference on Financial Innovation, FinTech and Information Technology (FFIT 2024)*, Advances in Computer Science Research 118,
https://doi.org/10.2991/978-94-6463-572-0_8

SVM to predict stock price movements. This method achieved good results in predicting the prices of multiple stocks. Leung et al. proposed a machine learning-based stock price prediction method by designing a structural support vector machine to classify stock price movements. Meanwhile, this method also used graph structures to cluster collaborating companies, improving prediction accuracy. This method can quickly predict the prices of multiple stocks in polynomial time and has shown good practical value. Reddy et al. [7], Jishtu et al. [8], Subasi et al. [10], and Alkafaween et al. [18] proposed machine learning (ML) methods that train from available stock data, then use the acquired knowledge for accurate predictions, and apply support vector machines (SVMs) to compare stock prices across different market sizes and financial markets, demonstrating the efficiency and effectiveness of SVMs. In addition, Gornall and Strebulaev [19] highlighted the contracting and valuation aspects of venture capital-backed companies, offering further insights into financial market prediction methods. Moreover, Li et al. [20] discussed the role of investor sentiment extracted from text in stock price prediction, further enhancing prediction accuracy with the use of deep learning techniques.

In conclusion, this study utilizes deep learning networks to predict stock prices. Firstly, we preprocess and select the original S&P500 data, choosing stocks from Apple, Google, and Amazon as experimental targets. Subsequently, we segment the data into 10-day periods. During training and testing, we use data from the previous 9 days as historical data to predict the closing price of the 10th day's stock. Our network consists of 2 layers of LSTM networks, optimized using the RMSProp optimizer, and finally outputs prediction results. We use mean squared error as the evaluation metric. From the experimental results, it can be seen that our method achieved mean squared errors of 0.081, 0.093, and 0.086 for Google, Apple, and Amazon stocks respectively, proving the effectiveness of LSTM networks in stock price prediction.

2 Method

2.1 Data Preprocessing

Firstly, we acquire publicly available S&P500 stock data through the internet, including opening price, closing price, low price, high price, trading volume, stock name, and timestamp. Table 1 below illustrates an example of the raw data. It can be observed that introducing unrelated data can decrease the discriminative power of the network model due to the varying scales among different data. Hence, we only select the closing price as the data feature. We choose Apple, Amazon, and Google from the S&P500 as evaluation indicators. The training set consists of stock data from January 1, 2013, to December 31, 2015. The test set comprises data from January 1, 2016, to August 1, 2017. During both training and inference, we segment the data into intervals of 10 days, and we predict the data for the tenth day using inputs from the previous nine days.

Table 1. Visualization of raw data in S&P500 stock.

·	Open	High	Low	Close	Volume
2012-08-13	92.29	92.59	91.74	92.4	2075391.0
2012-08-14	92.36	92.5	92.01	92.3	1843476.0
2012-08-15	92.0	92.74	91.94	92.54	1983395.0
2012-08-16	92.75	93.87	92.21	93.74	3395145.0
2012-08-17	93.93	94.3	93.59	94.24	3069513.0
2012-08-20	94.0	94.17	93.55	93.89	1640008.0
2012-08-21	93.98	94.1	92.99	93.21	2302988.0

2.2 Network Architecture

Long Short-Term Memory (LSTM) network is an improved type of recurrent neural network (RNN). Compared to traditional RNN networks, LSTM can better handle the issue of long-term dependencies in sequential data. Its key feature lies in its ability to pass information between data at different time steps while avoiding the vanishing or exploding gradient problems commonly encountered in traditional RNN networks.

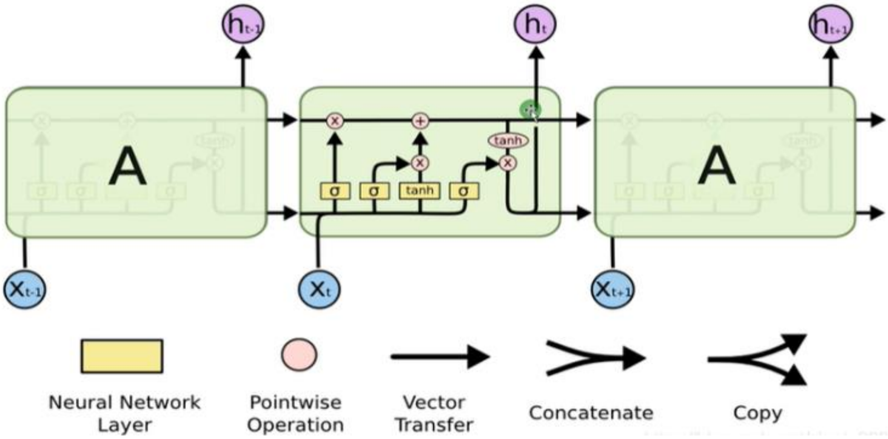

Fig. 1. Framework of LSTM

As shown in Figure 1, the LSTM network consists of three different types of gate mechanisms, including the forget gate, input gate, and output gate. The forget gate determines which information to discard from the previous time step's cell state.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

where W_f is the weight matrix of the forget gate, $h_{t-1}$is the hidden state at time t-1, x_t is the input at time t, and b_f is the bias matrix of the forget gate.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C_t} = tanh(W_c \cdot [h_{t-1}, x_t] + b_C \tag{3}$$

Next is the sigmoid layer of the input gate, which determines which values to update. The tanh layer creates a new candidate value vector $\widetilde{C_t}$, which can be added to the state.

Update the old cell state, C_{t-1} by entering the new state C_t. Then, multiply the old state by F_t to forget what was previously decided to forget and add the new candidate value.

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C_t} \tag{4}$$

Finally, the output gate begins with a sigmoid layer, which determines which parts of the cell state to output. Then, the cell state is passed through a tanh function (scaling values between -1 and 1) and multiplied by the output of the sigmoid gate.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

2.3 Training and Testing

In the initial setup of our training phase, we configured the neural network to run for a maximum of 100 epochs, with each batch containing 16 samples. We selected a learning rate of 0.001 to ensure a steady convergence. The network architecture includes a fully connected layer that projects the input data into a 128-dimensional feature space, facilitating complex pattern recognition. This model relies on the RMSProp optimization algorithm, which is particularly effective for maintaining a stable learning rate over numerous training cycles. Throughout the training, the primary metric for performance optimization is the mean squared error (MSE) loss, which quantifies the difference between the predicted and actual values.

During the testing phase, we modified the batch size to 4 to accommodate a more detailed analysis of the model's predictive capabilities on a smaller scale. The same computational techniques used in the training phase are applied here to assess the model's performance under test conditions. This process involves generating predictions based on the test dataset and comparing them against the true outcomes to evaluate accuracy.

The entire experiment was conducted using Pytorch version 2.2.1, a robust framework for building deep learning models, on an Apple M2Pro laptop. The coding was implemented in Python version 3.11.5, chosen for its compatibility and performance efficiency in handling advanced computational tasks associated with deep learning algorithms. This setup provides a well-equipped environment for both developing and testing the neural network model, ensuring the reliability and reproducibility of the results.

3 Experiments

3.1 Quantitative Results of the Experiment

Table 2. Comparison results on the evaluation stocks.

Method	MSE(Google)	MSE(Apple)	MSE(Amazon)
Linear Regression[11]	0. 281	0. 313	0.301
Polynomial Regression[12]	0. 273	0.308	0.294
Random Forest[13]	0. 100	0. 117	0. 138
Ours	**0.081**	**0.093**	**0.086**

As shown in Table 2, we compare our method with Linear Regression, Polynomial Regression, and Random Forest methods. The comparison experiments are conducted using data of the same dimension for testing. From the experimental results, it can be observed that compared to the currently best-performing method, Random Forest, our method achieves a lower MSE on the three mentioned stocks by 0.019, 0.024, and 0.052 respectively. The results demonstrate that, compared to traditional methods, the LSTM network based on deep learning can more accurately predict stock trends, validating the effectiveness of our approach.

3.2 Qualitative Results of the Experiment

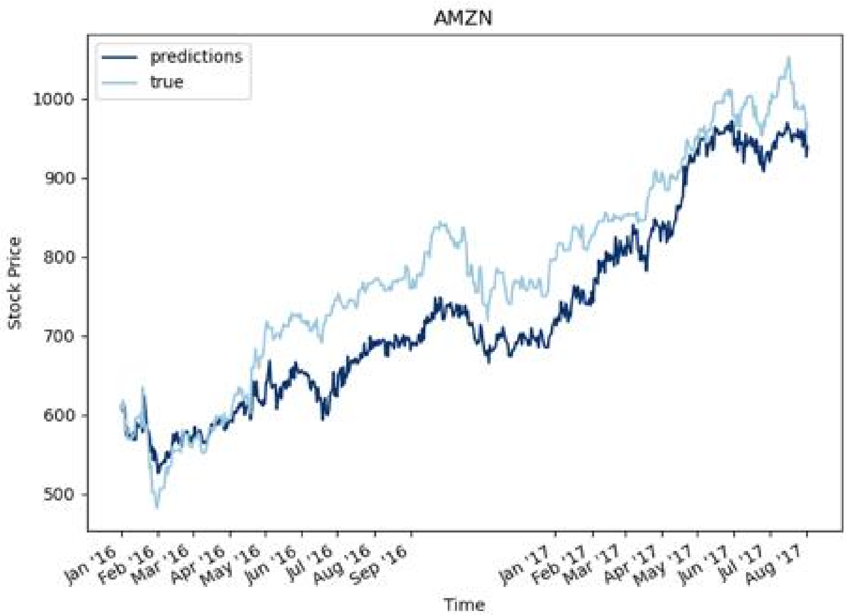

Fig. 2. Prediction results of our method on three evaluation stocks

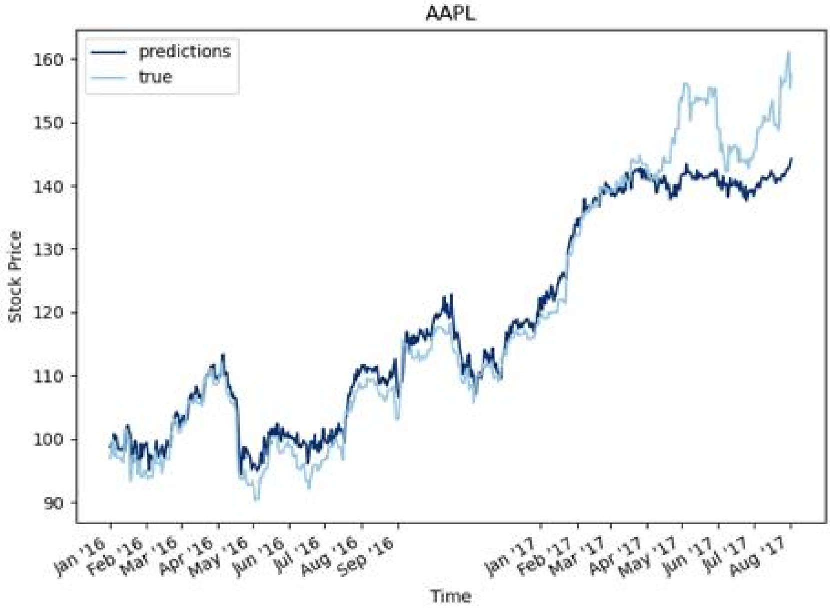

Fig. 3. Prediction results of our method on three evaluation stocks

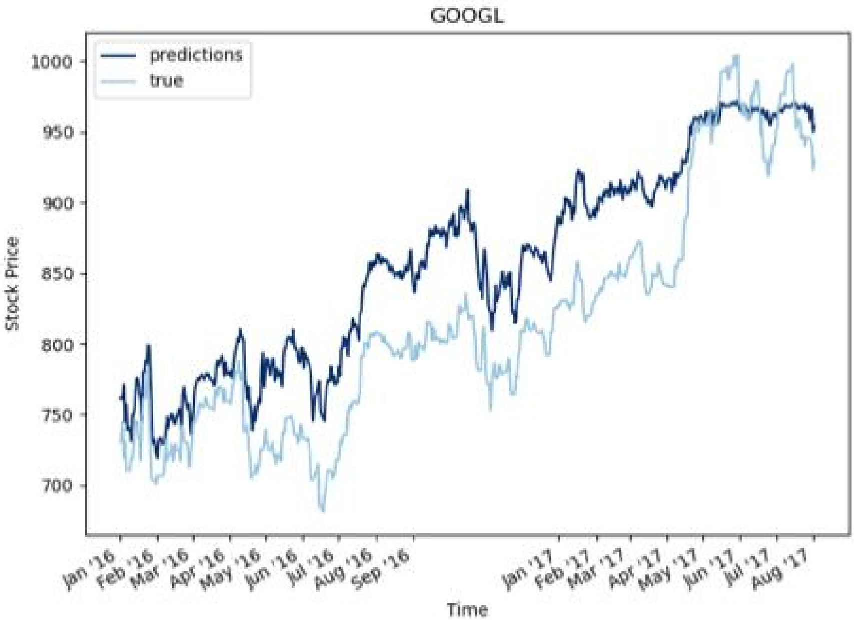

Fig. 4. Prediction results of our method on three evaluation stocks

As illustrated in Figure 2, our method demonstrates a robust capability to accurately predict stock price trends. This assertion is particularly evident when examining the results for Apple stock. The data reveals that our method not only captures the numerical values with high precision but also accurately reflects the overarching trends in the stock's movement. This level of accuracy is not limited to Apple stock alone; Referencing Figures 3 and 4, similar results were observed for Google and Amazon stock. The method's predictive performance on these diverse stocks indicates its generalizability and robustness across different market entities.

The precision in predicting Apple stock trends underscores the effectiveness of our LSTM-based model. The model successfully identifies and learns from the historical data patterns, enabling it to forecast future price movements with minimal error. This capability is crucial for stock market applications where even slight inaccuracies can lead to significant financial implications. The ability to predict both the exact numerical values and the general trend direction provides a comprehensive tool for investors, allowing for more informed decision-making processes.

Furthermore, the consistent performance of our method across multiple stocks showcases its versatility. By accurately predicting the trends of Google and Amazon stocks, our method proves its applicability to various market sectors. This cross-sector applicability is essential for developing a reliable stock prediction system that can be utilized in diverse financial environments. The method's ability to maintain high accuracy levels across different stocks suggests that it effectively captures the underlying market dynamics, which are often complex and multifaceted.

The experimental results presented in Figure 2 are indicative of the practical applicability of our method in real-world scenarios. The ability to predict stock prices with such accuracy can significantly benefit portfolio management, trading strategies, and risk assessment. For instance, accurate trend predictions can aid in optimizing entry and exit points for trades, thus maximizing returns and minimizing risks. Additionally, financial institutions can leverage this predictive power to enhance their market analysis and forecasting capabilities, leading to better-informed strategic decisions.

In conclusion, the experimental outcomes not only validate the theoretical underpinnings of our approach but also highlight its practical relevance. The method's robust performance across different stocks confirms its potential as a powerful tool for stock price prediction in real-world financial markets. This validation opens avenues for further research and development, aiming to refine and enhance the predictive accuracy and applicability of deep learning models in financial forecasting. Our findings advocate for the broader adoption of advanced deep learning techniques in stock market analysis, promising improved accuracy and efficiency in predicting market movements.

3.3 Experimental Analysis

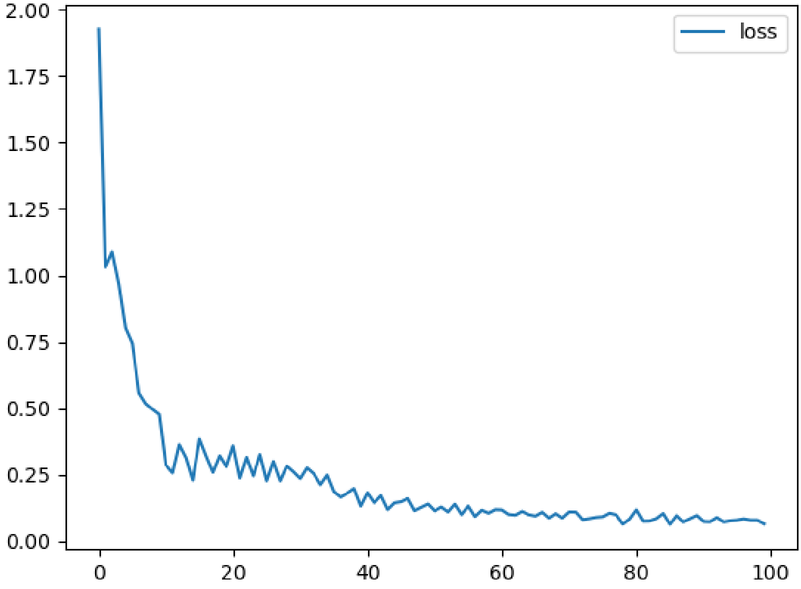

Fig. 5. Optimization curve of our method.

As shown in Figure 5, the loss curve illustrates the variation in the loss function during the training process of the deep learning model. The curve demonstrates a gradual decrease in the loss value as the number of training epochs increases. The horizontal axis represents the number of training epochs, while the vertical axis represents the loss value. According to the data depicted in the figure, the loss value decreases rapidly during the initial iterations, indicating that the model effectively learns the features from the data during the early stages. As the number of epochs increases, the loss value gradually stabilizes and remains at a low level, demonstrating the model's convergence process. This suggests that the model's learning capability becomes more stable in later stages, reducing the risk of overfitting.

The figure can be divided into three distinct stages. In the first stage (approximately the first 10 epochs), the loss value decreases rapidly, indicating that the model successfully learns the key features from the data. In the second stage (between 10 and 50 epochs), the rate of decrease in the loss value slows down, showing that the model is approaching its optimal parameter configuration. In the final stage (from 50 to 100 epochs), the loss value remains relatively stable, suggesting that the model has reached its learning limit, and further training provides limited improvement in the model's performance.

In summary, this figure demonstrates the process of the model progressing from an initial state to final convergence, indicating that the model has acquired a solid learning capability through training and does not exhibit significant signs of overfitting.

4 Conclusion

In this research, we explored the application of deep learning techniques to the challenging and dynamic problem of stock prediction, a field that is critically important within data science and has profound implications for stock investors and issuing companies alike. Our study leverages stock data spanning from January 1, 2013, to December 31, 2015, for training purposes, and from January 1, 2016, to August 1, 2017, for testing. The selected timeframes ensure that the model is trained on comprehensive and diverse market conditions, enhancing its robustness and predictive power.

We implemented a two-layer Long Short-Term Memory (LSTM) network, a type of recurrent neural network (RNN) known for its exceptional ability to model sequential data. LSTM networks are particularly suited for time series data due to their capability to remember patterns over long durations and manage the vanishing gradient problem effectively. This architecture allows our model to capture intricate stock trends and dependencies that simpler models might miss.

The effectiveness of our approach was evaluated using three high-profile stocks: Google (GOOGL), Apple (AAPL), and Amazon (AMZN). These stocks were chosen for their market significance and diverse behavior patterns, which provide a rigorous test for the model. The mean squared error (MSE) values achieved were 0.081 for Google, 0.093 for Apple, and 0.086 for Amazon. These results are indicative of the model's accuracy, suggesting that it can predict stock prices with a high degree of precision.

Our findings underscore the practicality and efficacy of using LSTM networks for stock prediction. The model's performance highlights several key advantages:

1. Modeling Complex Dependencies: The LSTM architecture excels at capturing long-term dependencies in stock price movements, which are crucial for accurate predictions. This is particularly important in stock markets, where historical data can have a significant impact on future trends.

2. Robustness to Market Variability: By training on a diverse dataset, the model learns to handle various market conditions, from bullish to bearish trends. This adaptability is critical for real-world applications, where market conditions are rarely stable.

3. Speed and Efficiency: The proposed method can rapidly process large datasets and generate predictions, making it suitable for real-time applications. This efficiency is vital for investors and companies that rely on timely and accurate predictions to make informed decisions.

4. Versatility Across Stocks: The consistent performance across different stocks demonstrates the model's versatility. Whether dealing with tech giants like Google, Apple, or Amazon, the model can adapt and provide reliable predictions.

5. Practical Implications: For investors, accurate stock predictions can significantly enhance portfolio management strategies, mitigate risks, and optimize returns. For issuing companies, understanding stock trends can inform financial strategies, investor relations, and market positioning.

Looking ahead, there are several avenues for further research and improvement. Incorporating additional features such as market sentiment data, macroeconomic indicators, and sector-specific trends could enhance the model's accuracy and robustness. Exploring advanced architectures, such as Transformer models, might also yield improvements in capturing complex temporal dependencies.

Moreover, extending the prediction horizon beyond short-term trends to include medium- and long-term forecasts could provide additional value for strategic planning. Implementing ensemble methods that combine multiple models might further enhance prediction accuracy and stability.

In conclusion, this work demonstrates that deep learning, particularly LSTM networks, offers a powerful tool for stock prediction. The promising results achieved with Google, Apple, and Amazon stocks suggest that these techniques can be broadly applied across various sectors and market conditions. By continuing to refine these models and integrating more diverse data sources, we can move closer to achieving highly reliable and actionable stock market predictions, ultimately benefiting both investors and companies in navigating the complex financial landscape.

References

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.
2. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.
3. Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., ... & Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR), 51(5), 1-36.
4. Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architectures. IEEE Access, 7, 53040-53065.
5. Pahwa, N., Khalfay, N., Soni, V., & Vora, D. (2017). Stock prediction using machine learning a review paper. *International Journal of Computer Applications*, *163*(5), 36-43.
6. Nandi, B., Jana, S. and Das, K.P. (2023). Machine learning-based approaches for financial market prediction: A comprehensive review. Journal of AppliedMath. 1, 2 (Aug. 2023), 134. DOI:https://doi.org/10.59400/jam.v1i2.134.
7. Chen W, Zhang H, Mehlawat MK, Jia L. Mean-variance portfolio optimization using machine learning—Based stock price prediction. Applied Soft Computing 2021; 100: 106943. doi: 10.1016/j.asoc.2020.106943.
8. Jishtu P, Prajapati H, Fiaidhi J. Prediction of the stock market based on machine learning and sentiment analysis. Available online: https://www.techrxiv.org/articles/preprint/Prediction_of_the_Stock_Market_Based_on_Machine_Learning_and_Sentiment_Analysis/21692852 (accessed on 16 June 2023).
9. Soni VK, Srivastava D. The use of supervised text classification techniques: A comprehensive study. In: Proceedings of the 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE 2022); 28–29 April 2022; Greater Noida, India.
10. Subasi A, Amir F, Bagedo K, et al. Stock market prediction using machine learning. Procedia Computer Science 2021; 194: 173–179. doi: 10.1016/j.procs.2021.10.071
11. Weisberg, S. (2005). *Applied linear regression* (Vol. 528). John Wiley & Sons.
12. Bradley, R. A., & Srivastava, S. S. (1979). Correlation in polynomial regression. *The American Statistician*, *33*(1), 11-14.
13. Rigatti, S. J. (2017). Random forest. Journal of Insurance Medicine, 47(1), 31-39.
14. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

15. Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and electronics in agriculture, 147, 70-90.
16. Buduma, N., Buduma, N., & Papa, J. (2022). Fundamentals of deep learning. " O'Reilly Media, Inc.".
17. Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. Electronic Markets, 31(3), 685-695.
18. Alkafaween, E., Hassanat A.B., Tarawneh, S. (2021). Improving Initial Population for Genetic Algorithm using the Multi Linear Regression Based Technique (MLRBT). Communications-Scientific Letters of the University of Zilina, 23 (1) (2021), pp. E1-E10.
19. Gornall, W., Strebulaev, I.A. (2022). The Contracting and Valuation of Venture Capital-Backed Companies. Forthcoming, Handbook of the Economics of Corporate Finance, 1 (2022).
20. Y. Li, H. Bu, J. Li, J. Wu. (2020). The role of text-extracted investor sentiment in Chinese stock price prediction with the enhancement of deep learning International Journal of Forecasting, 36 (4) (2020), pp. 1541-1562, 10.1016/j.ijforecast.2020.05.001.