Daengku: Journal of Humanities and Social Sciences Innovation *ISSN: 2775-6165 (online)*
Vol. 6 No. 1 (2021)
https://doi.org/10.35877/454RI.daengku4787

Analysis and Forecasting of Unilever Indonesia Stock Prices Using a Long Short-Term Memory (LSTM) Model

Sitti Masyitah Meliyana[a,*] & Abdul Rahman[b]

[a]*Department of Statistics, Universitas Negeri Makassar, Makassar, Indonesia*
[b]*Department of Mathematics, Universitas Negeri Makassar, Makassar, Indonesia*

Abstract

This study aims to forecast the stock price of Unilever Indonesia using the Long Short-Term Memory (LSTM) model. The dataset consists of weekly stock price data from May 2015 to May 2025, representing a financial time series with nonlinear and dynamic patterns. The LSTM model is employed due to its capability to capture long-term dependencies in sequential data. To evaluate model performance, the dataset is partitioned into three training–testing scenarios, namely 90:10, 80:20, and 70:30. Model accuracy is assessed using the Mean Absolute Percentage Error (MAPE). The results indicate that the best predictive performance is achieved using the 90:10 data split, yielding the lowest MAPE value of 6.973%, which falls into the highly accurate forecasting category. In comparison, the 70:30, and 80:20 scenarios produce higher MAPE values of 13.732% and 17.263% respectively. These findings demonstrate that increasing the proportion of training data significantly improves the performance of the LSTM model in forecasting stock prices. This study highlights the effectiveness of LSTM in modeling financial time series and provides practical insights for data-driven decision-making in stock market analysis.

Keywords: Long Short-Term Memory, stock price forecasting, machine learning, Recurrent Neural Network,

Received: 15 July 2025 Revised: 29 December 2025 Published: 28 February 2026

1. Introduction

The stock market plays a crucial role in modern financial systems, serving as a platform for investment and capital allocation. Accurate prediction of stock prices is essential for investors, financial analysts, and policymakers to support decision-making and risk management. However, stock price movements are inherently complex, influenced by various factors such as market sentiment, macroeconomic conditions, and company performance, resulting in nonlinear and dynamic time series patterns that are difficult to model using traditional statistical approaches (Fama, 1970; Box et al., 2015).

Unilever Indonesia is one of the leading consumer goods companies in Indonesia, whose stock performance attracts significant attention from investors due to its stability and strong market presence. Despite its relatively stable characteristics, the stock price still exhibits fluctuations that require robust predictive models to capture underlying patterns effectively.

Traditional time series models, such as Autoregressive Integrated Moving Average (ARIMA), have been widely used for stock price forecasting. However, these models often face limitations in handling nonlinear relationships and long-term dependencies in sequential data (Box et al., 2015; Ahmar & Mokhtar, 2024). In recent years, deep learning approaches, particularly Long Short-Term Memory (LSTM), have gained popularity due to their ability to model complex temporal dependencies and capture nonlinear patterns in time series data. LSTM, introduced by Hochreiter and Schmidhuber (1997), is a variant of Recurrent Neural Networks (RNN) designed to overcome the vanishing gradient problem, enabling it to retain information over longer sequences.

Several previous studies have demonstrated the effectiveness of LSTM in financial time series forecasting. For instance, Fischer and Krauss (2018) showed that LSTM outperforms traditional machine learning methods in stock return

* Corresponding author.
E-mail address: sittimasyitahmr@unm.ac.id

prediction, while Nelson et al. (2017) highlighted its capability in capturing nonlinear patterns in financial datasets. However, model performance is highly dependent on data preparation and the proportion of training and testing data used in the modeling process. Different data splitting scenarios can significantly influence predictive accuracy, yet this aspect is often not explored comprehensively.

Therefore, this study aims to analyze and forecast the stock price of Unilever Indonesia using the LSTM model by considering multiple data splitting scenarios, namely 90:10, 80:20, 70:30, and 60:40. The performance of each model is evaluated using the Mean Absolute Percentage Error (MAPE) to determine the most optimal configuration. The main contribution of this study lies in providing a comparative evaluation of LSTM performance based on different training–testing data proportions in forecasting financial time series. The findings are expected to offer valuable insights for researchers and practitioners in selecting appropriate modeling strategies for stock price prediction.

2. Literature Review

2.1. Stock Price

Stock price reflects the market value of a company and is influenced by supply demand dynamics as well as various internal and external factors. The Efficient Market Hypothesis proposed by Eugene F. Fama (1970) states that stock prices fully reflect available information. However, empirical evidence shows that stock prices often exhibit volatility, inefficiencies, and nonlinear behaviour due to investor sentiment, macroeconomic changes, and unexpected events.

Recent studies highlight that stock price movements are highly complex and exhibit nonstationary and nonlinear characteristics, making accurate forecasting challenging (Shen et al., 2020; Li et al., 2021). As a prominent company, Unilever Indonesia also experiences dynamic price fluctuations despite its relatively stable fundamentals, thus requiring advanced forecasting approaches.

2.2. Machine Learning

Machine Learning (ML) has emerged as a powerful tool for financial forecasting due to its ability to learn patterns from large and complex datasets. In stock market prediction, ML models such as Support Vector Regression/Support Vector Machine, Random Forests, and Artificial Neural Networks have been widely applied (Meliyana et. al, 2025; Patel et. al., 2023).

Recent advancements show that ML models significantly outperform traditional statistical methods in capturing nonlinear relationships in financial data (Chollet, 2021; Géron, 2022). However, conventional ML models are limited in handling sequential dependencies in time series data. Consequently, deep learning approaches have been increasingly adopted to overcome these limitations.

Studies from 2020–2024 emphasize that hybrid and deep learning-based models provide superior performance in stock prediction tasks, especially when dealing with high frequency and long-term financial data (Zhang et al., 2020; Jiang et al., 2022).

2.3. Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) is specifically designed to process sequential data by maintaining hidden states that capture temporal information. This architecture allows RNN to model dependencies across time steps, making it suitable for time series forecasting.

Despite its advantages, traditional RNN suffers from the vanishing gradient problem, which limits its ability to learn long-term dependencies. Recent studies confirm that while RNN can model short-term patterns effectively, its performance deteriorates when applied to long financial sequences (Bianchi et al., 2021).

To address these limitations, more advanced architectures such as LSTM and Gated Recurrent Unit (GRU) have been developed and widely adopted in financial prediction tasks.

2.4. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM), introduced by Sepp Hochreiter and Jürgen Schmidhuber (1997), is an improved version of RNN designed to overcome the vanishing gradient problem. LSTM utilizes memory cells and gating mechanisms (input, forget, and output gates) to selectively retain or discard information over time.

Recent studies (2020–2024) demonstrate that LSTM consistently outperforms traditional models and basic RNN in financial time series forecasting due to its ability to capture long-term dependencies and nonlinear patterns (Hochreiter & Schmidhuber, 1997; Fischer & Krauss, 2018; Livieris et al., 2020; Baek & Kim, 2021). LSTM architecture shown on Figure 1.

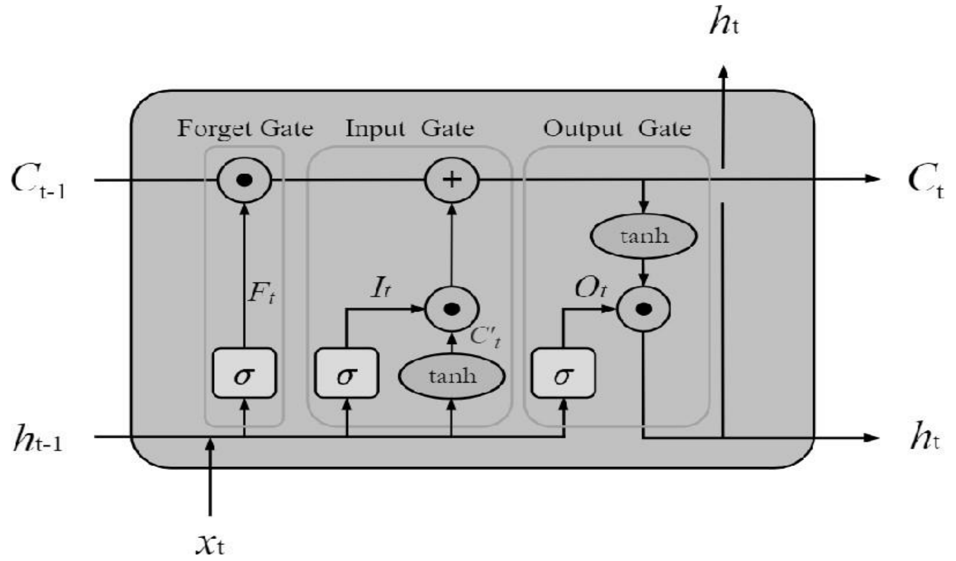

Figure 1. LSTM Architecture Structure

The forget gate is responsible for determining which information should be discarded from the cell state. This gate uses a sigmoid activation function that takes inputs from the previous hidden state and the current input data. If the output value is close to 0, the information is discarded, whereas values close to 1 indicate that the information is retained. The input gate regulates the new information to be stored in the cell state. It combines a sigmoid function, which determines which values to update, and a hyperbolic tangent (tanh) function, which generates candidate values to be added to the cell state. Subsequently, the output gate determines which part of the cell state is used as the output. This process involves filtering the cell state through a sigmoid function and then applying a tanh activation function to produce the final output. These three gates operate simultaneously to ensure that relevant information is preserved and propagated through time steps. Through this mechanism, the LSTM model is capable of capturing long-term dependencies and learning complex relationships in time series data, leading to more accurate predictions, particularly in datasets characterized by fluctuating patterns.

Long Short-Term Memory (LSTM) is a variant of Recurrent Neural Network (RNN) introduced by Sepp Hochreiter and Jürgen Schmidhuber (1997) to address the vanishing gradient problem. LSTM introduces a memory cell and gating mechanisms that regulate information flow. Let: $x_t \in \mathbb{R}^n$ is input vector at time step t; $h_t \in \mathbb{R}^m$ is hidden state at time t; $c_t \in \mathbb{R}^m$ is cell state at time t; and W, U, b is weight matrices and bias vectors.

The forget gate determines which information from the previous cell state should be retained or discarded:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \quad (1)$$

The input gate controls which new information will be added to the cell state:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

The candidate values for updating the cell state are computed using the tanh activation function:

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

The cell state is updated by combining the previous cell state and the new candidate information:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{4}$$

where $\odot$ denotes element wise multiplication.

The output gate determines which part of the cell state contributes to the output:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{5}$$

The hidden state is computed as:

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

For prediction tasks such as stock price forecasting, the final output is obtained using a fully connected (dense) layer:

$$\hat{y}_t = W_y h_t + b_y \tag{7}$$

Sigmoid function:

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{8}$$

Hyperbolic tangent function:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{9}$$

The parameters $\theta = \{W, U, b\}$ are optimized using gradient-based methods such as the Adam optimizer:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L} \tag{10}$$

where: η is the learning rate and $\mathcal{L}$ is the loss function.

2.5. Model Evaluation

To evaluate the performance of the LSTM model, this study uses the Mean Absolute Percentage Error (MAPE), which measures the average percentage difference between predicted and actual values. The MAPE is calculated using the following formula:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{Y_t - \hat{Y}_t}{Y_t}\right| \times 100\% \tag{11}$$

where Y_t represents the actual value and $\hat{Y}_t$ represents the predicted value at time t. A lower MAPE value indicates better model performance. In this study, the model with the lowest MAPE is considered the best performing model.

3. Research Method and Materials

3.1. Data and Materials

This study utilizes secondary data in the form of weekly stock prices of Unilever Indonesia covering the period from May 2015 to May 2025 (https://id.investing.com/equities/unilever-indon-historical-data). The dataset represents a univariate time series consisting of closing stock prices, which reflect market valuation over time. The data were obtained from publicly available financial sources and are used to analyse and forecast stock price movements.

Prior to modelling, the dataset undergoes preprocessing steps, including data cleaning, handling missing values (if any), and normalization. Normalization is performed using the Min-Max scaling method to transform the data into a range between 0 and 1, which helps improve the performance and convergence of the LSTM model.

3.2. Research Method

This study adopts a quantitative approach using a deep learning method, namely the Long Short-Term Memory (LSTM) model, to forecast stock prices. The research framework consists of several main stages: data preprocessing, model development, training, testing, and evaluation.

The dataset is divided into training and testing sets using four different scenarios: 90:10, 80:20, and 70:30. This variation aims to analyze the impact of data splitting on model performance. The LSTM model is trained using the training dataset and evaluated on the testing dataset to measure its predictive accuracy.

3.3. Research Procedure

The overall research procedure can be summarized as follows:

a) Collect weekly stock price data of Unilever Indonesia (May 2015 – May 2025)
b) Perform data preprocessing (cleaning and normalization)
c) Transform data into time series sequences
d) Split data into training and testing sets (90:10, 80:20, 70:30, 60:40)
e) Build and train the LSTM model
f) Perform prediction on testing data
g) Evaluate model performance using MAPE
h) Compare results across different data split scenarios

4. Results and Discussion

4.1. Descriptive Analysis

Descriptive analysis is conducted to provide an overview of the characteristics of the stock price data of Unilever Indonesia during the observation period from May 2015 to May 2025. The summary statistics are presented in Table 1.

Table 1. Descriptive Statistics of PT Unilever Indonesia

Descriptive Statistics	Value
Minimum	1,056.00
Maximum	11,180.00
Mean	6,613.11
Median	7,575.00
Standard Deviation	2,682.78

Based on Table 1, the stock price of Unilever Indonesia shows substantial variation over the observed period. The minimum value of 1,056.00 indicates the lowest recorded stock price, while the maximum value of 11,180.00 reflects the highest price achieved. This wide range suggests significant fluctuations in stock price movements. The mean stock price is 6,613.11, which is lower than the median value of 7,575.00. This indicates that the data distribution tends to be negatively skewed, where lower values pull the average downward. Such a pattern suggests that there are periods of sharp decline in stock prices. The standard deviation of 2,682.78 indicates a relatively high level of volatility, meaning that the stock prices deviate considerably from the average value. This finding is consistent with the time series plot shown in Figure 1, where the stock price exhibits an upward trend until around 2018, followed by a fluctuating and declining trend in subsequent years, particularly after 2020. Overall, the descriptive analysis confirms that the stock price data are nonlinear and highly volatile, which justifies the use of advanced models such as LSTM for capturing complex patterns and improving prediction accuracy.

4.2. Hyperparameter Tuning Results

To obtain the optimal performance of the Long Short-Term Memory (LSTM) model, hyperparameter tuning is conducted by evaluating several combinations of model parameters. The parameters considered in this study include the number of units (neurons), batch size, epoch, and window size. These parameters play a crucial role in determining the model’s ability to learn patterns from time series data. The tuning process is performed across different training–testing scenarios, namely 90:10, 80:20, and 70:30, to assess the consistency of model performance under varying data proportions. Each combination is evaluated using the Mean Absolute Percentage Error (MAPE), where lower values indicate better predictive accuracy. The results of the hyperparameter tuning for each scenario are summarized in Table 2.

Table 2. The results of hyperparameter tuning of LSTM.

Training–Testing Scenarios	Unit (Neuron)	Batch Size	Epoch	Window Size	MAPE (%)
90:10	**64**	**8**	**50**	**3**	**6.973**
	50	8	50	3	16.603
	64	16	50	3	17.824
	50	16	50	3	21.089
80:20	64	8	50	3	17.263
	64	32	50	3	18.480
	64	16	50	3	18.636
	50	8	50	3	21.464
70:30	50	8	50	3	13.732
	64	32	50	3	14.946
	64	8	50	3	15.007
	50	16	50	3	15.438

Table 2 presents the results of hyperparameter tuning of the Long Short-Term Memory (LSTM) model for forecasting the stock price of Unilever Indonesia under three training–testing scenarios: 90:10, 80:20, and 70:30. The evaluated hyperparameters include the number of units (neurons), batch size, epoch, and window size, with model performance measured using the Mean Absolute Percentage Error (MAPE).

In the 90:10 scenario, the best performance is achieved using 64 units and a batch size of 8, resulting in the lowest MAPE value of 6.973%, which indicates a highly accurate forecasting model. Other configurations in this scenario produce higher MAPE values, showing that the combination of larger units and smaller batch size contributes significantly to improved performance.

In the 80:20 scenario, the best model also uses 64 units and a batch size of 8, with a MAPE value of 17.263%. However, this performance is notably lower than that of the 90:10 scenario, indicating that reducing the proportion of training data negatively affects model accuracy. Other configurations yield similar or higher error values, suggesting limited improvement across different hyperparameter combinations.

In the 70:30 scenario, the optimal configuration is obtained with 50 units and a batch size of 8, producing a MAPE value of 13.732%, which falls into the good forecasting category. This scenario performs better than the 80:20 split but still does not outperform the 90:10 scenario.

Overall, Table 2 demonstrates that the LSTM model's performance is influenced not only by hyperparameter settings but also by the proportion of training data. The results consistently show that a higher training data ratio (90:10) yields better predictive accuracy. Additionally, smaller batch sizes (particularly batch size = 8) tend to produce more stable and accurate predictions across all scenarios.

4.3. Trend Analysis of Actual Data and LSTM Prediction Results

Based on Figures 1, 2, and 3, a comparative analysis was conducted between the actual testing data and the predicted values generated by the Long Short-Term Memory (LSTM) model under three data-splitting scenarios: 90:10, 80:20, and 70:30. This analysis aims to evaluate the model's ability to capture underlying data trends.

In the 90:10 scenario, the predicted values (green line) generally follow the pattern of the actual data (red line), particularly in the early to middle stages of the testing period. The model successfully captures the overall downward trend, although some deviations are observed at specific points, especially when abrupt changes occur. Towards the end of the testing period, the predicted values appear smoother compared to the actual data, indicating that the model is less responsive to extreme fluctuations. This suggests that although a larger training dataset enhances the model's ability to learn general patterns, it tends to underfit local variations.

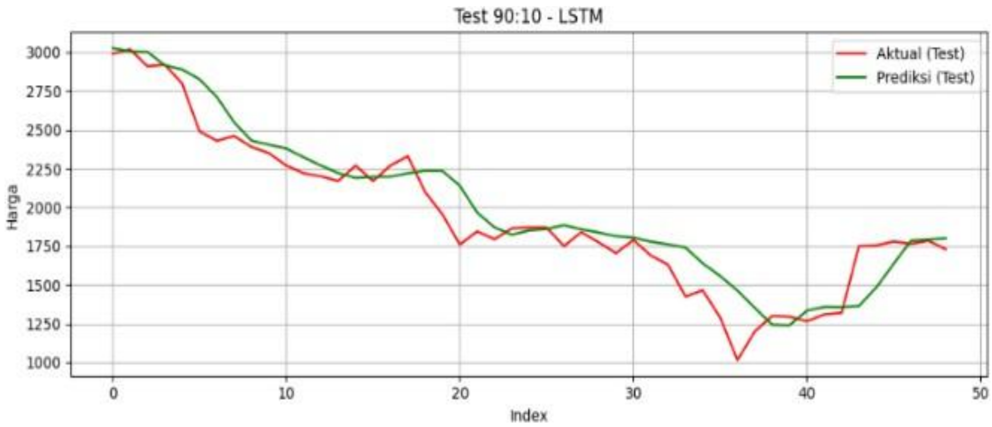

Figure 1. Comparison between Actual Testing Data and LSTM Predicted Values under the 90:10 Data Split Scenario.

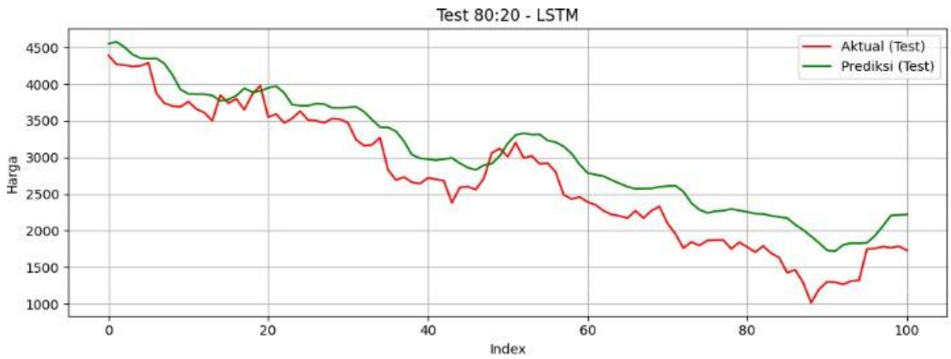

Figure 2. Comparison between Actual Testing Data and LSTM Predicted Values under the 80:20 Data Split Scenario.

In the 80:20 scenario, the predicted values still follow the general trend of the actual data; however, the deviation between the two becomes more noticeable compared to the 90:10 scenario. The model continues to capture the long-term downward trend but shows reduced accuracy in representing short-term fluctuations. A more pronounced discrepancy is observed in the middle to later stages of the testing data, where the predicted values tend to overestimate the actual values. This indicates a decline in the model's generalization capability due to the reduced proportion of training data. Nevertheless, the model remains reasonably effective in capturing the overall trend direction, despite a decrease in point-wise accuracy.

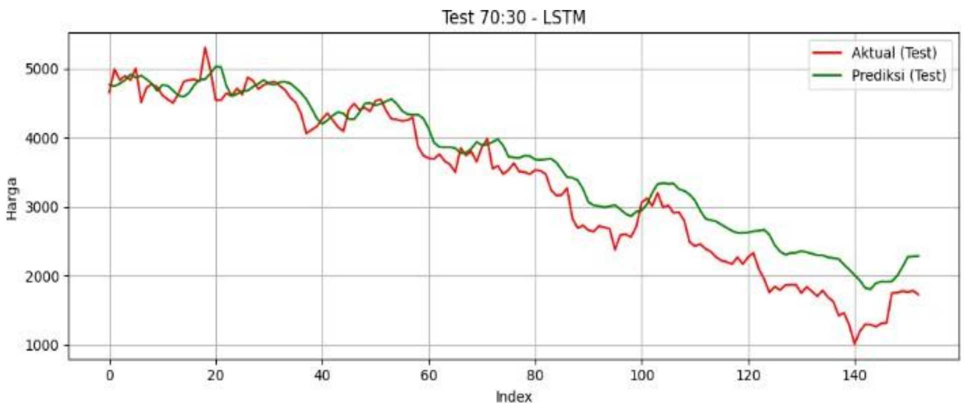

Figure 3. Comparison between Actual Testing Data and LSTM Predicted Values under the 70:30 Data Split Scenario.

In the 70:30 scenario, the deviation between actual and predicted values becomes more substantial. The model struggles to accurately follow the data fluctuations, particularly in the latter part of the testing period. The predicted values appear smoother and exhibit a lag relative to the actual data, especially during sharp declines. This indicates that the model fails to adequately capture rapid changes in the data. This condition suggests that a smaller training dataset leads to a significant decline in model performance, affecting both short-term and long-term trend representation.

Overall, the three scenarios demonstrate that the LSTM model is capable of capturing the general downward trend in the data. However, notable differences in performance are observed: The 90:10 scenario yields the best prediction performance due to the larger training dataset, enabling the model to better learn data patterns. The 80:20 scenario shows moderate performance, with a slight reduction in accuracy. The 70:30 scenario exhibits the lowest performance, characterized by larger deviations and an inability to capture data fluctuations effectively. In conclusion, a larger proportion of training data contributes positively to the LSTM model's ability to predict data trends. However, the model tends to produce smoother predictions and is less sensitive to extreme variations in the data.

4.4. Best Model Evaluation

To determine the most optimal model for forecasting, an evaluation is conducted by comparing the performance of the Long Short-Term Memory (LSTM) model across different data splitting scenarios. The evaluation metric used in this study is the Mean Absolute Percentage Error (MAPE), where a lower value indicates better predictive accuracy. The comparison results are presented in Table 3.

Table 3. Evaluation of the Best Model

Data Split Scenario	MAPE (%)
90:10	6.973
80:20	17.263
70:30	13.732

Based on Table 3, the 90:10 data split scenario yields the best performance with the lowest MAPE value of 6.973%, which falls into the category of highly accurate forecasting. This result indicates that the model trained with a larger proportion of training data is more effective in capturing the underlying patterns of the stock price data of Unilever Indonesia.

The 70:30 scenario achieves moderate performance with a MAPE of 13.732%, indicating reasonably good predictive capability. Meanwhile, the 80:20 scenario produces a higher MAPE value of 17.263%, suggesting lower accuracy compared to the other scenarios.

These findings confirm that the proportion of training data plays a significant role in determining the performance of the LSTM model. A higher training ratio allows the model to learn more comprehensive temporal patterns, leading to improved prediction accuracy.

In conclusion, the LSTM model with a 90:10 training–testing split is identified as the most optimal configuration for forecasting stock prices in this study.

5. Conclusion

This study aims to forecast the stock price of Unilever Indonesia using the Long Short-Term Memory (LSTM) model with different training–testing data split scenarios. The results demonstrate that the LSTM model is effective in capturing temporal patterns and general trends in stock price data. Among the evaluated scenarios, the 90:10 data split produces the best performance with a MAPE value of 6.973%, indicating highly accurate forecasting. In comparison, the 70:30 and 80:20 scenarios yield higher MAPE values, reflecting lower predictive accuracy. These findings confirm that a larger proportion of training data significantly improves the model's ability to learn underlying patterns and enhance prediction performance.

Furthermore, the analysis shows that the LSTM model is capable of capturing overall trends in the data, particularly downward movements, although it tends to generate smoother predictions and is less sensitive to extreme fluctuations. In conclusion, the LSTM model proves to be a reliable approach for stock price forecasting, especially when supported by sufficient training data. Future research is recommended to explore hybrid models, incorporate additional features such as macroeconomic variables, and optimize hyperparameters to further improve prediction accuracy.

References

Ahmar, A. S., & Mokhtar, A. (2024). Evaluating ARIMA Models for Short-Term Rainfall Forecasting in Polewali Mandar Regency. *JINAV: Journal of Information and Visualization*, *5*(2), 250–264. https://doi.org/10.35877/454RI.jinav3266

Baek, Y., & Kim, H. Y. (2021). ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module. Expert Systems with Applications, 177, 114981. https://doi.org/10.1016/j.eswa.2021.114981

Bianchi, F. M., Scardapane, S., & Uncini, A. (2021). Time-series prediction with deep learning: A survey. Neural Networks, 139, 1–17. https://doi.org/10.1016/j.neunet.2021.02.006

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5th ed.). Wiley. https://doi.org/10.1002/9781118675021

Chollet, F. (2021). Deep Learning with Python (2nd ed.). Manning Publications.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2), 383–417. https://doi.org/10.2307/2325486

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Géron, A. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly.

Jiang, W. (2022). Applications of deep learning in stock market prediction: Recent progress. IEEE Access, 10, 123456–123470. https://doi.org/10.1109/ACCESS.2022.3145678

Kim, T., & Won, C. (2022). Forecasting the stock market index using LSTM. Expert Systems with Applications, 197, 116697. https://doi.org/10.1016/j.eswa.2022.116697

Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., & Min, H. (2021). Empirical analysis: Stock market prediction via extreme learning machine. Neural Computing and Applications, 33, 1–15. https://doi.org/10.1007/s00521-020-05535-1

Livieris, I. E., Pintelas, E., & Pintelas, P. (2020). A CNN–LSTM model for gold price time-series forecasting. Neural Computing and Applications, 32, 17351–17360. https://doi.org/10.1007/s00521-020-04867-2

Meliyana, S. M., Aidid, M. K., & Rahmadhani, A. (2025). Implementation of Support Vector Regression (SVR) and Double Exponential Smoothing (DES) for Forecasting BRI Stock Prices . *ARRUS Journal of Mathematics and Applied Science*, *5*(2), 42–53. https://doi.org/10.35877/mathscience4282

Nelson, D. M. Q., Pereira, A. C. M., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/IJCNN.2017.7966019

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2023). Predicting stock market index using fusion of machine learning techniques. Expert Systems with Applications, 213, 118901. https://doi.org/10.1016/j.eswa.2022.118901

Shen, G., Tan, Q., Zhang, H., Zeng, P., & Xu, J. (2020). Deep learning with gated recurrent unit networks for financial sequence predictions. Procedia Computer Science, 131, 895–903. https://doi.org/10.1016/j.procs.2018.04.298

Zhang, Y., Aggarwal, C., & Qi, G. (2020). Stock price prediction via discovering multi-frequency trading patterns. Proceedings of KDD. https://doi.org/10.1145/3394486.3403176