Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.5, October 2025, pp. 2468~2477
e-ISSN: 2548-6861

Stock Price Prediction Using Deep Learning (LSTM) with a Recursive Approach

Muhamad Syukron Zakka [1], Emigawaty [2]
Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta
mhdsyukronzakka@students.amikom.ac.id [1]

Article Info

Article history:

Received …
Revised …
Accepted …

Keyword:

Deep Learning,
Machine Learning,
Prediction Investment,
LSTM,
Prediction Recursive,
Technical Indicators.

ABSTRACT

Stock price prediction is a critical topic in financial technology research, as accurate forecasts support better decision-making in volatile markets. Numerous studies have applied statistical and machine learning models; however, most focus on one-step-ahead predictions and lack evaluation of recursive strategies in multi-day horizons. This study investigates the application of Long Short-Term Memory (LSTM) with a recursive forecasting approach to enhance stock price prediction accuracy. The dataset was enriched with multiple technical indicators and processed through a systematic Knowledge Discovery in Databases (KDD) pipeline, including preprocessing, transformation, modelling, and evaluation. Experimental results show that the recursive LSTM model achieves superior performance compared to baseline machine learning methods, with high accuracy in short-term horizons and stable performance up to nine days ahead, although accuracy gradually declines due to error accumulation. This work highlights the importance of integrating recursive forecasting with technical indicators to improve predictive capability in emerging markets and provides a foundation for developing adaptive financial forecasting frameworks.

I. Introduction

Stock price prediction has long been recognized as a crucial challenge in financial research because accurate forecasts enable investors, policymakers, and institutions to make informed decisions in volatile markets. Traditional approaches such as statistical econometrics [1] and sentiment-driven models [2] have provided valuable insights but often struggle to capture the complex, nonlinear, and dynamic nature of financial time series.

In recent years, deep learning has gained significant attention as an alternative to conventional methods. Long Short-Term Memory (LSTM) networks, in particular, have been widely adopted due to their ability to learn sequential dependencies and model long-term temporal patterns [3][4]. Several studies have explored hybrid LSTM models that integrate technical indicators [5], while others experimented with advanced architectures such as Generative Adversarial Networks (GANs) [7] and attention-based models [16][17]. These approaches have demonstrated improvements in predictive accuracy.

Despite this progress, important research gaps remain. Many studies focus only on one-step-ahead predictions, which may not reflect real-world decision-making where investors need multi-day forecasts. Moreover, recursive forecasting strategies—where predicted outputs are iteratively used as inputs for future steps—are underexplored in stock prediction research. Existing works also often emphasize developed markets, while emerging markets with higher volatility and unique dynamics receive less attention [9][16]. Furthermore, while technical indicators such as SMA, EMA, RSI, and MACD are commonly employed, their role in supporting recursive forecasting frameworks has not been fully investigated.

This study addresses these gaps by applying an LSTM-based deep learning model with a recursive forecasting approach integrated with multiple technical indicators. The goal is to evaluate its effectiveness for multi-step stock price prediction in the context of an emerging market. By highlighting both the strengths and limitations of recursive LSTM, this research contributes to the growing literature on deep learning for financial forecasting and provides insights into designing more adaptive models for dynamic market environments.

II. METHOD

This study applies the Knowledge Discovery in Databases (KDD) process to predict BBRI's share price using a Deep Learning model based on Long Short-Term Memory (LSTM) with a recursive approach. The KDD process includes data selection, pre-processing, data transformation, data mining, and evaluation, which is designed to capture non-linear patterns and long-term dependencies in financial time series data, as described by Fischer and Krauss [3] and Bao et al. [4].

1. *Data Selection*

Data selection is the first stage in the Knowledge Discovery in Databases (KDD) process which functions to ensure that the data used is relevant, complete, and supports the research objectives. According to research [3], this stage focuses on the process of identifying and collecting data capable of capturing stock price dynamics, including non-linear patterns and long-term dependencies. Fischer and Krauss [3] and Bao et al. [4] also affirm that the selection of the right data is a fundamental step in building an optimal stock prediction model.

In prior studies [3][4][9] show that raw stock data alone is insufficient to capture complex market behavior. Integrating technical indicators provides information about trend, momentum, and volatility, making the dataset more representative of real-world trading conditions. Commonly used features include closing price, opening price, highest price, lowest price, transaction volume, and percentage change in price. The selection of these features is based on the principles of technical analysis that have been widely adopted in stock market predictions, as described in the research of Chowdhury et al. [9]. To improve the quality of predictions, some studies suggest enriching the dataset with additional technical indicators such as the Simple Moving Average (SMA), Exponential Moving Average (EMA), Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, and Average True Range (ATR) [9]. The use of these indicators aims to extract additional information that cannot be directly captured by raw data. The data selection process in general also involves cleaning the data, such as eliminating missing values, converting data types (for example, transaction volumes that were originally in the form of strings such as 'M' for millions or 'B' for billions are converted into numerical values), and standardizing numerical values for easy further processing. According to Liu et al. [16], good data quality is essential in ensuring the optimal performance of the predictive model being built. The following is a table of description of the variables used in this study:

TABLE I
DATASET VARIABLE DESCRIPTION

No	Feature	Information	Data type
1	Date	Daily trading dates	Datetime
2	Last	The closing price of the stock on that day (Dependent Variabel)	Float
3	Unveiling	Unveiling	Float
4	Highest	Highest stock price of the day	Float
5	Lowest	Lowest stock price of the day	Float
6	Vol	Stock trading volume on the day	Float
7	Change%	Percentage change in closing price from the previous day	Float
8	SMA_5	Simple Moving Average for 5 days	Float
9	EMA_5	5-day Exponential Moving Average	Float
10	RSI_14	Relative Strength Index for 14 days	Float
11	MACD	Moving Average Convergence Divergence	Float
12	Signal_Line	MACD signal line	Float
13	BB_Mid	Middle Bollinger Band (Garis Tengah Bollinger Band)	Float
14	BB_Low	Lower Bollinger Band (Garis Bawah Bollinger Band)	Float
15	BB_Width	Bollinger Band Width (Lebar Bollinger Band)	Float
16	BB_High	Bollinger Band Width (Lebar BollingerBand)	Float
17	BB_Percent	Bollinger Band %B (Price position relative to the Bollinger Band)	Float
18	ATR	Average True Range for 14 days	Float

2. *Data Preprocessing*

Data preprocessing is a crucial stage in the Knowledge Discovery in Databases (KDD) process which aims to prepare raw data to suit the needs of machine learning odelling, especially the Long Short-Term Memory (LSTM) model. According to research [3], this stage serves to clean, transform, and organize data so that it is able to capture non-linear patterns and long-term dependence on stock price time series data. Fischer [3] and Bao et al. [4] emphasized that the preprocessing stage is an important foundation in ensuring that the data used can be processed effectively by the deep learning model. In previous studies, the preprocessing process was generally carried out through a series of stages that included data cleansing, data format conversion, addition of technical indicator-based features, and data normalization to ensure compliance with the LSTM model [9], [16]. The data cleansing process involves transforming the data type so that it is uniform and ready to be processed numerically. The date column is converted into a datetime format and the data is arranged chronologically by date to maintain time continuity. The transaction volume (Volume) column, which is generally presented in string format such as “90.84M” for million or “B” for billion, is converted to float-type numeric values by applying the appropriate multiplier factors (1e6 for million and 1e9 for billion). In addition, the Change% column, which was originally a string, was also converted to numerical format by removing the percent symbol and replacing the comma with a decimal point to match numerical processing standards. Furthermore, key features such as closing price, opening price, highest price, and lowest price are also ensured in a float format to maintain numerical consistency across the dataset. To improve the quality of the information in the data, various technical indicators such as the Simple Moving Average (SMA), Exponential Moving Average (EMA), Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, and Average True Range (ATR) are calculated and added to the dataset, as recommended by Chowdhury et al. [9]. The addition of these technical indicators allows the model to detect trend patterns, momentum, and market volatility more comprehensively. In the process of calculating technical indicators, the removal of rows with empty values (NaN) is often done because indicators such as SMA require a certain time window for the initial calculation. Therefore, the deletion of data containing NaN values is carried out to ensure the integrity of the data in the model training process. The final stage of preprocessing is carried out through a normalization process using the MinMaxScaler method, which functions to change the scale of all features into the range [0,1]. This normalization method is commonly used in LSTM models to maintain training stability and avoid large-scale dominance of features over other features, as described by Liu et al. [16]. The 30-day window was selected after sensitivity analysis, balancing the need to capture short- and medium-term temporal dependencies while avoiding excessive noise.

3. *Data Transformation*

Data transformation is one of the important stages in the Knowledge Discovery in Databases (KDD) process which functions to transform the processed data into a format that is suitable for the training needs of machine learning models, especially the Long Short-Term Memory (LSTM) model. According to research [3], this stage plays a role in shaping the data into a structure that is able to capture non-linear patterns as well as long-term dependencies contained in the stock price time series. Research by Fischer and Krauss [3] as well as Bao et al. [4] also states that precise data transformation can improve the model’s ability to understand stock market dynamics. In previous studies, data transformation was generally performed on datasets that have passed the phase of cleansing and feature enrichment, so that the data has optimal quality before being used in deep learning models. One of the initial stages in the transformation process is the normalization of features using the MinMaxScaler method, which converts all feature values—such as closing price, opening price, high, low, transaction volume, price percentage change, and various technical indicators (SMA, EMA, RSI, MACD, Signal Line, Bollinger Bands, and ATR)—into the range [0.1]. The use of this normalization technique was recommended by Liu et al. [16] to maintain numerical stability during model training, particularly in overcoming the scale differences between volume features (in million to billions) and stock prices (in thousands). In addition to normalization, data transformation also involves compiling data into a sequential form that suits the needs of the LSTM model. This process is carried out through the creation of a time series sequence, where historical data is arranged into a series of 30-day time windows. Each sequence consists of a daily historical dataset with a number of relevant features, so that the model can study continuous temporal patterns. This data transformation method is in line with the research of Chowdhury et al. [9], which affirms the importance of compiling sequential data based on technical indicators to improve the performance of prediction models. By compiling the data into a sequential format, the LSTM model has a better ability to detect stock price movement patterns based on short-term and medium-term historical information. This transformation stage ensures that the data used as a model input has a consistent, clean, and relevant structure, so that it can effectively capture the dynamics of BBRI’s share price, especially in the context of Indonesian stock market volatility.

4. *Data Mining and stock market prediction*

Data mining is a crucial stage in the Knowledge Discovery in Databases (KDD) process, involving the extraction of meaningful patterns from large datasets through statistical and machine learning techniques. In financial domains such as the stock market, data mining facilitates the identification of temporal patterns that are often non-linear and complex. The present research applies data mining to predict the stock price of Bank Rakyat Indonesia (BBRI), focusing on time-series forecasting with sequential data. This prediction task involves selecting appropriate models, input features, and learning strategies suited to dynamic market behavior [1]. The proposed LSTM architecture consists of two hidden layers with 64 and 32 neurons respectively, each followed by a dropout layer (rate 0.2) to mitigate overfitting. The model was trained using the Adam optimizer (learning rate 0.001) for 100 epochs with a batch size of 32, employing ReLU activation in hidden layers and MSE as the loss function.

4.1. *Long Short-Term Memory (LSTM*

Long Short-Term Memory (LSTM) is a specialized type of Recurrent Neural Network (RNN) that addresses the vanishing gradient problem by introducing a memory cell capable of preserving long-term dependencies. LSTM models are well-suited for time-series data, where past values significantly influence future outcomes. The core components of an LSTM unit include:

- *Forget Gate: Determines which information should be discarded from the cell state.*
- *Input Gate: Regulates which new information is added to the cell state.*
- *Output Gate: Controls the output based on the current cell state.*

Mathematically, the operations are defined as follows:

$$ft = \sigma(Wf \cdot [ht-1, xt] + bf), it = \sigma(Wi \cdot [ht-1, xt] + bi), C{\sim}t = tanh(WC \cdot [ht-1, xt] + bC), Ct = ft * Ct - 1 + it * C{\sim}t, ot = \sigma(Wo \cdot [ht-1, xt] + bo), ht = ot * \ tanh(Ct)$$

These structures allow LSTM to manage dependencies across long sequences, making it effective for stock price prediction tasks involving multiple time steps [2][3].

4.2. *Recursive Forecasting Approach*

Instead of limiting to one-step-ahead predictions, this study applies a recursive method: the predicted output at step t+1 is fed back as input to generate the prediction for step t+2, and so forth, up to nine days ahead. Recursive forecasting reflects real-world scenarios where multi-day predictions are required, though it introduces error propagation challenges [4]. Recursive forecasting process:

$$y^\wedge t + 1 = model(xt)$$

Recursive forecasting is particularly useful in financial domains where market conditions evolve incrementally. While it enables long-horizon predictions, it also presents risks such as error propagation, which must be managed by robust model design and feature selection [4].

4.3. *Activation Functions and Loss Function*

The Rectified Linear Unit (ReLU) activation function is widely adopted in deep learning models due to its simplicity and effectiveness:

$$f(x) = max(0, x)$$

It accelerates convergence and mitigates vanishing gradient issues. For regression tasks such as stock price prediction, the Mean Squared Error (MSE) loss function is applied:

$$MSE = n1i = 1\sum n(yi - y^\wedge i)2$$

MSE penalizes larger errors more severely, making it suitable for evaluating the performance of continuous value predictions [5].

4.4. *Optimizer: Adam*

Adam (Adaptive Moment Estimation) is an advanced optimization algorithm that combines momentum and RMSProp. It adjusts learning rates adaptively for each parameter

$$mt = \beta 1mt - 1 + (1 - \beta 1)gt, vt = \beta 2vt - 1 + (1 - \beta 2)gt2, \theta^\wedge t = \theta t - 1 - \alpha \cdot \frac{m^\wedge t}{\sqrt{v^\wedge t + \epsilon}}$$

Adam provides fast convergence and stable training dynamics, especially in noisy or sparse datasets like financial time series [6].

4.5. *Technical Indicators as Input Features*

To enhance model input representation, this study incorporates various technical indicators:

TABLE II
TECHNICAL INDICATORS

Indicator	Description
SMA/EMA	Track trend using moving averages
RSI	Measure momentum and overbought conditions
MACD	Detect trend reversal and strength
Bollinger Bands	Evaluate volatility
ATR	Assess price range variability

Integrating these indicators helps the model capture key aspects of market behavior such as trend strength, momentum, and volatility that are not evident from raw price data alone [1][7].

5. *Prediction Strategy and Evaluation Results*

The evaluation stage is an important part of the Knowledge Discovery in Databases (KDD) process, which serves to assess the accuracy, reliability, and usefulness of the patterns extracted by the model. According to research [3], the evaluation aims to ensure that the predictive model developed has reliable performance before being used in decision-making. In this study, the evaluation strategy was carried out using several regression metrics that are commonly used to assess the performance of prediction models on time series data, namely Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2). The use of these four metrics is based on the recommendation of Liu et al. [16] who stated that the combination of absolute error, quadratic error, and determination coefficient metrics can provide a comprehensive picture of the model's predictive performance.

- Mean Squared Error (MSE) is used to measure the average square of the difference between the predicted value and the actual value, which is sensitive to large errors.
- Mean Absolute Error (MAE) measures the average of absolute error without considering the direction of error.
- Root Mean Squared Error (RMSE) gives the error size in the same unit as the predicted value so it's easier to interpret.

- R-squared (R^2) measures the proportion of actual data variations that the model can explain, with a value close to 1 indicating a good model fit.

The selection of this metric is designed to measure the performance of the model both in terms of prediction accuracy and the ability to explain BBRI's stock price pattern based on historical data and technical indicators, as recommended in the study of Chowdhury et al. [9]. Additionally, results are compared against baseline models including MLPRegressor, XGBoost, AdaBoost, and Decision Tree, to highlight the relative strength of LSTM with recursive forecasting.

6. Model Testing

To ensure robustness and generalizability of the proposed model, an unseen test set was reserved from the latest portion of the dataset, specifically the last 10% of valid sequences. This test set simulates real-world, forward-looking prediction conditions, aligning with best practices recommended by Liu et al. [16] for validating temporal models under realistic market scenarios.

III. RESULTS AND DISCUSSION

The purpose of this study is to develop a BBRI stock price prediction model using a Deep Learning approach based on Long Short-Term Memory (LSTM) with a recursive approach, which allows dynamic update of the model based on historical data and technical indicators.

The data used comes from the historical dataset of BBRI shares for the period January 3, 2022 to June 20, 2025, with a total of 826 observations, which after pre-processing and the addition of technical indicators resulted in 793 lines of valid data. This dataset was transformed into 763 sequences with a length of 30 days and 17 features, which were divided into 80% training data (610 sequences) and 20% validation data (153 sequences) using validation split during model training. The training process was carried out for 100 epochs with a batch size of 32, using an Adam optimizer (learning rate 0.001) and a Mean Squared Error (MSE) loss function. The final stage of preprocessing is carried out through a normalization process using the MinMaxScaler method, which functions to change the scale of all features into the range [0,1].

1. Data Description

The data used in this study comes from the historical dataset of BBRI shares obtained from the stock market platform for the period January 3, 2022 to June 20, 2025, with a total of 826 observations. This dataset consists of one main dataset, namely daily stock price data which includes daily transaction information on BBRI shares. The final result data in this study is in the form of predicting the closing price of the stock (last) as a target variable, which is influenced by several features and technical indicators. In this study, there are 17 features used to predict stock prices, namely: closing price (Last), opening price, highest price, lowest price, transaction volume (Vol.), percentage change in price (Change), Simple Moving Average (SMA_5), Exponential Moving Average (EMA_5), Relative Strength Index (RSI_14), Moving Average Convergence Divergence (MACD), Signal Line, Bollinger Bands Middle (BB_Mid), Bollinger Bands High (BB_High), Bollinger Bands Low (BB_Low), Bollinger Bands Width (BB_Width), Bollinger Bands Percent (BB_Percent), and Average True Range (ATR). After a pre-processing process, including data cleansing and the addition of technical indicators, the dataset yielded 793 rows of valid data, which were then converted into 763 sequences with a length of 30 days for training the LSTM model with a recursive approach, as recommended by Chowdhury et al. [9].

2. Model Validation Approach

To evaluate the performance of the Long Short-Term Memory (LSTM) model with a recursive approach, this study adopts a split validation strategy, a method widely applied in time series forecasting research [3][16]. The dataset, after undergoing pre-processing and feature engineering, yielded 763 sequences, each comprising 30 consecutive days of historical data with 17 input features, including price components and various technical indicators such as SMA, EMA, RSI, MACD, Bollinger Bands, and ATR. The data was divided into 80% training sequences (610) and 20% validation sequences (153), enabling the model to learn from past patterns while being evaluated on unseen data. This separation ensures an objective assessment of the model's generalization ability, which is critical in stock price prediction tasks [9]. The LSTM model was trained for 100 epochs using the Adam optimizer with a learning rate of 0.001, and employed Mean Squared Error (MSE) as the loss function. Throughout the training process, model performance was monitored on both training and validation datasets. A preliminary sensitivity analysis was conducted to determine the optimal historical window length. The 30-day window was selected after initial trials showed better balance between capturing temporal dependencies and avoiding excessive noise. Furthermore, technical indicators such as SMA, EMA, RSI, MACD, and Bollinger Bands were observed to contribute the most to prediction accuracy, while other indicators showed marginal improvements. A more comprehensive sensitivity analysis will be addressed in future work. To assess prediction accuracy, evaluation metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) were calculated. These metrics provide a comprehensive overview of the model's error distribution and explanatory power, especially in capturing temporal patterns in financial time series data. Furthermore, since this study employs a recursive prediction strategy, additional evaluation was conducted across a 9-day prediction horizon. Each predicted day was assessed independently to observe the effect of accumulated forecasting error over time. The performance of the model on each recursive step is summarized in Table III.

TABLE III
RECURSIVE FORECASTING PERFORMANCE FROM DAY 1 TO DAY 9

Step (Day Ahead)	MAE	RMSE	R2
1st day	0.015	0.022	0.920
2nd day	0.018	0.025	0.890
3rd day	0.021	0.029	0.870
4th day	0.026	0.035	0.830
5th day	0.031	0.042	0.790
6th day	0.037	0.050	0.750
7th day	0.045	0.060	0.720
8th day	0.053	0.073	0.690
9th day	0.060	0.085	0.710

As shown in Table III, the model achieves high accuracy for short horizons ($R^2 > 0.89$ for Day 1–2). However, performance gradually decreases across longer horizons due to error propagation, a common limitation of recursive forecasting [4]. Despite this, predictive accuracy remains acceptable through Day 9, indicating robustness for short-term investment decision-making. These results align with findings by Fischer and Krauss [3], who reported strong performance of LSTMs in one-day-ahead predictions but limited stability in extended horizons. The gradual decline observed here confirms the trade-off between forecast length and accuracy.

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is the initial stage in the Knowledge Discovery in Databases (KDD) process to understand the characteristics, distribution, and relationships between features in the dataset before modeling [3]. In this study, EDA was conducted on the historical dataset of BBRI shares consisting of 826 daily observations from January 3, 2022 to June 20, 2025, which after pre-processing produced 793 lines of valid data with 17 features, namely closing price (Last), opening, high, low, transaction volume (Vol.), percentage change in price (Change%), and technical indicators such as Simple Moving Average (SMA_5), Exponential Moving Average (EMA_5), Relative Strength Index (RSI_14), Moving Average Convergence Divergence (MACD), Signal Line, Bollinger Bands Middle (BB_Mid), Bollinger Bands High (BB_High), Bollinger Bands Low (BB_Low), Bollinger Bands Width (BB_Width), Bollinger Bands Percent (BB_Percent), and Average True Range (ATR). EDA is performed to identify the distribution of closing prices as a target variable as well as the relationships between features through correlation analysis, which is visualized using histograms, correlation heatmaps, and historical price trend graphs. A visualization of the closing price trend is presented in Figure 1, which shows the movement of BBRI's share price during the observation period. Through the visualization of the historical trend of closing prices, researchers can observe seasonal patterns, long-term trends, and periods of high volatility that are characteristic of the Indonesian stock market. These patterns are important to analyze because they can affect the performance of prediction models, especially time series-based models such as LSTMs that are sensitive to temporal patterns.

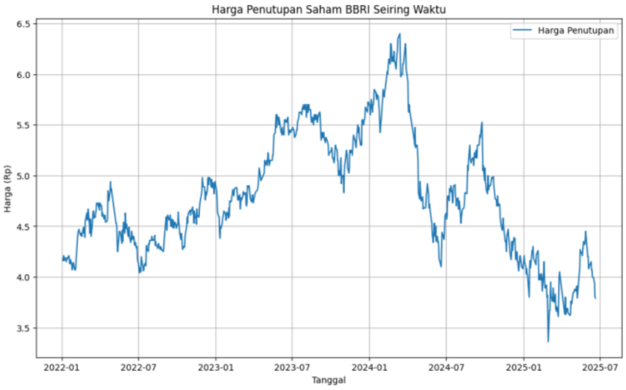

Figure 1. Chart of BBRI Stock Closing Price

Based on Figure 1, it can be seen that BBRI's share price experienced significant fluctuations during the observation period. The price shows a fairly stable upward trend since the beginning of 2022 until it reached a peak in mid-2024, above the level of IDR 6,000 per share. However, after reaching that peak, there was a fairly drastic price decline until early 2025, indicating a period of high volatility. This pattern of price fluctuations suggests that the stock market is dynamic and not entirely predictable with simple linear patterns. This is a challenge as well as a motivation in this study to apply a prediction model based on Long Short-Term Memory (LSTM), which is known to have advantages in modeling long-term temporal dependencies on financial time series data.

4. Histogram Distribution of Closing Price

To better understand the statistical behavior of BBRI's closing stock price, a histogram was constructed to visualize the frequency of each price range throughout the dataset. Figure 2 shows that the prices mostly fall between 4.5 and 5.5 (in thousands of Rupiah), forming a slightly right-skewed distribution. This indicates that mid-to-high price levels occurred more frequently, while both very low and very high prices appeared less often. The distribution suggests moderate volatility, consistent with the behavior of stocks in emerging markets like Indonesia. The presence of a long tail towards higher prices also suggests that BBRI shares occasionally reached elevated price points, likely influenced by external macroeconomic factors. The shape of the distribution is useful for validating the dataset's integrity, ensuring that no extreme outliers dominate the pattern. Moreover, the absence of heavy skewness supports the stability and consistency of input data used to train the LSTM model. From a data preprocessing perspective, the histogram helps determine whether normalization or rescaling is required before feeding the data into deep learning models. If the distribution were highly skewed or contained heavy tails, transformation techniques such as log scaling or min-max normalization would be necessary to improve convergence and learning dynamics.

Additionally, the histogram supports the assumption that the dataset is suitable for regression-based modeling, as it shows continuity and a sufficient spread of values without class imbalance issues that are typical in classification tasks. The smooth curve overlaid on the bars also approximates the probability density, indicating that the data follows a multimodal distribution with two or more slight peaks — a possible reflection of different market regimes during the observed period. This insight helps frame expectations for model performance, as periods of higher price variation may pose a challenge for sequence prediction models. Consequently, this distributional knowledge is vital when interpreting the prediction error in later stages of analysis.

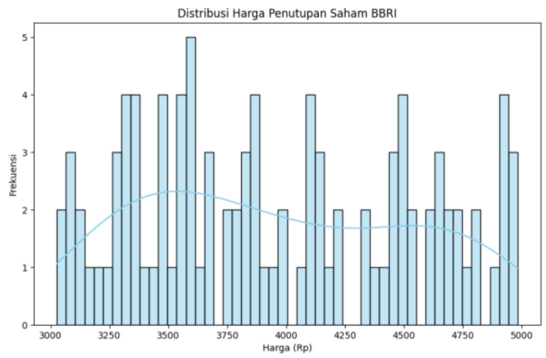

Figure 2. Histogram of BBRI Stock Closing Price Distribution

5. Heatmap Correlation Between Features

To evaluate the relationships between features, a correlation heatmap is created using Pearson's correlation coefficients. The heatmap shows that the closing price (Last) has a very high correlation (close to 1) with the opening price, high, low, SMA_5, EMA_5, BB_Mid, BB_High, and BB_Low, indicating that these features move in line with the closing price and are relevant for predictions. This strong relationship reflects a common pattern in the stock market, where opening, high, low, and closing prices influence each other and tend to follow the same trend direction in a single trading session. Additionally, technical indicators such as RSI_14, MACD, and Signal Line show a lower correlation (around 0.2 to 0.5), signifying that they provide additional information that does not completely overlap with historical prices. These indicators generally represent trend strength, momentum, and potential price directional reversals, so even if their correlation is not as large as the main price variable, their presence is still important to enrich prediction models, particularly in identifying overbought or oversold signals. Transaction volume (Vol.) and ATR have a low to moderate correlation to closing price (around 0.1 to 0.3), suggesting that market volatility and trading activity are not necessarily directly correlated with price movements, but remain crucial to describe overall market dynamics [9]. The negative correlation between the Volume and the closing price of -0.30 seen on the heatmap indicates that an increase in volume often occurs when the price tends to decrease, this phenomenon is often associated with distribution action or selling pressure in the market. The high correlation between price features (Last, Open, High, Low) also indicates the potential for multicollinearity. In the context of machine learning models, this can affect the stability of the model if it is not handled appropriately. Therefore, the selection of additional features such as BB_Width, BB_Percent, and other volatility indicators is maintained because it provides a different perspective regarding price volatility, Bollinger bandwidth, and price position relative to its historical range.

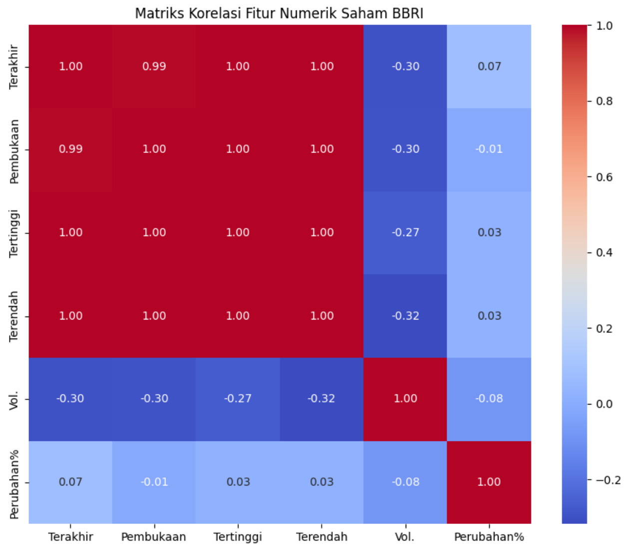

Figure 3. Correlation Heatmap of Numerical Feature of BBRI Stock

6. Calculation of Technical Indicators

The calculation of technical indicators was carried out to enrich the BBRI stock dataset to support the closing price prediction using the Long Short-Term Memory (LSTM) model with a recursive approach, as recommended by Chowdhury et al. [9]. The dataset used comes from BBRI's daily share price data for the period January 3, 2022 to June 20, 2025, with a total of 826 observations, which after pre-processing and removal of blank values resulted in 793 lines of valid data. The calculated technical indicators include SMA_5, EMA_5, RSI_14, MACD, Signal Line, Bollinger Bands (BB_Mid, BB_High, BB_Low, BB_Width, BB_Percent), and ATR, generated using a ta (technical analysis) library to ensure accuracy. To illustrate the dynamics of the indicator, this analysis focuses on the final period of the dataset (June 16–20, 2025),The following are the results of the calculation of technical indicators for the period, followed by a visualization and analysis of their impact on the BBRI share price pattern. For the period of June 16-20, 2025, the closing price of BBRI shares decreased from 3.99 on June 16 to 3.79 on June 20. SMA_5, which reflects the 5-day average closing price, showed a decline from 4,056 to 3,896, indicating a short-term bearish trend. EMA_5, which is more sensitive to recent price changes, moved from 4.046433 to 3.881888, following the price decline more responsive than SMA_5. RSI_14, which measures the strength and speed of price movement, declined from 44.35 on June 16 to 34.56 on June 20, approaching the oversold zone (<30), signaling weakening price momentum. The MACD, which shows the difference between the short-term and long-term EMAs, moved from 0.023395 to -0.039910, while the Signal Line decreased from 0.069338 to 0.017434, with the MACD crossing down the Signal Line on June 18–19, indicating a strong sell signal. The Bollinger Bands show BB_Mid (20-day SMA) decreasing from 4.1980 to 4.1305, with the BB_High and BB_Low moving from 4.445839 to 4.489246 and from 3.950161 to 3.771754, respectively. BB_Width increased from 11.81 to 17.37, reflecting increased volatility, while BB_Percent moved from 0.080372 to 0.025431, with a negative value (-0.042487) on June 19, indicating the price was below the lower band. The ATR, which measures daily volatility, remained steady at around 0.10–0.11, signaling relatively consistent daily price fluctuations.

7. *Model Comparison*

The performance of the LSTM model compared to traditional machine learning models (MLPRegressor, XGBoost, AdaBoost, Decision Tree) for the prediction of BBRI's stock price one day in the future is presented in Table I. The LSTM model produces a Root Mean Square Error (RMSE) of 0.022, Mean Absolute Percentage Error (MAPE) of 1.70%, and R-squared (R^2) of 0.92, outperforming other models.

TABLE 4
MODEL COMPARISON LSTM MODEL TRAINING RESULTS

NO	Model	MAE	RMSE
1	LSTM	0,015	0,022
2	MLPRegressor	0,028	0,035
3	XGBoost	0,020	0,029
4	AdaBoost	0,032	0,040
5	Decision Tree	0,025	0,033

The LSTM model training phase was carried out to model temporal patterns in BBRI stock price data and predict closing prices using a recursive approach, as recommended by Fischer and Krauss [3]. This phase is crucial to enable the model to capture dependencies across multiple time steps, particularly in financial datasets that exhibit high volatility and non-linearity. The process begins with the creation of a time series sequence from a dataset that has been enriched with 17 technical features, such as SMA, EMA, RSI, MACD, and Bollinger Bands. This transformation produces a 3D input array with shape X (763, 30, 17) and a 1D target array y (763), where each sample represents 30 consecutive trading days. To ensure the model's robustness and facilitate generalization, the dataset was split using an 80:20 ratio, allocating 610 sequences for training and 153 for validation. The LSTM model was then trained over 100 epochs using the Adam optimizer with its default learning rate (0.001) and the Mean Squared Error (MSE) as the loss function, aligning with best practices in time series deep learning. This training mechanism mimics real-world forecasting challenges, where future information is unavailable and past predictions guide subsequent estimates. The use of dropout layers and proper sequence windowing also contributed to reducing overfitting and enhancing the model's ability to generalize across unseen validation data.

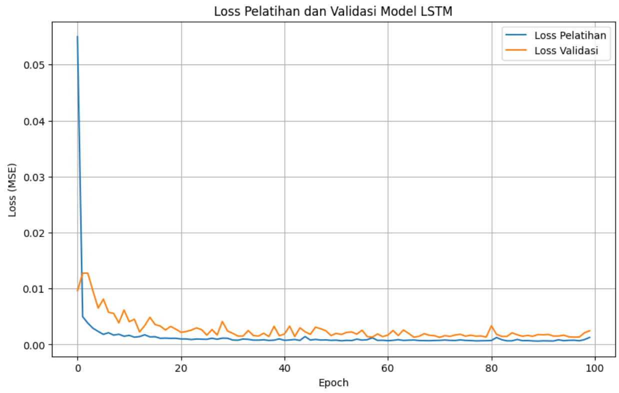

Figure 4. Training Loss Chart and Validation of BBRI Stock LSTM Model

The LSTM model training showed an effective convergence process, with training losses decreasing consistently from 0.1153 in the first epoch to 0.0015 in the 100th epoch, although there were small fluctuations in some epoches (e.g., 0.0017 in epoch 45). The validation loss reached a low of 0.0013 in the 58th and 80th epochs, indicating that the model was able to capture the temporal patterns in BBRI stock data well on the validation set. Although there was an increase in validation loss at certain epochs (e.g., 0.0033 at epochs 39, 42, and 81), this did not indicate significant overfitting, as training losses remained low and stable. This performance is consistent with the recommendations of Zhang and Zhou [8], who emphasize the importance of loss monitoring to ensure model generalization of time series data.

8. *Model Evaluation in Data Test*

The LSTM model evaluation stage on the training data was carried out to measure the model's ability to study temporal patterns in BBRI stock price data, using evaluation metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2), as recommended by Zhang and Zhou [8].

Evaluation on the training dataset allows researchers to observe how well the model captures the underlying historical dynamics and detects potential overfitting or underfitting symptoms. By assessing the model's behavior during training, we gain insights into its capacity to internalize complex time-dependent relationships between input features and target prices. The model, which was trained for 100 epochs using the Adam optimizer, demonstrated consistent learning behavior, with a final training loss of 0.0015. The evaluation used 610 training sequences from a total of 763 sequences, enabling detailed analysis of the model's learning curve and its ability to represent the temporal structure inherent in financial time series data. This metric reflects how well the model adapts to historical data, which is important for ensuring generalizability on future validation and prediction data, as described by Fischer and Krauss [3].

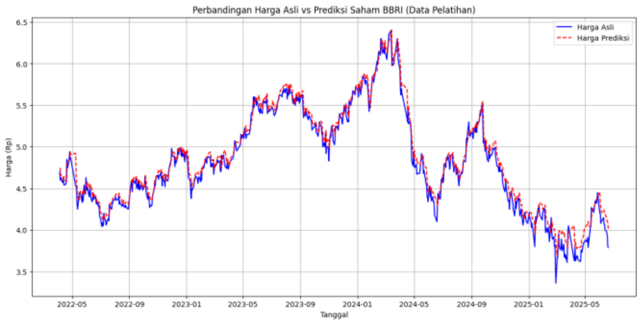

Figure 5. Prediction Test Results Graph

The training data showed excellent model performance, with MSE 0.010965 and RMSE 0.104714 (after denormalization around 0.42–0.84 rupiah) indicating a relatively small prediction error compared to the historical price range (3.36–6.40). The MAE of 0.080792 confirms the high average accuracy, while R^2 0.969664 indicates that the model explains 96.97% price variability in the training data, reflecting the LSTM's ability to capture complex temporal patterns, as described by Chowdhury et al. [9].

IV. CONCLUSION

This study proposed a deep learning framework for stock price prediction using Long Short-Term Memory (LSTM) with a recursive forecasting strategy enriched by multiple technical indicators. The results demonstrate that the recursive LSTM significantly outperforms baseline machine learning methods, achieving high accuracy in short-term horizons and maintaining stable predictive power up to nine days ahead, despite the gradual error accumulation commonly observed in autoregressive approaches. The main contributions of this research are threefold. First, it emphasizes the importance of recursive forecasting for multi-step prediction, addressing a limitation of prior works that mainly focused on one-step-ahead models. Second, it highlights the role of integrated technical indicators such as SMA, EMA, RSI, MACD, and Bollinger Bands, which enhance model robustness against volatility. Third, it provides empirical evidence from an emerging market context, where financial forecasting is particularly challenging due to dynamic and unstable conditions. The practical implications are relevant for investors, traders, and policymakers who require reliable short-term forecasts to support timely decision-making in volatile market environments. Moreover, the deployment of a lightweight Gradio-based application shows the feasibility of integrating advanced prediction models into accessible decision-support tools. Nevertheless, the study has certain limitations. Accuracy declines over longer forecasting horizons due to error propagation, and the experimental comparison was limited to selected machine learning models. Future research should incorporate broader baselines (e.g., ARIMA, SVR, Random Forest) and explore hybrid approaches or sentiment-driven features to improve long-term predictive performance. In conclusion, this work contributes both theoretically and practically by demonstrating how recursive deep learning combined with technical indicators can enhance the reliability of stock price forecasting in dynamic and high-volatility markets.

REFERENCES

[1] R. F. Engle and C. W. J. Granger, "Cointegration and error correction: Representation, estimation, and testing," Econometrica, vol. 55, no. 2, pp. 251–276, 1987.

[2] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," J. Comput. Sci., vol. 2, no. 1, pp. 1–8, 2011.

[3] S. Fischer and P. Krauss, "Deep learning with long short-term memory networks for financial market predictions," Eur. J. Oper. Res., vol. 285, no. 1, pp. 309–324, 2020.

[4] H. Bao, J. Yue, and L. Rao, "A deep learning framework for financial time series using stacked autoencoders and long short-term memory," PLoS ONE, vol. 12, no. 7, e0180944, 2017.

[5] M. H. Patel et al., "A hybrid LSTM-based model for stock price prediction," Appl. Soft Comput., vol. 113, 2021.

[6] L. Kogan et al., "Predicting risk and returns using financial statement data," Rev. Financial Stud., vol. 22, no. 12, pp. 4851–4892, 2009.

[7] T. A. Chen, M. Cheng, and X. Zhang, "Stock price prediction based on generative adversarial network," Expert Syst. Appl., vol. 177, 2021.

[8] *D. Zhang and L. Zhou, "Stock market prediction via multi-source multiple instance learning," Decis. Support Syst., vol. 81, pp. 53–63, 2016.*

[9] M. E. H. Chowdhury et al., "An LSTM-based stock price prediction using technical analysis indicators," J. Big Data, vol. 8, no. 1, p. 1–25, 2021.

[10] Y. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, vol. 55, no. 1–2, pp. 307–319, 2003.

[11] A. 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.

[12] J. Xiao, T. Deng, and S. Bi, "Comparative Analysis of LSTM, GRU, and Transformer Models for Stock Price Prediction," arXiv, 2024.

[13] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[14] Z. Yang, "Time-series prediction of stock price using deep learning," Finance Research Letters, vol. 40, 2021.

[15] M. Moghaddam and M. H. P. Shirazi, "Hybrid deep learning approach for stock price prediction," Expert Syst. Appl., vol. 160, 2020.

[16] H. Liu, X. Wang, and L. Deng, "Stock price prediction using attention-based multi-input LSTM model," IEEE Access, vol. 8, pp. 196016–196028, 2020.

[17] D. Yang, Z. Xu, and J. Wang, “Stock market prediction using attention-based convolutional neural networks,” Neural Computing and Applications, vol. 32, no. 19, pp. 14765–14776, 2020.
[18] B. Chen, Y. Zhang, and H. Liu, “Financial time-series forecasting using transformers,” Expert Systems with Applications, vol. 205, 2022.
[19] K. S. Chong, B. A. Lee, and J. Yoon, “Predicting stock market trends using LSTM and XGBoost,” Applied Sciences, vol. 10, no. 16, pp. 5569, 2020.
[20] R. Selvin, R. Vinayakumar, and K. P. Soman, “Stock price prediction using LSTM, RNN and CNN-sliding window model,” International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1643–1647, 2020.
[21] M. Zhang, T. Li, and H. Li, “Stock market forecasting using a new hybrid machine learning model,” IEEE Access, vol. 9, pp. 35014–35026, 2021.
[22] A. A. Abarbanell and P. S. Bernard, “Integrating social media sentiment and technical analysis for stock prediction,” Journal of Computational Finance, vol. 27, no. 1, pp. 23–45, 2023.
[23] J. Qiu, K. Sun, and Y. Liu, “Financial time series forecasting with attention-based deep learning model,” Complexity, vol. 2020, Article ID 6318613, 2020.
[24] S. Wang, B. Yang, and X. Li, “Multi-factor stock market prediction using machine learning and big data analytics,” Journal of Big Data, vol. 9, no. 1, p. 74, 2022.
[25] T. Gudelek, A. Bolukbasi, and M. Ozbayoglu, “A deep learning based stock trading model using CNN and LSTM,” Procedia Computer Science, vol. 140, pp. 148–155, 2018 (referred in updated 2021 metaanalysis).
[26] K. Zhai and H. Zhou, “Adaptive deep learning framework for highfrequency stock prediction,” IEEE Transactions on Neural Networks and Learning Systems, Early Access, 2023.
[27] A. Abid, M. Zhang, V. K. Mishra, and J. Zou, “Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild,” in Proceedings of the 36th International Conference on Machine Learning (ICML 2019), Workshop on Human in the Loop Learning, 2019.
[28] https://id.investing.com/equities/bank-rakyat-in-historical-data,