International Journal of Computer, Network Security and Information System (IJCONSIST)
Vol. 7 No. 1, September 2025

Stock Price Prediction Using Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) Methods

Riza Akhsani Setyo Prayoga
Information Technology Education, Faculty of Engineering, State University of Surabaya
rizaprayoga@unesa.ac.id

Fery Almas Ariansyah
Department of Informatics, Faculty of Computer Science, University of Pembangunan Nasional Veteran Jawa Timur, Surabaya, East Java, Indonesia
20081010017@student.upnjatim.ac.id

Muhammad Falikhuddin Daffa
Department of Informatics, Faculty of Computer Science, University of Pembangunan Nasional Veteran Jawa Timur, Surabaya, East Java, Indonesia
20081010023@student.upnjatim.ac.id

Steven Angkawidjaja
Department of Informatics, Faculty of Computer Science, University of Pembangunan Nasional Veteran Jawa Timur, Surabaya, East Java, Indonesia
20081010092@student.upnjatim.ac.id

Laqma Dica Fitrani
Department of Digital Business, Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Surabaya, East Java, Indonesia
laqma_dica.bd@upnjatim.ac.id

Masti Fatchiyah Maharani
Department of Data Science, Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, East Java, Indonesia
masti_fatchiyah.sada@upnjatim.ac.id

Angga Lisdiyanto
Department of Informatics, Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Surabaya, East Java, Indonesia
angga_lisdiyanto.mti@upnjatim.ac.id

Abstract*—This research aims to improve the accuracy of stock price prediction through the application of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) methods, focusing on stocks from the Composite Stock Price Index (CSPI) referred to as the IDX Composite. The research process includes comprehensive steps, including data collection and preprocessing, dataset creation with emphasis on stock closing prices, and division of the dataset into training and test data. The LSTM and GRU models were designed with a recurrent layer and a Dense layer and then trained for 100 epochs with a batch size of 32. Model evaluation was performed by comparing key metrics such as Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE) on the test set. The EPOCH-RMSE graph provides an overview of the changes in the RMSE value during training. The best result of the LSTM model was achieved at the 96th epoch with RMSE 40.36, MSE 1385.97, and MAE 30.09, while GRU achieved peak performance at the 92nd epoch with RMSE 37.33, MSE 908.29, and MAE 25.42. In conclusion, GRU can be considered as a more effective option in predicting JCI stock prices based on performance evaluation using various metrics such as RMSE, MSE, and MAE.

Keywords*—*Stock price prediction, LSTM, GRU

I. INTRODUCTION

Investment in the stock market is one form of investment that is in great demand by the public because it has great profit potential [1]. However, investing in the stock market also has a high risk due to stock price fluctuations that cannot be predicted with certainty. Therefore, many studies have been conducted to develop accurate stock price prediction methods that can help investors in making investment decisions [2]. This research was conducted as the final project of the Informatics Research Course to develop a more accurate and effective stock price prediction method.

Several studies related to stock price prediction using the Long Short-Term Memory (LSTM) method have been conducted in recent years [3]. One of the studies conducted by used the LSTM method to optimize stock price predictions with technical analysis parameters. This study successfully developed an accurate prediction model with an RMSE of 5.20877667554 and a MAPE of 0.08658576985 [4]. In addition, the study also used JII daily stock price data from 2019 to 2021. The results showed that the LSTM model can provide good predictions using historical JII stock price data. However, in this study, there was no test whether the prediction results had obtained optimal results and were in accordance with the original stock price chart, because the use of existing secondary data should be possible.

There is also research from [5] which uses the Long Short-Term Memory (LSTM) method in building cryptocurrency price prediction models [6]. The test results indicate that the LSTM model yields accurate predictions, with an RMSE value of 0.0630 for the Dogecoin coin type [7]. This research does not provide information on how the data is separated into training data and test data, and how the LSTM model validation process is carried out. This is important because the proper data separation and validation process can affect the accuracy and reliability of the LSTM model [8].

Research by [9] used the Long Short-Term Memory (LSTM) method in building a cryptocurrency price prediction model. LSTM is considered superior compared to other algorithms in managing time series data [10]. The test results indicate that the LSTM model yields accurate predictions, with an RMSE value of 0.0630 for the Dogecoin coin type [10]. However, in this study, there is a lack of discussion about adjusting LSTM parameters, such as the number of hidden layers, epochs, batch sizes, and dropouts in the hidden layer, to improve model performance [11].

Research from [12] also tried to apply the LSTM (Long Short-Term Memory) method to predict cryptocurrency prices. In this study, the LSTM model was trained using Bitcoin and Ethereum price data from 2018 to 2020. The training results show that LSTM has good performance in predicting cryptocurrency prices, with high accuracy and low MAPE values. In this study, there are no parameter settings, such as the number of units, batch size, and epoch, to provide optimal results.

Research conducted by [13] on the use of the Gated Recurrent Unit (GRU) method in predicting cryptocurrency. The study found that GRU has faster training capabilities compared to Long Short-Term Memory (LSTM). The research shows that GRU can be a more efficient method for predicting cryptocurrency trends. One thing that is not discussed in this journal related to the LSTM method is the performance comparison between LSTM and GRU in handling time series data that has high fluctuations.

From the descriptions above, there are still many studies related to the LSTM and GRU methods, but the direct comparison between the two is not exploratory enough. Therefore, this study aims to fill the gap by evaluating the performance of the Gated Recurrent Unit (GRU) method and comparing it with the Long Short-Term Memory (LSTM) method in stock price prediction. This study hypothesizes that the use of the GRU method can produce better prediction performance compared to the LSTM method.

The introduction contains the purpose of the article/research that is formulated and presented by an adequate introduction and avoids detailed references and research result presentations. The research urgency, supporting facts, and data must be included. A preliminary research result should be explained as the basis of the research. Before mentioning the objective/s, a gap analysis must be elucidated. The gap analysis states the difference/s between the research and other previous studies. At this point, the novelty will be apparent. The research stance must be included, whether it corrects, debates, or supports the previous research.

II. Methodology

The research methodology used in this study can be described using the block diagram in Figure 1.

Figure 1. Block Diagram of Stock Price Prediction

This research begins with the collection of a dataset that includes stock prices with the ^JKSE index, also known as IDX Composite. After that, the dataset undergoes a preprocessing stage to optimize the raw data before it is applied to the machine learning model. The normalization process of the min-max method is performed to maintain the consistency of the data scale. The next step involves model building using LSTM and GRU. The processed data is then visualized and predicted using these models. Evaluation uses an evaluation matrix; in this research, the author uses RMSE, MSE, and MAE [14]. Through the implementation of this research, the author can compare the performance between the LSTM and GRU models to determine which model is superior to predicting future stock prices. The prediction results can be used as a reference by investors for making investment decisions.

A. Data Collection

This study will use daily stock prices from the Jakarta Composite Index (JCI), which reflects the average stock prices of various companies listed on the Indonesia Stock Exchange (IDX). The data is taken from Yahoo Finance with the code ^JKSE as index identification. The data used in this study involves a historical time span starting from October 08, 2021, to October 06, 2023. The top three and bottom three data structures that will be analyzed and explored in the study, these data structures are included to provide an initial overview of the data to be processed.

Table 1. The Dataset

No	Date	Open	High	Low	Close	Adj.Close	Volume
1	2021-10-08	6440.000000	6497.655762	6436.517090	6481.769043	6481.769043	231302300
2	2021-10-11	6482.957031	6506.091797	6451.678223	6459.696777	6459.696777	204779900
3	2021-10-12	6462.314941	6504.013184	6460.086914	6486.267090	6486.267090	204607300
⋮		⋮					
484	2023-10-04	6940.887207	6943.273926	6839.857910	6886.576172	6886.576172	196090300
485	2023-10-05	6886.985840	6934.804199	6874.826172	6874.826172	6874.826172	167955300
486	2023-10-06	6875.099121	6913.012207	6875.099121	6888.518066	6888.518066	142303700

Description:

Open = Opening price on the day
High = The highest price reached on that day
Low = The lowest price reached on that day
Close = Closing price on the day
Adj. Close = Adjusted closing price
Volume = Total number of shares traded that day

B. Data Preprocessing

This research uses Feature Scaling as a form of data preparation. Feature Scaling is a process of normalizing data to provide a consistent range of values. The feature scaling step is applied to the complete set of stock price data that has been adapted to the LSTM and GRU models. In this research, the feature scaling strategy used is min-max normalization. The data values are transformed by min-max normalization, resulting in a data range of [0, 1]. The equation of min-max normalization is [15].

$$x' = \frac{x - minimum(x)}{maksimum(x) - minimum(x)} \tag{2.1}$$

where x is the value in the dataset, x' is the value after normalizing to a range of 0 to 1.

If applied to the stock data contained in Table 1, it can be observed that the highest closing data is in data number 486, which will be the maximum(x) with a value of 6888.518066. Conversely, the smallest value in the close column, which will be the minimum(x), with a value of 6459.696777, can be found in data number 2.

After determining the smallest and largest values, the next process is to determine the x that will be normalized. In this case, the x value represents a certain value in the column that will be used as a variable for the normalization process using the min-max equation. For example, entry number 1 with a close value of 6481.769043 will be normalized. Substitute the given value,

$$x' = \frac{6481.769043 - 6459.696777}{6888.518066 - 6459.696777} \tag{2.2}$$

$$x' = \frac{22.072266}{428.821289} \tag{2.3}$$

$$x' = 0.0515 \tag{2.4}$$

The close value of entry number 1 after normalization is about 0.0515 These normalization steps are applied repeatedly to all stock price data. This triggers the consistency of close values so that the stock data is within a standardized range of values suitable for use in analysis and prediction.

C. Modelling

a) Long Short-Term Memory (LSTM)

According to (Nilsen, 2022) in his research, Long Short-Term Memory (LSTM) is an architecture in Recurrent Neural Network (RNN) used in the field of sequential data modeling. LSTM was created by Hochreiter and Schmid Huber in 1997 as a response to the long-term dependency problem faced by RNN. LSTM consists of several networks that will perform repetitive processes. In the study (Carnegie et al., 2023), explained that the LSTM module performs several mathematical calculations such as addition, subtraction, merging, vector multiplication, and the application of mathematical functions such as tanh and sigmoid. Unlike RNNs that connect neurons linearly and forward, LSTMs forward information and consider neurons that are behind them.

LSTM is adapted to overcome the problems that arise in RNN, especially the problem of losing important information at the beginning of the sequence when the sequence length reaches a certain level. RNN can experience vanishing gradients in the backward propagation phase, where the gradient value becomes so small that it does not contribute significantly to the gate. With LSTM, the solution is to add a state cell and three additional gates.

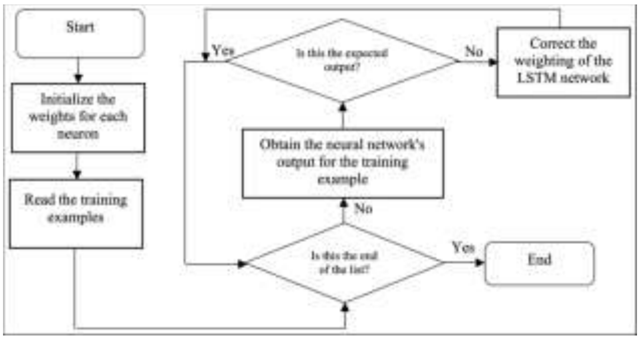

Figure 3. Flowchart of LSTM Method

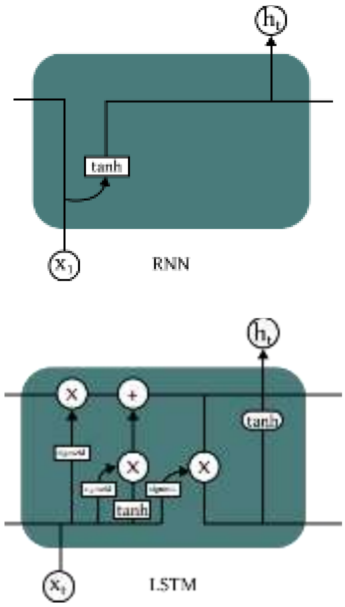

Figure 4. Comparison of RNN and LSTM

Forget Gate f_t is the initial part of the LSTM. The forget gate task is to calculate the amount of data to be transferred from the previous cell state (C_{t-1}) to the current cell state at time $t(C_t)$. Since the forget gate function has a sigmoid function (σ(-)), the forget gate output is limited to a value between 0 and 1. The forget gate equation is [15].

Forget Gate

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f) \qquad (2.5)$$

The input gate and candidate state comprise the second half of the LSTM. The input gate's job is to decide which values need to be updated. The candidate state's responsibility is to collect and update new data temporarily. The role of the candidate state is identical to the cell state in RNN. Candidate state exists in LSTM and is used to update cell state, whereas cell state in RNN passes information directly to the next cell. According to [15] the equation of input gate and candidate state is:

Input Gate

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \qquad (2.6)$$

Candidate State

$$\acute{C}_t = tanh(U_c x_t + W_c h_{t-1} + b_c) \qquad (2.7)$$

The fourth part is the cell state, the cell state is used to combine the calculation results of the input gate, candidate state, and forget gate. The equation of the cell state is [6]

Cell State

$$C_t = f_t * C_{t-1} + i_t * \acute{C}_t) \qquad (2.8)$$

Next, there is state h_t and output gate. The output gate functions to regulate how much information needs to be forwarded to the next cell. For state h_t has the function of delivering information to the next cell. The equation of state h_t and the output gate is in [15].

Output Gate

State h_t

$$h_t = o_t * tanh(C_t) \qquad (2.9)$$

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \qquad (2.10)$$

Explanation:

σ	= Sigmoid Function
tan$h$	= Hyperbolic Tangent Function
x_t	= Input Vector
h_{t-1}	= Hidden State Vector
$\acute{C}_t$	= Cell Input Activation Vector
C_t	= Cell State Vector
$U_{f,i,o,c}, W_{f,i,o,c}, b_{f,i,o,c}$	= Weighting Matrix

b) Gated Recurrent Unit (GRU)

According to [16]The Gated Recurrent Unit (GRU) was first introduced with the main purpose of overcoming the vanishing gradient problem that often arises in RNN. The vanishing gradient problem itself occurs when the gradient value continues to decrease over time, which makes the gradient value too small so that the contribution to the machine learning process becomes minimal. By using two gates, namely the Update Gate and the Reset Gate, GRU designs a mechanism that allows the determination of information that can be forwarded to the output.

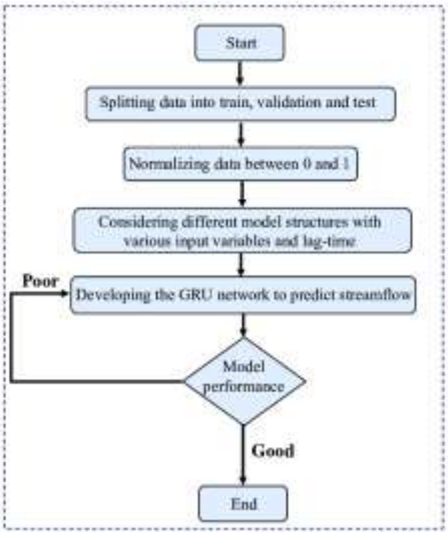

Figure 5. Flowchart of GRU Method

The Update Gate and Reset Gate in GRU play an important role in deciding how information is stored and passed on. During training, both gates can be set to keep relevant information from the past without deleting or ignoring unrelated information.

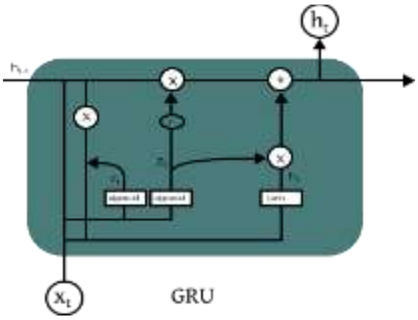

Figure 6. GRU Diagram

The update gate processes the input before it is processed by the GRU model. The update gate uses a sigmoid activation function σ(-) to calculate the amount of historical data that should be deflected into the future. The equation of the update gate (U_t) is [15]:

$$u_t = \sigma(W_{hu}.h_{t-1} + W_{xu}.x_t + b_u) \quad (2.11)$$
$$r_t = \sigma(W_{hr}.h_{t-1} + W_{xr}.x_t + b_r) \quad (2.12)$$

Next, the candidate state ($\widehat{h_t}$) is calculated using the reset gate calculation result. The hidden state at time t (h_t) is calculated using the candidate state. The hyperbolic tangent activation function is used to calculate the number of candidate state parts *(tanh(·))*. The equation of the candidate state is as follows [15].

$$\widehat{h_t} = tanh(W_{hh}.(r_t \odot h_{t-1}) + W_{xh}.x_t + b_h) \quad (2.13)$$

The symbol ⊙ In equation 2.13 is known as a hadamard product. The Hadamard product multiplication function multiplies the elements of two equal-sized matrices that are in the same row and column. GRU performs the calculation of the hidden state (h_t) following the calculation of the update gate, reset gate, and candidate state. The output time t, which is sent to the next GRU unit, is stored by the hidden state. [15] The hidden state equation is as follows:

$$h_t = (1-u) \odot h_{t-1} + u_t \odot \widehat{h_t} \quad (2.14)$$

The next step involves comparing the error values of each model after modeling using the training dataset and making predictions with the testing dataset. This process is done for each model and the dataset of stock prices listed on the ^JKSE index. In this comparison, the values of the testing dataset for Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) are tested. Furthermore, the smallest values of RMSE, MSE, and MAE are identified.

D. Data Prediction and Visualization

In this step, the LSTM and GRU models are applied to forecast stock price movements. After going through the preprocessing stage, the data is prepared to be processed using the LSTM and GRU models. This process will provide an up-to-date prediction of stock price changes using both models. In addition, the prediction results will also be visualized so that they can be more easily understood and analyzed.

E. Evaluation

Evaluation is essential for assessing performance and identifying the strengths and weaknesses of a method, as well as allowing for improvement through adjustments and modifications. Evaluation of a method involves using statistics with available data to analyze the degree of variation and determine performance. Among the statistical methods often used to conduct evaluations are:

1) Root Mean Square Error (RMSE)

RMSE is used to calculate the amount of error in predicting a method and calculates the difference between the actual value and the expected value. The total result obtained is divided by the predicted value, then the square root is taken.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (yi-yi)^2}{n}} \quad (2.15)$$

2) Mean Square Error (MSE)

MSE calculates the average square error between the actual and predicted values. This method is commonly used to assess the approximate error value in a prediction. A small MSE value or an MSE value close to zero indicates that the prediction results are in accordance with the actual data so that it is suitable for calculating predictions for future periods.

$$MSE = \sum \frac{|\hat{y}i - yi|}{n} \quad (2.16)$$

3) Mean Absolute Error (MAE)

MAE represents the average absolute difference between the actual value and the predicted value in the data set. It measures the average residual in the data set.

III. Result and Discussion

This research comprehensively involves several essential stages that detail every aspect of the process. First, data collection is conducted, where the dataset includes daily stock price data from the Composite Stock Price Index (CSPI) with the code ^JKSE.

A. Preprocessing

The dataset undergoes a preprocessing stage that includes Min-Max normalization to maintain the consistency of the data scale before being applied in the machine learning model.

1) Stock Price Visualization

Detailing stock price dynamics and movements through visual representations is an inevitable first step in the process of in-depth analysis of financial market data. Through informative graphical depictions, analysts can visually understand stock price fluctuations over time, open a window of insight into market trends, and identify significant events that affect stock values.

In Figure 6, the Share Price graph provides an informative visual representation of the evolution of the JCI closing price over time. With the X-axis showing specific dates and the Y-axis reflecting the closing price value, the graph provides a clear and structured picture of stock price fluctuations, allowing analysts to easily identify trends or significant events within the observed period.

2) Dataset Generation

In the dataset building process, the focus was on exploring the 'Close' column of the JCI stock price dataset. This choice was made deliberately to simplify the dataset, bringing a centralized focus on the key variable, the closing price of the stock. This decision was based on the understanding that closing price plays a central role in financial analysis, and by filtering out this variable, we can optimize the dataset to illustrate the dynamics of stock price movements more clearly and understandably. As a visual representation, the dataset has been illustrated in Figure 4.

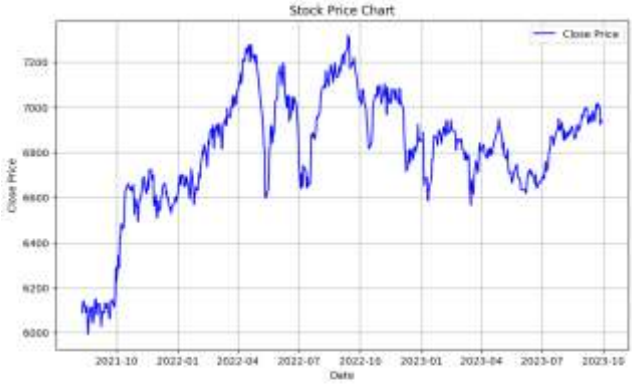

Figure 6. Share Price Chart

3) Dataset Separation

The dataset that has been created is divided into two, namely training data (train) and test data (test). This division is important to train the model using part of the data and test its performance on data that has never been seen before. A commonly used ratio is 80% for training data and 20% for test data. After this division, the training dataset consists of 417 data points, while the testing dataset consists of 105 data.

Figure 7. Distribution of Training and Testing Datasets

In the visualization in Figure 7., the dashed red line shows the division between the training dataset and the testing dataset. You can see how the data is organized over time, and the split reflects the predefined proportion (80% training, 20% testing).

4) Data Scaling

The next process is data scaling using the Min-Max scaling method. The purpose of this process is to normalize the values in the dataset into the range of 0 to 1. This is done so that the model can cope with scale differences and improve stability and convergence during the training process

Figure 8. Scaled Cover Price

The visualization in Figure 8., shows the comparison between the original closing price and the scaled closing price. In this graph, the blue color represents the original closing price, while the red dotted line represents the scaled closing price.

B. Modelling

a) LSTM

The Long Short-Term Memory (LSTM) model is designed using an architecture that involves one LSTM layer consisting of 50 units, followed by a Dense layer with one unit for output. The model training process was conducted for 100 epochs with a batch size of 32 to ensure that the model learns complex patterns in the data and can provide generalized results over a wide range of inputs [17]. After the training phase, the results were analyzed through visualization of Root Mean Squared Error (RMSE) graphs on the training and testing sets. The RMSE graph gives an idea of the extent of the model's prediction error on both data sets [18]. The best result, measured by the lowest RMSE value on the test set, is recorded and used as a reference for prediction on the test set. This approach provides confidence that the model has learned well from the training data and can make accurate stock price predictions on new, never-before-seen data [19].

From the graph in Figure 9, the LSTM model achieves the minimum RMSE value on the test dataset at a certain epoch, which is the benchmark for optimal model selection. The relationship between epochs and RMSE is that during the early stages of training, the RMSE increases, indicating the early stages of the model and its struggle to produce correct predictions. Subsequently, the RMSE decreases over time, signaling the model's acquisition of knowledge from the training data, resulting in more accurate predictions. Eventually, the RMSE reaches a saturation point, indicating that the model is no longer acquiring further insights from the training data, and consequently, the RMSE stabilizes without further reduction.

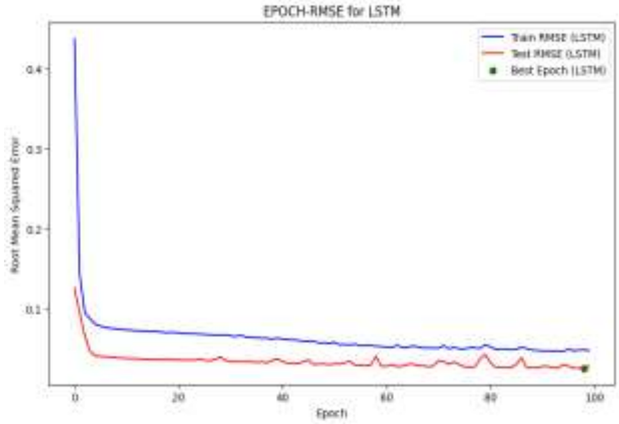

Figure 9. Grafik EPOCH-RMSE untuk LSTM

After training the LSTM model, stock price predictions are made on the test set. The graph of the prediction results is compared with the actual data on the test set and the prediction on the training set. This graph provides a visual representation of how well the LSTM model can predict stock prices.

Figure 10. LSTM Prediction Results

By detailing Figure 10, it can be concluded that the stock price prediction model using LSTM shows satisfactory performance. This can be seen from the Predictions graph (model prediction results), which closely follows and is in the same direction as the stock price visualization graph in Figure 6. Through Figure 10, the graph of LSTM prediction results provides a deeper understanding of the extent to which the model can adjust to the behavior of stock prices during the test period.

b) GRU

At the model development stage, the Gated Recurrent Unit (GRU) Model architecture was built with a similar approach to the LSTM model [20]. The configuration involves one GRU layer consisting of 50 units, followed by a Dense layer with one unit to generate the output. Subsequently, the model goes through a training process that is performed for 100 epochs, with a batch size setting of 32. Once the training process is complete, the best results, measured by the best performance on the test set, are recorded and used as the basis for making predictions on the test set. This approach ensures that the GRU model can provide reliable and accurate predictions of stock prices on new data that has never been seen before, after a sufficient training phase.

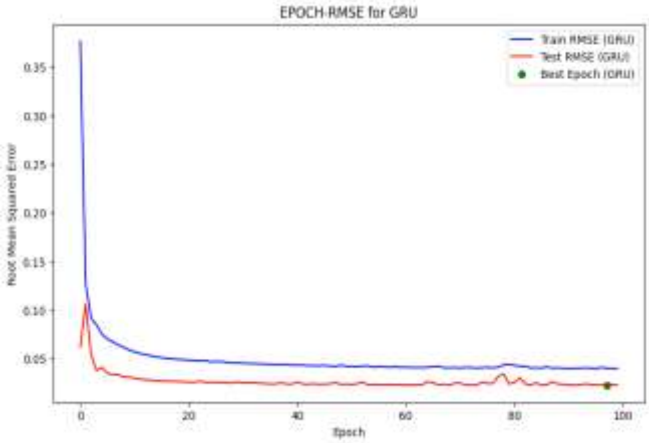

Figure 11. EPOCH-RMSE for GRU

Through the graph analysis in Figure 11, it is illustrated that the Gated Recurrent Unit (GRU) model managed to achieve the lowest RMSE value at a certain epoch in the test set. This minimum RMSE value serves as a crucial reference in determining the optimal model for stock price prediction. In the initial training phase, RMSE tends to be high as the model is still in the early learning stage, grappling with the challenge of making accurate predictions. However, as the epochs progress, the RMSE consistently decreases, signifying the model's continuous extraction of information from the training data and the refinement of its skills in generating more accurate predictions. At a certain point, the RMSE reaches a saturation point, indicating that the GRU model has attained maximum understanding of the training data, and further enhancements to the quality of predictions become less significant.

After training the GRU model, stock price predictions are made on the test set. The graph of the prediction results is compared with the actual data on the test set and the prediction on the training set. This graph provides a visual representation of how well the GRU model can predict stock prices. By detailing Figure 12, it can be concluded that the stock price prediction model using GRU shows satisfactory performance. This can be seen from the Predictions graph (model prediction results), which closely follows and is in the same direction as the stock price visualization graph in Figure 4. The comparison between model predictions and actual data not only provides a strong visual representation but also provides valuable insights into the performance of the GRU model in anticipating stock price changes.

Figure 12. GRU Prediction Results

C. Evaluation

After looking at the results of the two models, a comparison was made between LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit) in terms of stock price prediction performance. The evaluation is done by considering three main metrics: Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE) for each model in the test set.

Table 2. Evaluation of the model

Model	RMSE	MSE	MAE
LSTM	37.2286	1385.968	30.087
GRU	30.1377	908.286	25.418

From Table 2 above, we can directly compare the performance of the two models for each metric. Some analysis and conclusions can be drawn from this comparison:

1) RMSE (Root Mean Squared Error)

GRU has a lower RMSE value (30.1377) compared to LSTM (37.2286), indicating that GRU tends to provide more accurate predictions in measuring the absolute change in stock prices.

2) MSE (Mean Squared Error)
GRU shows a lower MSE value (908.286) compared to LSTM (1385.968), indicating that GRU is more successful in overcoming large errors in stock price prediction.

3) MAE (Mean Absolute Error)
The GRU model has a lower MAE value (25,418) compared to LSTM (30,087), indicating that GRU provides more accurate predictions with smaller errors on average.

D. Conclusion

Based on the results and discussion, this study concludes that the GRU method consistently provides more accurate JCI stock price predictions compared to the LSTM method. The evaluation shows a lower error rate in GRU, especially in the testing period, with RMSE reaching 30.14. In contrast, LSTM has an RMSE of 37.23 at Epoch 98.

In addition to RMSE, evaluations were conducted using MSE and MAE. The results show that GRU has an MSE of 908.29 and an MAE of 25.42, while LSTM has an MSE of 1385.97 and an MAE of 30.09. Both methods can provide predictions that are in accordance with the actual behavior of stock prices within the specified time range.

References

[1] S. Gaffar and A. T. Uleng Akal, "Combination of Fundamental and Technical Analysis of Shares: Revealing the Profit Potential in the Stock Market of PT. United Tractors, Tbk," *Golden Ratio Financ. Manag.*, vol. 2, no. 2, pp. 108–119, 2022.

[2] S. Abolmakarem, F. Abdi, K. Khalili-Damghani, and H. Didehkhani, "A multi-stage machine learning approach for stock price prediction: Engineered and derivative indices," *Intell. Syst. with Appl.*, vol. 24, no. September, p. 200449, 2024.

[3] D. R. Chandranegara, R. A. Afif, C. S. K. Aditya, W. Suharso, and H. Wibowo, "Prediksi Harga Saham Jakarta Islamic Index Menggunakan Metode Long Short-Term Memory," *J. Edukasi dan Penelit. Inform.*, vol. 9, no. 1, p. 129, 2023.

[4] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment Analysis Using Word2vec and Long Short-Term Memory (LSTM) for Indonesian Hotel Reviews," *Procedia Comput. Sci.*, vol. 179, no. 2020, pp. 728–735, 2021.

[5] Z. Chen, C. Li, and W. Sun, "Bitcoin price prediction using machine learning: An approach to sample dimension engineering," *J. Comput. Appl. Math.*, vol. 365, p. 112395, 2020.

[6] X. Sha, "Time Series Stock Price Forecasting Based on Genetic Algorithm (GA)-Long Short-Term Memory Network (LSTM) Optimization," *Adv. Econ. Manag. Polit. Sci.*, vol. 91, no. 1, pp. 142–149, 2024.

[7] N. W. Meri Aryati, I. K. A. G. Wiguna, N. W. S. Putri, I. K. K. Widiartha, and N. L. W. S. R. Ginantra, "Komparasi Metode LSTM dan GRU dalam Memprediksi Harga Saham," *J. Media Inform. Budidarma*, vol. 8, no. 2, p. 1131, 2024.

[8] R. Herrera Casanova and A. Conde, "Enhancement of LSTM models based on data pre-processing and optimization of Bayesian hyperparameters for day-ahead photovoltaic generation prediction," *Comput. Electr. Eng.*, vol. 116, no. January 2023, p. 109162, 2024.

[9] M. F. Rizkilloh and S. Widyanesti, "Cryptocurrency Price Prediction Using Long Short Term Memory (LSTM) Algorith," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 25–31, 2022.

[10] Y. D. Rosita and L. S. Moonlight, "Comparative Analysis of LSTM Neural Network and SVM for USD Exchange Rate Prediction: A Study on Different Training Data Scenarios," *Sci. J. Informatics*, vol. 11, no. 1, pp. 195–206, 2024.

[11] N. M. H. Alamri, M. Packianather, and S. Bigot, "Optimizing the Parameters of Long Short-Term Memory Networks Using the Bees Algorithm," *Appl. Sci.*, vol. 13, no. 4, 2023.

[12] R. Hidayat and R. D. Irawan, "Prediksi Harga Saham Syariah Menggunakan Metode Deep Learning GRU dan LSTM," vol. 20, no. 1, pp. 37–50.

[13] W. Hastomo, A. S. B. Karno, N. Kalbuana, E. Nisfiani, and L. ETP, "Optimasi Deep Learning untuk Prediksi Saham di," ... *(Jurnal Edukasi dan ...*, vol. 7, no. 2, pp. 133–140, 2021.

[14] A. Pranolo *et al.*, "Enhanced Multivariate Time Series Analysis Using LSTM: A Comparative Study of Min-Max and Z-Score Normalization Techniques," *Ilk. J. Ilm.*, vol. 16, no. 2, pp. 210–220,

2024.

[15] A. Nilsen, “Perbandingan Model RNN, Model LSTM, dan Model GRU dalam Memprediksi Harga Saham-Saham LQ45,” *J. Stat. dan Apl.*, vol. 6, no. 1, pp. 137–147, 2022.

[16] M. D. A. Carnegie and C. Chairani, “Perbandingan Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU) Untuk Memprediksi Curah Hujan,” *J. Media Inform. Budidarma*, vol. 7, no. 3, p. 1022, 2023.

[17] Y. Ashari and A. Suhendar, “Implementasi Algoritma Long Short-Term Memory (Lstm) Untuk Memprediksi Harga Beras Di Jawa Tengah Berdasarkan Cuaca,” *Djtechno J. Teknol. Inf.*, vol. 5, no. 3, pp. 624–636, 2024.

[18] V. Plevris, G. Solorzano, N. P. Bakas, and M. E. A. Ben Seghier, “Investigation of Performance Metrics in Regression Analysis and Machine Learning-Based Prediction Models,” *World Congr. Comput. Mech. ECCOMAS Congr.*, no. June, 2022.

[19] Furizal, A. Ritonga, A. Ma’arif, and I. Suwarno, “Stock Price Forecasting with Multivariate Time Series Long Short-Term Memory: A Deep Learning Approach,” *J. Robot. Control*, vol. 5, no. 5, pp. 1322–1335, 2024.

[20] E. J. Prasetyo and K. D. Hartomo, “Multi-industry stock forecasting using GRU-LSTM deep transfer learning method,” *J. Infotel*, vol. 15, no. 2, pp. 30–43, 2023.