Faktor Exacta

Vol. 19, No. 1, March 2026, pp. 48~59
eISSN: 2502-339X, pISSN: 1979-276X, https://doi.org/10.30998/faktorexacta.v19i1.2036

BBCA Stock Price Forecasting Using Statistical, Machine Learning, and Deep Learning Models with Technical Indicator Features: A Comparative Analysis

Rendi Prasetya[1], Herlinda[2]
[1, 2] Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Indraprasta PGRI, Indonesia

Article Info

Article history:
Received 29/01/2026
Revised 17/06/2026
Accepted 26/06/2026

Keywords:
Stock price prediction
Gaussian process regression
Long short-term memory
Relative strength index
Time-series prediction

ABSTRACT

Due to high volatility, non-linearity, and complex dynamics, stock price forecasting remains a challenging task in financial time-series analysis. This study quantitatively compares the forecasting performance of six methods: Moving Average (MA), Linear Regression (LR), Support Vector Regression (SVR), Random Forest (RF), Gaussian Process Regression (GPR), and Long Short-Term Memory (LSTM), utilizing daily BBCA stock data from May 2011 to April 2026. Preprocessing encompassed data validation, chronological ordering, Min-Max normalization, and feature engineering via lagged price values, MA, and the Relative Strength Index (RSI). Model evaluation relied on Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE), supplemented by historical visualization and a 30-trading-day future projection. Experimental results indicate that GPR achieved superior performance (MAE: 58.68, MSE: 8,940.29, RMSE: 94.55, MAPE: 1.12%). RF yielded competitive results (MAE: 113.36, MSE: 86,609.19, RMSE: 294.29, MAPE: 1.54%), followed by LSTM (MAE: 88.75, MSE: 16,269.79, RMSE: 127.55, MAPE: 1.92%). Conversely, LR and SVR exhibited higher prediction errors, failing to capture complex price movements. Visualizations confirm that GPR, RF, and LSTM closely mirrored actual price trends, with GPR and LSTM generating more stable 30-day trajectories than linear or conventional statistical approaches. Overall, non-linear learning-based models, particularly GPR, offer reliable alternatives for short-term stock forecasting.

Corresponding Author:
Rendi Prasetya,
Teknik Informatika, Fakultas Teknik dan Ilmu Komputer
Universitas Indraprasta PGRI,
Jl. Nangka No. 58 C, Tanjung Barat, Jagakarsa, Jakarta Selatan.
Email: prasetyarendi@gmail.com

1. INTRODUCTION

Stock markets exhibit highly volatile, non-linear, and stochastic characteristics influenced by various factors, including economic conditions, geopolitical events, investor sentiment, and market dynamics. Such characteristics make stock price forecasting a challenging task in financial time-series analysis. Accurate prediction of stock price movements plays an important role in investment decision-making, portfolio optimization, and financial risk management. However, stock price data frequently contain noise, structural changes, and temporal dependencies that complicate forecasting processes and reduce prediction reliability [1], [2], [3].

In the context of the Indonesian capital market, BBCA stock represents one of the blue-chip stocks with long historical trading records and is widely considered relevant for stock price forecasting analysis. Based on the historical BBCA stock data used in this study from May 2011 to April 2026, the closing price exhibited substantial movement over the observation period. The initial closing price was IDR 1,500, while the final closing price in April 2026 was approximately IDR 5,850. This indicates an increase of approximately 290%

from the initial closing price. During the observation period, the maximum closing price reached approximately IDR 10,800. This wide price range indicates that BBCA stock contains long-term trend patterns, short-term fluctuations, and dynamic price behavior. Therefore, BBCA stock is considered relevant for evaluating forecasting models that are expected to capture both linear and non-linear patterns in financial time-series data.

Traditional forecasting approaches initially dominated stock market prediction studies due to their relatively simple implementation and low computational cost. However, increasing market complexity has accelerated the development of advanced approaches capable of modeling dynamic and non-linear relationships within financial datasets. Machine learning and deep learning techniques have gained significant attention because of their ability to improve forecasting performance and capture complex market patterns [4], [5], [6].

Recent studies have explored hybrid forecasting architectures integrating Long Short-Term Memory (LSTM), attention mechanisms, and hybrid deep learning frameworks to improve forecasting accuracy and model robustness [7], [8], [9], [10], [11]. Furthermore, comparative studies involving Bidirectional Long Short-Term Memory (BiLSTM) and temporal forecasting models have demonstrated improvements in learning sequential patterns and long-term dependencies within financial datasets [12], [13].

Despite substantial progress in stock market forecasting research, many previous studies primarily focus on improving individual forecasting architectures rather than systematically comparing traditional statistical, machine learning, and deep learning approaches under identical experimental settings. Furthermore, studies involving Indonesian blue-chip stocks, particularly BBCA, remain relatively limited despite their importance in the Indonesian capital market. Consequently, identifying forecasting approaches capable of providing robust and reliable performance for practical investment applications remains challenging.

This study aims to quantitatively compare the forecasting performance of six stock price prediction methods, namely Moving Average, Linear Regression, Support Vector Regression, Random Forest, Gaussian Process Regression, and Long Short-Term Memory, using BBCA historical stock price data. The comparison is conducted based on four error metrics: MAE, MSE, RMSE, and MAPE. In addition, this study incorporates lagged price features, Moving Average, and Relative Strength Index to represent temporal price behavior, trend information, and market momentum in short-term BBCA stock price forecasting.

The contribution of this study lies in providing a comparative evaluation of statistical, machine learning, and deep learning approaches under identical dataset, preprocessing, feature engineering, and evaluation settings for BBCA stock forecasting. Additionally, this study applies technical indicator-enhanced feature representation and evaluates not only historical prediction accuracy but also future forecasting behavior through a 30-trading-day projection analysis.

1.1 Traditional Statistical Methods

Moving Average (MA) is a statistical indicator used to calculate average stock prices over a specified time window. The primary purpose of MA is to smooth short-term fluctuations and identify market trends. Although MA is computationally efficient and easy to interpret, its lagging nature limits its capability to capture complex non-linear relationships in stock price movements.

Linear Regression (LR) is a statistical method employed to model linear relationships between dependent and independent variables. Due to its simplicity and interpretability, LR has been widely adopted in forecasting studies. However, the assumption of linearity often limits its effectiveness in highly volatile financial environments because stock market behavior typically exhibits dynamic and non-linear characteristics [1], [4].

1.2 Machine Learning Methods

Support Vector Regression (SVR) is a regression-based extension of Support Vector Machines designed to minimize prediction errors while preserving model generalization capability. Through kernel functions, SVR can model non-linear relationships in financial time-series data and is commonly applied in stock price prediction studies [4].

Random Forest (RF) is an ensemble learning approach that combines multiple decision trees to improve predictive accuracy and model robustness. Random Forest uses bagging and aggregation mechanisms to reduce prediction variance and is recognized for its capability to model complex interactions among variables while maintaining robustness against noisy data [14].

Gaussian Process Regression (GPR) is a probabilistic non-parametric regression approach that models a distribution over possible prediction functions. GPR provides flexibility in representing non-linear relationships and enables uncertainty estimation [15], making it relevant for financial forecasting applications involving dynamic market behavior and risk considerations. Machine learning-based approaches have also been shown to be effective in empirical asset pricing and financial prediction tasks [6].

1.3 Deep Learning Methods

Long Short-Term Memory (LSTM) is a variant of Recurrent Neural Networks designed to overcome vanishing gradient problems in sequence learning tasks. LSTM employs memory cells and gating mechanisms that enable efficient learning of long-term temporal dependencies [5], [16], [17].

Previous studies reported that BiLSTM and transformer-based architectures demonstrated improvements in modeling long-term dependencies and contextual information in financial time-series forecasting tasks [7], [12]. Hybrid approaches integrating convolutional neural networks and attention mechanisms further improved feature extraction and forecasting stability in stock price prediction tasks [8].

In addition to predictive architectures, incorporating technical indicators such as Relative Strength Index (RSI) and Moving Average has become a common strategy for improving feature representation in stock forecasting. These indicators enrich the input features by providing additional information regarding market momentum and trend behavior, which may contribute to better prediction performance in financial time-series forecasting [18], [19].

2. METHOD

This study employed a quantitative approach with a comparative experimental design to evaluate the performance of various stock price forecasting models. The research framework consisted of data acquisition, data preprocessing, feature engineering, model development and training, performance evaluation, and future stock price forecasting. Historical data acquisition and preprocessing were performed using Python, while predictive modeling and experimental analysis were conducted using MATLAB.

2.1 Data Acquisition

The dataset utilized in this study consisted of historical daily stock data of BBCA (PT Bank Central Asia Tbk). Historical stock data were automatically collected from Yahoo Finance using the *yfinance* library in Python. The dataset covered approximately fifteen years, spanning from May 2011 to April 2026. Daily observations were selected to capture both short-term fluctuations and long-term stock price movements. The acquired dataset included several stock attributes, namely opening price, highest price, lowest price, closing price, adjusted closing price, and trading volume. The collected data were stored in CSV format for subsequent preprocessing, feature engineering, model training, and performance evaluation.

Before preprocessing, the dataset still retained its original structure as obtained from Yahoo Finance. The raw data consisted of daily trading records arranged chronologically based on trading dates. A sample of the raw BBCA historical stock data before preprocessing is presented in Table 1. The table shows several observations from the early period, middle period, and final period of the dataset to illustrate the variation of stock price movement over time.

Table 1. Sample of Raw BBCA Historical Stock Data Before Preprocessing

No.	Period Date	Open	High	Low	Close	Volume
1	02-May-2011	1,500	1,500	1,480	1,500	11,520,000
2	03-May-2011	1,450	1,500	1,450	1,450	29,397,500
3	04-May-2011	1,470	1,470	1,420	1,470	66,220,000
4	05-May-2011	1,430	1,450	1,430	1,430	48,987,500
5	06-May-2011	1,460	1,460	1,410	1,460	39,665,000
6	09-May-2011	1,440	1,460	1,440	1,440	17,860,000
7	10-May-2011	1,460	1,470	1,440	1,460	8,327,500
8	11-May-2011	1,470	1,470	1,440	1,470	32,395,000
9	12-May-2011	1,440	1,460	1,440	1,440	21,392,500
10	13-May-2011	1,460	1,470	1,430	1,460	56,490,000
11	16-May-2011	1,420	1,440	1,410	1,420	61,670,000
12	18-May-2011	1,430	1,460	1,420	1,430	93,630,000
13	19-May-2011	1,450	1,480	1,440	1,450	45,402,500
14	20-May-2011	1,460	1,480	1,440	1,460	50,617,500
15	23-May-2011	1,430	1,450	1,410	1,430	57,972,500

As shown in Table 1, the raw dataset contained numerical stock price attributes and trading volume before any preprocessing procedures were applied. The selected sample illustrates the structure of the original dataset and the variation in BBCA stock prices across the observation period. These data were then processed through several stages, including chronological validation, missing value handling, normalization, and feature construction. The preprocessing stage was required to ensure data consistency and to transform the original stock data into a suitable input format for the forecasting models.

2.2 Data Preprocessing

The acquired historical data underwent preprocessing procedures before model development. These procedures included data validation, chronological ordering of observations, normalization, denormalization preparation, and training-testing data splitting. Data validation was conducted to ensure that each observation contained valid trading date information and closing price values. Observations with invalid dates or missing closing price values were excluded from the dataset to maintain data consistency and reliability. The data validation process is expressed as follows:

$$I_i = \begin{cases} 1, & if\ t_i \neq NaT\ and\ y_i \neq NaN \\ 0, & otherwise \end{cases}$$

where (I_i) denotes the validity index of the (i)-th observation, (t_i) represents the trading date, and (y_i) represents the closing price. Only observations with $(I_i = 1)$ were retained for further processing.

After data validation, the observations were arranged chronologically based on trading dates to preserve the temporal structure of the stock price data. Chronological ordering was required because time-series forecasting depends on the sequence of historical observations. Subsequently, stock price values were normalized into the range of ([0,1]) using the Min-Max normalization method to improve model convergence and prevent scale dominance during model training [16]. The Min-Max normalization equation is defined as follows:

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

where (x') denotes the normalized value, (x) denotes the original stock price value, (x_{min}) denotes the minimum value, and $(x_{max}\})$ denotes the maximum value in the dataset.

The normalization parameters were retained and later applied during the denormalization process for prediction analysis. Denormalization was performed to transform normalized prediction values back into the original stock price scale. The denormalization equation is defined as follows:

$$x = x'(x_{max} - x_{min}) + x_{min}$$

where (x) denotes the denormalized value and (x') denotes the normalized prediction value.

The dataset was then divided into training and testing data using a time-based splitting strategy. The first 80% of the observations were used as training data, while the remaining 20% were used as testing data. The number of training observations is expressed as follows:

$$N_{train} = \lfloor 0.8 \times N \rfloor$$

where (N) denotes the total number of observations and (N_{train}) denotes the number of training observations. This splitting strategy was applied to preserve the chronological order of stock price data and to simulate realistic forecasting conditions.

2.3 Feature Engineering

Feature engineering was performed specifically for machine learning and deep learning models, including LSTM, SVR, Random Forest, and GPR. The extracted features consisted of lagged values, Moving Average, and Relative Strength Index. These features were used to capture temporal dependencies, trend characteristics, and momentum information in stock price movements. The extracted features consisted of:

a. Lagged values were constructed from previous normalized closing prices to represent sequential relationships within the stock price series [20]. In this study, five lagged values were used as input features. The lag feature is expressed as follows:

$$X_t = [y'_{t-5}, y'_{t-4}, y'_{t-3}, y'_{t-2}, y'_{t-1}]$$

where (X_t) denotes the lag feature vector at time (t), and (y'_{t-k}) denotes the normalized closing price from the previous (k)-th observation.

b. Moving Average was calculated to represent trend characteristics and smooth short-term price fluctuations. In this study, a 30-day Moving Average window was used. The Moving Average equation is defined as follows:

$$MA_t = \frac{1}{30}\sum_{i=1}^{30} y'_{t-i}$$

where (MA_t) denotes the Moving Average value at time (t), and (y'_{t-i}) denotes the normalized closing price from previous observations.

c. Relative Strength Index was utilized as a momentum indicator to identify the strength of stock price movements and represent overbought or oversold market conditions [1]. RSI was calculated using a 14-day period. The price change is defined as follows:

$$\Delta y_t = y_t - y_{\{t-1\}}$$

The gain and loss values are calculated as follows:

$$Gain_t = max(\Delta\, y_t, 0)$$
$$Loss_t = |min(\Delta\, y_t, 0)|$$

The average gain and average loss over a 14-day period are expressed as follows:

$$Average\ Gain_t = \frac{1}{14}\sum_{i=1}^{14} Gain_{t-i}$$
$$Average\ Loss_t = \frac{1}{14}\sum_{i=1}^{14} Loss_{t-i}$$

The relative strength value is calculated as follows:

$$RS_t = \frac{Average\ Gain_t}{Average\ Loss_t + \epsilon}$$

where (ϵ) is a very small value used to avoid division by zero. The RSI value is then calculated using the following equation:

$$RSI_t = 100 - \frac{100}{1 + RS_t}$$

The RSI value was normalized into the range of ([0,1]) using the following equation:

$$RSI_t = \frac{RSI_t}{100}$$

For SVR, Random Forest, and GPR, the final input feature vector consisted of five lagged normalized closing prices, one Moving Average feature, and one normalized RSI feature. The machine learning feature vector is expressed as follows:

$$X_t = [y'_{t-5}, y'_{t-4}, y'_{t-3}, y'_{t-2}, y'_{t-1}, MA_t, RSI_{t-1}\,]$$

while the prediction target is defined as follows:

$$Y_t = y'_t$$

For the LSTM model, the input sequence consisted of normalized closing price and normalized RSI values over the previous 10 trading days. The LSTM input sequence is expressed as follows:

$$X_t^{LSTM} = \begin{bmatrix} y'_{t-10} & y'_{t-9} & \cdots & y'_{t-1} \\ RSI_{t-10} & RSI_{t-9} & \dots & RSI_{t-1} \end{bmatrix}$$

and the target output is expressed as follows:

$$Y_t^{LSTM} = y'_t$$

All extracted features were normalized to maintain uniform scaling across variables. The Linear Regression model utilized time index values as predictor variables, whereas the Moving Average model operated directly on historical stock prices without additional features. Previous studies suggested that technical indicators such as RSI and moving averages can significantly improve predictive performance by enriching market trend and momentum information [18], [19].

2.4 Model Development and Training

The dataset was divided into training and testing datasets using a time-based splitting strategy, where earlier observations were allocated for model training and later observations for model evaluation. This approach was employed to replicate realistic financial forecasting scenarios and preserve temporal relationships within stock market data.

The forecasting models developed in this study consisted of:

a. Moving Average (MA): implemented as a baseline statistical approach using average stock prices within a predefined window. The Moving Average model generated predictions by calculating the average of normalized closing prices from the previous 30 trading days.

$$\hat{y}_t = \frac{1}{30}\sum_{i=1}^{30} y'_{t-i}$$

b. Linear Regression (LR): modelled the relationship between the time index and normalized stock prices using a linear function.

$$\hat{y}_t = \beta_0 + \beta_1 x_t + \varepsilon_t$$

c. Long Short-Term Memory (LSTM): implemented using a recurrent neural network architecture consisting of sequence input, LSTM, fully connected, and regression layers. Model training employed the Adam optimization algorithm.

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right)$$
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$$
$$C_t = f_t C_{t-1} + i_t \tilde{C}_t$$
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$
$$h_t = o_t \tanh(C_t)$$
$$\hat{y}_t = W_y h_t + b_y$$

d. Support Vector Regression (SVR): implemented using a Radial Basis Function (RBF) kernel. Hyperparameter tuning was performed to minimize forecasting errors and improve prediction accuracy [4]. Support Vector Regression was implemented using a Gaussian Kernel to capture non-linear relationship between input features and target values.

$$\hat{y}_t = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x_i, x_t) + b$$
$$K(x_i, x_t) = \exp(-\gamma \|x_i - x_t\|^2)$$

e. Random Forest (RF): developed using an ensemble learning approach combining multiple decision trees through bagging techniques to improve predictive robustness. Random Forest generated predictions by averaging the outputs of 100 regression trees.

$$\hat{y}_t = \frac{1}{100} \sum_{b=1}^{100} T_b(x_t)$$

f. Gaussian Process Regression (GPR): implemented using a squared exponential kernel function to model complex non-linear stock price patterns. Gaussian Process Regression models stock price movement probabilistically using a squared exponential kernel.

$$f(x) \sim GP\left(m(x), k(x, x')\right)$$
$$k(x, x') = \sigma_f^2 \exp\left(-\frac{\|x - x'\|^2}{2l^2}\right)$$

2.5 Model Performance Evaluation

Model performance was evaluated using four forecasting error metrics, namely Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). These evaluation metrics were computed by comparing the actual closing price values with the predicted values generated by each forecasting model. Only valid prediction values were used in the evaluation process to ensure that the error calculation was performed consistently across all models.

a. Mean Absolute Error (MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

b. Mean Squared Error (MSE)

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

c. Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

d. Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

Where (y_i) denotes the actual closing price value, $(\hat{y}_i)$ denotes the predicted closing price value, and (n) denotes the number of valid observations used in the evaluation. Lower MAE, MSE, RMSE, and MAPE values indicate better forecasting performance.

These evaluation metrics were computed using testing datasets to objectively compare prediction performance among the developed forecasting models.

2.6 Future Stock Price Forecasting

In addition to historical prediction evaluation, future BBCA stock prices were projected for the subsequent 30 trading days. Forecasting was conducted using an iterative multi-step forecasting strategy, where prediction outputs generated at previous steps were recursively utilized as input variables for subsequent forecasting periods.

The forecasting process considered trading-day observations only to reflect realistic stock market conditions. Predicted values were transformed back to their original scale using the denormalization process based on previously stored normalization parameters.

2.7 Implementation Environment

The modeling, evaluation, visualization, and future forecasting processes were conducted using MATLAB R2022a. The implementation was supported by Deep Learning Toolbox for LSTM model development, Statistics and Machine Learning Toolbox for Linear Regression, Support Vector Regression, Random Forest, and Gaussian Process Regression, and Financial Toolbox for financial time-series analysis support. Historical BBCA stock data were obtained from Yahoo Finance and stored in CSV format. Python with the yfinance and pandas libraries was used to support historical data acquisition and preliminary data preparation, while the main preprocessing, feature engineering, model training, performance evaluation, and forecasting analysis were conducted in MATLAB to ensure consistency across all evaluated models.

3. RESULTS AND DISCUSSION

This section presents the experimental results of BBCA stock price forecasting using six forecasting methods, namely Moving Average, Linear Regression, Support Vector Regression, Random Forest, Gaussian Process Regression, and Long Short-Term Memory. The results are presented according to the research stages described in the method section, including dataset description, preprocessing results, feature engineering results, model evaluation, historical prediction analysis, and future stock price forecasting.

3.1 Dataset and Preprocessing Results

The dataset used in this study consisted of daily BBCA stock data collected from Yahoo Finance for the period of May 2011 to April 2026. The raw dataset contained daily trading attributes, including opening price, highest price, lowest price, closing price, and trading volume. Before model development, the dataset was validated to remove invalid observations and arranged chronologically to preserve the temporal structure of the stock price series.

The closing price was used as the main target variable in the forecasting process. Based on the raw data sample presented in Table 1, BBCA stock price exhibited daily variations in open, high, low, close, and volume values. These variations indicate that stock price movement contains temporal patterns that need to be represented properly before model training.

After validation and chronological ordering, the closing price values were normalized using Min-Max normalization into the range of [0,1]. This normalization process was applied to improve model convergence and prevent scale dominance during model training. The normalization parameters were stored and later used to transform predicted values back into the original stock price scale through denormalization. The dataset was then divided into training and testing data using an 80:20 time-based splitting strategy, where the earlier observations were used for training and the later observations were used for model evaluation.

3.2 Feature Engineering Results

Feature engineering was conducted to improve the representation of temporal patterns, trend characteristics, and momentum information in BBCA stock price movements. The constructed features consisted of lagged closing prices, Moving Average, and Relative Strength Index.

Five lagged normalized closing prices were used to represent short-term temporal dependencies in the stock price series. A 30-day Moving Average was used to capture trend characteristics and smooth short-term fluctuations. Meanwhile, a 14-day RSI was used to represent price momentum and market movement strength.

For SVR, Random Forest, and GPR, the input feature vector consisted of five lagged normalized closing prices, one Moving Average feature, and one normalized RSI feature. For LSTM, the input sequence consisted of normalized closing price and normalized RSI values over the previous 10 trading days. This feature construction enabled the forecasting models to learn historical price behavior not only from previous prices but also from trend and momentum indicators.

3.3 Historical Analysis and Model Performance Comparison

Table 2 presents the comparative forecasting error metrics obtained from the six evaluated models. Model performance was evaluated using MAE, MSE, RMSE, and MAPE. These metrics were computed by comparing actual closing price values and predicted values generated by each model.

Table 2. Predictive Error Comparison for BBCA Historical Analysis

Method	MAE	MSE	RMSE	MAPE (%)
Moving Average	157.0094	52,846.01	229.8826	3.1082
Linear Regression	602.2523	611,527.96	782.0025	14.4214
Support Vector Regression	535.8636	1,743,372.44	1320.3700	7.7478
Random Forest	113.3626	86,609.19	294.2944	1.5414
Gaussian Process Regression	58.6839	8,940.29	94.5531	1.1187
Long Short-Term Memory	88.7468	16,269.79	127.5531	1.9169

Based on the results presented in Table 2 and Figure 1, machine learning and deep learning approaches generally achieved lower prediction errors than traditional statistical methods. This observation indicates that learning-based approaches possess stronger capabilities for modeling non-linear and dynamic characteristics of stock market data. Similar findings have been reported by Patel [4], Sezer [2], Jiang [3] and Nti [1], which suggested that machine learning techniques frequently outperform conventional statistical methods in financial forecasting applications.

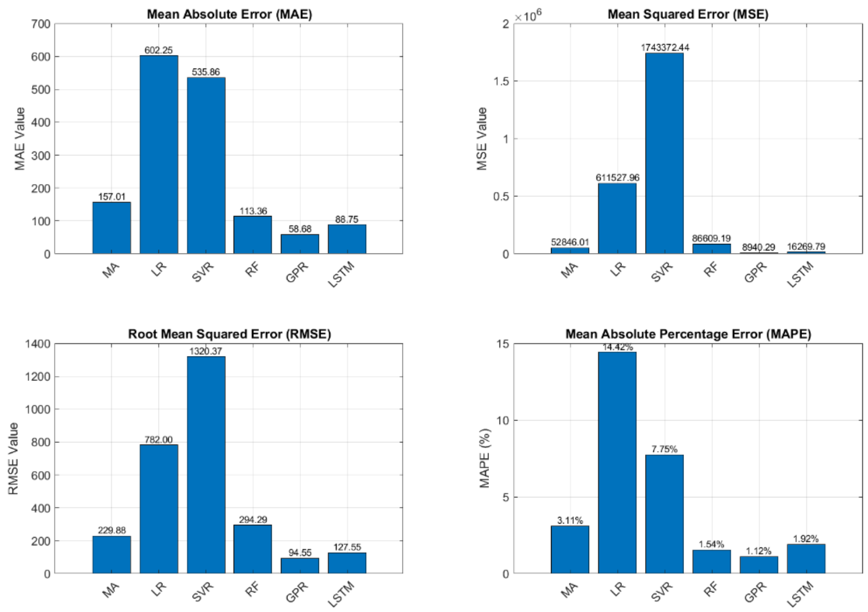

Figure 1. Prediction Error Comparison for BBCA Historical Analysis

Among the evaluated forecasting methods, Gaussian Process Regression (GPR) achieved the best predictive performance by obtaining the lowest error values across most evaluation metrics, particularly MAE (58.68), RMSE (94.55), and MAPE (1.12%). Random Forest achieved the second-best performance with a MAPE value of 1.54%, followed by Long Short-Term Memory (LSTM) at 1.92%.

In contrast, Linear Regression and Support Vector Regression generated significantly larger prediction errors. Linear Regression produced the highest MAPE value of 14.42%, indicating poor capability in modeling complex stock market behavior. The relatively weak performance of Linear Regression can be attributed to its assumption of linear relationships between predictor variables and stock prices, whereas stock market data typically exhibit non-linear, dynamic, and volatile characteristics.

The superior performance of Gaussian Process Regression may be attributed to its probabilistic non-parametric learning capability, which enables flexible modeling of non-linear relationships within stock market data [6]. Additionally, GPR effectively captures local variations and uncertainty characteristics commonly present in financial time-series data. Previous studies also demonstrated that probabilistic regression approaches can achieve strong predictive performance when applied to noisy financial datasets.

Furthermore, incorporating technical indicators such as Moving Average and Relative Strength Index (RSI) contributed positively to prediction performance by enriching feature representation with trend and momentum information. Feature engineering has been widely recognized as an important factor for improving forecasting quality in financial time-series applications [19].

3.4 LSTM Training Behavior

The LSTM learning curve shown in Figure 2 demonstrates a stable learning process during model training. Initially, both loss and RMSE values rapidly decreased, indicating that the network successfully adapted to learning stock market patterns during the early training stage.

As the training process progressed, both RMSE and loss values gradually stabilized and converged toward small values without significant oscillation. Minor fluctuations remained visible throughout the training iterations; however, no severe divergence was observed. This behavior indicates that the model maintained stable convergence characteristics and learned temporal dependencies embedded within historical stock price sequences effectively.

The training process reached the maximum epoch configuration with a relatively smooth convergence trend, suggesting that the selected learning parameters contributed positively to training stability. Similar findings have been reported in previous LSTM-based financial forecasting studies, where appropriate optimization strategies and network configurations improved learning stability and forecasting performance [13], [16].

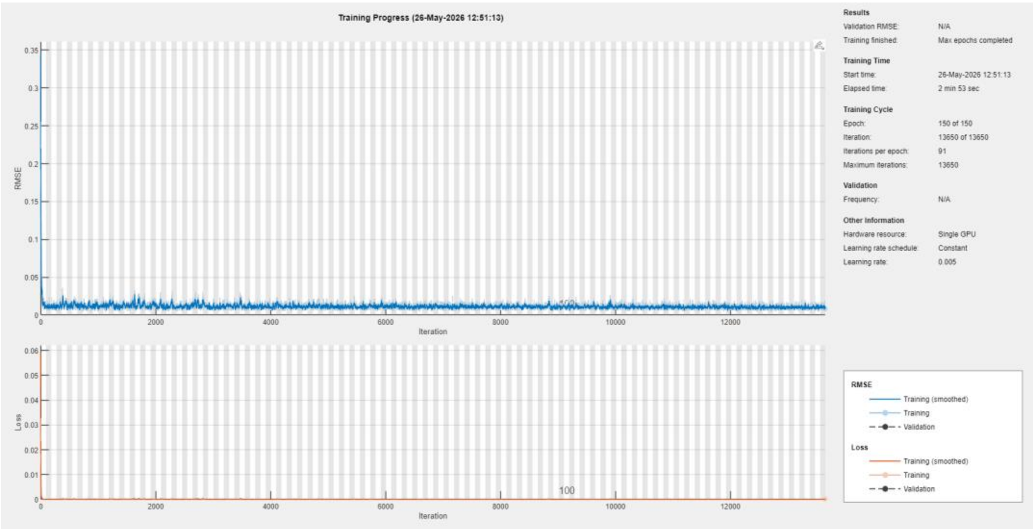

Figure 2. LSTM Learning Curve

3.5 Historical Prediction Visualization

Figure 3 illustrates historical prediction results generated by each forecasting model. The visual analysis supports the quantitative findings obtained from the error metrics discussed previously.

The historical prediction results reveal distinct forecasting behaviors among the evaluated methods. The Moving Average model generated relatively smooth prediction trajectories and generally followed the long-term movement of stock prices. However, prediction outputs tended to lag behind rapid price fluctuations due to the averaging mechanism used by the method.

Linear Regression primarily captured long-term trends but showed limitations in following short-term price dynamics. The prediction line exhibited a relatively linear pattern and gradually deviated from actual stock price movements, particularly during periods of increased volatility. This limitation is likely caused by the assumption of linear relationships between predictor variables and stock prices.

Support Vector Regression showed relatively unstable behavior in later periods of the historical data. Significant deviations between predicted and actual values were observed, especially during highly volatile market conditions. This instability may indicate sensitivity to parameter configuration and challenges in representing complex market patterns within the selected feature space.

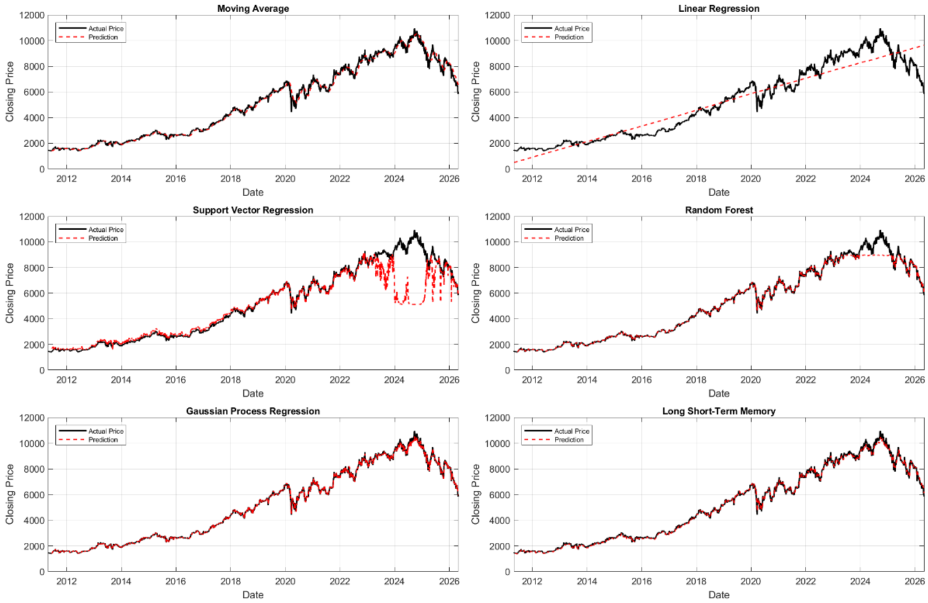

Figure 3. Historical Stock Price Prediction Results Using Six Forecasting Methods

Random Forest, Gaussian Process Regression, and Long Short-Term Memory produced predictions that closely followed actual stock price trajectories. Among these approaches, Gaussian Process Regression demonstrated the closest overlap with historical price movements, visually confirming its superior quantitative performance previously reported via MAE, RMSE, and MAPE.

LSTM also exhibited strong prediction capability by successfully learning temporal dependencies and sequential information contained within stock price movements. However, slight deviations remained visible during periods with abrupt market changes.

Overall, the visual results shown in Figure 3 support the quantitative findings that learning-based approaches generally outperform traditional statistical methods for stock price forecasting.

3.6 Future Stock Price Forecasting

In addition to historical prediction evaluation, this study also conducted 30-trading-day future forecasting. The future forecasting results are presented in Table 3 and Figure 4. The forecasting process was performed using an iterative multi-step strategy, where predicted values from previous steps were recursively used as input for subsequent prediction periods.

Table 3. Selected Future BBCA Stock Price Forecast Results for 30 Trading Days

Trading Day	Date	MA	LR	LSTM	SVR	RF	GPR
Day 1	2026-05-01	6506.70	9623.60	6015.40	5824.20	5904.40	5887.10
Day 5	2026-05-07	6461.80	9633.50	5996.50	5234.50	5823.20	6024.50
Day 10	2026-05-14	6422.20	9645.90	6034.80	5105.10	5778.00	6077.20
Day 15	2026-05-21	6397.30	9658.30	6249.90	5039.00	5721.50	6090.10
Day 20	2026-05-28	6359.60	9670.60	6431.90	5019.40	5561.30	6156.50
Day 25	2026-06-04	6339.00	9683.00	6527.50	4947.40	5468.80	6180.90
Day 30	2026-06-11	6345.70	9689.00	6572.90	4932.00	5409.30	6233.00

The future forecasting results show different prediction behaviors among the evaluated models. Linear Regression generated a strong upward projection, with predicted values approaching approximately IDR 9,700. This result indicates a tendency toward overestimation caused by long-term linear extrapolation.

Support Vector Regression produced a downward prediction pattern, indicating instability when applied to recursive future forecasting. Random Forest also showed a moderate downward trend, while Moving Average generated relatively stable forecasts due to its smoothing mechanism.

Gaussian Process Regression generated stable future predictions around the latest historical price level, whereas LSTM produced a moderate upward trend over the 30-trading-day forecasting period. These results indicate that GPR and LSTM produced more reasonable future forecasting trajectories compared with Linear Regression and SVR. However, future stock price projections should be interpreted as decision-support information rather than deterministic outcomes because stock prices are affected by external factors that are not fully represented by historical price data.

These findings suggest that future stock price forecasting remains highly dependent on model characteristics and underlying assumptions. Different learning mechanisms may produce substantially different future projections even when trained using identical historical data.

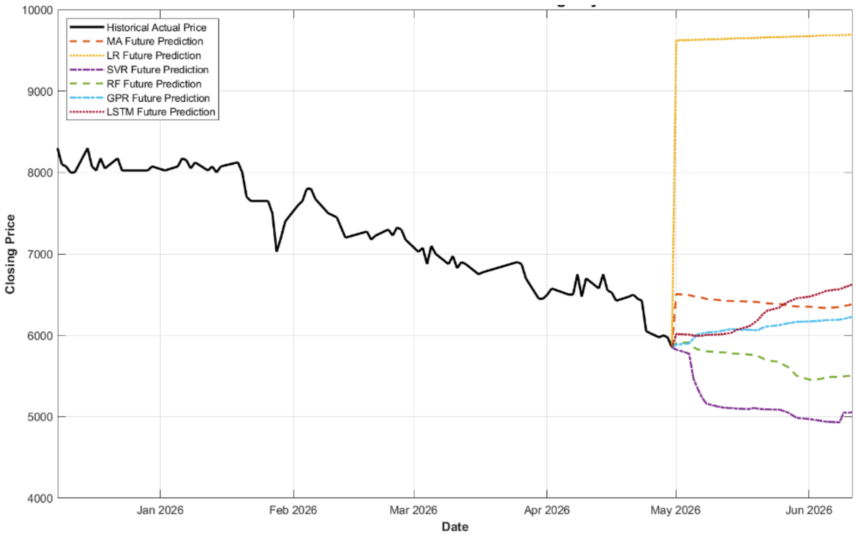

Figure 4. BBCA Stock Price Projection for the Next 30 Trading Days

Therefore, model selection should not rely solely on prediction accuracy but should also consider interpretability, generalization capability, and robustness under uncertain market conditions. Moreover, stock forecasting results should be interpreted as decision-support information rather than deterministic future outcomes because stock prices remain affected by external economic conditions, market sentiment, geopolitical events, and other factors that cannot be fully represented by historical data alone.

4. CLOSING

This study conducted a comparative analysis of six BBCA stock price forecasting methods, namely Moving Average (MA), Linear Regression (LR), Support Vector Regression (SVR), Random Forest (RF), Gaussian Process Regression (GPR), and Long Short-Term Memory (LSTM). The experiment used historical daily BBCA stock price data from May 2011 to April 2026. Model performance was evaluated using four error metrics, namely Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE), supported by historical prediction visualization and 30-trading-day future forecasting analysis.

The experimental results show that learning-based models generally achieved better forecasting performance than conventional statistical approaches. Among the evaluated methods, Gaussian Process Regression achieved the best predictive performance, with MAE of 58.68, MSE of 8,940.29, RMSE of 94.55, and MAPE of 1.12%. Random Forest also produced competitive results with MAPE of 1.54%, followed by LSTM with MAPE of 1.92%. These findings indicate that non-linear learning-based models are more suitable for capturing dynamic and non-linear patterns in BBCA stock price movements than purely linear or conventional statistical models.

The historical prediction visualization further supports the quantitative evaluation results. GPR, Random Forest, and LSTM produced prediction patterns that were closer to the actual stock price movement, while Linear Regression and Support Vector Regression showed larger deviations in several periods. Moving Average was able to capture the general trend direction but tended to lag behind rapid price changes due to its smoothing mechanism.

The use of lagged price values, Moving Average, and Relative Strength Index provided additional feature representation related to temporal dependency, trend behavior, and price momentum. However, the findings should be interpreted within the scope of the experimental setting because this study did not separately evaluate the individual contribution of each technical indicator through feature ablation analysis. Therefore, the results demonstrate the effectiveness of the complete feature-based forecasting framework rather than the isolated effect of each feature.

Despite the promising results, this study has several limitations. The forecasting process primarily relied on historical price data and technical indicators, while external variables such as market sentiment, macroeconomic indicators, company fundamentals, trading volume-based indicators, and financial news events were not incorporated into the model. As a result, several market behaviors influenced by external factors may not be fully captured by the forecasting models.

Future studies are recommended to integrate additional data sources, apply advanced hyperparameter optimization techniques, evaluate feature contribution through ablation testing, and explore hybrid forecasting architectures. Furthermore, the implementation of Explainable Artificial Intelligence (XAI) methods may improve model interpretability, while real-time trading simulation and multi-stock evaluation across different sectors may enhance the generalizability and practical relevance of the forecasting framework.

REFERENCES

[1] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A systematic review of fundamental and technical analysis of stock market predictions," *Artif. Intell. Rev.*, vol. 53, pp. 3007–3057, 2020.

[2] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial time series forecasting with deep learning: A systematic literature review (2005–2019)," *Appl. Soft Comput.*, vol. 90, p. 106181, 2020.

[3] W. Jiang, "Applications of deep learning in stock market prediction: Recent progress," *Expert Syst. Appl.*, vol. 184, p. 115537, 2021.

[4] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 259–268, 2015.

[5] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 654–669, 2018.

[6] S. Gu, B. Kelly, and D. Xiu, "Empirical asset pricing via machine learning," *Review of Financial Studies*, vol. 33, no. 5, pp. 2223–2273, 2020.

[7] B. Lim, S. Arik, N. Loeff, and T. Pfister, "Temporal Fusion Transformers for interpretable multi-horizon time series forecasting," *Int. J. Forecast.*, vol. 37, no. 4, pp. 1748–1764, 2021.

[8] M. Li, X. Huang, and S. Zhu, "A hybrid deep learning model for stock price prediction based on CNN-BiLSTM and attention mechanism," *Expert Syst. Appl.*, vol. 216, p. 119330, 2023.

[9] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Advances in Neural Information Processing Systems*, 2021, pp. 22419–22430.

[10] H. Zhou *et al.*, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2021, pp. 11106–11115.

[11] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[12] K. Althelaya, E.-S. El-Alfy, and S. Mohammed, "Evaluation of bidirectional LSTM for short-term and long-term stock market prediction," in *2018 International Conference on Information and Communication Systems*, 2018, pp. 151–156.

[13] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," in *2019 IEEE International Conference on Big Data*, 2019, pp. 3285–3292.

[14] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[15] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[16] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long short-term memory," *PLoS One*, vol. 12, no. 7, p. e0180944, 2017.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] M. Wen, P. Li, L. Zhang, and Y. Chen, "Stock market trend prediction using high-order information of time series," *IEEE Access*, vol. 7, pp. 28299–28308, 2019.

[19] J. Liu, Y. Li, and Y. Chen, "Stock price prediction based on BiLSTM and technical indicators," *IEEE Access*, vol. 10, pp. 10329–10338, 2022.

[20] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *2017 International Conference on Advances in Computing, Communications and Informatics*, 2017, pp. 1643–1647.