International Journal of Informatics and Computation (IJICOM)
Vol. 5, No.1, August 2023
ISSN: 2685-8711, E-ISSN: 2714-5263 DOI : 10.35842/ijicom

Robust Stock Price Prediction using Gated Recurrent Unit (GRU)

Hamzah[1], Poly Endrayanto Eko Chrismawan[2], Sugeng Winardi[3] , Rainbow Tambunan[4]

Abstract

Forecasting the direction of price movement of the stock market could yield significant profits. Traders use technical analysis, which is the study of price by scrutinizing past prices, to forecast the future price of the nickel stock price. Therefore, in this study, we propose Gated Recurrent Units (GRU) to predict nickel stock price trends. This research aims to produce an accurate nickel stock price trend prediction model. The research method utilized historical data on nickel stock prices from Yahoo Finance. The research results show that the model developed accurately predicted nickel stock price trends. From the RMSE, MAE, and MSE analysis results, the RMSE value was 0.0123, the MAE value was 0.0089, and the MSE value was 0.0002 on the test data.

Keywords:
Deep Learning, GRU, Stock Price, Prediction.

1. Introduction

Stock price prediction is challenging, as it depends on numerous factors, including market dynamics, news events, and investor sentiment. Stock market investment is essential to the global financial system, with individuals and institutions relying on accurate forecasting techniques to guide their investment decisions. The ability to predict future stock prices has been a topic of interest and research for many years. Various methods, from traditional statistical models to machine learning algorithms, have been developed and applied to stock price prediction [1].

Moving averages have long been used as a simple and intuitive technique for identifying trends in stock prices. By calculating the average of a specified number of past prices, they provide a smoothed representation of the price movement, allowing traders to identify potential buying or selling opportunities based on patterns in the data. However, MA provides a simplified model of stock price trends. Still, it is based solely on historical data and does not account for other factors that could influence the market, such as news events or changes in market sentiment. Furthermore, the choice of the moving average period can significantly impact the accuracy and timing of predictions, making it a subjective decision for traders and investors [2].

Linear regression, on the other hand, aims to model the relationship between an independent variable (such as time) and a dependent variable (stock prices) using a

Corresponding Author: Hamzah[1] (hamzah@respati.ac.id), Universitas Respati Yogyakarta
1. Sugeng Winardi, Universitas Respati Yogyakarta
2. Poly Endrayanto Eko Chrismawan, Universitas Respati Yogyakarta
3. Rainbow Tambunan, Universitas Respati Yogyakarta

straight line. This technique assumes a linear relationship between the variables and seeks to estimate the best-fit line that minimizes the errors. Linear regression takes a linear relationship between the independent and dependent variables, which may not be valid for all stock price movements. Stocks often exhibit non-linear patterns with sudden spikes or periods of high volatility, making linear regression less accurate in capturing these complexities [3].

Autoregressive Integrated Moving Average (ARIMA) models are widely used in time series analysis and forecasting. ARIMA models incorporate the concepts of autocorrelation (the relationship between a variable and its past values) and differencing (transforming the data to remove trends or seasonality) to model and predict future values. However, ARIMA models suffer from certain limitations as well. These models assume that the data is stationary, meaning that the statistical properties of the data remain constant over time. However, stock prices often exhibit non-stationarity, with trends and seasonality, requiring additional data transformations or more sophisticated models to capture these patterns accurately [4].

GARCH (Generalized Autoregressive Conditional Heteroskedasticity) models have been introduced to capture the volatility clustering and time-varying volatility observed in financial time series. These models allow for examining the conditional variance of stock prices, providing insights into the risk and uncertainty associated with price movements. GARCH models capture the volatility clustering and time-varying nature of stock prices. However, these models are based on the assumption that volatility is the only source of risk in the market, neglecting other factors that may influence stock price movements, such as external shocks or changes in market structure. Additionally, GARCH models tend to struggle to predict extreme stock market events or sudden shifts in volatility, which can lead to inaccurate predictions during periods of market turbulence [7].

SVM is a machine learning algorithm that aims to find the optimal hyperplane that separates data points into different classes. SVM has been employed successfully in various prediction tasks, including stock price forecasting, by transforming the original data into higher-dimensional feature spaces. SVM are powerful machine learning algorithms, but they can be computationally intensive and require careful tuning of parameters to achieve optimal performance. SVMs may also struggle when faced with highly noisy or complex data, as they rely on finding the best-separating hyperplane, which becomes more challenging as the data becomes more irregular [6].

Given the myriad of techniques available for stock price prediction, it is essential to compare and evaluate their effectiveness to identify the most reliable approaches. This study aims to provide a comprehensive comparative analysis of moving averages, linear regression, ARIMA models, GARCH models, and SVMs regarding their accuracy, robustness, and suitability for predicting stock prices. By identifying the strengths and weaknesses of each technique, this research seeks to contribute to developing more reliable and effective stock price prediction models, ultimately assisting investors in making informed decisions. These challenges and limitations highlight the need for further research and improvement in stock price prediction techniques. By addressing these problems and developing more robust models, investors and traders can have access to more accurate and reliable predictions, enabling them to make better-informed investment decisions [10].

Several studies have been conducted to predict stock price trends using various methods, including learning techniques. In this paper, we proposed GRU as a deep-learning method to predict stock price trends. GRU has advantages in overcoming vanishing gradient problems and short-term and long-term memory [4]. However, other ways, such as LSTM and CNN, have been widely used in research to predict stock price trends. This research aims to predict nickel stock price trends using the GRU and compare the results with other methods. Stock price prediction using GRU is a popular application of deep learning in finance. GRU is a type of recurrent neural network (RNN) widely used for sequential data analysis and has shown promising results in time series forecasting tasks [9].

2. Related Works

Stock price prediction is a growing research area using traditional techniques involving statistical and econometric models. Traditional methods commonly include Moving Averages, Linear Regression, Autoregressive Integrated Moving Average (ARIMA), GARCH, and Support Vector Machines [9][10][11][18]. Moving Averages is a technique that calculates the average price of a stock over a specified period. It identifies trends and predicts future price movements based on historical averages [4]. Linear regression is a statistical model that assumes a linear relationship between the dependent variable (stock price) and one or more independent variables (such as time, volume, or market factors). It estimates the coefficients of the independent variables to predict future stock prices [5].

ARIMA is a popular time series model used for stock price forecasting. It incorporates three components: autoregressive (AR), moving average (MA), and differencing (I). ARIMA models capture patterns, trends, and seasonality in time series data to produce forecasts [6]. GARCH Models: GARCH (Generalized Autoregressive Conditional Heteroskedasticity) models are commonly used to predict stock market volatility. GARCH models capture the variance and volatility clustering in financial time series to predict future volatility, which can indirectly inform stock price movements [7]. Support Vector Machines (SVM): SVM is a supervised machine learning algorithm that can be applied to stock price prediction. It constructs a hyperplane that separates different classes of data, based on which it can predict whether the price will increase or decrease [8].

A study proposed MA as a technical analysis tool that helps the trader see the trend and identify key price points for stock trading. While a trend indicator, the moving average is also a lagging indicator. A lagging indicator is a financial signal that is given only after a large price shift has occurred. The paper adopted ML techniques on a technical indicator. The proposed model will apply Regression on Moving averages to reduce the latency of the trading signal given and thus overcome its drawback. The model can predict the reversal by predicting the trading signal provided by the moving averages [2].

Another study presented an extensive process of building a stock price predictive model using the ARIMA model. Published stock data from the New York Stock Exchange (NYSE) and the Nigeria Stock Exchange (NSE) are used with a developed stock price predictive model. Results revealed that the ARIMA model has a strong potential for short-term prediction and can compete favorably with existing techniques for stock price prediction [11]. A paper explored the GARCH model to predict stock prices using two steps. First is the least square (LS) method, and the smallest absolute deviation (LAD) is considered to identify a correlation between mean and median. The GARCH model is proposed to calculate the error distribution in stock returns based on the correlation between mean and median. They have varied the degree of t-distribution parameters to identify the appropriate error distribution. In the second step, the volatility of stock prices is given as input to the GARCH model to forecast future crisis events. They have considered Infosys and SBI stock to carry out the proposed experiment. Experiment results reduce the error in predicting stock crisis events [12].

Several traditional techniques have been explored in finance for a long time and provide a foundation for stock price prediction. However, they may not capture complex patterns or sudden changes in the market as effectively as deep learning models. Combining traditional techniques with more advanced approaches to improve prediction accuracy is often beneficial. Additionally, market knowledge, fundamental analysis, and other external factors should be considered for better decision-making. Therefore, it requires more sophisticated approaches to deal with the issues [8].

Current communities explored deep learning to address stock price prediction. A paper proposed a deep learning technique to predict the stock market. Since RNN has the advantage of processing time series data, it is very suitable for forecasting stocks. Therefore, they employed use the RNN to predict network stock price in the past ten years. Experiments show that the prediction accuracy is over 95%, and the loss is close to 0.1% [13]. Another study proposed the LSTM model to examine the future price of a stock. The paper predicted stock market prices to make more acquainted and precise investment decisions [14].

The current paper explored a stock price prediction model based on CNN, which has obvious self-adaptability and self-learning ability. Combining the characteristics of CNN and the Thai stock market, the data set is trained and tested after pretreatment. On this basis, three stocks (BBL, CAPLL&PTT) listed on the Thai Stock Exchange are tested and compared with the actual stock price. The results show that the model based on CNN can effectively identify the changing trend of stock price and predict it, which can provide a valuable reference for stock price forecast [15]. Another paper discussed CNN mode for an experimental study on the S&P 500 index. The CNN model structure transferred from inception v3 with three additional layers, and the technical indicators used in the input chart image are simple moving averages (25 days). The label data used in the model are categorical - either up, flat, or down. The model has 50% accuracy on the test set when conducting a three-day ahead forecast, which is higher than the simple momentum and contrarian strategies, indicating its high alpha-generating potential [16].

3. Proposed Method

GRU is a recurrent neural network (RNN) developed to overcome the vanishing gradient problem often occurring in traditional RNN networks. The GRU (Gated Recurrent Unit) model is a type of recurrent neural network (RNN) that aims to address the vanishing gradient problem faced by traditional RNNs. It is particularly effective in sequence modeling tasks, including natural language processing, speech recognition, and time series forecasting [17].

Basics of Recurrent Neural Networks (RNNs): RNNs are designed to process sequential data by maintaining internal memory. They operate on one input at a time from a sequence and use the previous hidden state to capture information about the preceding inputs. This allows them to handle variable-length inputs. Hidden state and Update Gate: Like traditional RNNs, the GRU maintains a hidden state. However, it introduces an additional component called the "update gate." This gate determines how much past hidden state information should be passed to the current time step. It takes the current input and the previous hidden state as inputs and outputs a value between zero and one.

The GRU also introduces a "reset gate" that decides how much of the previous hidden state should be forgotten. It takes the current input and the previous hidden state as inputs, similar to the update gate. The reset gate outputs a value between zero and one. The reset gate resets the previous hidden state, creating a new "candidate hidden state." This candidate's hidden state blends the previous and current input. The reset gate determines the proportion of each component. The candidate's hidden state is combined with the update gate to produce the new one. Depending on the specific task, the hidden state of the final time step or a specific intermediate time step. It can be fed into a fully connected layer for further processing and prediction. Fig. 1 shows the architecture of the GRU.

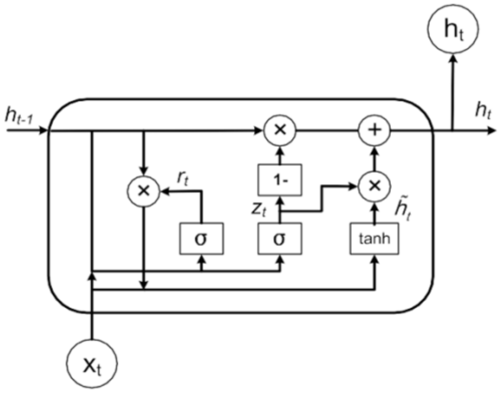

Fig. 1 Architecture of GRU with the hidden state of the final time step

GRU has three gates in doing computing, namely: (1) an update gate, in equation one to help the model determine how much information needs to be continued to the next stage;

$$z_t = \sigma(w_z x_t + u_z h_{t-1} + b_z) \qquad (1)$$

(2) Reset the gate in equation two to decide how much previous information to delete.

$$r_t = \sigma(w_r x_t + u_r h_{t-1} + b_r) \qquad (2)$$

(3) current memory content in equation three, where a reset gate will be used to restore information from a previous time.

$$h^{\sim} = \tan h\,(w_h x_t + r_h * u_h h_{t-1} + b_h) \quad (3)$$

And (4) the final memory in equation four uses the update gate to store the current unit information and the information from the previous step to connect to the following network.

$$h_t = z_t * h_{(t-1)} + (1 - z_t) * h^{\sim} \qquad (4)$$

GRU can collect relationships across temporal scales and features gating units that influence the flow of information within the unit but without the need for separate memory cells. The GRU model enhances traditional RNNs by introducing update and reset gates. These gates control the flow of information within the network, allowing the model to retain or forget information from the past selectively. The GRU addresses the vanishing gradient problem by selectively updating the hidden state and performing sequence modeling tasks better.

4. Experimental Setup

A stock price dataset consists of historical data for a specific stock, including the date, opening price, closing price, highest price, lowest price, and trading volume. It records how the stock's price has changed and is used to analyze trends, predict future price movements, and evaluate investment strategies. These datasets are available from financial data providers and can be used in finance and investment for various purposes. To conduct this experiment, we gathered a stock price dataset from Yahoo Finance, including the date, opening price, lowest price, highest price, and closing price of PT Aneka Tambang Tbk (ANTM.JK). To employ GRU for stock price prediction, we undergo several stages:

1. Data Preparation: Gather historical stock price data, along with relevant features, such as trading volume, indicators, and news sentiment. Split the data into training and testing datasets.
2. Data Preprocessing: Normalize or scale the input data to ensure all features are comparable. This step helps the model to converge faster during training.
3. Model Architecture: Design a GRU-based neural network for stock price prediction. The GRU model takes sequences of historical data as input and learns to capture patterns and dependencies in the data. It can have multiple layers of GRU units and other layers, such as dense layers, for prediction.
4. Training: Train the GRU model on the training dataset. During training, the model adjusts its parameters to minimize the difference between predicted and actual stock prices. The loss function used for training can be mean squared error (MSE) or other appropriate regression loss functions.
5. Prediction: Once the model is trained, use it to make predictions on the testing dataset or real-time data. The predicted stock prices can be compared with the actual prices to evaluate the model's performance.

5. Result and Analysis

The GRU can obtain the best trend and momentum using the dataset based on the experimental result. The GRU fluctuates above and below the zero line as the moving averages converge, cross, and diverge. Hence, traders can look for signal line crossovers, centerline crossovers, and divergences to generate signals. Historical data sequences are a series of stock prices from the past, which usually consist of a closing price within a specified period, such as a day, week, or month. The length of historical data sequences has an essential impact on the performance of stock price prediction models.

The number of periods used as input will affect the ability of the model to capture patterns and trends in the data. Shorter sequence lengths may be better at identifying short-term trends, while longer sequences may be better at identifying long-term trends. However, remember that using historical data sequences that are too short can lose important context in long-term trend analysis. At the same time, too long sequences can make the model more complex and prone to overfitting. Therefore, selecting the length of historical data sequences must be a careful consideration, which must reflect the purpose of analysis and the characteristics of the data you have. In this experiment, we tune the decision between using a small or large 'seq_length' depending on the features of the data. Both have advantages and disadvantages to consider:

Small 'seq_length':
1. Excess:
   - Allows models to more quickly recognize simpler and clearer patterns in data.
   - Minimizes the problem of complex calculations and higher memory.
   - It may be better suited for recurring short-term trends in data.
2. Lack:
   - May miss longer and more complex patterns in the data.
   - Unable to catch long-term trends well.

Big 'seq_length':
1. Excess:
   - Capable of capturing long-term patterns and trends in data.
   - Can handle larger variations in trends.
2. Lack:
   - It requires more computation and memory, which can constrain training time.
   - It is more likely to overfit the training data.

A comparison table of the various 'seq_length' values with the evaluation metrics MAE, MSE, MAPE, and PCC can help you understand how changing the length of the input data sequence affects model performance. Table 1 depicts MAE, MSE, MAPE, and PCC evaluation metrics with different Sequence Lengths.

Table 1: MAE, MSE, MAPE, and PCC evaluation metrics with different Sequence Length

Seq.Length	MAE	MSE	MAPE	PCC
15	0.0181	0.0007	2.94%	0.9743
30	0.0215	0.0011	3.45%	0.9567
60	0.0257	0.0017	4.12%	0.9251
90	0.0314	0.0025	5.05%	0.8846

From the comparison table that you provide, we can draw the following conclusions:
1. In terms of Mean Absolute Error (MAE) and Mean Squared Error (MSE):
   - The model with 'seq_length' 15 performs best regarding MAE and MSE.
   - Increasing 'seq_length' tends to increase MAE and MSE values, indicating that the model may face difficulties accurately predicting when the input data sequence is extended.
2. In terms of Mean Absolute Percentage Error (MAPE):
   - The model with 'seq_length' 15 has the lowest MAPE value, indicating that the model prediction has the lowest percentage error of the actual value.
   - MAPE values increase as 'seq_length' increases, indicating a more significant percentage error in long-term predictions.
3. In terms of Pearson Correlation Coefficient (PCC):

- The model with 'seq_length' 15 has the highest PCC value, indicating a strong correlation between predicted results and actual values.
- PCC values tend to decrease as 'seq_length' increases, which could indicate that the model may have difficulty maintaining strong correlations in long-term predictions.

Based on evaluating the given metrics, a smaller 'seq_length' (15) provides better short-term prediction accuracy and correlation between prediction and actual. However, deciding the optimal 'seq_length' value must consider your analysis objectives, data characteristics, and the trade-off between short-term accuracy and performance. Fig. 2 depicts the Prediction Chart for the Next 30 Days.

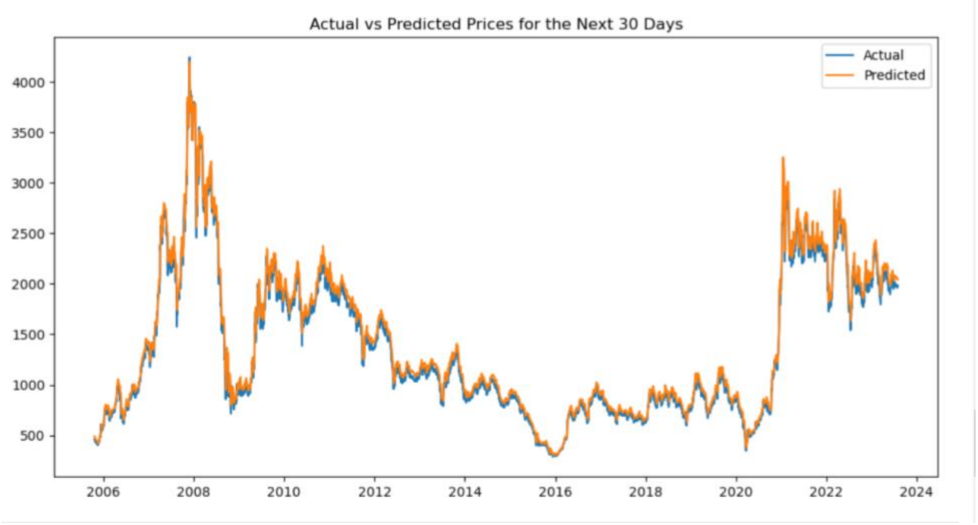

Fig. 2 Prediction Chart for Next 30 Days.

The overall model performance evaluation results are very positive:

- A low MSE value indicates that the model accurately forecasts stock prices.
- Low MAE and MAPE indicate that prediction errors are generally limited, and the percentage of errors is also low.
- A high Pearson correlation indicates that the model tends to follow the actual trend of stock price movements.

MSE is the sum of squared difference between the values that are fitted by the model, and observed values that are divided by the number of historical points, minus the number of parameters in the model. The number of parameters in the model is subtracted from the number of historical points to be consistent with an unbiased model variance estimate. MAE computed as the average absolute difference between the values fitted by the model (one-step ahead in-sample forecast), and the observed historical data. MAPE is the average absolute percent difference between the values that are fitted by the model and the observed data values. Table 2 depicts the Classification Report

Table 2: Classification Report of Proposed Model

	Precision-Recall		F1-Score	Support
0	1.00	1.00	1.00	130
1	1.00	1.00	1.00	69
accuracy			1.00	199
macro avg	1.00	1.00	1.00	199
weighted avg	1.00	1.00	1.00	199

The results from the classification report show that the model you have trained performs very well in classification. All metrics (precision, recall, and F1-score) have a value of 1.00, which indicates that the model can classify data perfectly. This may indicate problems or overfitting, especially if the dataset has a significant class imbalance.

6. Conclusion

Stock market predictions help investors benefit in the financial markets. Various papers have proposed different techniques in stock market forecasting, but no model can provide accurate predictions. In this study, we developed a stock price prediction model using a GRU architecture. To create a stock price prediction model, we collect the dataset, preprocess it, extract features, evaluate the model, and then deploy the GRU method to predict stock prices in real time. The research results show that the model developed accurately predicted nickel stock price trends. From the RMSE, MAE, and MSE analysis results, the RMSE value was 0.0123, the MAE value was 0.0089, and the MSE value was 0.0002 on the test data.

Acknowledgment

We are deeply grateful for the Universitas Respati Yogyakarta's support throughout this research. We are thankful for the guidance and opportunities provided by the university.

References

[1] Gandhmal, D.P., Kumar, K.: "*Systematic analysis and review of stock market prediction techniques*." Comput. Sci. Rev. 34, 100190, 2019.

[2] S. Dinesh, N. R. R, S. P. Anusha and S. R, "Prediction of Trends in Stock Market using Moving Averages and Machine Learning," 2021 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2021, pp. 1-5.

[3] Chan, W.N.: "*Time series data mining: comparative study of ARIMA and prophet methods for forecasting closing prices of Myanmar stock exchange*". J. Comput. Appl. Res. 1, 2020, pp: 75–80.

[4] Edwards, R. D., & Magee, J." *Technical Analysis of Stock Trends."* Journal of Technical Analysis, 56(3), 2001, pp: 224-239.

[5] Engle, R. F., & Sargan, J. D. C." *A Linear Market Model with Recursive Estimation of Predictive Coefficients.*", Journal of Financial Economics, 21(1), 2003, pp:135-171.

[6] Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. "*Time Series Analysis: Forecasting and Control."* Hoboken, NJ: Wiley. 2015.

[7] Bollerslev, T. “*ARCH Modeling in Finance: A Review of the Theory and Empirical Evidence*”. Journal of Econometrics, 31(3), 1986, pp: 5-59.

[8] Nakanishi, M., Morimoto, Y., & Morishita, S.” *A Self-Adaptive Support Vector Machine and Time Series Prediction*.” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 36(1), 2006, pp:101-107.

[9] Yan, H., Ouyang, H.: “*Financial time series prediction based on deep learning. Wireless Pers. Commun*.” 102, 2018, pp:683–700.

[10] Ou, J.A., Penman, S.H.: “*Financial statement analysis and the prediction of stock returns. J. Account”,* Econ. 11, 1989, pp: 295–329.

[11] A. A. Ariyo, A. O. Adewumi and C. K. Ayo, "*Stock Price Prediction Using the ARIMA Model*," 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 2014, pp. 106-112, doi: 10.1109/UKSim.2014.67.

[12] Nagaraj Naik, Biju R Mohan, Rajat Aayush Jha, “*GARCH Model Identification for Stock Crises Events*,” Procedia Computer Science, Vol. 171, 2020, pp 1742-1749.

[13] Yongqiong Zhu, “*Stock price prediction using the RNN model*” J. Phys.: Conf. Ser. 1650 032103 2020, DOI 10.1088/1742-6596/1650/3/032103

[14] K. J, H. E, M. S. Jacob and D. R, "*Stock Price Prediction Based on LSTM Deep Learning Model*," 2021 International Conference on System, Computation, Automation and Networking (ICSCAN), Puducherry, India, 2021, pp. 1-4, doi: 10.1109/ICSCAN53069.2021.9526491.

[15] L. Sayavong, Z. Wu and S. Chalita, "*Research on Stock Price Prediction Method Based on Convolutional Neural Network,*" 2019 International Conference on Virtual Reality and Intelligent Systems (ICVRIS), Jishou, China, 2019, pp. 173-176, doi: 10.1109/ICVRIS.2019.00050.

[16] Komori, Yohei, “*Convolutional Neural Network for Stock Price Prediction Using Transfer Learning*.”, Dec, 29, 2020.

[17] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y.”*Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation*”. arXiv preprint arXiv:1406.1078, 2014.

[18] Hamzah, et al, “*Effective Stock Prediction Model Using MACD Method.”,* International Journal of Informatics and Computation,Vol. 4, no. 2, dec. 2022, pp: 1-6.