JURNAL INFORMATIKA, Vol.11 No.2 October 2024, Page 80-89
ISSN (Print) 2355-6579 | ISSN (Online) 2528-2247
DOI: https://doi.org/10.31294/inf.v11i2.22156

Stock Price Prediction On IDX30 Index Using Long Short-Term Memory Algorithm

Ken William[1]*, Dionisia Bhisetya Rarasati[2]

[1,2] Faculty of Technology and Design, Bunda Mulia University
Alam Sutera, Jl. Jalur Sutera Bar. No.Kav.7-9, Tangerang City, Banten 15143, Indonesia

Correspondence e-mail: kenwilliam1404@gmail.com

Submission:	Revision:	Acceptance:	Available Online:
23-05-2024	15-07-2024	25-07-2024	06-09-2024

Abstract

The capital market plays a significant role in a country's economy, facilitating corporate financing and providing investment opportunities for the public. One popular investment instrument is stocks, yet many investors struggle to make profitable investment decisions due to a lack of understanding of stock investments. Therefore, predicting stock prices can be a way to determine the future value of a stock. This research aims to address this issue by applying the Long Short-Term Memory (LSTM) algorithm to predict stock prices on the IDX30 index. LSTM is capable of processing sequential data, such as stock price data, complexly because it can store information over long periods. The testing is conducted using various parameters in layers, epochs, and time steps to obtain the best prediction model. The LSTM architecture used consists of four layers: the LSTM layer with 128 neurons, dropout and dense layers with 64 neurons, and an additional dense layer that converts the output from the previous layer into prediction results. This study demonstrates that the LSTM algorithm can accurately predict stock prices based on evaluation metrics such as Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE). The best results for PT Bank Central Asia Tbk show a MAPE of 1.14% and RMSE of 137.71, PT Bank Rakyat Indonesia Tbk shows a MAPE of 1.58% and RMSE of 87.4, and PT Bank Mandiri Tbk shows a MAPE of 1.64% and RMSE of 88.26.

Keywords: Stocks, Prediction, Stock prices, Long Short-Term Memory (LSTM), MAPE, RMSE.

1. Introduction

The capital market plays an important role in the economy by serving as a means for companies to obtain alternative funding sources beyond banks or financial institutions, particularly through Initial Public Offerings (IPOs). Companies often choose this method because it is considered more cost-effective compared to obtaining credit. Additionally, the capital market provides opportunities for the general public to invest their funds by purchasing shares of listed companies, with the aim of achieving capital gains. The capital market serves as an alternative investment option for the public, who can also choose other instruments such as insurance policies, gold bullion, or foreign currency savings (Dwi Septiani et al., 2020).

Stocks are one of the investment instruments highly favored by the public because they offer various investment goals and provide numerous benefits (Rustiana et al., 2022). According to SIKAPI Otoritas Jasa Keuangan, stocks represent ownership rights in a company, and stock indices are created to monitor the overall market performance.

The stock market is a place where various types of stocks can be traded, and stock prices can change daily (Toranggi Berton et al., 2022). One of the main indicators of capital market fluctuations is the Composite Stock Price Index (IHSG). The IHSG provides a general picture of the condition of the capital market in Indonesia because it reflects the average stock prices of all companies listed on the Indonesia Stock Exchange (Orca & Setiawan, 2022). Based on the capital market statistics in Indonesia in November 2023, the stock market showed significant growth in the number of single investor identifications (SID). PT Bursa Efek Indonesia on December 29, 2023, also reported a remarkable increase in the Composite Stock Price Index (IHSG), indicating growing interest in stock investment.

Stock return is the anticipated rate of return from investing in a particular stock or stock portfolio. This return rate heavily depends on the company's financial performance and the overall market conditions. The better the company's financial performance, the more likely its stock price will rise, providing investors with profit through changes in the stock price (Riza & Sudibyo, 2020).

According to PT Bursa Efek Indonesia, the IDX30 Index is one of the important stock indices on the Indonesia Stock Exchange (IDX), reflecting the overall performance of the stock market and providing a benchmark for stock investors (Verkino et al., 2020). However, difficulties in selecting stocks are a common problem faced by investors (Tanto & Kurniawan, 2022).

Fundamental analysis is a technique used by investors to assess how well a company is performing so they can make more confident investment decisions. One approach in this analysis is using financial ratios as indicators, such as solvency and profitability ratios. The solvency ratio indicates a company's ability to pay its debts, while the profitability ratio shows how effectively a company generates profit from its operations. These two ratios are important as measures of the risks faced by the company concerning its obligations and the potential profits from its operational activities (Veny & Gunawan, 2022).

With technological advancements, many investors utilize machine learning techniques to analyze and predict future stock price movements (Belyadi & Haghighat, 2021). One algorithm that can be used for this purpose is Long Short-Term Memory (LSTM), a derivative of the Recurrent Neural Network (RNN) algorithm (Khumaidi & Ayu Nirmala, 2022). To address this challenge, this study aims to apply the Long Short-Term Memory (LSTM) algorithm to predict stock prices on the IDX30 index.

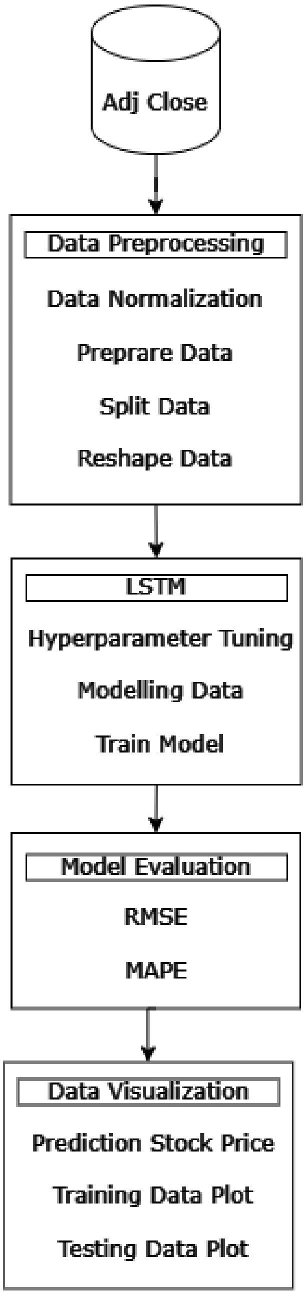
Research related to stock price prediction using the Long Short-Term Memory algorithm was conducted by Freddy Toranggi Berton et al. in 2022 by predicting Indosat's stock. The results of the last 30 days of testing yielded a MAPE of 2.02%. From the hyperparameter tuning process, it was found that the optimal parameters for the LSTM model are 16 hidden layers, 2 LSTM layers, 200 epochs, and a learning rate of 0.05, with the lowest MSE Loss value of 0.000885280198417604 (Toranggi Berton et al., n.d.). Another study on price prediction using Long Short-Term Memory was also conducted by Laras Wiranda and Mujiono Sadikin in 2019. This study applied Long Short-Term Memory (LSTM) to sales data of drug "X" at PT. Metiska Farma to forecast future sales. The experiment yielded the best performance with an RMSE of 13,762,154.00 and a MAPE of 12%. The trial used a data composition of 90% training data and 10% test data, an interval of [-1,1], and 1500 epochs (Wiranda & Sadikin, n.d.). Thus, it has been proven that Long Short-Term Memory has a high level of accuracy in predicting future prices.

This research is expected to contribute to the development of knowledge, especially in the fields of artificial intelligence and the stock market, as well as provide practical benefits for stock investors in making more accurate investment decisions.

2. Research Methodology

The methodology in this study consists of several stages, namely data collection, data processing, model creation, model evaluation, and data visualization.

The steps taken in this study to analyze stock price prediction on the IDX30 index can be seen in Figure 1.



Source: Research Process

Figure 1. Research Stages

The methodology employed in this study involves several stages, illustrated in Figure 1, for predicting stock prices on the IDX30 index:

1. Data Collection
   The data collection phase involves gathering stock data from various sources, which will subsequently be processed for testing purposes. This dataset includes stock information from PT Bank Central Asia Tbk (BBCA), PT Bank Rakyat Indonesia Tbk (BBRI), and PT Bank Mandiri Tbk (BMRI), with data sourced from Yahoo Finance. The period covered is from April 27, 2019, to April 27, 2024. The collected dataset comprises the dates and adjusted closing prices of these stocks, providing a comprehensive historical context for analysis.
2. Data Preprocessing
   The data phase entails preparing the data to be suitable for training and testing the stock price prediction model. This involves several critical steps:
   - Data Normalization
     The data is normalized to a range of 0-1 using the MinMax Scaler to ensure consistency in scale and to prevent larger features from dominating the model's learning process.
   - Prepare Data
     Time sequences are created as features and targets to train the model, allowing the model to capture temporal dependencies in the data.
   - Split Data
     The dataset is divided into two parts, with 80% allocated for training and 20% for testing. This split helps in evaluating the model's performance on unseen data.
   - Reshape Data
     The data is transformed into a three-dimensional format suitable for LSTM models, enabling the model to understand the temporal relationships between input sequences.
3. LSTM
   The third stage, the Long Short-Term Memory (LSTM) model implementation, is pivotal for stock price prediction. LSTM, an advanced form of the Recurrent Neural Network (RNN), addresses issues such as vanishing gradients and is adept at capturing long-term dependencies through its memory cells and gates (Okut, 2021). The process includes:
   - Hyperparameter Tuning
     Adjusting model parameters during training, such as learning rate, number of epochs, and model architecture, to optimize performance.
   - Modelling Data
     Constructing an appropriate model architecture, which may include LSTM layers for capturing sequential dependencies, dense layers for mapping to the output space, and dropout layers to mitigate overfitting.
   - Train Model
     The model is trained using the training dataset, optimizing parameters through an optimizer and minimizing prediction errors using a loss function.

4. Model Evaluation
   Model evaluation is conducted using metrics such as Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) to assess the model's accuracy and error rates.
   - MAPE
     This metric evaluates the percentage difference between predicted and actual values, providing a measure of prediction accuracy relative to actual values (Nabillah & Ranggadara, 2020).

     The formulas to calculate MAPE (Putro et al., 2021) are as follows:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{actual(t) - predicted(t)}{actual(t)}\right| \times 100\%$$

     This metric allows for the assessment of forecast accuracy in percentage terms, facilitating comparison across different datasets or models. Table 1 summarizes the range of MAPE values, offering a framework for evaluating the accuracy of forecasting models based on their performance percentages.

     Table 1 summarizes the range of Mean Absolute Percentage Error (MAPE) values, providing a framework for evaluating the accuracy of forecasting models based on their performance percentages.

Table 1. Range of MAPE Values

Value (%)	Description
<10	Very Accurate Forecasting Model
10-20	Accurate Forecasting Model
20-50	Acceptable Forecasting Model
>50	Not Acceptable Forecasting Model

Source: (Budiprasetyo et al., 2023)

   - RMSE
     This metric measures the square root of the average squared errors produced by MAPE, evaluating the model's ability to

predict values that correspond closely with observed data(Prasetyo et al., 2021).

The formulas to calculate RMSE (Prasetyo et al., 2021) are as follows:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(actual(t) - predicted(t))^2}{n}}$$

This metric provides a clear indication of prediction errors, with larger deviations having a greater impact due to the squaring of errors. A lower RMSE value signifies better predictive performance, making it a preferred choice when assessing the reliability of forecasting models.

5. Data visualization
The final stage, data visualization, involves displaying the predicted stock price results. This includes plotting historical actual and predicted price data and evaluating the model's performance. Visualization aids in understanding the model's effectiveness and the stock price trends, providing insights into the accuracy and reliability of the predictions.

3. Results and Discussion

This section discusses the research findings based on each stage conducted.

3.1 Data Collection

Stock data collection was conducted by retrieving data from Yahoo Finance. Users were prompted to input the stock symbols they wished to retrieve data for, such as BBCA, BBRI, and BMRI. Users were also asked to specify the start and end dates for the data retrieval. In this study, data was collected from April 27, 2019, to April 27, 2024. Upon receiving user input, the program downloaded stock data from Yahoo Finance based on the specified stock symbols and date range. The collected data included adjusted closing prices for the requested period.

Figure 2(a) shows the stock data retrieval results for BBCA for the period from April 27, 2019, to April 27, 2024.

```
Masukkan Simbol Saham (BBCA.JK): BBCA.JK
Masukkan tanggal mulai (format: yyyy-mm-dd): 2019-04-27
Masukkan tanggal akhir (format: yyyy-mm-dd): 2024-04-27
[*********************100%%**********************]  1 of 1 completedDate
2019-04-29    5107.948242
2019-04-30    5166.349609
2019-05-01    5166.349609
2019-05-02    5107.948242
2019-05-03    5098.962891
                 ...
2024-04-22    9350.000000
2024-04-23    9725.000000
2024-04-24    9950.000000
2024-04-25    9775.000000
2024-04-26    9625.000000
Name: Adj Close, Length: 1220, dtype: float64
```

(a)

Figure 2(b) shows the stock data retrieval results for BBRI for the period from April 27, 2019, to April 27, 2024.

```
Masukkan Simbol Saham (BBCA.JK): BBRI.JK
Masukkan tanggal mulai (format: yyyy-mm-dd): 2019-04-27
Masukkan tanggal akhir (format: yyyy-mm-dd): 2024-04-27
[*********************100%%**********************]  1 of 1 completedDate
2019-04-29    3075.949951
2019-04-30    3097.212402
2019-05-01    3097.212402
2019-05-02    3097.212402
2019-05-03    3104.299561
                 ...
2024-04-22    5300.000000
2024-04-23    5300.000000
2024-04-24    5225.000000
2024-04-25    5150.000000
2024-04-26    4830.000000
Name: Adj Close, Length: 1220, dtype: float64
```

(b)

Figure 2(c) shows the stock data retrieval results for BMRI for the period from April 27, 2019, to April 27, 2024.

```
Masukkan Simbol Saham (BBCA.JK): BMRI.JK
Masukkan tanggal mulai (format: yyyy-mm-dd): 2019-04-27
Masukkan tanggal akhir (format: yyyy-mm-dd): 2024-04-27
[*********************100%%**********************]  1 of 1 completedDate
2019-04-29    2994.811279
2019-04-30    2975.551758
2019-05-01    2975.551758
2019-05-02    2965.922363
2019-05-03    2946.663086
                 ...
2024-04-22    6725.000000
2024-04-23    6825.000000
2024-04-24    7050.000000
2024-04-25    6950.000000
2024-04-26    6750.000000
Name: Adj Close, Length: 1220, dtype: float64
```

(c)

Source: Research Process

Figure 2. Stock Data: (a) BBCA Stock (b) BBRI Stock (c) BMRI Stock

3.2 Data Preprocessing

At this stage, the first step undertaken is data normalization. This is done to transform the range of data values into the range of 0-1. Normalization is crucial for maintaining scale consistency and preventing features with large values from dominating the model training process. Data normalization can be performed using the MinMaxScaler technique as described by the following equation (Rosyd et al., 2024):

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Where x' is the scaled data in the new range, x is the data value to be normalized, x_{max} is the maximum value of the variable, and x_{min} is the minimum value of the variable.

Here is an example calculation of normalization for the adjusted closing price data of BBCA stock from April 27, 2019, to April 27, 2024, with a maximum value of 10,092 and a minimum value of 3,992.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{5107 - 3992}{10092 - 3992} = \frac{1115}{6100} \approx 0.1828$$

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{5166-3992}{10092-3992} = \frac{1174}{6100} \approx 0.1923$$

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{5166-3992}{10092-3992} = \frac{1115}{6100} \approx 0.1923$$

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{9950-3992}{10092-3992} = \frac{5958}{6100} \approx 0.9766$$

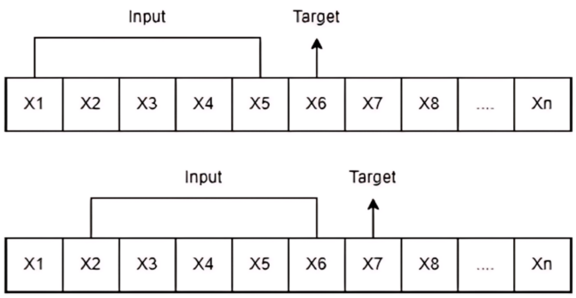
$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{9975-3992}{10092-3992} = \frac{5958}{6100} \approx 0.9479$$

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{9625-3992}{10092-3992} = \frac{5958}{6100} \approx 0.9233$$

Next, the data is passed into the prepare_data function to iterate through the normalized data, forming sequences of X initialized with 30 previous time steps and y as the next time step. Each iteration allows the formation of corresponding sequences of X and y, which are then appended to lists X and y.

These steps ensure that the data is prepared in a suitable format for training the LSTM model, where previous time sequences (X) are used as features and the next value (y) is used as the target. All sequences of X and values of y are then converted into numpy arrays for use in the model training and testing processes.

Figure 3 illustrates the time steps involved in data analysis.



Source: (Budiprasetyo et al., 2023)

Figure 3. Illustration of Time Steps

The next process, data splitting, divides the prepared data, formatted as time sequences (X) and target values (y), into two subsets: training data (train) and testing data (test). The splitting process uses an 80:20 ratio, where 80% of the data is used for training the model (X_train, y_train), and 20% is used to evaluate the model's performance (X_test, y_test). This division is crucial for assessing the model's performance: the training data is used to train the model, while the testing data evaluates how well the model can predict target values that it has not seen before.

Here are the calculations for dividing the data into training and testing sets with a time step of 30 and a total of 1220 data points:

total_data = number_of_data_points - time_steps
train_size = 1220 - 30 = 1190
*train_size = total_data * 0.8*
*train_size = 1190 * 0.8 = 952*

test_size = total_data - train_size
test_size = 1190 - 952 = 238

In this study, 30 time steps are used to form one sample, so only data from 1 to 1190 can be used to form samples, while the last 30 data points (1191 to 1220) are used as targets for each sample. Therefore, the total number of samples that can be created is 1190. Based on this calculation, the number of data points for training (X_train, y_train) consists of 952 samples, while the number of data points for testing (X_test, y_test) consists of 238 samples.

After splitting the data, the next step is to reshape the input data to convert it into the three-dimensional format required by the LSTM model. Reshaping is done to ensure that the model understands the time relationship between input data. Thus, the input data is transformed into a three-dimensional array with the format [number of samples, time_steps, features]. Consequently, each sample will have time_steps and one feature representing the normalized stock values.

3.3 Long Short-Term Memory

Implementation of the Long Short-Term Memory (LSTM) model is performed after the data has completed all preprocessing stages. The LSTM model is built using the Sequential object from Keras. This model consists of several layers, starting with an LSTM layer with 128 neurons and ReLU activation function, which takes input of shape (n_steps, 1). Next, a Dense layer with 64 neurons and ReLU activation function is added, followed by a dropout layer with a dropout rate of 0.2 to prevent overfitting. Finally, a Dense layer with one unit is added, which serves as the output layer of the model.

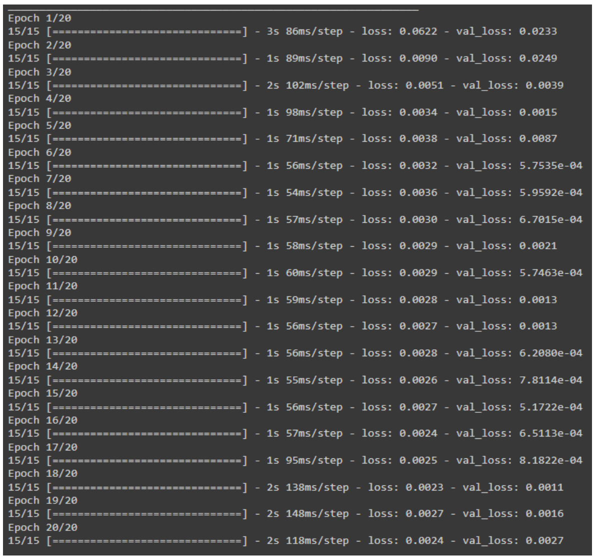
Figure 4 shows the model summary results for BBCA stock.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_8 (LSTM)               (None, 128)               66560

 dense_6 (Dense)             (None, 64)                8256

 dropout_4 (Dropout)         (None, 64)                0

 dense_7 (Dense)             (None, 1)                 65

=================================================================
Total params: 74881 (292.50 KB)
Trainable params: 74881 (292.50 KB)
Non-trainable params: 0 (0.00 Byte)
```

Source: Research Process

Figure 4. Model Summary

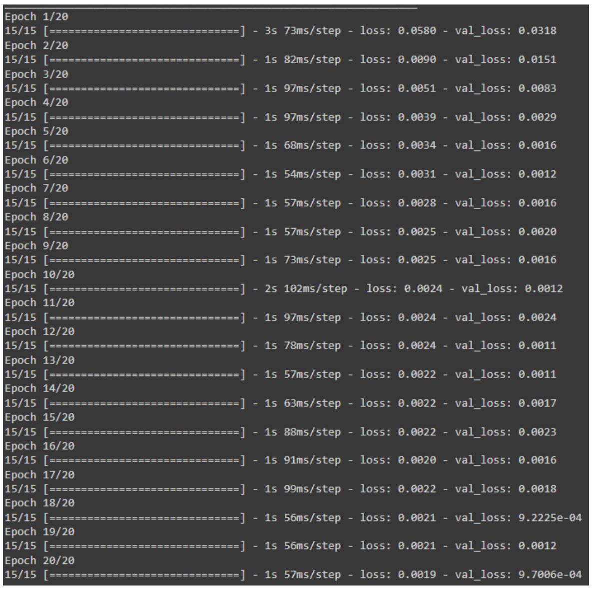
Figure 5(a) presents the results of the training process, illustrating the relationship between epochs, loss, and validation loss for BBCA stock.



```
Epoch 1/20
15/15 [==============================] - 3s 86ms/step - loss: 0.0622 - val_loss: 0.0233
Epoch 2/20
15/15 [==============================] - 1s 89ms/step - loss: 0.0090 - val_loss: 0.0249
Epoch 3/20
15/15 [==============================] - 2s 102ms/step - loss: 0.0051 - val_loss: 0.0039
Epoch 4/20
15/15 [==============================] - 1s 98ms/step - loss: 0.0034 - val_loss: 0.0015
Epoch 5/20
15/15 [==============================] - 1s 71ms/step - loss: 0.0038 - val_loss: 0.0087
Epoch 6/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0032 - val_loss: 5.7535e-04
Epoch 7/20
15/15 [==============================] - 1s 54ms/step - loss: 0.0036 - val_loss: 5.9592e-04
Epoch 8/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0030 - val_loss: 6.7015e-04
Epoch 9/20
15/15 [==============================] - 1s 58ms/step - loss: 0.0029 - val_loss: 0.0021
Epoch 10/20
15/15 [==============================] - 1s 60ms/step - loss: 0.0029 - val_loss: 5.7463e-04
Epoch 11/20
15/15 [==============================] - 1s 59ms/step - loss: 0.0028 - val_loss: 0.0013
Epoch 12/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0027 - val_loss: 0.0013
Epoch 13/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0028 - val_loss: 6.2080e-04
Epoch 14/20
15/15 [==============================] - 1s 55ms/step - loss: 0.0026 - val_loss: 7.8114e-04
Epoch 15/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0027 - val_loss: 5.1722e-04
Epoch 16/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0024 - val_loss: 6.5113e-04
Epoch 17/20
15/15 [==============================] - 1s 95ms/step - loss: 0.0025 - val_loss: 8.1822e-04
Epoch 18/20
15/15 [==============================] - 2s 138ms/step - loss: 0.0023 - val_loss: 0.0011
Epoch 19/20
15/15 [==============================] - 2s 148ms/step - loss: 0.0027 - val_loss: 0.0016
Epoch 20/20
15/15 [==============================] - 2s 118ms/step - loss: 0.0024 - val_loss: 0.0027
```

(a)

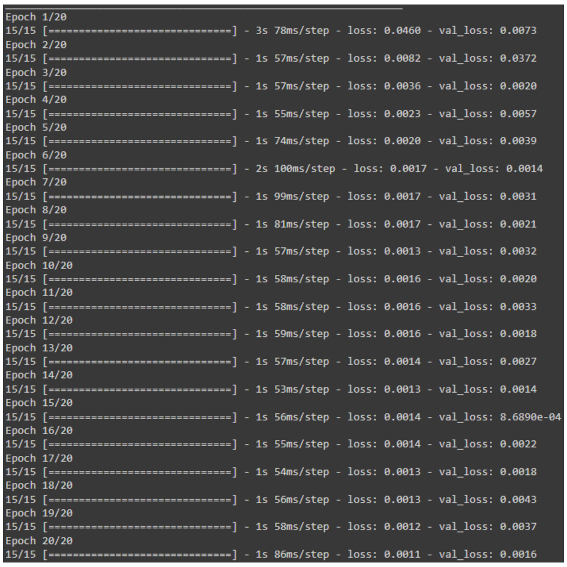
Figure 5(b) presents the results of the training process, illustrating the relationship between epochs, loss, and validation loss for BBRI stock.



```
Epoch 1/20
15/15 [==============================] - 3s 73ms/step - loss: 0.0580 - val_loss: 0.0318
Epoch 2/20
15/15 [==============================] - 1s 82ms/step - loss: 0.0090 - val_loss: 0.0151
Epoch 3/20
15/15 [==============================] - 1s 97ms/step - loss: 0.0051 - val_loss: 0.0083
Epoch 4/20
15/15 [==============================] - 1s 97ms/step - loss: 0.0039 - val_loss: 0.0029
Epoch 5/20
15/15 [==============================] - 1s 68ms/step - loss: 0.0034 - val_loss: 0.0016
Epoch 6/20
15/15 [==============================] - 1s 54ms/step - loss: 0.0031 - val_loss: 0.0012
Epoch 7/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0028 - val_loss: 0.0016
Epoch 8/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0025 - val_loss: 0.0020
Epoch 9/20
15/15 [==============================] - 1s 73ms/step - loss: 0.0025 - val_loss: 0.0016
Epoch 10/20
15/15 [==============================] - 2s 102ms/step - loss: 0.0024 - val_loss: 0.0012
Epoch 11/20
15/15 [==============================] - 1s 97ms/step - loss: 0.0024 - val_loss: 0.0024
Epoch 12/20
15/15 [==============================] - 1s 78ms/step - loss: 0.0024 - val_loss: 0.0011
Epoch 13/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0022 - val_loss: 0.0011
Epoch 14/20
15/15 [==============================] - 1s 63ms/step - loss: 0.0022 - val_loss: 0.0017
Epoch 15/20
15/15 [==============================] - 1s 88ms/step - loss: 0.0022 - val_loss: 0.0023
Epoch 16/20
15/15 [==============================] - 1s 91ms/step - loss: 0.0020 - val_loss: 0.0016
Epoch 17/20
15/15 [==============================] - 1s 99ms/step - loss: 0.0022 - val_loss: 0.0018
Epoch 18/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0021 - val_loss: 9.2225e-04
Epoch 19/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0021 - val_loss: 0.0012
Epoch 20/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0019 - val_loss: 9.7006e-04
```

(b)

Figure 5(c) presents the results of the training process, illustrating the relationship between epochs, loss, and validation loss for BMRI stock.



```
Epoch 1/20
15/15 [==============================] - 3s 78ms/step - loss: 0.0460 - val_loss: 0.0073
Epoch 2/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0082 - val_loss: 0.0372
Epoch 3/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0036 - val_loss: 0.0020
Epoch 4/20
15/15 [==============================] - 1s 55ms/step - loss: 0.0023 - val_loss: 0.0057
Epoch 5/20
15/15 [==============================] - 1s 74ms/step - loss: 0.0020 - val_loss: 0.0039
Epoch 6/20
15/15 [==============================] - 2s 100ms/step - loss: 0.0017 - val_loss: 0.0014
Epoch 7/20
15/15 [==============================] - 1s 99ms/step - loss: 0.0017 - val_loss: 0.0031
Epoch 8/20
15/15 [==============================] - 1s 81ms/step - loss: 0.0017 - val_loss: 0.0021
Epoch 9/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0013 - val_loss: 0.0032
Epoch 10/20
15/15 [==============================] - 1s 58ms/step - loss: 0.0016 - val_loss: 0.0020
Epoch 11/20
15/15 [==============================] - 1s 58ms/step - loss: 0.0016 - val_loss: 0.0033
Epoch 12/20
15/15 [==============================] - 1s 59ms/step - loss: 0.0016 - val_loss: 0.0018
Epoch 13/20
15/15 [==============================] - 1s 57ms/step - loss: 0.0014 - val_loss: 0.0027
Epoch 14/20
15/15 [==============================] - 1s 53ms/step - loss: 0.0013 - val_loss: 0.0014
Epoch 15/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0014 - val_loss: 8.6890e-04
Epoch 16/20
15/15 [==============================] - 1s 55ms/step - loss: 0.0014 - val_loss: 0.0022
Epoch 17/20
15/15 [==============================] - 1s 54ms/step - loss: 0.0013 - val_loss: 0.0018
Epoch 18/20
15/15 [==============================] - 1s 56ms/step - loss: 0.0013 - val_loss: 0.0043
Epoch 19/20
15/15 [==============================] - 1s 58ms/step - loss: 0.0012 - val_loss: 0.0037
Epoch 20/20
15/15 [==============================] - 1s 86ms/step - loss: 0.0011 - val_loss: 0.0016
```

(c)

Source: Research Process

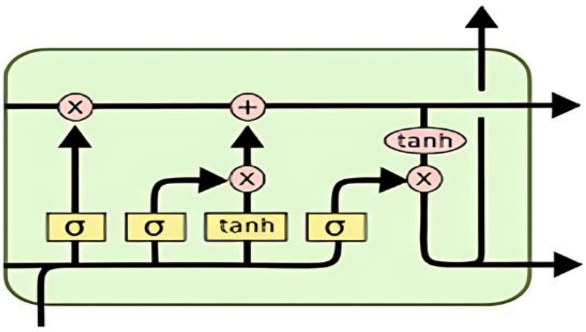
Figure 5. Epoch, loss, and val_loss results: (a) BBCA Stock (b) BBRI Stock (c) BMRI Stock

Next, the model is compiled using the Adam optimizer and Mean Squared Error (MSE) loss function to measure its performance. Once the model is built, it is trained using the fit method, providing training data (X_train and y_train) for 20 epochs (training iterations), with a batch_size of 64.

Several equations are used in Long Short-Term Memory (LSTM), including (Ali et al., 2022):

$$f_t = \sigma\,(W_f S_{t-1} + W_f X_t)$$
$$I_t = \sigma\,(W_i S_{t-1} + W_i X_t)$$
$$O_t = \sigma\,(W_t S_{t-1} + W_t X_t)$$
$$\hat{c}_t = tanh\,(W_c S_{t-1} + W_c X_t)$$
$$c_t = (i_t * \hat{c}_t + f_t * c_{t-1})$$
$$h_t = O_t * \tanh(c_t)$$

In the equations above, f_t represents the forget gate value that regulates the deletion of information from the previous cell state, I_t is the input gate that regulates the information to be added to the current cell state, O_t is the output gate that regulates the information to be conveyed from the current cell state, c_t is the current cell state value that holds long-term information, $\hat{c}_t$ is the candidate value which is potential information to be added to the current cell state, W represents the weight values that connect inputs and outputs within the network, and h_t is the hidden state value that stores short-term information.



Source: (Kurnia Sari et al., 2020)

Figure 6. LSTM Architecture

In this section, we detail the manual calculations performed for the Long Short-Term Memory (LSTM) model applied to PT Bank Central Asia Tbk. The calculations involve determining the values of the forget gate, input gate, candidate value, cell state, output gate, and hidden state. The following weight values are initialized for each gate: 0.8 for the forget gate, 0.6 for the input gate, 0.7 for the output gate, and 0.9 for the candidate value. The bias values are set as follows: 0.6 for the forget gate, 0.8 for the input gate, 0.5 for the output gate, and 0.3 for the candidate value. The previous state S_{t-1} or the previous state is initialized to 0.

The following presents the results of the manual calculations for the LSTM algorithm.

1. Calculation of Forget Gate (f_t)
   The forget gate controls how much of the previous cell state should be retained. Using the sigmoid activation function, the calculation is as follows:
   $$\begin{aligned} f_t &= \sigma\,(W_f S_{t-1} + W_f X_t) \\ &= \sigma\ (0.8 * 0 + 0.8 * 0.1828) \\ &= \sigma\ (0.8 * 0.1828) \\ &= \sigma\ (0.14624) \\ &\approx 0.53645 \end{aligned}$$
2. Calculation of Input Gate (I_t)
   The input gate determines the amount of new information to be added to the current cell state:
   $$\begin{aligned} I_t &= \sigma\,(W_i S_{t-1} + W_i X_t) \\ &= \sigma\ (0.6 * 0 + 0.6 * 0.1828) \\ &= \sigma\ (0.6 * 0.1828) \\ &= \sigma\ (0.10968) \\ &\approx 0.52738 \end{aligned}$$
3. Calculation of Candidat Value ($\hat{c}$)
   The candidate value provides potential new information for the cell state:
   $$\begin{aligned} \hat{c}_t &= tanh\,(W_c S_{t-1} + W_c X_t) \\ &= tanh\ (0.9 * 0 + 0.9 * 0.1828) \\ &= tanh\ (0.9 * 0.1828) \\ &= tanh\ (0.16452) \\ &\approx 0.16359 \end{aligned}$$
4. Calculation of state cell
   The cell state is updated based on the forget and input gates:
   $$\begin{aligned} c_t &= (i_t * \hat{c}_t + f_t * c_{t-1}) \\ &= (0.10968 * 0.16452 + 0.14624 * 0) \\ &\approx 0.01805 \end{aligned}$$
5. Calculation of output gate
   The output gate decides the next hidden state based on the current cell state:
   $$\begin{aligned} O_t &= \sigma\,(W_t S_{t-1} + W_t X_t) \\ &= \sigma\ (0.7 * 0 + 0.7 * 0.1828) \\ &= \sigma\ (0.7 * 0.1828) \\ &= \sigma\ (0.12796) \\ &\approx 0.53191 \end{aligned}$$
6. Calculation of hidden state
   Finally, the hidden state is computed using the output gate and the cell state:
   $$\begin{aligned} h_t &= O_t * \tanh(c_t) \\ &= 0.10968 * \tanh(0.01805) \\ &\approx 0.00198 \end{aligned}$$

These calculations illustrate the functioning of the LSTM unit, showcasing how it manages the flow of information over time, which is crucial for tasks involving sequential data analysis.

3.4 Model Evaluation

Model evaluation is a crucial process to assess how well the model performs after implementation. In the above code, the model evaluation is conducted to examine how accurate the predictions are compared to the actual values. First, the normalized predictions are transformed back to their original scale using the inverse function of the earlier normalization process. Subsequently, evaluation metrics are calculated, namely RMSE and MAPE, each measuring the level of prediction error.

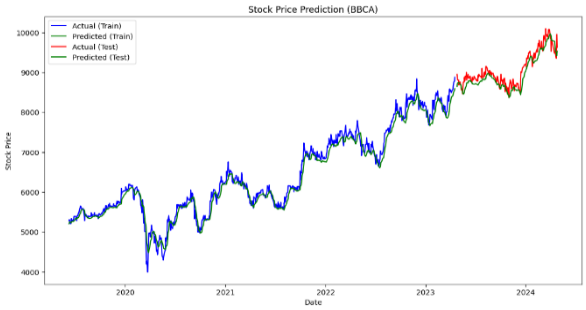
Based on the evaluation results of the model for BBCA, BBRI, and BMRI stocks from April 27, 2019, to April 27, 2024, using an architecture comprising an LSTM layer with 128 neurons, a Dense layer with 64 neurons, a dropout layer with a dropout rate of 0.2, and a final Dense layer as the output.

BBCA stock achieved an RMSE of 137.71 and an MAPE of 1.14%. BBRI stock achieved an RMSE of 87.4 and an MAPE of 1.58%. BMRI stock achieved an RMSE of 88.26 and an MAPE of 1.64%.

3.5 Data Visualization

After going through all the stages, the final step is to visualize the obtained data. Below are the graphs showing the actual and predicted prices of BBCA, BBRI, and BMRI stocks from April 27, 2019, to April 27, 2024.

Figure 7(a) illustrates the actual and predicted stock prices for BBCA.

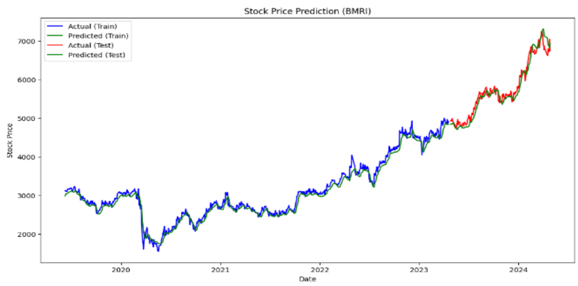


(a)

Figure 7(b) illustrates the actual and predicted stock prices for BBRI.

(b)

Figure 7(c) illustrates the actual and predicted stock prices for BMRI.



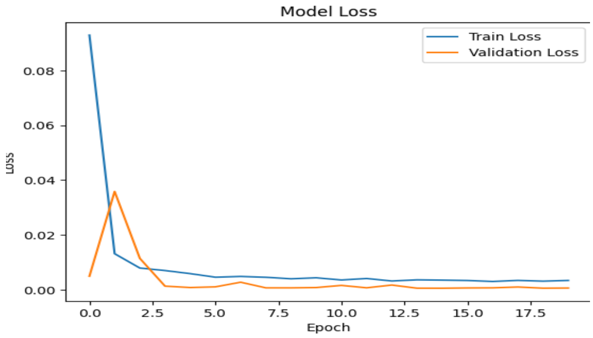
(c)

Source: Research Process

Figure 7. Prediction Graphs: (a) BBCA Stock (b) BBRI Stock (c) BMRI Stock

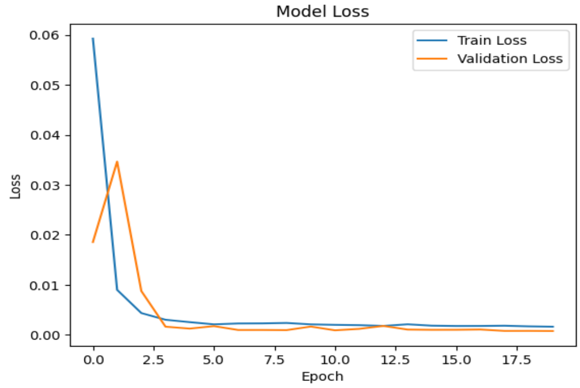
Based on the graph, the actual and predicted prices of BBCA, BBRI, and BMRI stocks were compared using the LSTM model. The Pearson correlation coefficients for BBCA, BBRI, and BMRI were found to be 0.9503399284605313, 0.9422497616573802, and 0.9805907558242081, respectively, indicating very high accuracy as they approach the value of 1.

Figure 8(a) presents the test loss results for BBCA, highlighting the performance of the LSTM model during testing.



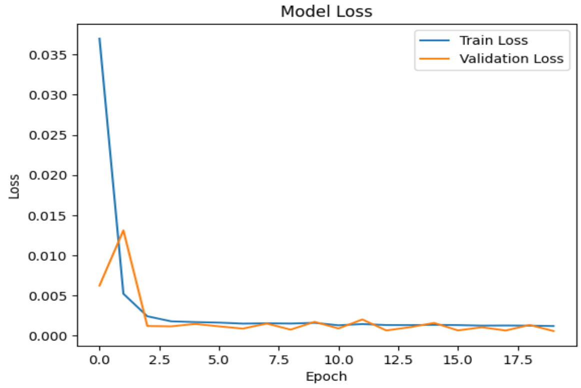
(a)

Figure 8(b) presents the test loss results for BBRI, showing the performance of the LSTM model during testing.



(b)

Figure 8(c) displays the test loss results for BMRI, highlighting the LSTM model's performance during testing.



(c)

Figure 8. Test Loss Results: (a) BBCA Stock (b) BBRI Stock (c) BMRI Stock

Figure 8 illustrates the comparison between the test loss (loss) and validation loss (val_loss) data from the LSTM model testing results for BBCA, BBRI, and BMRI stocks. According to the graph, BBCA stock achieved a test loss of 0.0006073624826967716, BBRI stock had a test loss of 0.0009700612863525748, and

BMRI stock had a test loss of 0.0015719049843028188.

Figure 9(a) illustrates the stock price prediction results for BBCA, showcasing the anticipated price movements over the next 30 days based on the LSTM model's testing outcomes.

```
Date: 2024-04-27, Predicted Price: 9610.6826171875
Date: 2024-04-28, Predicted Price: 9604.53125
Date: 2024-04-29, Predicted Price: 9598.6103515625
Date: 2024-04-30, Predicted Price: 9592.7216796875
Date: 2024-05-01, Predicted Price: 9586.841796875
Date: 2024-05-02, Predicted Price: 9580.7216796875
Date: 2024-05-03, Predicted Price: 9574.716796875
Date: 2024-05-04, Predicted Price: 9568.4638671875
Date: 2024-05-05, Predicted Price: 9561.8037109375
Date: 2024-05-06, Predicted Price: 9554.830078125
Date: 2024-05-07, Predicted Price: 9547.6240234375
Date: 2024-05-08, Predicted Price: 9540.0107421875
Date: 2024-05-09, Predicted Price: 9531.93359375
Date: 2024-05-10, Predicted Price: 9524.0205078125
Date: 2024-05-11, Predicted Price: 9515.744140625
Date: 2024-05-12, Predicted Price: 9506.8857421875
Date: 2024-05-13, Predicted Price: 9498.1474609375
Date: 2024-05-14, Predicted Price: 9488.8544921875
Date: 2024-05-15, Predicted Price: 9479.3427734375
Date: 2024-05-16, Predicted Price: 9469.482421875
Date: 2024-05-17, Predicted Price: 9460.2138671875
Date: 2024-05-18, Predicted Price: 9450.51171875
Date: 2024-05-19, Predicted Price: 9440.080078125
Date: 2024-05-20, Predicted Price: 9429.5302734375
Date: 2024-05-21, Predicted Price: 9418.7138671875
Date: 2024-05-22, Predicted Price: 9407.640625
Date: 2024-05-23, Predicted Price: 9396.21875
Date: 2024-05-24, Predicted Price: 9384.8779296875
Date: 2024-05-25, Predicted Price: 9373.46484375
Date: 2024-05-26, Predicted Price: 9361.6025390625
```

(a)

Figure 9(b) presents the stock price prediction results for BBRI, highlighting the expected price trends over the next 30 days as determined by the LSTM model's testing.

```
Date: 2024-04-27, Predicted Price: 5126.8603515625
Date: 2024-04-28, Predicted Price: 5099.31005859375
Date: 2024-04-29, Predicted Price: 5075.14501953125
Date: 2024-04-30, Predicted Price: 5053.99755859375
Date: 2024-05-01, Predicted Price: 5036.03564453125
Date: 2024-05-02, Predicted Price: 5020.51123046875
Date: 2024-05-03, Predicted Price: 5007.06494140625
Date: 2024-05-04, Predicted Price: 4994.82958984375
Date: 2024-05-05, Predicted Price: 4983.6845703125
Date: 2024-05-06, Predicted Price: 4973.42138671875
Date: 2024-05-07, Predicted Price: 4964.119140625
Date: 2024-05-08, Predicted Price: 4955.6123046875
Date: 2024-05-09, Predicted Price: 4947.4521484375
Date: 2024-05-10, Predicted Price: 4939.6904296875
Date: 2024-05-11, Predicted Price: 4932.23095703125
Date: 2024-05-12, Predicted Price: 4925.0302734375
Date: 2024-05-13, Predicted Price: 4918.02685546875
Date: 2024-05-14, Predicted Price: 4911.224609375
Date: 2024-05-15, Predicted Price: 4904.4677734375
Date: 2024-05-16, Predicted Price: 4897.8388671875
Date: 2024-05-17, Predicted Price: 4891.40966796875
Date: 2024-05-18, Predicted Price: 4884.94287109375
Date: 2024-05-19, Predicted Price: 4878.423828125
Date: 2024-05-20, Predicted Price: 4872.00537109375
Date: 2024-05-21, Predicted Price: 4865.67724609375
Date: 2024-05-22, Predicted Price: 4859.25
Date: 2024-05-23, Predicted Price: 4852.82666015625
Date: 2024-05-24, Predicted Price: 4846.38671875
Date: 2024-05-25, Predicted Price: 4839.9423828125
Date: 2024-05-26, Predicted Price: 4833.4248046875
```

(b)

Figure 9(c) presents the stock price prediction results for BMRI, highlighting the expected price trends over the next 30 days as determined by the LSTM model's testing.

```
Date: 2024-04-27, Predicted Price: 6742.3701171875
Date: 2024-04-28, Predicted Price: 6726.4404296875
Date: 2024-04-29, Predicted Price: 6709.82275390625
Date: 2024-04-30, Predicted Price: 6692.5419921875
Date: 2024-05-01, Predicted Price: 6674.787109375
Date: 2024-05-02, Predicted Price: 6656.87255859375
Date: 2024-05-03, Predicted Price: 6638.80615234375
Date: 2024-05-04, Predicted Price: 6620.22607421875
Date: 2024-05-05, Predicted Price: 6600.6767578125
Date: 2024-05-06, Predicted Price: 6581.140625
Date: 2024-05-07, Predicted Price: 6561.57958984375
Date: 2024-05-08, Predicted Price: 6541.91064453125
Date: 2024-05-09, Predicted Price: 6521.88623046875
Date: 2024-05-10, Predicted Price: 6501.93310546875
Date: 2024-05-11, Predicted Price: 6481.27294921875
Date: 2024-05-12, Predicted Price: 6460.22021484375
Date: 2024-05-13, Predicted Price: 6438.9501953125
Date: 2024-05-14, Predicted Price: 6416.42724609375
Date: 2024-05-15, Predicted Price: 6393.55322265625
Date: 2024-05-16, Predicted Price: 6370.1142578125
Date: 2024-05-17, Predicted Price: 6346.70068359375
Date: 2024-05-18, Predicted Price: 6322.9638671875
Date: 2024-05-19, Predicted Price: 6298.4775390625
Date: 2024-05-20, Predicted Price: 6273.78759765625
Date: 2024-05-21, Predicted Price: 6249.35498046875
Date: 2024-05-22, Predicted Price: 6224.59326171875
Date: 2024-05-23, Predicted Price: 6199.6708984375
Date: 2024-05-24, Predicted Price: 6174.82470703125
Date: 2024-05-25, Predicted Price: 6150.36279296875
Date: 2024-05-26, Predicted Price: 6125.4482421875
```

(c)

Source: Research Process

Figure 9. Stock Price Prediction Results: (a) BBCA Stock (b) BBRI Stock (c) BMRI Stock

Figure 9 depicts the results of the stock price predictions for BBCA, BBRI, and BMRI for the next 30 days based on the LSTM model testing.

4. Conclusion

This research analyzed the stock price predictions for BBCA, BBRI, and BMRI using adjusted close data from April 27, 2019, to April 27, 2024. The LSTM model employed consisted of an LSTM layer with 128 neurons, a Dense layer with 64 neurons, a dropout layer with a dropout rate of 0.2, and a final Dense layer for output. The results indicate that the LSTM model effectively captures the dynamics of these stock prices.

The comparison of actual versus predicted prices reveals a high degree of similarity, with Pearson correlation coefficients of 0.950 for BBCA, 0.942 for BBRI, and 0.981 for BMRI. Model evaluation metrics indicate strong performance, with RMSE values of 137.71 for BBCA, 87.4 for BBRI, and 88.26 for BMRI, and MAPE values of 1.14%, 1.58%, and 1.64%, respectively.

Despite these promising results, this study acknowledges certain limitations, including the restricted time frame and focus on only three stocks. Future research could explore predictions for a broader range of companies listed on the Indonesia Stock Exchange to identify diverse patterns and trends. Additionally, experimenting with alternative predictive algorithms may yield further insights and enhance comparative analyses.

References

Belyadi, H., & Haghighat, A. (2021). Introduction to machine learning and Python. *Machine Learning Guide for Oil and Gas Using Python*, 1–55. https://doi.org/10.1016/B978-0-12-821929-4.00006-8

Khumaidi, A., & Ayu Nirmala, I. (2022). *Algoritma Long Short Term Memory dengan Hyperparameter Tuning: Prediksi Penjualan Produk*. Deepublish.

Budiprasetyo, G., Hani'ah, M., & Aflah, D. Z. (2023). Prediksi Harga Saham Syariah Menggunakan Algoritma Long Short-Term Memory (LSTM). *Jurnal Nasional Teknologi Dan Sistem Informasi*, *8*(3), 164–172. https://doi.org/10.25077/teknosi.v8i3.2022.164-172

Dwi, S., Martono, A., Ferdiansyah, F., & Karlina, L. (2020). *Manajemen investasi dan pasar modal bagi siswa/i dan guru Akuntansi SMK*. Bintang Nusantara.

Kurnia Sari, W., Palupi Rini, D., Firsandaya Malik, R., & Saladin Azhar, I. B. (2020). Klasifikasi Teks Multilabel pada Artikel Berita Menggunakan Long Short-Term Memory dengan Word2Vec. *JURNAL RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *4*(2).

Nabillah, I., & Ranggadara, I. (2020). Mean Absolute Percentage Error untuk Evaluasi Hasil Prediksi Komoditas Laut. *JOINS (Journal of Information System)*, *5*(2), 250–255. https://doi.org/10.33633/joins.v5i2.3900

Okut, H. (2021). *Deep Learning for Subtyping and Prediction of Diseases: Long-Short Term Memory*. www.intechopen.com

Orca, O. O., & Setiawan, T. (2022.). *Apakah Return Bagi Investor Saham Akan Terdampak Akibat Pengumuman Pandemik Covid 19 Di Maret 2020?* https://doi.org/10.30813/digismantech.v1i2.3256

Prasetyo, V. R., Lazuardi, H., Mulyono, A. A., & Lauw, C. (2021). Penerapan Aplikasi RapidMiner Untuk Prediksi Nilai Tukar Rupiah Terhadap US Dollar Dengan Metode Linear Regression. *Jurnal Nasional Teknologi Dan Sistem Informasi*, *7*(1), 8–17. https://doi.org/10.25077/teknosi.v7i1.2021.8-17

Putro, E. A. N., Rimawati, E., & Vulandari, R. T. (2021). Prediksi Penjualan Kertas Menggunakan Metode Double Exponential Smoothing. *Jurnal Teknologi Informasi Dan Komunikasi (TIKomSiN)*, *9*(1), 60. https://doi.org/10.30646/tikomsin.v9i1.548

Riza, F., & Sudibyo, W. (2020). Faktor-Faktor Yang Mempengaruhi Tingkat Stock Returns Pada Perusahaan Subsektor Otomotif Dan Komponen Yang Terdaftar Di Bursa Efek Indonesia Periode 2015-2018. In *Journal of Business and Applied Management* (Vol. 13, Issue 1). http://journal.ubm.ac.id/

Rosyd, A., Irma Purnamasari, A., & Ali, I. (2024). Penerapan Metode Long Short Term Memory (Lstm) Dalam Memprediksi Harga Saham Pt Bank Central Asia. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 8, Issue 1).

Rustiana, D., Ramadhani, S., & Batubara, M. (2022). *Strategi di Pasar Modal Syariah*.

Tanto, V. M., & Kurniawan, T. A. (2022). *Pengembangan Sistem Rekomendasi Investasi Saham berbasis Web (Studi Kasus: Reliance Sekuritas Malang)* (Vol. 6, Issue 6). http://j-ptiik.ub.ac.id

Toranggi Berton, F., Aasya Aldin Islamy, M., Ramadhan Fitra, R., Gede Angga Dinata, I., & Yudistira, N. (2022). *Prediksi Harga Saham Indosat Menggunakan Algoritma LSTM*.

Veny, V., & Gunawan, Y. (2022). Perubahan Harga Saham Dilihat Dari Faktor Fundamental Perusahaan Makanan Dan Minuman. *Jurnal Akuntansi Bisnis*, *15*(1). https://doi.org/10.30813/jab.v15i1.2874

Verkino, B., Sinaga, B. M., & Andati, T. (2020). Portofolio Optimal Investasi Saham dari 6 Sektor pada Indeks LQ45 Periode 2015-2018. *Jurnal Aplikasi Bisnis Dan Manajemen*. https://doi.org/10.17358/jabm.6.2.389

Wiranda, L., & Sadikin, M. (2019). *Penerapan Long Short Term Memory Pada Data Time Series Untuk Memprediksi Penjualan Produk Pt. Metiska Farma* (Vol. 8).