

Stock Prediction System using Technical Indicators with the LSTM Method

Revelin Angger Saputra ^{1*}, Donni Richasdy ², Yuliant Sibaroni, ³

^{1,2,3}*School of Computing, Telkom University
Telekomunikasi St, Bandung, Indonesia*

revanggerev@gmail.com

Abstract

The capital market industry in Indonesia is developing in a better direction so that the growth of new investors is also increasing. Until the end of February 2021, operational data from the Indonesian Stock Exchange (IDX) and data from the Indonesian Central Securities Depository (KSEI) recorded that the number of new capital market investors had increased by 16.35% or 634,350 investors, from the previous 3,880,753 investors. to 4,515,103 investors. The development of the capital market industry in Indonesia, which has increased investor interest in investing, is expected to mobilize public funds to support national economic development. Some companies that are familiar to the community are BCA, BNI, BRI and MANDIRI. This study attempts to forecast banking stock prices on the LQ45 index, using the Long Short-Term Memory (LSTM) method. LSTM is one of the Recurrent Neural Networks (RNN) which has good accuracy in predictions. The identified fields are Close, Open, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Moving Average (MA). The evaluation method used in this prediction system is MAPE in the form of percent output. Based on the analysis of the average evaluation values from each test, test 1 had an average evaluation value of 0.86%, test 2 had an average value of 0.70%, and test 3 had an average value of 0.77%. Test 2 using the *Close, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Moving Average (MA)* fields had a better evaluation value with an average value of 0.70% compared to test 1 using the *Close and Open* fields and test 3 using the *Close, Open, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Moving Average (MA)* fields.

Keywords: Prediction System, Stocks, LSTM, Technical Indicators, MAPE

I. INTRODUCTION

Currently, the investment world, especially the capital market, is growing in Indonesia. The capital market brings together those who need long-term funding with those who want to invest in financial products such as stocks, bonds, mutual funds, and others. The Indonesian Stock Exchange/IDX is the trading place for each stock. Thanks to the capital market, companies can obtain large amounts of funds in a short period of time. A company gets funds through investors who invest their capital in the company by buying the company's stocks. [1].

According to Manoj S Hegde in [2], the stock market is where publicly listed company stocks are traded. The uncertain stock market makes it a serious area of study and analysis. Although considered unpredictable, it is said to be influenced by certain factors such as historical stock trends, trending news, social media sentiment, and many other factors. Therefore, by expanding the range of factors that affect stock prices, greater accuracy can be achieved. Although sometimes unpredictable due to certain factors such as stock trends, current news,

social media sentiment, global events, and public figures who consciously or unconsciously become trendsetters such as Elon Musk in Dodge Coin.

According to data gathered from IDX [3], the capital market in Indonesia is starting to grow, so the number of new investors is also increasing. Until the end of February 2021, the operational data of the Indonesia Stock Exchange (IDX) and the data of the Indonesia Central Securities Depository (KSEI) recorded an increase of 16.35% or 634,350 new investors, from the previous 3,880,753 investors to 4,515,103 investors. The increasing interest in securities/stocks is due to the belief that investing in the capital market has high-profit prospects. The development of the capital market in Indonesia, which attracts investor interest, is expected to mobilize the public to support national economic development.

In addition to getting profits, investors can also suffer losses when choosing the wrong investment in stocks, such as capital loss, not receiving dividends, and liquidity risk. Companies will not provide dividends if they decide not to distribute dividends to investors and choose to use them to increase the company's capital and assets. Capital gain is the difference between the increase in shareholder value, and capital loss is the difference between the decrease in stock when buying or selling. There is a great potential for losses when choosing to invest in a company's stock, so investing in stocks in the capital market recommends that investors have good analytical skills in order to maximize profits and minimize losses.

Quoting from the book *Pintar Pasar Modal Indonesia* [4], tools for analysis are generally divided into 2. (1) Fundamental Analysis, where the analysis is based on the company's economic fundamentals. (2) Technical Analysis, which is an analysis carried out to determine price movements or observation of stock price movements in real-time within a certain time frame that is described in the form of a graph.

Based on the background, the system for predicting the movement of company stock prices in Indonesia using the LSTM method is expected to provide investors with insights to make decisions. This research is also expected to be a reference for further research on specific company stock analysis.

This stock prediction system will use technical indicators on the history of the stock, limiting its variables. By using technical factors, investors can make decisions in choosing companies to invest in. This research uses technical indicators such as RSI, MACD, and MA to be implemented in the stock prediction system that will be created. The method of the prediction system used is Long Short-Term Memory Network (LSTM). The LSTM algorithm considers past data and past output. Because of its characteristics, LSTM is suitable for sequential data that has a dependence on the previous output, such as stock values.

A. Topics and Limitations

The system being developed is intended for desktop/PC and is being worked on in the Colab platform, which aims to facilitate usage for the public. The dataset being used consists of historical stock data for BCA, BNI, BRI, and Mandiri. The output produced is in the form of graphs, evaluation values, and the stock prices for the last 5 days based on predictions.

B. Objective

The aim of this study is to develop a time-series prediction system using stock data that can help provide insight to investors in the form of prediction graphs to determine stock selection for several banking companies. This prediction system also provides output for the stock prices for the last 5 days.

Objective of this study is building a stock prediction system that provides predicted and actual stock price graphs, evaluation metrics, and stock prices for the past 5 days based on the prediction results, with a several scenario testing. The first test will use only the "Close" and "Open" fields to generate output in the form of predicted and actual price graphs, evaluation scores, and stock prices for the last 5 days based on predictions. The second test will use the "Close," "RSI," "MACD," and "MA" fields to generate output in the form of predicted and actual price graphs, evaluation scores, and stock prices for the last 5 days based on predictions. The third test will use the "Close," "Open," "RSI," "MACD," and "MA" fields to generate output in the form of predicted and actual price graphs, evaluation scores, and stock prices for the last 5 days based on predictions.

The purpose of conducting these three scenarios is to find the optimal testing approach for the problem at hand. By testing different combinations of features and variables, the aim is to identify the most effective approach for predicting stock prices and providing accurate outputs in the form of predicted and actual price charts, evaluation metrics, and stock prices for the past five days. The ultimate goal is to develop a reliable stock prediction system that can provide valuable insights for investors.

II. LITERATURE REVIEW

The stock market is one of the important elements of the economy that is used as a benchmark for the progress or even decline of a country's economy. A country is considered advanced if it has a well-developed stock market, which can be seen through the composite stock price index (IHSG) that reflects the status of issuers. Therefore, the IHSG can be seen as a reflection of the country's economic situation. A country in an economic crisis is reflected in a sharp decline in the composite stock price index. The stock market can be used to attract foreign investors and foreign funds to help improve the country's economy by increasing the composite stock price index or IHSG as a benchmark for each issuer.

To increase their capital, companies that have become issuers can conduct initial public offerings (IPOs), in which they sell some or all of the company's shares through the stock exchange with the help of the exchange company. On the other hand, people who have available funds can invest in the stock exchange by buying stocks, bonds, derivative products, or mutual funds. Although investing in the stock exchange carries greater risks than investing in banking products such as savings and deposits, it also has the potential for greater profits. However, there is a risk of loss and even a risk of losing all invested funds. Therefore, wise and careful analysis is required before investing in the stock market. Experience and knowledge in the stock market are also very important for managing investments properly.

	2020	2021	2021			
			Q1	Q2	Q3	Q4
EQUITIES						
Jakarta Composite Index						
High	6,325.406	6,723.386	6,435.205	6,107.538	6,286.943	6,723.386
Low	3,937.632	5,760.584	5,862.352	5,760.584	6,026.020	6,228.845
Close	5,979.073	6,581.482	5,985.522	5,985.489	6,286.943	6,581.482
Stock Market Summary						
Trading Day	242	247	61	59	63	64
Listed Companies, eop	713	766	724	735	750	766
Listed Shares, m. shares	6,054,120	7,227,250	6,277,728	6,433,201	6,727,696	7,227,250
Market Capitalization, b. IDR	6,970,009	8,255,624	7,071,300	7,107,007	7,715,390	8,255,624
Stock Trading, total						
Volume, m. Shares	2,752,471	5,096,450	1,210,408	1,040,616	1,326,840	1,518,586
Value, b. IDR	2,228,798	3,302,932	969,068	639,855	792,847	901,162
Frequency, times	163,937,983	319,821,157	84,238,953	62,779,320	87,080,608	85,722,276
Stock Trading, daily average						
Volume, m. Shares	11,374	20,633	19,843	17,638	21,061	23,728
Value, b. IDR	9,210	13,372	15,886	10,845	12,585	14,081
Frequency, times	677,430	1,294,822	1,380,966	1,064,056	1,382,232	1,339,411
Stock Foreign Trading Value, b. IDR						
Buying	677,316	850,807	226,283	181,503	190,125	252,896
Selling	725,129	812,833	214,419	176,650	180,333	241,432
Net	(47,813)	37,974	11,864	4,853	9,793	11,465

Figure 1. Trading volume data in Indonesia.
(IDX, Indonesia Stock Exchange, n.d.)

According to Purnomo in his book [5], the products traded in the stock market are securities or intangible instruments. In the development of information and communication technology, securities transactions in the stock market no longer require paper documents and can be done online through remote trading. The products traded include stocks, bonds, derivatives, and mutual funds. To raise capital, a company can sell some of its shares or issue bonds through the stock market, which is considered more efficient than applying for a bank loan.

The statistical report of the stock market by KSEI [3] also show the Indonesian stock market industry has experienced positive development, as seen from the growth in the number of new investors. Operational data from the Indonesia Stock Exchange (BEI) and the Indonesian Central Securities Depository (KSEI) show that by the end of February 2021, the number of new investors had increased by 16.35% or about 634,350 investors, from 3.880.753 investors to 4.515.103 investors. This positive development is expected to encourage people to support national economic development through investment in the stock market. Many people are interested in investing in the stock market because it is considered to have a high profit prospect, especially in investing in securities or stocks.

Based on Figure 1, IDX recorded an almost two-fold increase in average daily trading volume in 2020 to 2021. And for the total trading volume in 2021, it increased to 1,074,134 from 2020. This shows a positive trend in Indonesia society which is starting to explore the stock market.

A. Technical Analysis

The basis of technical analysis is to use past stock price movement data to determine where the price movement will go next. What needs to be emphasized from technical analysis is its ability to identify trends as early as possible.

- **Relative Strength Index**

Relative Strength Index (RSI) is a technical indicator used in financial market analysis to measuring the strength of trends and price volatility. This indicator calculates the ratio of the average price increase in a certain period with the average price decrease in the same period. RSI can be used on various types of assets such as stocks, currencies, commodities, and market indices. RSI is used to identify whether an asset is traded in an overbought or oversold condition. When RSI is above 70, the asset is considered overbought and is likely to experience a price decrease in the near future. Conversely, when RSI is below 30, the asset is considered oversold and is likely to experience a price increase in the near future. RSI can also be used to confirm price trends, with rising RSI values indicating a bullish trend, while falling RSI values indicating a bearish trend. [6] The time period used in calculating RSI can vary depending on the trader's or analyst's preferences, but a 14-day time period is the most commonly used. RSI is calculated using a formula that involves the calculation of positive and negative price changes over a period of time and is expressed as a percentage. RSI is usually displayed in the form of a line chart below the price chart. RSI has the formula:

$$RSI = 100 - 100 \times \frac{AL}{AL + AG}$$

Where

AL : Average Loss

AG : Average Gain

- **Moving Average (MA)**

Moving Average (MA) is a technical indicator used in financial market analysis to smooth out price movements and identify market trends. This indicator calculates the average price of an asset over a certain period of time and displays this value as a line on a price chart. Moving averages can be used in different time periods, such as daily, weekly, or monthly. For example, a 50-day moving average calculates the average price over the last 50 days, while a 200-day moving average calculates the average price over the last 200 days.

There are two common types of moving averages: Simple Moving Average (SMA) and Exponential Moving Average (EMA). SMA calculates the average price simply by adding prices over a certain period of time and dividing it by the number of time periods. On the other hand, EMA gives greater weight to more recent prices and less weight to older prices.

Moving averages can help traders and analysts identify market trends, whether bullish or bearish. When the price is above the moving average line, the trend is considered to be bullish, and when the price is below the moving average line, the trend is considered to be bearish. In addition, the crossover of moving average lines can also provide buy or sell signals, depending on whether the shorter moving average line crosses the longer moving average line from bottom to top or from top to bottom.

- **Moving Average Convergence Divergence**

Moving Average Convergence Divergence (MACD) is a technical indicator used in financial market analysis to measure the strength of trends and price volatility. The indicator measures the difference between two different exponential Moving Averages (MAs), namely the 12-period MA and the 26-period MA, and displays this value as the MACD line on the price chart. MACD can also generate a signal line, which is an exponential average of the MACD itself, typically calculated with a 9-period. When the MACD line crosses the signal line from bottom to top, this is considered a buy signal, while when the MACD line crosses the signal line from top to bottom, this is considered a sell signal. MACD can help investors and analysts identify market trends, both upward and downward. The indicator can also help identify divergences between MACD and price, where there is a difference between price movement and MACD movement, which can provide signals of trend changes. MACD can also be used to identify overbought and oversold conditions. When the MACD line is above the signal line and the histogram line is positive, this indicates an overbought condition, while when the MACD line is below the signal line and the histogram line is negative, this indicates an oversold condition.

B. Dataset

LQ45 is an index that measures the price performance of 45 stocks with high liquidity and large market capitalization, supported by good company fundamentals. The banking stocks that will be analyzed are only the stocks from four out of five banks that are included in the LQ45 and have the highest number of shares for the index, namely Bank Central Asia (BBCA) with 36,038,997,951 shares, Bank Mandiri (BMRI) with 18,461,519,999 shares, Bank Rakyat Indonesia (BBRI) with 70,190,307,940 shares, and Bank Negara Indonesia (BBNI) with 7,375,636,872 shares. This data is based on the LQ45 major evaluation issued by the Indonesian Stock Exchange on January 25, 2023. The dataset that will be used is downloaded using Yahoo Finance within the timeframe of 09/11/2017 - 08/11/2022. The downloaded dataset will then be processed by adding several technical indicator fields, namely RSI, MACD, and MA. This process is done using Excel. The timeframe that will be analyzed is within the range of December 15, 2017 to August 08, 2022.

C. LSTM

LSTM (Long Short-Term Memory) is a type of neural network model used in time series analysis and forecasting. LSTM is designed to overcome the long-term dependency problem that occurs in conventional

RNN (Recurrent Neural Network) models. LSTM allows the model to learn and remember information from a longer time period through unique memory cells. These memory cells are connected to gate structures, including input gates, forget gates, and output gates, which control the flow of information in and out of the memory cells. The input gate controls how much new information should enter the memory cells, while the forget gate controls how much old information should be deleted from the memory cells. The output gate controls how much information should be outputted from the memory cells to the next layer. In each memory cell, there are three sigmoid layers and one hyperbolic tangent (tanh) layer. The sigmoid layers produce values between 0 and 1, which determine how much information should be passed through, while the tanh layer produces values between -1 and 1 that determine how much information should be stored in the memory cell. LSTM is particularly suitable for processing time series data with complex patterns and interactions between variables, and is capable of overcoming the vanishing gradient problem that occurs in conventional RNN models. The sigmoid layer displays values on a scale of zero to one, indicating how much each component should be passed through. A value of zero means "not allowing anything to pass through", while a value of one means "allowing everything to pass through". The sigmoid will have a value of zero if it is located in the range $0 \leq \sigma \leq 0.5$, and will have a value of one if it is located in the range $0.5 < \sigma \leq 1$.

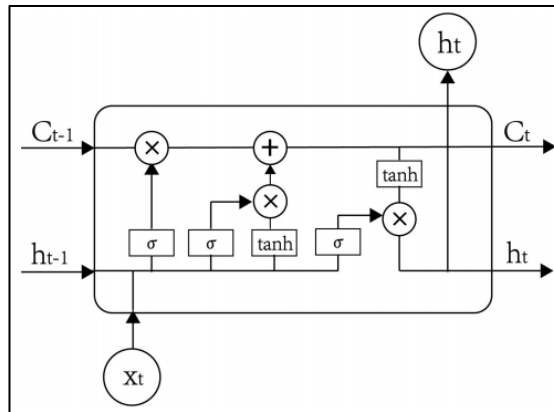


Figure 2. Structure of the LSTM Model
(Source : Qiu, Wang, & Zhou, 2020)

Figure 2 show the structure of the LSTM model, for more details will be explained below using a mathematical formula.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$\tanh(x) = 2\sigma(2x) - 1$$

Where x is the input data and e is a mathematical constant.

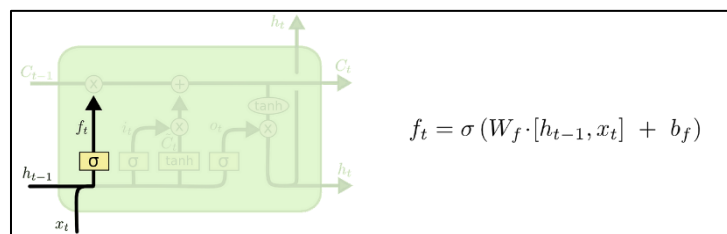


Figure 3. Forget Gate Structure in LSTM
(Source: Olah, 2015)

On figure 3 explain that the forget gate in LSTM unit determines which information from the cell state will be discarded from the model. As shown in Figure 3, the memory cell receives the h_{t-1} output from the previous moment and the external information x_t from the current moment as input and combines them into a long vector $[h_{t-1}, x_t]$. The notation $[h_{t-1}, x_t]$ represents the concatenation operation, [8] which means adding rows of x_t with rows of h_{t-1} through the σ transformation to become:

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

Where

f_t : Forget gate

σ : Sigmoid

W_t : weight values for forget gate

h_{t-1} : Output values before order - t

x_t : Input values on - t

b_f : bias value at forget gate

The main function of the forget gate in an LSTM (Long Short-Term Memory) network is to determine how much information from the previous memory cell (C_{t-1}) should be stored or ignored when generating the current memory cell state (C_t). The forget gate processes input from the previous output (h_{t-1}) and the current input (x_t), then outputs a value between 0 and 1, where 1 indicates full information retention and 0 indicates the information will be completely erased. In this way, the forget gate helps control the flow of information in the LSTM network and ensures that irrelevant or unnecessary information is not stored in the memory cell.

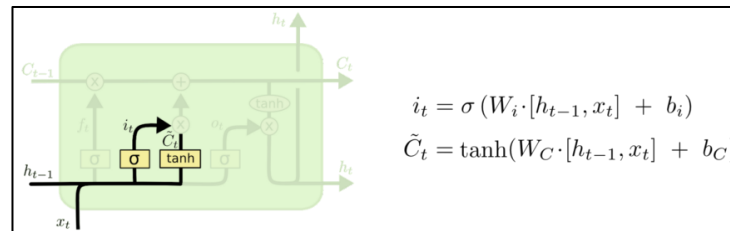


Figure 4. Gate Input Structure in LSTM
(Source: Olah, 2015)

Based on Figure 4, the main function of the input gate in a LSTM (Long Short-Term Memory) network is not to determine the amount of input at the current time. The input gate controls the flow of information into the memory cell (C_t) by determining how much new information (x_t) will be added to the memory cell. The input gate processes input from the previous input (h_{t-1}) and the current input (x_t) through a sigmoid function, which produces a value between 0 and 1. This value indicates how much information should be updated in the previous memory cell status (C_{t-1}) to update the current memory cell status (C_t). The input gate also helps filter out irrelevant or insignificant information before it is stored in the memory cell. In this way, the input gate helps control and improve the information that enters the memory cell, so that the information stored in the memory cell becomes more relevant and useful for the LSTM network. [8] as shown in this formula:

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

i_t : Input gate

σ : Sigmoid

W_i : Weight values for input gate

h_{t-1} : Output values before orde - t

x_t : Input values on orde - t

b_i : Bias value on input gate

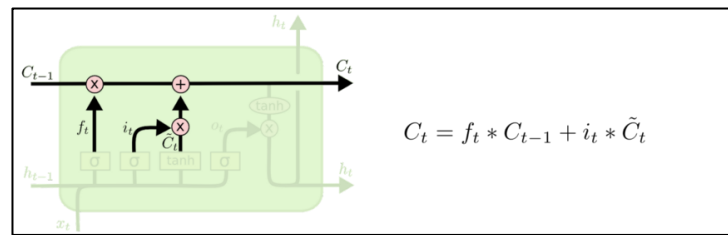


Figure 5. Cell State LSTM Structure
(Source : Olah, 2015)

Figure 5 show for other functions, it is to update the information from b to the updated state of the cell. A new candidate vector C_t is created through the tanh layer to control how much new information is added, as in the equation below, which is used to update the state of the memory cell. [8]

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

\hat{C}_t : The new value added to the cell state.
 tanh : tanh function
 W_c : Weight value for cell state
 h_{t-1} : Output value before orde - t
 x_t : Input value on orde - t
 b_c : The bias value on the cell state.

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t$$

C_t : The new value added to the cell state.
 f_t : forget gate
 C_{t-1} : Cell state before orde t
 i_t : input gate
 \hat{C}_t : The new value added to the cell state.

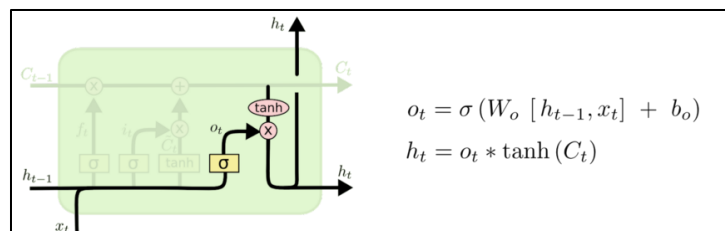


Figure 6. Output Gate LSTM Structure
(Source : Olah, 2015)

Figure 6 show the output gate controls how much of the current cell state is discarded. The first output information is determined by the sigmoid layer, then the cell state is processed by tanh and multiplied by the output of the sigmoid layer to obtain the final output part. [8]

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

O_t : Output gate

σ : Sigmoid Function

W_0 : Weight value for output gate

h_{t-1} : Output value before orde - t

x_t : Input value on orde - t

b_0 : The bias value on the cell state.

Final output defined as :

$$h_t = O_t * \tanh (C_t)$$

h_t : Output value orde t

O_t : Output gate

\tanh : Tanh function

C_t : cell state

D. Evaluation Method

MAPE (Mean Absolute Percent Error) It is a calculation used to determine the average percentage of absolute error, with the formula:

$$MAPE = \sum (|Actual - Forecast | / Actual) * 100 / n$$

From the above formula, $\sum (| Actual - Forecast | / Actual)$ is the result of subtracting the actual value from the prediction, and dividing by the actual value for each period, then summing up the results. And "n" is the number of periods used for calculation. The lower the MAPE value, the better the forecasting model used, and for MAPE there is a range of values that can be used as a measure of the ability of a forecasting model, the range of values can be seen in the Table 1.

TABLE I
MAPE ASSESSMENT CLASSIFICATION.
(Source : Yuhan Xie & Yunfie Yang. 2022)

Range MAPE	Definition
< 10 %	Very good forecasting model capability.
10 - 20 %	Good forecasting model capability.
20 - 50 %	Sufficient forecasting model capability.
> 50 %	Bad forecasting model capability.

III. RESEARCH METHOD

This forecasting system is built using a dataset obtained from Yahoo Finance. Using this data, a machine learning model is used as the backbone of the computation. The model contains an LSTM module to capture trends in stock closing prices history. The data obtained from Yahoo Finance is not directly processed. The acquired data will be pre-processed to adjust its constituent components, such as variables, the use of dots or commas in decimals, and others. An overview of the process of this recommendation system can be seen in Figure 7.

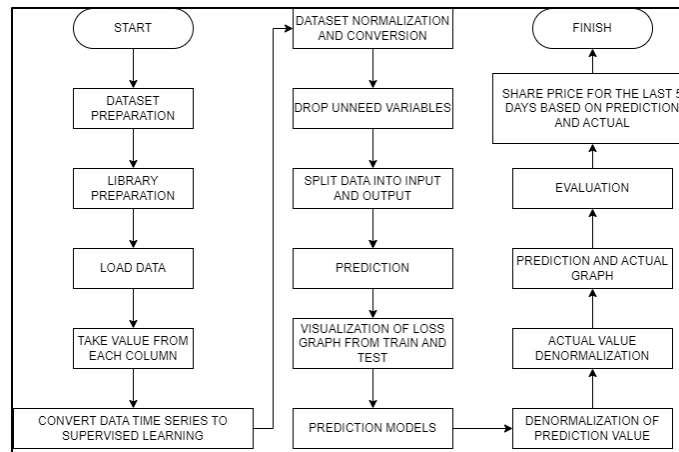


Figure 7. Workflow Diagram of the System.

In *Dataset Preparation*, the obtained dataset has several fields and is in CSV format. Adjustments were made by converting the file format to Excel. The fields in the newly downloaded dataset from Yahoo Finance are Date, Open, High, Low, Close, Adj Close, and Volume. Then, several fields were added to obtain the values of technical indicators such as RSI, MACD, and MA. The added fields are Gain, Loss, Average Gain, Average Loss, RS, RSI, EMA12, EMA26, MACD, and MA.

To obtain the RSI value, an additional field is needed, namely Gain, Loss, Average Gain, Average Loss, and RS. To obtain Gain, the difference between the latest price and the price yesterday is calculated from the Close field, and the same is done for the Loss field by finding the difference in stock prices between yesterday and today. After that, the Average Gain and Average Loss are calculated by finding the average of 14 days from the Gain and Loss fields. RS is obtained by dividing Average Gain by Average Loss. And RSI is obtained from the formula $=100-100/(1+RS(n))$.

The MACD value is obtained from EMA12 and EMA26. EMA 12 is the average of the Close field value for 12 days, and EMA 26 is the average of the Close field value for 26 days. And MACD is obtained by subtracting EMA 12 from EMA26. MA is obtained from the average of the Close field value every 3 days. Then, the value of each field is increased by 1 level to obtain T-2, the value of T-2 is increased by 1 level to obtain T-3, and the value of T-3 is increased by 1 level to obtain T-4. Library Preparation First, the stock data that will be used as a dataset is downloaded from Yahoo Finance and grouped into 1 folder to facilitate work.

The next step is to import several libraries commonly used in data science and machine learning, including pandas, numpy, datetime, math, matplotlib, scikit-learn, and keras. These libraries provide tools for data manipulation, analysis, visualization, modeling, and neural network training. Specifically, the code imports modules for reading and manipulating data, scaling and encoding variables, dividing data into training and test sets, defining and training neural network models, and evaluating model performance using mean squared error.

The LSTM layer from the Keras library is used to build a recurrent neural network, which is very useful for time series forecasting problems. Upload Prepared Dataset The dataset that has been adjusted and added technical indicator fields is uploaded to Colab and ready to use in the system. The system will read the Excel file with the filename used, using the `read_excel()` function from the pandas library. The dataset is then stored in the dataset variable. Using `header=0` indicates that the first row of the Excel file is the header, so it is used as the column name in the pandas DataFrame. Then `index_col=0` is used to indicate that the first column in the Excel file is the index, so it is used as the row index in the pandas DataFrame. Extracting Values from Dataset

After reading the Excel file, the dataset will be converted into a NumPy array format using the values attribute of the DataFrame object. By performing `values = dataset.values`, the values variable will contain a NumPy array consisting of the values in the dataset. This NumPy array can be used to perform various mathematical and statistical operations on the dataset.

The `series_to_supervised()` function is used to convert datasets into inputs and outputs that are suitable for recurrent neural network models, especially LSTM. This function takes four parameters, namely `data`, `n_in`, `n_out`, and `dropnan`. `data` is initialized with the dataset to be converted. If the dataset consists of one variable, the `data` parameter is filled with a list. If the dataset consists of several variables, the `data` parameter is filled with a `DataFrame` object. `n_in` is the number of past time steps to be used as input to the model. `n_out` is initialized as the number of past time steps to be used as output from the model, with `dropnan` being a boolean parameter that determines whether rows with empty (NaN) values should be removed from the output. This function produces a `DataFrame` that contains input and output variables. The `DataFrame` column names depict time $(t - n, t - n + 1, \dots, t - 1, t, t + 1, \dots, t + n - 1)$ for each input and output variable.

Normalization, conversion, and dropping columns in the dataset. Variables in the dataset are normalized using the `MinMaxScaler` from `scikit-learn`. `MinMaxScaler` is used to normalize numeric variables into a specified range. In this study, `feature_range=(0.1, 0.9)` is used to indicate that the data will be normalized into the range of 0.1 to 0.9. After the data is normalized, the resulting variables are stored in a variable called `scaled`. This variable is then used to prepare the data for training the LSTM model. To view the scaled values, the command `print(scaled)` can be added to view the values in this variable.

Splitting data into input and output The data is then divided into a training set and a testing set, with 80% or 981 rows used as the training set and the rest as the testing set. The training set is then divided into input (`train_X`) and output (`train_y`) sets, while the testing set is divided into input (`test_X`) and output (`test_y`) sets. The input set is reshaped into a 3D array with a shape of `[samples, timesteps, features]`.

Defining the LSTM model Next, the LSTM network is designed and compiled using the `Keras` library. The network consists of one LSTM layer with 20 units and one dense layer with one output unit. The network is compiled with the mean squared error (`mse`) loss function and the `Adam` optimizer. The network is then fitted to the training data with 500 epochs and a batch size of 72. The trained model is saved to disk as `"savedModel"`. Training the LSTM model To train the LSTM model with the previously prepared data, several steps are required. `"model.fit()"` is a `Keras` method used to train the model with the prepared dataset. The first and second parameters in this method are variables that contain the training data (`train_X` and `train_y`). This system uses 500 epochs. Epochs indicate the number of iterations during training, where the entire dataset is given to the model at each iteration. The batch size indicates the number of samples that will be processed at each epoch, in this prediction system, the batch size used is 72. To determine the test data, this prediction system uses `validation_data`. This parameter must contain variables that contain the test data (`test_X` and `test_y`). The `model.fit()` method will evaluate the model at each epoch using the test data. `Verbose` indicates how detailed information is displayed during training. In this study, `verbose` is used as 2, which means that the information displayed is per epoch. After the training is complete, the model will be saved in the directory specified by the `model.save()` command. The storage directory of this model can adjust parameters on the `model.save()` command.'

Prediction Model To estimate the output value of the model, use `test_X` data as input. The output of the model will be stored in a customizable variable, defined as `yhat` in this prediction system. Next, to reshape the `test_X` array, where the shape of the `test_X` array was previously `(number_of_samples, number_of_timesteps, number_of_features)`, the shape of the `test_X` array is changed to `(number_of_samples, number_of_features)` by ignoring the number of timesteps. This is done so that scaling inversion can be performed on the data and output of the model.

Visualize Loss step is used to create a plot of the model's training history. Defining `history.history` as a dictionary containing the training metric values at each epoch. Then plot the loss value at each epoch for the training and validation data. `"pyplot.plot()"` is used to create the plot of both loss values. Labels are used to give each curve a label. Then `pyplot.legend()` is used to display the legend on the plot, and `pyplot.show()` is used to display the plot on the screen.

Denormalize Prediction Values Next, to return the value `yhat` from the normalization scale to its original scale. First, combine the `yhat` value with a part of the original data used as input in `test_X`, starting from the second column (`test_X[:, 1:]`), so that it returns to its original shape with the same number of features as the

original data. Next, return the inv_yhat value from the normalization scale to the original scale using the $inverse_transform$ method of the scaler object. To obtain the actual values from the original data of the first column resulting from $inverse_transform$, $inv_yhat = inv_yhat[:,0]$ is performed, because the first column is the column that contains the target or output value in the original data.

Denormalize Actual Values is the step to return the actual $test_y$ value from the normalization scale to its original scale. First, combine the $test_y$ value with a part of the original data used as input in $test_X$, starting from the second column ($test_X[:, 1:]$), so that it returns to its original shape with the same number of features as the original data. Next, return the inv_y value from the normalization scale to the original scale using the $inverse_transform$ method of the scaler object. To obtain the actual values from the original data of the first column resulting from $inverse_transform$, $inv_y = inv_y[:,0]$ is performed because the first column is the column that contains the target or output value in the original data.

Visualize Prediction Results and Valid Values is the step to visualize the model's prediction results and actual values from the original data, use the matplotlib module to create a plot. This function receives one or more numeric arrays that represent data and can determine the color, label, and type of plot to be used. Draw a graph that shows the actual values from the original data plotted as a blue line, draw a graph that shows the model's prediction results plotted as an orange line, and add a legend to the graph with a customizable position.

Evaluation Calculate the MAPE using NumPy. Calculate the absolute difference between the actual value and the predicted value, then divide that difference by the actual value. MAPE is calculated by taking the average of all absolute percentage errors and multiplying it by 100 to get the result in percentage form. The smaller the MAPE value, the better the model's performance in predicting actual values.

Provide the last 5 days of stock prices Display the predicted stock price for the last 5 days and will display the actual stock price for the last 5 days. This output is useful to determine how accurate the model is in predicting the stock prices for the last 5 days and comparing it with the actual stock prices during the same period.

IV. RESULTS AND DISCUSSION

In this program, users are required to upload a dataset to be used and enter the file name in the "load data" section. The testing is done using the historical stock data of BCA, BNI, BRI, and Mandiri, within the date range of 2017/12/15 - 2022/08/08. The testing will be done three times, where each test will use different variables. In the first test, only the Close and Open fields will be used, in the second test, Close, RSI, MACD, and MA fields will be used, and in the last test, Close, Open, RSI, MACD, and MA fields will be used.

The first test (Test 1), only using the Close and Open fields. The evaluation results can be seen in Table II.

TABEL II
THE EVALUATION RESULT OF TEST 1

	BCA	BNI	BRI	MANDIRI
MAPE	0.75%	1.04%	0.40%	1.26%

Based on the first experiment, the MAPE values for BCA were 0.75%, 1.04% for BNI, 0.40% for BRI, and 1.26% for Mandiri. To facilitate visualization of the predictions, Figure 8 show the actual stock prices with predictions.

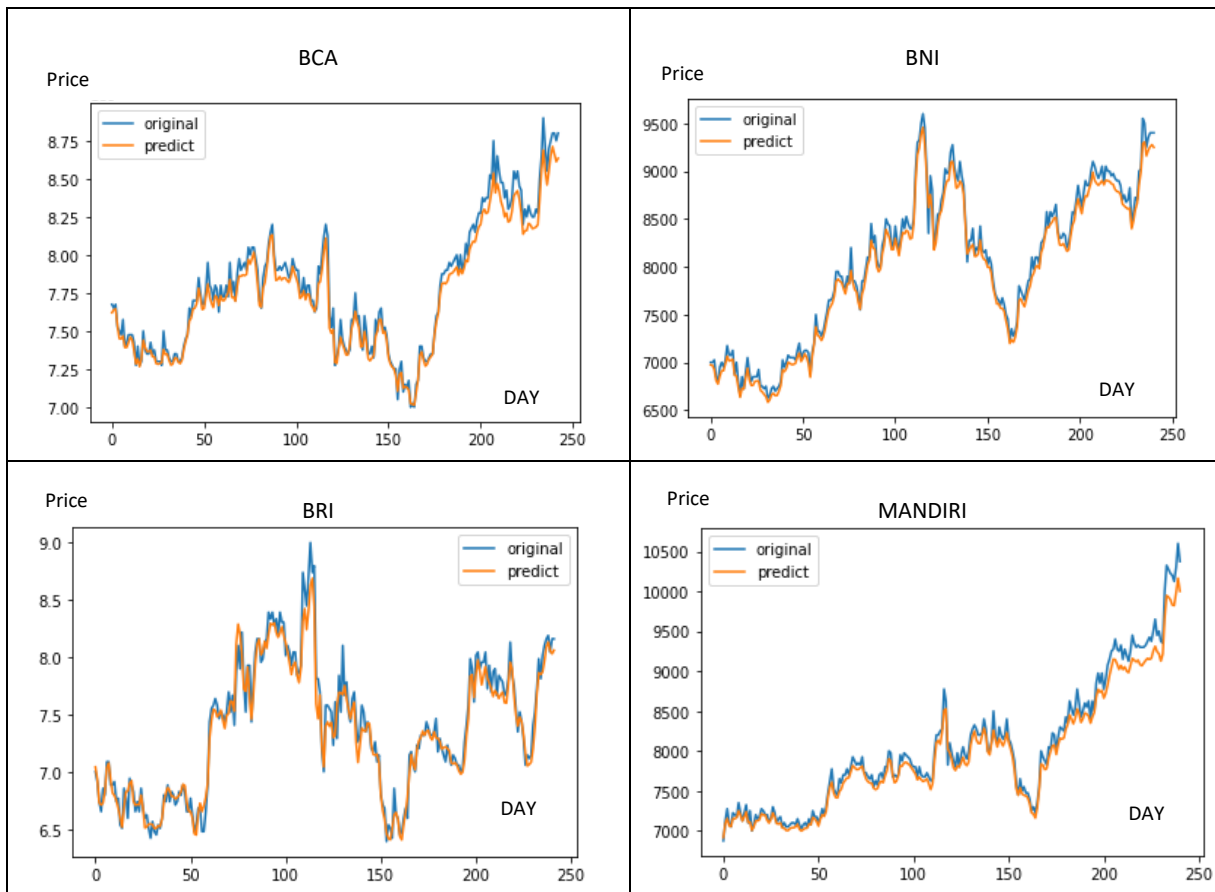


Figure 8. Prediction Graph of BCCA, BNI, BRI, MANDIRI Using Test 1

Figure 8 show after this prediction system has been successfully created and running properly, the second test will be conducted, with a scenario using Close, RSI, MACD, and MA fields.

TABEL III.
THE EVALUATION RESULT OF TEST III

	BCA	BNI	BRI	MANDIRI
MAPE	0.48%	1.40%	0.40%	0.51%

Based on the second experiment, the MAPE values obtained are BCA at 0.48%, BNI at 1.40%, BRI at 0.40%, and MANDIRI at 0.51%. To facilitate the visualization of predictions, Figure 9 show the actual stock price graphs with the predictions.

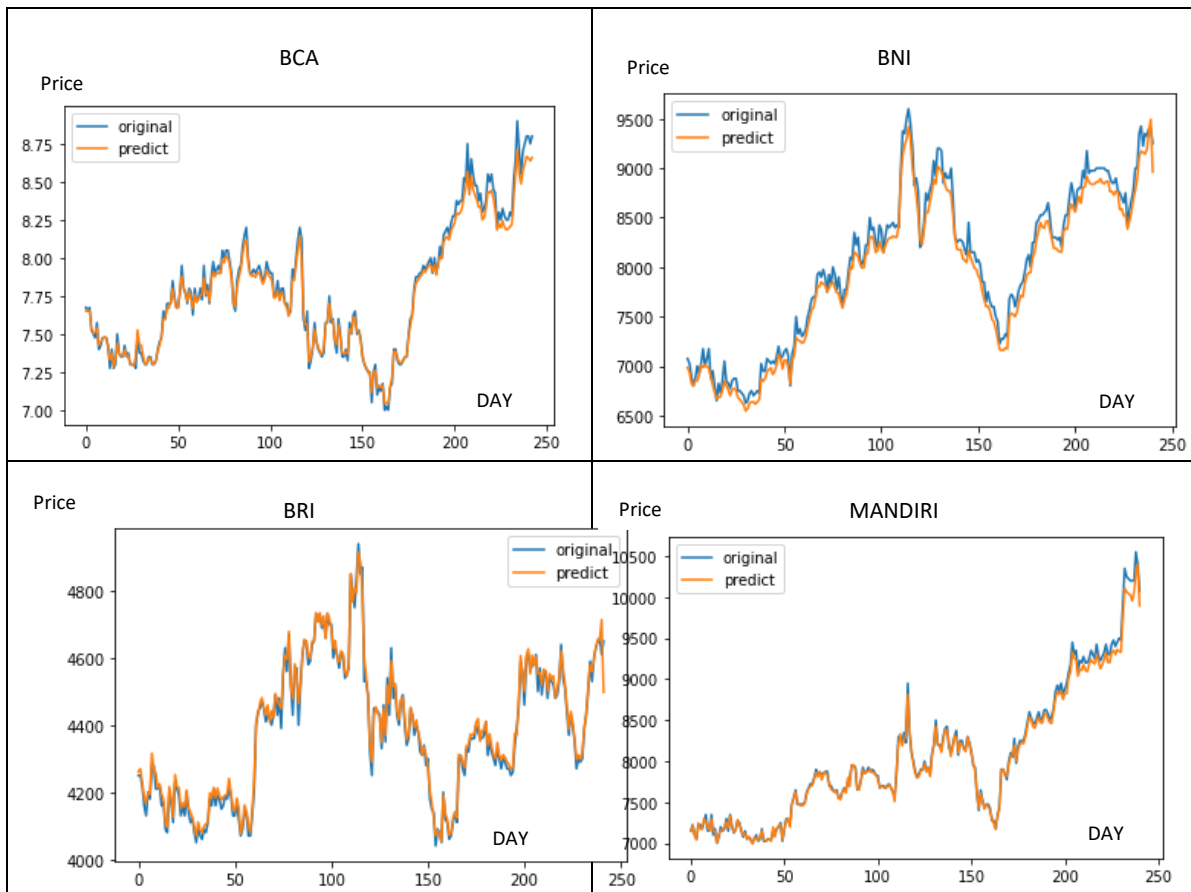


Figure 9. Prediction Graph of BCCA, BNI, BRI, MANDIRI Using Test 2

Next, the third test will be conducted, with a scenario using Close, Open, RSI, MACD, and MA fields.

TABEL IV.
 THE EVALUATION RESULT OF TEST 3

	BCA	BNI	BRI	MANDIRI
MAPE	1.07%	0.83%	0.60%	0.59%

Based on the third experiment, the MAPE values for BCA, BNI, BRI, and Mandiri are 1.07%, 0.83%, 0.60%, and 0.59%, respectively. To facilitate the visualization of the predictions, Figure 10 show the actual stock price graphs with the predictions.

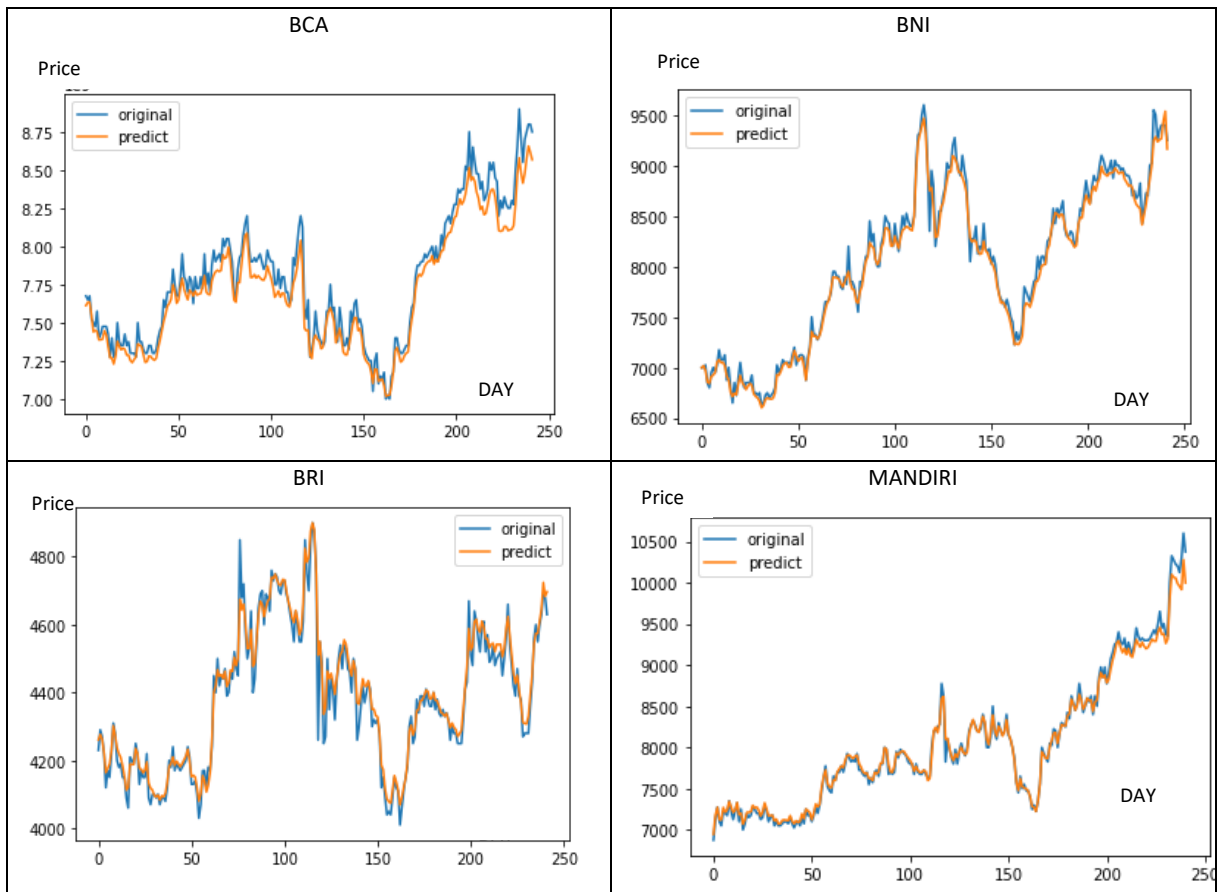


Figure 10. Prediction Graph of BCCA, BNI, BRI, MANDIRI Using Test 3

After conducting Test 1, Test 2, and Test 3, satisfactory results were obtained. Test 1 used Close and Open fields, test 2 used Close, RSI, MACD, and MA fields, and Test 3 used Close, Open, RSI, MACD, and MA fields. The evaluation results of the prediction system were obtained. To facilitate analysis of the test results, the following table contains the test data to be analyzed further.

TABEL V
ANALYSIS OF EVALUATION RESULTS FOR EACH SHARE

	BCA	BNI	BRI	MANDIRI	Average
MAPE 1	0.75%	1.04%	0.40%	1.26%	0.86%
MAPE 2	0.48%	1.40%	0.40%	0.51%	0.70%
MAPE 3	1.07%	0.83%	0.60%	0.59%	0.77%

Based on the analysis of the average evaluation values from each test, test 1 had an average evaluation value of 0.86%, test 2 had an average value of 0.70%, and test 3 had an average value of 0.77%. Test 2 using the Close, RSI, MACD, and MA fields had a better evaluation value with an average value of 0.70% compared to test 1 using the Close and Open fields and test 3 using the Close, Open, RSI, MACD, and MA fields.

V. CONCLUSION

LSTM can be a very useful tool for predicting stock prices. However, stock price predictions cannot be used as the sole definitive guide for making investment decisions without further analysis. based on three experiments conducted on 4 banks, showing different MAPE values for each bank. However the MAPE values obtained were below 10%, which indicates that the forecasting model has very good predictive ability.

From the three experiments that have been conducted, it can be concluded that the second experiment using Close, RSI, MACD, and MA fields had a better evaluation score with an average value of 0.70% compared to the first experiment using only Close and Open fields and the third experiment using Close, Open, RSI, MACD, and MA fields. Therefore, the forecasting system for timeseries is more optimal when using Close, RSI, MACD, and MA fields only.

Testing is only done 3 times based on predetermined indicators. Indicators are determined based on indicators that are popularly used. The limited testing time is also a factor in this study, only 3 tests were carried out. The main limitation of using machine learning algorithms for stock price prediction is that this research can only backtest historical data and price movements do not always follow historical trends under various unexpected conditions such as political factors, global crises, wars, and individual influences. Further analysis is needed to produce predictions that generate more profits for investors.

ACKNOWLEDGMENT

Praise and gratitude be to Allah SWT who has bestowed His blessings and mercy so that the author can complete this final project and journal. This final project and journal were carried out in order to fulfill one of the requirements to complete the practical work course at Telkom University. The author realizes that without the help and guidance from various parties, from the study period to the preparation of this journal, it would have been very difficult for the author to complete it. Therefore, the author would like to express gratitude to:

1. God Almighty for His blessings and gifts that have enabled the author to be in this position.
2. Prophet Muhammad SAW who has been a mercy to all the universe.
3. Dr. Z.K Abdurahman Baizal, S.Si, M.Kom as the Dean of the Faculty of Informatics, Telkom University;
4. Dr. Erwin Budi Setiawan, S.Si., M.T as the Head of the Informatics Undergraduate Program, Telkom University;
5. Donni Richasdy S.T, M.T as the first supervisor who has provided time, energy, and thoughts to guide the author in preparing this Final Project;
6. Dr. Yuliant Sibaroni, S.Si., M.T as the second supervisor who has provided time, energy, and thoughts to guide the author in preparing this Final Project;
7. The author's parents and family who have helped the author both materially and morally;
8. Diana Laili Putri who has provided support and attention;
9. Friends who have provided support.

REFERENCE

- [1] C. Felita, "Pembangunan Sistem Rekomendasi Saham Berdasarkan Analisis Fundamental Menggunakan Metode Weighted Product," *Pembangunan Sistem Rekomendasi Saham Berdasarkan Analisis Fundamental Menggunakan Metode Weighted Product*, P. 9, 2018.
- [2] PES Institute of Technology , "An Ensemble Stock Predictor and Recommender System," *IEEE*, p. 1, 2018.
- [3] KSEI, "StatistikPasarModal Indonesia," Finance , February 2021. [Online]. Available: https://www.ksei.co.id/files/Statistik_Publik_-_Februari_2022.pdf. [Diakses 21 November 2022].
- [4] R. Ang, *Buku Pintar Pasar Modal Indonesia (The Intelligent Guide to, Jakarta: Mediasoft Indonesia, 1997.*
- [5] Purnomo, *Pasar uang & pasar valas, Jakarta: Gramedia Pustaka Utama, 2013.*
- [6] Zipmex, "Definisi MACD dan Perbedaannya dengan RSI," *Teknologi*, 2021. [Online].
- [7] B. W. C. Z. Jiayu Qiu, "Forecasting stock prices with long-short term," *Plos*, 2019.
- [8] K. D. Larasati, "Medium," 30 June 2022. [Online]. Available: <https://medium.com/@dhea.larasati326/multivariate-long-short-term-memory-dengan-python-c7170f443bd9>. [Diakses 15 February 2023].
- [9] M. A. Maricar, "Analisa Perbandingan Nilai Akurasi Moving Average dan Exponential Smoothing untuk Sistem Peramalan Pendapatan pada Perusahaan XYZ," *JURNAL SISTEM DAN INFORMATIKA*, p. 39, 2019.