ISSN: 3064-4372
DOI: 10.21512/ijcshaijournal.v3i1.14936

Optimizing CNN-LSTM Models for Stock Price Prediction in a Multi-Sector Holding Company

Muhamad Fajar Suprapto[1]*, Septi Andryana[2]

[1]Department of Information Systems,
[2]Department of Information Technology,
Universitas Nasional,
Jakarta, Indonesia 12520
muhamadfajarsuprapto2022@student.unas.ac.id;
septi.andryana@civitas.unas.ac.id

Abstract—**Accurate stock price forecasting is difficult because financial time-series data usually demonstrate nonlinear relationships, irregular fluctuations, and interdependent temporal patterns. This research investigates the predictive performance of three neural network models based on deep learning: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid CNN-LSTM architecture for forecasting stock prices of a multi-sector holding company. The dataset used in this study contains daily historical price observations collected from 2015 to 2025, where sequential samples are generated using a sliding window approach. To obtain appropriate model settings, hyperparameter optimization is carried out using a grid search procedure. Model performance is assessed using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Experiments are first performed using an 80:20 training-testing split and followed by a robustness evaluation using a 70:30 data split. Under the primary evaluation scheme, the experimental results indicate that the LSTM model yields the lowest prediction error, reflected by an RMSE value of 77.86, MAE of 58.23, and MAPE of 1.28%. Meanwhile, the hybrid CNN-LSTM model demonstrates more stable performance across different data proportions, achieving an RMSE of 75.71 and MAPE of 1.23% during the robustness test. The results indicate that LSTM is effective in capturing sequential dependencies inherent in financial time-series data, integrating convolutional feature extraction with sequential learning can improve prediction stability under varying training conditions. The results provide empirical insights into the selection of deep learning architectures for stock price prediction in the context of multi-sector holding companies.**

Keywords—***Stock Price Prediction; Convolutional Neural Network; Long Short-Term Memory; Hybrid CNN-LSTM; Financial Time-Series Forecasting***

I. INTRODUCTION

The capital market serves as an important indicator of economic activity and investment dynamics within a country. Stock prices are primarily formed through supply-demand interactions that reflect various economic and corporate conditions, including company performance, macroeconomic policy, and investor expectations. In recent years, stock markets worldwide have experienced increased volatility following the COVID-19 pandemic, rising global inflation, and geopolitical tensions. These conditions have intensified uncertainty in financial markets and have made the analysis of stock price movements increasingly complex [1], [2]. As a result, predicting stock prices remains a challenging task, particularly because financial time-series data often exhibit nonlinear relationships, non-stationary behavior, and high levels of noise [3].

The complexity of stock price dynamics becomes more pronounced in the case of multi-sector holding companies. Such firms operate across multiple industries, causing their financial performance to be influenced simultaneously by different sectoral trends. PT Astra International Tbk (ASII) represents one of the largest multi-sector holding companies in Indonesia, with business activities spanning automotive manufacturing, heavy equipment, financial services, agribusiness, infrastructure, and technology sectors. Due to this diversification, fluctuations in ASII's stock price are affected by multiple economic factors across industries. Consequently, the resulting price patterns are often irregular and difficult to capture using conventional statistical forecasting methods [4]. This characteristic makes ASII a relevant case study for evaluating predictive models based on historical financial financial time-series data from the Indonesian stock market.

Financial forecasting has traditionally relied on statistical approaches such as linear regression, ARIMA, and GARCH models. Despite their widespread use, these techniques frequently struggle to capture nonlinear dynamics and long-range dependencies inherent in stock market data. Early machine learning models also present limitations in modeling sequential relationships within time-series datasets [5], [6]. Recent advances in deep learning provide alternative approaches that are better suited to these characteristics. Convolutional Neural Networks (CNN) are capable of extracting short-term local patterns from sequential data, whereas Long Short-Term Memory (LSTM) networks are designed to model long-term temporal relationships in sequential data. Because of these complementary properties, recent studies have increasingly examined hybrid CNN-

Received: Dec. 31, 2025; received in revised form: Mar. 16, 2026; accepted: Mar. 18, 2026; available online: Mar. 29, 2026.

Corresponding: muhamadfajarsuprapto2022@student.unas.ac.id

LSTM architectures for financial time-series forecasting, as they combine local feature extraction with temporal dependency learning [7]-[11].

Several recent studies have demonstrated the effectiveness of deep learning architectures in improving stock price prediction performance. For instance, Seun et al. [12] evaluated several deep learning models and reported that hybrid CNN-LSTM architectures can achieve higher prediction accuracy compared with standalone CNN or LSTM models. Similarly, Zhao et al. [13] showed that hybrid CNN-LSTM models are capable of capturing both spatial and temporal information from stock price sequences, which leads to improved forecasting performance. In addition, Wu et al. [19] proposed a graph-based CNN-LSTM model that incorporates leading indicators to enhance prediction capability in financial markets. Zhang et al. [9] also demonstrated that CNN-LSTM architectures can effectively model complex nonlinear relationships in stock price data. These findings highlight the growing interest in hybrid deep learning models for financial time-series forecasting. Despite these developments, empirical studies that systematically compare CNN, LSTM, and hybrid CNN-LSTM architectures in the context of Indonesian multi-sector holding companies remain limited. Most existing studies primarily focus on prediction accuracy without thoroughly examining the stability and robustness of model performance under different experimental settings.

In addition, many previous studies primarily focus on prediction accuracy while paying less attention to the stability and robustness of model performance under different training conditions. Financial time-series models may produce different results depending on the proportion of the training-testing data split used in the experiment. However, systematic robustness evaluations using multiple data-splitting schemes are still relatively uncommon in the literature. This limitation indicates the need for further research that not only compares predictive accuracy across deep learning architectures but also evaluates the consistency of model performance when exposed to different data configurations.

Based on these considerations, this study evaluates and compares the performance of Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and hybrid CNN-LSTM models for predicting the stock price of PT Astra International Tbk using adjusted closing price data from 2015 to 2025. Hyperparameter optimization is performed using a grid search method to determine appropriate model configurations. Model evaluation is conducted using an 80%-20% training-testing split and further assessed through a robustness test with a 70%-30% split to examine prediction stability. This study contributes by systematically comparing CNN, LSTM, and hybrid CNN-LSTM architectures for stock price prediction in a multi-sector holding company listed in the Indonesian stock market. This study contributes by providing empirical insights into the impact of hyperparameter optimization and the predictive capability of hybrid deep learning architectures in financial time-series forecasting.

II. PROPOSED METHODS

A. *Research Workflow*

The overall research workflow employed in this study is illustrated in Figure 1. The study begins with the collection of historical daily stock price data of PT Astra International Tbk. The collected data are then processed through a preprocessing stage that includes data cleaning, normalization using the Min-Max scaling method, and sequence construction using a sliding window method to prepare the dataset for time-series forecasting models.

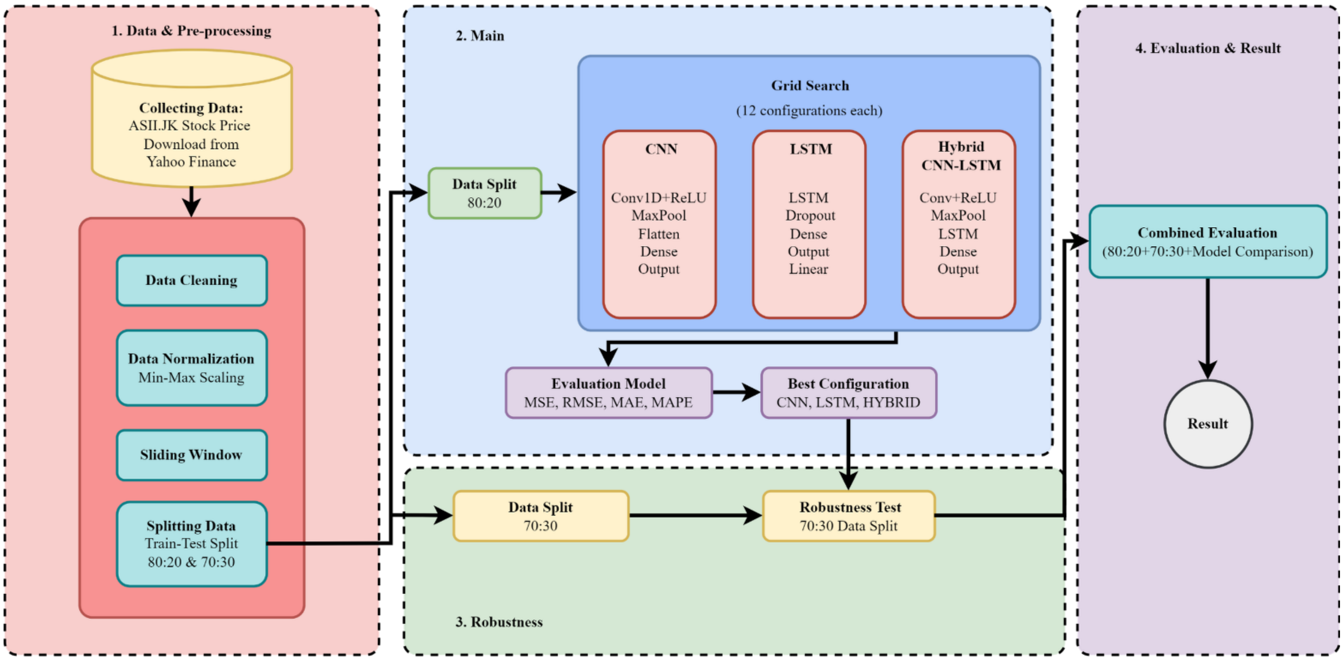

Figure 1. Research workflow consisting of dataset preparation, model training, hyperparameter tuning, and prediction performance assessment

As illustrated in Figure 1, the dataset is divided into training and testing subsets using an 80%-20% split for the primary experiment. The prepared sequential dataset is then used to train three deep learning architectures, namely CNN, LSTM, and a hybrid CNN-LSTM model. Hyperparameter tuning is carried out through a grid search approach to identify the most appropriate configuration for each model.

The models are evaluated based on four error metrics, namely MSE, RMSE, MAE, and MAPE. To further assess model stability, an additional robustness test is conducted using a 70%-30% data split. The final stage involves comparing prediction results across models to identify the architecture that provides the best forecasting performance.

B. Dataset and Data Source

This study utilizes historical stock price data of PT Astra International Tbk (ASII) as the case study object. ASII is selected because it operates as a multi-sector holding company with a diversified business portfolio spanning automotive, heavy equipment, financial services, agribusiness, infrastructure, and technology sectors. Such diversification causes the company's stock performance to be influenced by multiple sectoral dynamics, making it a relevant case study for evaluating prediction models designed to capture complex financial time-series patterns [4].

The dataset consists of daily stock price observations using the adjusted closing price variable. This variable incorporates corporate actions such as stock splits and dividend distributions, providing a more consistent representation of historical stock value compared with the regular closing price. The use of adjusted closing prices has been widely adopted in machine learning-based stock price prediction studies because it reduces bias caused by structural price adjustments [14].

The data used in this study cover the period from January 1, 2015 to September 30, 2025, consisting of 2,668 daily trading observations prior to sequence generation. The historical price data were obtained from Yahoo Finance using the ticker ASII.JK. The relatively long observation period allows the models to learn both short-term variations and long-term temporal dependencies inherent in nonlinear financial time-series data [15].

C. Data Preprocessing

The preprocessing stage is performed to prepare the historical stock price data so that they can be effectively used as inputs for deep learning models. Financial time-series data often contain missing values and varying scales, which may negatively affect the training process if not handled properly [8].

The preprocessing stage begins by sorting the dataset chronologically to preserve the sequential nature of time-series observations. Missing values are handled using a combination of forward filling and the removal of incomplete records through a dropna procedure. After this step, the number of usable observations is reduced from 2,668 to 2,647 records.

After the cleaning process, the data are normalized using the Min-Max scaling technique within the range of [0, 1] to standardize the data scale and prevent large values from dominating the training process. This normalization technique is widely used in neural network-based financial prediction models because it preserves the original distribution of the data while improving model convergence during training [9], [16]. The normalization process is defined as:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

Where x denotes the original stock price value, $x_{\max}$ and $x_{\min}$ represent the maximum and minimum values within the dataset, respectively. The variable x' indicates the normalized value after the scaling process.

The final step in the preprocessing stage is sequence construction using a sliding window approach with a window size of 60 trading days. In this procedure, each sequence consisting of 60 historical observations is used as input to estimate the stock price for the following trading day. This sliding window method is widely applied in financial time-series modeling because it enables the model to learn temporal relationships among consecutive observations [17].

D. Model Architecture

This study evaluates three deep learning architectures, namely Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid CNN-LSTM model, which have been widely applied in time-series forecasting and stock price prediction studies [15], [16]. All models receive sequential input data with a window length of 60 trading days and a single feature representing the adjusted closing price. The selection of these architectures aims to compare their capabilities in capturing short-term local patterns and long-term temporal dependencies in nonlinear and volatile stock price data.

Figure 2 presents the architectures of the CNN, LSTM, and hybrid CNN-LSTM models used in this study.

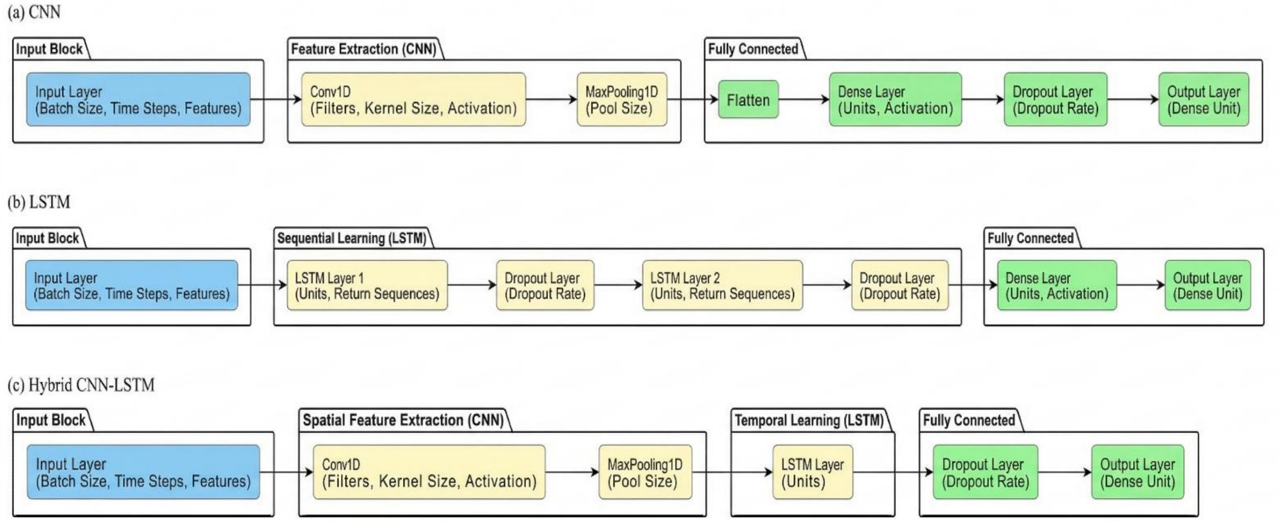

Figure 2. Architecture diagrams of the deep learning models used in this study: (a) CNN, (b) LSTM, and (c) CNN-LSTM hybrid model

As presented in Figure 2, the CNN model is used to learn local temporal characteristics from stock price sequences. The input sequence is first passed through a one-dimensional convolutional layer (Conv1D) that employs multiple filters to identify short-term patterns in the time-series data. The convolution layer uses the ReLU activation function to capture nonlinear relationships. The resulting feature maps are then reduced through a MaxPooling1D layer to emphasize the most dominant features while reducing dimensionality. The extracted features are subsequently flattened and passed to a dense fully connected layer, with a dropout layer subsequently introduced to minimize overfitting. Finally, a dense output layer produces the predicted stock price value. CNN-based architectures have been widely used for extracting local features from sequential data in financial time-series prediction tasks [15].

The LSTM architecture is designed to capture long-term temporal dependencies in sequential data. In this study, a stacked LSTM structure is employed, consisting of two LSTM layers that enable the model to learn hierarchical temporal representations from historical stock price sequences. To reduce overfitting during the training process, dropout layers are introduced after each LSTM layer. The extracted features are subsequently forwarded to a dense layer that produces the final prediction output. LSTM networks are particularly well suited for modeling sequential financial data because they can preserve relevant information across extended time intervals [8].

The hybrid CNN-LSTM architecture combines the capabilities of CNN and LSTM within a single modeling framework. In the first stage, a one-dimensional convolutional layer is applied to learn local features from the input sequence, followed by a MaxPooling layer that compresses the feature representation and reduces dimensionality. The resulting feature maps are then provided as input to an LSTM layer, which is responsible for learning long-term temporal relationships within the sequential data. To enhance model generalization, a dropout layer is introduced before the final prediction stage. The processed features are subsequently fed into a dense layer that produces the predicted stock price value. Through the integration of convolution-based feature extraction and sequential temporal modeling, the hybrid architecture can capture both short-term price variations and longer-term trends in stock price movements, which has been shown to improve forecasting performance in recent studies [10]-[12].

To ensure a fair comparison among the three architectures, all models are trained using the same input window length, training procedure, and evaluation metrics. This standardized experimental setup allows differences in prediction performance to be attributed primarily to the intrinsic characteristics of each model architecture rather than variations in the training configuration.

E. Hyperparameter Optimization

Hyperparameter optimization is conducted to determine the most suitable configurations for the CNN, LSTM, and hybrid CNN-LSTM models. The selection of appropriate hyperparameters is a crucial step in deep learning experiments because the choice of model parameters can significantly influence prediction accuracy and training stability. Proper hyperparameter tuning also ensures that the comparison among different model architectures is performed under consistent experimental conditions, allowing the observed performance differences to reflect the intrinsic capabilities of each model.

In this study, a grid search approach is applied to systematically examine different hyperparameter combinations within a predefined search space. This method is commonly adopted in machine learning and deep learning research because it offers a clear and reproducible procedure for determining suitable model configurations. Under this strategy, multiple combinations of candidate hyperparameters are exhaustively explored, and each configuration is trained using the training dataset. The predictive performance of each configuration is then evaluated using the predefined evaluation metrics to determine the most suitable parameter settings.

The hyperparameter search space is defined according to the architectural characteristics of each model. For the CNN model, the optimization focuses on the number of convolution filters and kernel sizes. These parameters influence the model’s ability to capture local temporal patterns from the time-series input. A larger number of filters can potentially capture more complex features, while different kernel sizes allow the model to detect patterns over varying temporal ranges.

For the LSTM model, the tuning process concentrates on the number of LSTM units and the dropout rate. The number of units determines the model’s capacity to learn temporal relationships from sequential data, whereas the dropout mechanism is applied to mitigate overfitting by randomly disabling a portion of neurons during the training phase. Proper adjustment of these parameters helps maintain a balance between model complexity and its ability to generalize to unseen data.

The hybrid CNN-LSTM architecture integrates the main parameters from both CNN and LSTM components. In this case, convolution filters and kernel sizes are used for local feature extraction, while the LSTM units are responsible for capturing long-term temporal dependencies. To maintain training stability and reduce computational complexity, the search space for the hybrid architecture is slightly constrained compared with the individual CNN and LSTM models.

Several training parameters are standardized across all models to ensure fair performance comparison. The training parameters considered in this study include batch size, learning rate, number of epochs, loss function, and optimizer. All models are trained using the Adam optimizer with a learning rate of 0.001, which is commonly used in deep learning because of its adaptive learning mechanism and stable convergence properties. In addition, Mean Squared Error (MSE) is adopted as the loss function since it is well suited for regression tasks such as stock price prediction.

To enhance training efficiency and reduce the risk of overfitting, an early stopping technique with a patience value of 10 epochs is implemented during the training process. This mechanism continuously monitors the validation performance and stops the training procedure automatically when no improvement is detected over several consecutive epochs. Early stopping helps reduce unnecessary training iterations and ensures that the model does not continue learning after reaching its optimal performance.

The hyperparameter search spaces for each model are summarized in Table 1, which presents the parameter ranges evaluated during the grid search procedure.

Table 1. Hyperparameter search space used for optimizing the CNN, LSTM, and hybrid CNN-LSTM models

Parameter	Model		
	CNN	**LSTM**	**Hybrid CNN-LSTM**
Filters	64, 128	-	64, 128
Kernel Size	3, 5, 7	-	3, 5, 7
Units (LSTM)	-	64, 128, 256	128, 256
Dropout Rate	-	0.2, 0.3	0.2
Batch Size	32	32, 64	32
Learning rate	0.001	0.001	0.001
Epochs	50	50	50
Early stopping	Patience = 10	Patience = 10	Patience = 10
Loss Function	MSE	MSE	MSE
Optimizer	Adam	Adam	Adam

The optimal hyperparameter configuration for each model is determined based on the lowest RMSE value obtained on the testing dataset under the primary evaluation scheme, which uses an 80% training set and 20% testing set. After the optimal configurations are obtained, the same hyperparameters are reused in the robustness evaluation using a 70%-30% training-testing split without conducting additional hyperparameter searches.

F. Experimental Setup and Evaluation Metrics

The experiments conducted in this study aim to assess the stock price prediction performance of three deep learning architectures: CNN, LSTM, and the hybrid CNN-LSTM model under consistent and reproducible evaluation conditions. All experiments utilize time-series data without shuffling in order to maintain the chronological sequence of observations.

The initial evaluation scheme adopts an 80%-20% training-testing data split. To further investigate model robustness, an additional experiment is conducted using a 70%-30% training-testing split without performing additional hyperparameter searches. This robustness evaluation is intended to examine the stability of model performance under different training data proportions.

Model performance is assessed using four widely adopted error metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). These metrics are chosen because they offer complementary perspectives for evaluating prediction accuracy, including sensitivity to large deviations and the average magnitude of prediction errors. The mathematical expressions of these metrics are presented as follows.

Mean Squared Error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{2}$$

Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \tag{4}$$

Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{5}$$

where y_i refers to the actual stock price, $\hat{y}_i$ represents the value predicted by the model, and n indicates the total number of observations in the testing dataset. The combined use of these four metrics enables a comprehensive assessment of prediction errors by capturing both the magnitude of the deviations and the relative percentage differences between the predicted and actual values.

III. EXPERIMENTAL RESULT

This section reports the evaluation results of the CNN, LSTM, and hybrid CNN-LSTM models on the testing dataset. The experiments are carried out under two evaluation settings: the primary scheme using an 80% training set and a 20% testing set, and a robustness evaluation using a 70% training set and a 30% testing set. Model performance is assessed using four error metrics, namely Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Employing multiple evaluation metrics enables a more comprehensive analysis of prediction performance from different perspectives.

A. Results of Main Experiment

The primary experiment employs an 80% training dataset and a 20% testing dataset. The predictive performance of the three deep learning architectures CNN, LSTM, and hybrid CNN-LSTM is assessed using the optimal hyperparameter settings obtained during the optimization stage. Model performance is evaluated on the testing dataset using four error metrics: RMSE, MSE, MAE, and MAPE.

Table 2 summarizes the performance comparison of the three models based on their respective optimal configurations. In general, all models are able to capture the overall movement patterns of the stock price series with relatively low prediction errors.

Table 2. Performance comparison of CNN, LSTM, and hybrid CNN-LSTM models under the 80%-20% train-test split

Model	Best Configuration	Evaluation Metrics			
		RMSE	**MSE**	**MAE**	**MAPE**
CNN	Filters = 128, Kernel Size = 7, Batch Size = 32, LR = 0.001	81.97	6,719.04	60.89	1.34%
LSTM	Units = 256, Dropout = 0.3, Batch Size = 32, LR = 0.001	77.86	6,062.89	58.23	1.28%
Hybrid	Filters = 128, Kernel Size = 7, LSTM Units = 256, Dropout = 0.2, Batch Size = 32, LR = 0.001	78.43	6,150.89	57.76	1.27%

Based on Table 2, the LSTM model achieves the lowest RMSE value (77.86) and MSE value (6,062.89), indicating the best overall predictive performance under the primary evaluation scheme. This outcome can be explained by the capability of LSTM networks to learn long-term temporal relationships in sequential data. Since stock price movements often contain temporal correlations and sequential structures, the memory mechanism in LSTM enables the model to preserve relevant historical information over extended periods, which contributes to better prediction accuracy.

The hybrid CNN-LSTM model demonstrates competitive performance, achieving an RMSE of 78.43 and an MSE of 6,150.89, which are close to the values obtained by the LSTM model. Although its RMSE is slightly higher, the hybrid architecture achieves the lowest MAE and MAPE values (57.76 and 1.27%, respectively). These results imply that the hybrid model achieves smaller average prediction errors relative to the actual values. The combination of convolution-based feature extraction and sequential learning allows the model to capture both short-term local fluctuations and long-term temporal dependencies, which contributes to stable prediction performance.

The CNN model records the highest error values among the three models, with an RMSE of 81.97 and an MSE of 6,719.04. This result is expected because CNN primarily focuses on extracting local patterns within short temporal windows and does not explicitly model long-term sequential dependencies. Consequently, while CNN can detect short-term price movements, it may struggle to capture longer temporal relationships present in financial time-series data.

Despite these differences, the relatively small gap among the error values indicates that all three models achieve comparable predictive capability under the primary evaluation scheme.

A visual comparison between the observed stock prices and the predicted values generated by the three models is presented in Figure 3.

Figure 3. Comparison between the actual stock prices and the predicted values generated by the CNN, LSTM, and hybrid CNN-LSTM models on the testing dataset

As shown in Figure 3, the predicted curves produced by all models closely follow the overall trend of the actual stock price movements during the testing period. The LSTM and hybrid CNN-LSTM models appear to capture trend reversals and upward price trends more accurately than the CNN model. This visual observation also corresponds with the quantitative evaluation results shown in Table 2.

B. Robustness Test Results

To further evaluate model stability, a robustness test is conducted by modifying the proportion of training and testing data. In this evaluation scenario, the dataset follows a 70%-30% training-testing split, while maintaining the same hyperparameter configurations obtained from the primary evaluation scheme. Hyperparameter tuning is not repeated in this stage to examine whether the model performance remains consistent under different training-testing data proportions.

Table 3 summarizes the performance results of the CNN, LSTM, and hybrid CNN-LSTM models under the robustness testing scheme.

Table 3. Performance comparison of CNN, LSTM, and hybrid CNN-LSTM models under different train-test split ratios

Model	**Split**	**Evaluation Metrics**			
		RMSE	**MSE**	**MAE**	**MAPE**
CNN	80:20	81.97	6,719.04	60.89	1.34%
	70:30	78.15	6,108.11	59.02	1.27%
LSTM	80:20	77.86	6,062.89	58.23	1.28%
	70:30	77.57	6,017.00	58.95	1.27%
Hybrid	80:20	78.43	6,150.89	57.76	1.27%
	70:30	75.71	5,732.11	56.99	1.23%

The findings in Table 3 suggest that the prediction performance of all models remains relatively consistent under different training-testing data splits. Under the 70%-30% split, the RMSE values for the CNN, LSTM, and hybrid CNN-LSTM models are 78.15, 77.57, and 75.71, respectively. Notably, the hybrid CNN-LSTM model achieves the lowest RMSE and MSE values in this robustness testing scheme.

This result indicates that the hybrid architecture generalizes more effectively across different data proportions. The convolutional component identifies local patterns in the data, whereas the LSTM component captures temporal dependencies over longer time horizons, allowing the model to maintain stable prediction accuracy even with reduced training data.

For the MAPE metric, all models maintain relatively low error levels, with values around 1.23%-1.27%, indicating that the prediction errors remain small relative to the actual stock price values. Overall, these findings demonstrate that the performance of the CNN, LSTM, and hybrid CNN-LSTM models remains consistent and robust across different data split configurations, whereas the hybrid CNN-LSTM model exhibits slightly higher stability with respect to overall prediction accuracy.

IV. CONCLUSION

This study investigates the effectiveness of three deep learning architectures, including CNN, LSTM, and hybrid CNN-LSTM, in predicting stock prices based on historical time-series data of PT Astra International Tbk. The experiments are conducted under two evaluation schemes: a primary evaluation using an 80%-20% training-testing data split and a robustness evaluation using a 70%-30% split, based on the optimal hyperparameter configurations obtained through the grid search optimization process. The experimental results demonstrate that all models are capable of producing stable predictions with relatively low error values across multiple evaluation metrics.

Under the primary evaluation scheme, the LSTM model achieves the lowest RMSE value (77.86), indicating strong capability in capturing long-term temporal dependencies in financial time-series data. Meanwhile, the hybrid CNN-LSTM architecture produces the lowest MAE and MAPE values among the evaluated models, suggesting improved accuracy in terms of average prediction deviation. In the robustness evaluation, the hybrid CNN-LSTM model shows the most stable performance, achieving the lowest RMSE value of 75.71, which indicates its ability to maintain prediction accuracy under different training-testing data proportions.

This study contributes by providing a systematic comparative analysis of CNN, LSTM, and hybrid CNN-LSTM architectures for stock price prediction in the context of a multi-sector holding company listed in the Indonesian capital market. The results demonstrate that while LSTM effectively captures temporal dependencies, the hybrid CNN-LSTM architecture can provide more stable prediction performance by combining local feature extraction from CNN with sequential temporal learning from LSTM. These findings provide empirical insights that may assist researchers and practitioners in selecting appropriate deep learning architectures for financial time-series forecasting tasks.

From a practical perspective, the findings suggest that hybrid deep learning architectures can serve as a promising approach for supporting data-driven decision making in stock market analysis, particularly in complex market environments where price movements are influenced by multiple sectoral dynamics.

Despite the encouraging findings, this study has several limitations that should be acknowledged. First, this study relies solely on historical adjusted closing price data without incorporating additional explanatory variables such as technical indicators, macroeconomic factors, or investor sentiment. Second, the experiment focuses on a single company as a case study, which may restrict the generalizability of the findings across different stocks or varying market conditions.

Future research may extend this work by incorporating multi-feature financial datasets, including technical indicators, macroeconomic variables, and sentiment-based features, to improve model predictive capability. In addition, future studies may explore advanced deep learning architectures or optimization strategies, such as attention-based architectures, transformer-based forecasting approaches, or automated hyperparameter optimization methods, to further improve prediction accuracy and model generalization.

ACKNOWLEDGMENT

The authors would like to thank Universitas Nasional, particularly the Faculty of Information and Communication Technology, for the academic support provided in the completion of this undergraduate thesis research.

REFERENCES

[1] A. Jafarova, Y. Guliyeva, and T. Rzayeva, “Analysis of the impact of financial market in economy,” *Sci. Collect. InterConf+*, vol. 31, no. 147, pp. 69-76, 2023, doi: 10.51582/interconf.19-20.03.2023.006.

[2] J. A. Ho, J. Wuri, L. B. Harnoto, and Y. R. Hardanti, “Macroeconomic Determinants of Stock Market Performance: Evidence From Indonesia,” *Int. J. Econ. Bus. Manag. Res.*, 2025, doi: 10.51505/ijebmr.2025.9812.

[3] O. Shirke, A. Chaudhari, D. Tandel, and S. Bari, “Stock Market Price Prediction using Machine Learning,” *Int. J. Comput. Appl.*, 2022, doi: 10.5120/ijca2022922092.

[4] E. S. Nugraha, Z. Alika, and D. A. Hamzah, “Forecasting the Stock Price of PT Astra International Using the LSTM Method,” *J. RESTI*, 2024, doi: 10.29207/resti.v8i3.5699.

[5] Z. Xu, “Stock Price Prediction with Traditional Statistics and Machine Learning,” *Adv. Econ. Manag. Polit. Sci.*, vol. 6, no. 1, pp. 272-282, 2023, doi: 10.54254/2754-1169/6/20220138.

[6] M. Zheng, “Studying Stock Prediction in the Context of Machine Learning Exemplified by Analyzing SVM and LSTM Models,” *Highlights Sci. Eng. Technol.*, 2024, doi: 10.54097/wxga9q92.

[7] J. Verma and S. Agarwal, “A hybrid deep neural network model for time series forecasting,” *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 11, no. 1, pp. 20-25, 2022, doi: 10.30534/ijatcse/2022/051112022.

[8] S. Zaheer *et al.*, “A Multi Parameter Forecasting for Stock Time Series Data Using LSTM and Deep Learning Model,” *Mathematics*, vol. 11, no. 3, p. 590, 2023, doi: 10.3390/math11030590.

[9] J. Zhang, W.-H. Chan, and Y. Lin, “Stock Price Prediction Research Based on CNN-LSTM,” *Highlights Business, Econ. Manag.*, 2024, doi: 10.54097/cyqrxb84.

[10] Y. Ding, “Enhancing Stock Price Prediction Method Based on CNN-LSTM Hybrid Model,” *Highlights Business, Econ. Manag.*, vol. 21, pp. 774-781, 2023,

doi: 10.54097/hbem.v21i.14760.

[11] X. Zhao, “Exploring the Performance of the CNN-LSTM Model in Stock Prediction,” *Highlights Business, Econ. Manag.*, vol. 45, pp. 903-907, 2024, doi: 10.54097/x9m1cm10.

[12] E. Seun *et al.*, “Stock Price Prediction: Evaluating the Efficacy of CNN, LSTM, CNN-LSTM, and CNN-BILSTM Models,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 4, pp. 4259-4268, 2024, [Online]. Available: http://www.ijisae.org

[13] J. Zhao, Z. Wang, and H. Li, “Stock Price Prediction Based on Hybrid CNN-LSTM Model,” *Highlights Business, Econ. Manag.*, vol. 24, pp. 120-126, 2024, doi: 10.54097/hbem.v24i.14231.

[14] J. Wei, Q. Xu, and C. He, “Deep Learning of Predicting Closing Price through Historical Adjustment Closing Price,” *Procedia Comput. Sci.*, vol. 202, pp. 379-384, 2022, doi: 10.1016/j.procs.2022.04.052.

[15] H. Jin, “Data Time Spans Impact on Stock Price Prediction: An Empirical Study Across Multiple Markets Based on BP Neural Networks,” *Adv. Econ. Manag. Polit. Sci.*, vol. 191, no. 1, pp. 1-9, 2025, doi: 10.54254/2754-1169/2025.bj24010.

[16] X. Ma, “The Investigation of LSTM-Random Search with Various Standardization and Normalization Technologies,” *Highlights Sci. Eng. Technol.*, 2024, doi: 10.54097/xgeyhr93.

[17] P. Chatigny, J.-M. Patenaude, and S. Wang, “Spatiotemporal Adaptive Neural Network for Long-Term Forecasting of Financial Time Series,” *Int. J. Approx. Reason.*, vol. 132, pp. 70-85, 2021, doi: 10.1016/j.ijar.2020.12.002