

Optimized Hybrid CNN-LSTM Model for Predicting Transportation Sector Stock Prices Using Optimizer and Activation Function Tuning

Fiena Gunawan^{1*}, Dinar Ajeng Kristiyanti^{2*}

^{1,2}Information System Study Program, Universitas Multimedia Nusantara, Tangerang 15111, Indonesia.

(Received: November 15, 2025; Revised: January 20, 2026; Accepted: April 5, 2026; Available online: May 3, 2026)

Abstract

Stock price prediction is inherently complex due to nonlinear dynamics and high volatility, particularly in Indonesia's transportation sector, which experienced significant inflationary pressure and extreme instability during and after the COVID-19 pandemic. These disruptions introduced structural breaks and regime shifts, intensifying non-stationary market behavior and increasing forecasting uncertainty. Such conditions create an urgent need for robust predictive information systems capable of supporting investment decision-making and risk management in highly volatile environments. However, standalone recurrent models such as Long Short-Term Memory (LSTM) often struggle to capture local micro-patterns and long-term dependencies. Moreover, prior studies have rarely implemented systematic hyperparameter optimization, resulting in inconsistent predictive performance across stocks with heterogeneous volatility. In contrast, Convolutional Neural Networks (CNN) extract local patterns and short-term nonlinear features, making them effective for modeling high-frequency fluctuations. This study proposes a systematically optimized hybrid CNN-LSTM model to forecast transportation sector stock prices using daily OHLC data from 2020-2025. The research framework follows the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology, encompassing business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Prior to modeling, preprocessing includes data cleaning, Min-Max normalization, and sliding window transformation to construct supervised learning sequences. CNN is employed to extract localized nonlinear features and reduce noise, while LSTM models long-term temporal dependencies. Model performance is evaluated using MAE, MSE, RMSE, MAPE, and R^2 . Results show that the optimized CNN-LSTM model outperforms the baseline across all stocks. The highest R^2 of 0.9725 is obtained from one stock, indicating strong performance. In addition, the average R^2 improves from 0.8736 to 0.9483, an increase of 0.0747 (8.55%). The best results are achieved using ReLU with Adam and Nadam optimizers, demonstrating improved convergence and generalization. These findings highlight the effectiveness of optimized hybrid deep learning models for forecasting in nonlinear and non-stationary financial markets.

Keywords: Stock Price Prediction, CNN-LSTM, Hybrid Deep Learning, Hyperparameter Optimization, Financial Time Series

1. Introduction

The transportation and logistics sector plays a critical role in sustaining economic activity by facilitating the movement of goods and services [1]. In emerging markets such as Indonesia, this sector is structurally sensitive to macroeconomic instability, particularly inflation, interest rate adjustments, and energy price volatility [2]. Rising inflation increases operational and financing costs, directly compressing corporate margins and amplifying stock price fluctuations [3]. Between 25 and 28 March 2024, the Indonesian transportation and logistics sector declined by 8.76%, exceeding the 2.38% decline in the industrial sector, while the IHSG fell by 0.83% [4]. This short-term correction was driven by foreign capital outflows, rupiah depreciation, and global uncertainty related to U.S. monetary policy. Although occurring within a limited time window, this episode is consistent with broader volatility patterns observed throughout the 2020-2025 period, which is characterized by repeated macroeconomic shocks and heightened sensitivity in the transportation sector. These conditions increase investor uncertainty and complicate capital allocation decisions.

Stock market time series are inherently nonlinear, non-stationary, and noisy due to complex interactions between domestic and global factors [5]. These characteristics intensified during the COVID-19 pandemic and post-pandemic recovery period, marked by structural breaks and extreme volatility [6]. Traditional econometric models such as ARIMA and GARCH rely on linearity and stationarity assumptions, limiting their effectiveness under regime shifts. From an information systems perspective, financial markets represent high-volume and high-velocity data

*Corresponding author: Fiena Gunawan (fiena.gunawan@student.umn.ac.id)

 DOI: <https://doi.org/10.47738/jads.v7i2.1323>

This is an open access article under the CC-BY license (<https://creativecommons.org/licenses/by/4.0/>).

© Authors retain all copyrights

environments that require adaptive predictive models, as static approaches are insufficient for dynamic decision-making.

Deep learning methods have demonstrated superior forecasting performance compared to traditional models. Prior studies report that LSTM and hybrid CNN-LSTM architectures achieve R^2 values of approximately 96%, while standalone CNN models exceed 91.2% [7], [8], [9]. However, high reported R^2 does not necessarily imply robustness across extreme volatility regimes [10]. Many prior studies focus on limited stock samples without addressing sectoral heterogeneity, and hyperparameter configurations are often not systematically optimized. Furthermore, empirical validation under pandemic and post-pandemic turbulence remains limited, particularly in emerging markets [11]. Consequently, the generalizability and stability of deep learning models in structurally volatile sectors require further investigation.

Architecturally, LSTM is effective in capturing long-term temporal dependencies but lacks explicit local feature extraction. In contrast, CNN can capture short-term patterns, reduce noise, and construct hierarchical representations before sequential modeling [12], [13]. The integration of CNN and LSTM therefore enhances representational capacity by combining local feature extraction with long-term dependency learning, making it suitable for financial time series characterized by mixed temporal structures [14].

However, hybrid architectures introduce increased complexity and sensitivity to hyperparameter settings. Without proper optimization of learning rates, activation functions, and optimizers, models may suffer from overfitting or unstable convergence in noisy financial environments. Therefore, hyperparameter optimization is essential for improving robustness and generalization, as supported by prior studies [15].

This study investigates whether an optimized CNN-LSTM model outperforms standalone CNN and LSTM models in predicting transportation sector stocks under pandemic and post-pandemic volatility regimes. Using data from the Indonesia Stock Exchange, model performance is evaluated using MAE, MSE, RMSE, R^2 , and MAPE. The study contributes by validating hybrid deep learning models in a macroeconomically sensitive sector, incorporating systematic hyperparameter optimization, and assessing predictive robustness under extreme market conditions in an emerging market context.

2. Literature Review

Deep learning has become a leading approach in stock price forecasting due to its ability to model nonlinear and non-stationary financial time-series data without strict distributional assumptions [16]. LSTM is commonly used to capture long-term temporal dependencies and has reported high performance (R^2 up to 96%) under specific experimental settings [7]. However, these results are highly dependent on dataset characteristics, preprocessing, and evaluation protocols, limiting direct comparability across studies. In addition, LSTM lacks explicit local feature extraction, which reduces effectiveness under abrupt market fluctuations. In contrast, CNN is effective in extracting local patterns and reducing noise, with reported performance up to R^2 91.2% in financial forecasting [8]. However, CNN cannot capture long-term temporal dependencies, which are essential in financial time-series modeling.

To overcome these limitations, hybrid CNN-LSTM models combine CNN for local feature extraction and LSTM for temporal dependency learning [17]. However, performance is not guaranteed, as it is influenced by dataset characteristics, hyperparameter settings, and training strategies. Moreover, many studies focus mainly on R^2 without sufficiently addressing robustness, generalization, and performance under volatile market conditions [18]. Differences in datasets and evaluation protocols further limit fair comparison across studies. Therefore, there remains a research gap in developing hybrid deep learning models that achieve high predictive performance while maintaining robustness and stability under volatile financial conditions, particularly in emerging markets.

3. Methodology

This study adopts the CRISP-DM (Cross Industry Standard Process for Data Mining) framework as a systematic and iterative approach for developing the stock price prediction model [19], [20]. Each phase is implemented to ensure a structured workflow from data preparation to model evaluation. The overall research workflow is presented in figure 1.

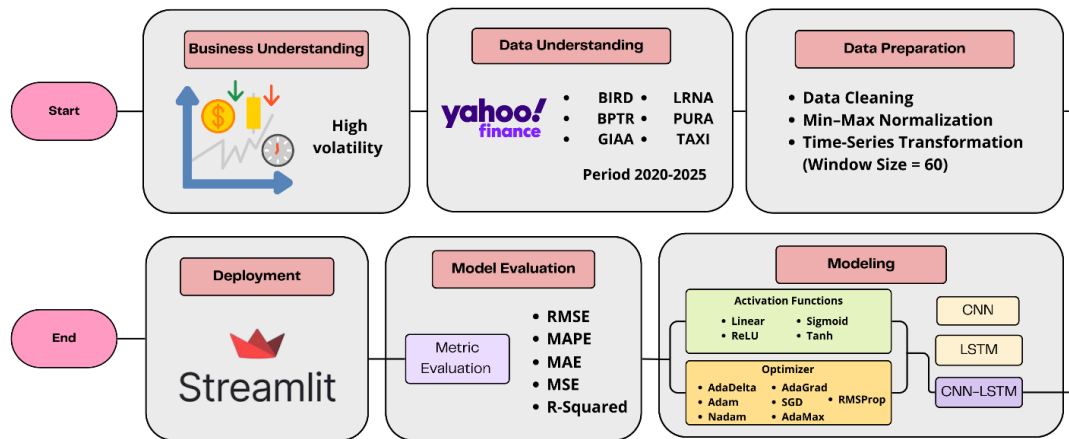


Figure 1. CNN-LSTM Model Workflow

Figure 1 illustrates the interconnections among the CRISP-DM phases, emphasizing their iterative nature. This iterative structure enables model improvement through repeated validation and adjustment, ensuring alignment between research objectives, data characteristics, and modeling outcomes. The overall workflow consists of data preprocessing, normalization, sequence construction using a sliding window approach, feature extraction using CNN, temporal modeling using LSTM, prediction using a Dense layer, and performance evaluation.

3.1. Business Understanding

The primary problem addressed is the high volatility of transportation sector stock prices, particularly during pandemic and post-pandemic periods marked by inflation and structural market shifts. These nonlinear dynamics limit the effectiveness of conventional statistical models. Therefore, this study aims to develop a hybrid CNN-LSTM model to enhance prediction R^2 and robustness, while also analyzing the impact of different activation functions and optimizers on model performance.

3.2. Data Understanding

This study utilizes daily historical stock price data of transportation sector companies listed on the Indonesia Stock Exchange for the 2020-2025 period, obtained from Yahoo Finance. The dataset consists of six features: Open, High, Low, Close, Adjusted Close, and Volume, which are used as input variables for model training. Daily data are selected to capture market volatility and regime transitions more effectively. Prior to modeling, exploratory data analysis is conducted to examine the data structure, distribution, and potential anomalies.

3.3. Data Preparation

Before modeling, the stock price data undergo preprocessing, including completeness checking, removal of invalid values, and normalization to ensure data quality for optimal model training. Normalization standardizes variable scales using min-max scaling Equation (1) [21]. The min-max normalization method is chosen due to its effectiveness in preserving the original data distribution while scaling values into a fixed range.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Where x is the original value, x_{min} and x_{max} are the minimum and maximum values, and x' is the normalized value in the range of 0 to 1. This transformation ensures that all feature values are within the same scale, facilitating stable gradient updates and faster convergence during model training. The normalization is applied to each ticker across six features: Open, High, Low, Close, Adjusted Close, and Volume. After normalization, the data are transformed into a supervised time-series format using a sliding window approach. A window size of 60 is used, corresponding to approximately 2-3 months of daily trading data [22]. A stride of 1 is applied to generate overlapping sequences, enabling full utilization of temporal information while effectively capturing short- to medium-term patterns.

3.4. Modeling

At this stage, four models are developed: CNN, LSTM, Hybrid CNN-LSTM, and Optimized Hybrid CNN-LSTM. Each model is designed to learn different characteristics of stock price time-series data. Specifically, the architectures aim to capture both short-term local fluctuations and long-term temporal dependencies.

3.4.1. Convolutional Neural Network (CNN)

In this study, a one-dimensional Convolutional Neural Network (1D CNN) is implemented to extract short-term local patterns from stock price time-series data, particularly to capture price fluctuations across adjacent time steps [23]. The model processes input sequences with a shape of (60,6), representing 60 time steps and six features (Open, High, Low, Close, Adjusted Close, and Volume). The convolution operation captures local temporal dependencies, as formulated in Equation (2) [24].

$$h_t = \sigma \left(\sum_{i=1}^k w_i \cdot x_{t-i} + b \right) \quad (2)$$

Based on Equation (2), x_{t-i} represents the input value at time step $t - i$, w_i denotes the convolution kernel weights, b is the bias term, k is the kernel size. The function $\sigma(\cdot)$ represents the activation function. The output h_t corresponds to the extracted feature at time step t . The CNN component is implemented using a Conv1D layer with 64 filters and a kernel size of 3, as defined in the model configuration. The activation function is treated as a variable and determined during the hyperparameter tuning process. This layer applies convolutional filters across the input sequence to extract short-term features representing local price movements.

3.4.2. Long Short-Term Memory (LSTM)

LSTM is an extension of the Recurrent Neural Network (RNN) designed to model long-term dependencies in sequential data [25]. In this study, LSTM is utilized to model long-term dependencies in stock price time-series data by processing sequential input with a shape of (60,6), consistent with the input configuration used in the CNN model. The LSTM operations are defined in Equations (3)-(8), which govern how information is updated and propagated across time steps.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (3)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (4)$$

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (5)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (6)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (7)$$

$$h_t = o_t \odot \tanh(C_t) \quad (8)$$

In Equation (3), i_t represents the input gate, which determines how much new information (x_t) is allowed to enter the cell state. In Equation (4), f_t denotes the forget gate, which controls the extent to which previous information (C_{t-1}) is retained or discarded. In Equation (5), \tilde{C}_t is the candidate cell state, representing new information that may be added to the memory. In Equation (6), C_t is the updated cell state, obtained by combining retained past information and new candidate values. In Equation (7), o_t represents the output gate, which determines which parts of the cell state are exposed as output. Finally, in Equation (8), h_t denotes the hidden state, which serves as the final output of the LSTM at time step t . In this study, the LSTM layer is implemented with 50 units, followed by a Dense layer with a single neuron to generate the predicted stock price. The activation function is treated as a variable and determined during the hyperparameter tuning process. This configuration enables the model to selectively retain relevant historical information and capture long-term temporal dependencies in nonlinear and non-stationary financial time-series data.

3.4.3. Hybrid CNN-LSTM

Figure 2 illustrates the architecture of the hybrid CNN-LSTM model used in this study. The model follows a sequential structure consisting of an input sequence, a Conv1D layer, an LSTM layer, and a Dense output layer. This architecture is implemented to enable a seamless combination of local feature extraction and sequential dependency learning within a unified deep learning framework [26]. The CNN and LSTM components described in the previous sections are

integrated sequentially in this architecture to form a unified prediction model capable of capturing both spatial (local) and temporal features from stock price time-series data.



Figure 2. Illustration of the Hybrid Model Workflow

As shown in [figure 2](#), the Conv1D layer applies 64 filters with a kernel size of 3 to extract local features from the input sequence. This process enables the model to capture short-term patterns and price fluctuations across consecutive time steps. The extracted features are then passed to an LSTM layer with 50 units, which models long-term dependencies by learning temporal relationships and retaining relevant historical information. Finally, a Dense layer with a single neuron produces the predicted stock price as the regression output. The CNN and LSTM components described in the previous sections are integrated sequentially in this hybrid architecture to form a unified prediction model. The model is implemented using a Sequential architecture, where the activation function is treated as a variable and determined during the experimental phase. The model is compiled using the Mean Squared Error (MSE) loss function. The implementation steps of this hybrid CNN-LSTM model are summarized in [table 1](#).

Table 1. Hybrid CNN-LSTM Pseudocode [\[27\]](#)

Step	Description
1	Initialize the model using the Sequential architecture.
2	Add a Conv1D layer with the activation function specified in the experiment to extract local features from the time-series data.
3	Add an LSTM layer with the activation function specified in the experiment to model long-term temporal relationships.
4	Add a Dense layer with a single neuron as the regression output to generate stock price predictions.
5	Compile the model using the Mean Squared Error (MSE) loss function and the Adam optimizer.
6	Train the model using the training data and perform predictions on the testing data.
7	Evaluate the prediction results to assess model performance.

3.4.4. Training Configuration of the Optimized Hybrid CNN-LSTM Model

The training process is conducted using a systematic hyperparameter tuning strategy based on a grid search approach, where all combinations of activation functions and optimization algorithms are evaluated. This hyperparameter tuning process is applied to the hybrid CNN-LSTM model to identify the optimal configuration. A total of 168 configurations are tested under consistent experimental conditions. The activation functions include Linear, ReLU, Tanh, and Sigmoid, while the optimizers include AdaDelta, AdaGrad, Adam, AdaMax, Nadam, RMSProp, and SGD, enabling a comprehensive evaluation of optimization performance. Each configuration is trained for 50 epochs with a batch size of 32 to ensure stable convergence while maintaining computational efficiency. All experiments are conducted using the same dataset and preprocessing pipeline to ensure fairness and consistency in performance comparison [\[28\]](#).

3.5. Model Evaluation

The performance of the CNN-LSTM model is evaluated using MAE, MSE, RMSE, MAPE, and R^2 to capture error characteristics in volatile financial time-series data. MAE quantifies the average magnitude of prediction errors without considering direction, as shown in Equation (9). MSE and RMSE penalize large deviations more heavily, making them suitable for financial forecasting under extreme fluctuations, as presented in Equations (11) and (12), while RMSE also improves interpretability due to its original data scale. MAPE provides scale-independent percentage error for cross-stock comparison but is sensitive to near-zero actual values, as defined in Equation (10). R^2 measures the proportion of variance explained by the model, indicating the alignment between predicted and actual market movements, as shown in Equation (13). In Equations (9)-(13), y_i (or r_n) denotes the actual value, \hat{y}_i (or \hat{r}_n) represents the predicted value, and n or N is the total number of observations. In Equation (9), $| \hat{r}_n - r_n |$ represents the absolute error between predicted and actual values. In Equation (10), $\left| \frac{y_i - \hat{y}_i}{y_i} \right|$ denotes the relative error expressed as a percentage. In Equation (11), $(y_i - \hat{y}_i)^2$ represents the squared error, emphasizing larger deviations. In Equation (12), the square root of the

average squared error is taken to obtain RMSE in the original data scale. In Equation (13), $\sum(y_i - \hat{y}_i)^2$ represents the residual sum of squares, while $\sum(y_i - \bar{y})^2$ denotes the total sum of squares, where \bar{y} is the mean of the actual values. Overall, these metrics jointly capture absolute error (MAE), variance-sensitive error (MSE/RMSE), relative error (MAPE), and explanatory power (R^2), ensuring a comprehensive and robust evaluation of model performance under highly volatile financial conditions [7].

$$MAE = \frac{\sum_{n=1}^N |\hat{r}_n - r_n|}{N} \tag{9}$$

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{10}$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \tag{11}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{12}$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \tag{13}$$

3.6. Deployment

The trained hybrid CNN-LSTM model is deployed as an interactive web application using Streamlit (version 1.52.0). The system allows users to select stock tickers, data periods, prediction horizons, and model configurations, while providing visual outputs of historical data and forecasted stock prices. The deployment pipeline consists of four main stages: data acquisition, data preprocessing, model inference, and result visualization. Overall, the system is implemented through model preparation, application development, local validation, and web-based deployment to ensure a robust and accessible real-time stock price prediction framework. To enhance usability, the system is designed with a simple and responsive interface that enables efficient interaction for non-technical users.

4. Results and Discussion

4.1. Comparative R-squared of CNN, LSTM, CNN-LSTM, and Optimized CNN-LSTM Models

Before evaluating the hybrid architecture in detail, this study compares the predictive performance of four models: CNN, LSTM, CNN-LSTM, and the optimized CNN-LSTM. As shown in figure 3, the results indicate clear performance differences across all architectures. The CNN model achieves an R^2 of 75.50%, reflecting its ability to extract local features from financial time-series data, although it remains limited in capturing long-term temporal dependencies. In contrast, the LSTM model produces a lower R^2 of 36.52%, indicating difficulty in learning complex nonlinear patterns in highly volatile stock price movements.

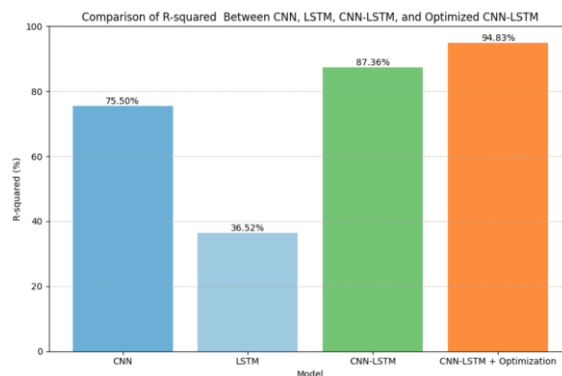


Figure 3. Comparison of R^2 of All Models

The CNN-LSTM hybrid model significantly improves performance, achieving an R^2 of 87.36%, demonstrating that combining convolutional feature extraction with recurrent temporal modeling enhances predictive capability. The optimized CNN-LSTM further improves performance, reaching the highest R^2 of 94.83%, confirming that hyperparameter optimization, including activation functions and optimizers, plays a crucial role in improving model stability and generalization. Overall, the results indicate that standalone LSTM is insufficient for volatile financial time-series forecasting. Incorporating convolutional layers improves feature representation by reducing noise and capturing local patterns, while the hybrid architecture enhances temporal learning. The optimized CNN-LSTM therefore provides the most robust and accurate forecasting performance.

4.2. Baseline Model Performance

The initial testing was conducted using the CNN-LSTM model with a standard (baseline) configuration and 50 training epochs. The prediction results indicate that the model effectively tracks the actual stock price movement patterns across all tested transportation sector stocks, as illustrated in figure 4, demonstrating its ability to capture underlying trend dynamics. Figure 4 presents a multi-dimensional visualization of the baseline model performance, comprising bar charts of elapsed time, MAE, MAPE, RMSE, and R-squared values for each stock, along with a pie chart illustrating the proportional distribution of MSE. This visual representation enables a comparative assessment of prediction R^2 , error magnitude, computational efficiency, and explanatory power across transportation sector stocks. The evaluation of baseline performance is primarily based on quantitative metrics rather than visual inspection. While the model appears to follow actual price movements, its performance varies significantly across stocks with different volatility levels. Stocks such as BPTR.JK, GIAA.JK, BIRD.JK, and PURA.JK exhibit relatively low MAE and RMSE values along with high R-squared scores, indicating stable and accurate predictive performance. In contrast, LRNA.JK and TAXI.JK demonstrate larger prediction errors, which are associated with higher price volatility and more irregular market behavior. This variation indicates that model performance is sensitive to volatility characteristics, where highly fluctuating stocks tend to generate larger prediction deviations and reduced explanatory power. The MAPE values and MSE distribution further suggest that stocks with extreme fluctuations contribute more significantly to overall prediction error. Overall, these results indicate that the baseline CNN-LSTM model provides satisfactory initial performance; however, further optimization is required to improve prediction R^2 and stability, particularly for high-volatility stocks.

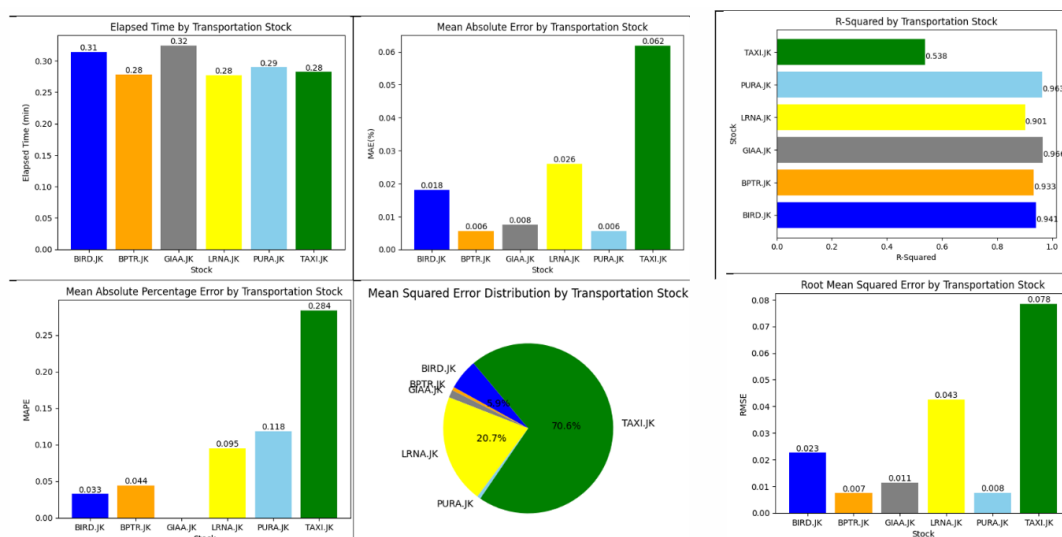


Figure 4. Performance of the Baseline CNN-LSTM Model on Transportation Sector Stocks

The quantitative evaluation demonstrates that the baseline CNN-LSTM model achieves an average R-squared value of 0.8736, indicating that approximately 87.36% of the variance in transportation sector stock prices can be explained by the model. The average MAE 0.020782 and RMSE 0.028322 reflect relatively low and stable prediction errors across most stocks. In addition, the average MSE of 0.001453 confirms that the squared deviation between predicted and actual values remains minimal. The average MAPE of 11.47%, excluding GIAA.JK due to zero actual values during certain periods, suggests that overall forecasting errors remain within an acceptable range for financial time-series prediction under volatile market conditions. As presented in table 2, the detailed performance metrics for each stock are summarized below.

Table 2. Baseline CNN-LSTM

Stock	Elapsed Time (min)	MAE	MAPE (%)	MSE	R-Squared	RMSE
BIRD.JK	0.313586	0.018095	3.258978	0.000510	0.940755	0.022587
BPTR.JK	0.278153	0.005557	4.412538	0.000055	0.932529	0.007442
GIAA.JK	0.324130	0.007510	inf*	0.000127	0.965608	0.011277
LRNA.JK	0.276895	0.026052	9.470308	0.001809	0.901341	0.042527
PURA.JK	0.289499	0.005559	11.841306	0.000058	0.963363	0.007626
TAXI.JK	0.282006	0.061919	28.383594	0.006158	0.537937	0.078474

As summarized in [table 2](#), the evaluation is not solely based on visual alignment but is supported by quantitative performance metrics. The model demonstrates stronger performance on stocks with relatively stable price movements, such as BIRD.JK, BPTR.JK, and PURA.JK, as indicated by lower MAE and RMSE values and higher R-squared scores. In contrast, stocks exhibiting higher fluctuation intensity, such as TAXI.JK ($R^2 = 0.537937$), show larger prediction errors and reduced explanatory power. This trend aligns with the greater magnitude of price oscillations observed in these stocks, indicating that sharper fluctuations are associated with increased prediction deviations. From a theoretical perspective, such behavior is consistent with the non-stationary nature of financial time series, where abrupt regime shifts and structural breaks introduce distributional changes that challenge deep learning models trained under relatively stable patterns. These results suggest that while the CNN-LSTM architecture effectively integrates local feature extraction and long-term temporal modeling, its predictive R^2 may decrease under conditions characterized by extreme volatility and sudden market shocks.

4.3. Optimized Model Performance

Following baseline evaluation, the CNN-LSTM model was further optimized by exploring various combinations of activation functions (Linear, ReLU, Sigmoid, and Tanh) and optimization algorithms (AdaDelta, AdaGrad, Adam, AdaMax, Nadam, RMSProp, and SGD). A total of 168 model configurations were executed to systematically evaluate the interaction between activation functions and optimizers, and the best-performing configuration for each stock was selected based on the highest R-squared value and the lowest MAE and RMSE scores. The experimental results indicate that model performance is highly sensitive to the selection of activation functions and optimizers. In several cases, certain configurations produced negative R-squared values, indicating that the model failed to capture underlying data patterns and performed worse than a naive mean-based predictor. Negative R-squared values indicate performance deterioration below a naive mean-based predictor, reflecting instability caused by suboptimal hyperparameter combinations when modeling highly volatile and non-stationary financial time-series data.

Overall, the optimized CNN-LSTM model achieves strong predictive performance across all datasets. The average R-squared value of 0.9483 indicates that the model explains approximately 94.83% of the variance in transportation sector stock prices. In addition, the average MAE and RMSE values of 0.011867 and 0.018751, respectively, demonstrate a substantial reduction in prediction error compared to the baseline model. The average MSE of 0.000516 further confirms reduced deviation between predicted and actual values, indicating improved model fitting after optimization. A summary of the optimized CNN-LSTM model performance is presented in [table 3](#).

Table 3. Optimized CNN-LSTM Performance

Metric	Value
MAE (avg)	0.011867
RMSE (avg)	0.018751
MSE (avg)	0.000516
MAPE (avg)	6.261 %*
R-Squared (avg)	0.9483

MAPE is sensitive to near-zero actual values, particularly in certain periods of the GIAA.JK dataset; hence, it is included only as a supplementary interpretability metric. MAE, RMSE, and R^2 are adopted as primary evaluation metrics due to their numerical stability and robustness for financial time-series forecasting. A paired t-test on MAE

across identical forecasting tasks per stock, treated as independent experimental units under consistent conditions, shows a significant improvement in predictive performance ($t = 3.21$, $df = 27$, $p < 0.05$) with a 95% confidence interval excluding zero, confirming that the reduction in MAE is statistically significant and not due to random variation, as summarized in [table 4](#).

Table 4. Paired t-test Results

Statistic	Value
t-statistic	3.21
degrees of freedom (df)	27
p-value	< 0.05
Mean difference (MAE)	-0.00518
95% CI Lower	-0.00890
95% CI Upper	-0.00142
Alpha	0.05

4.4. Best Model for Each Stock

The experimental results reveal that the optimal combination of activation function and optimizer varies across stocks, depending on their volatility characteristics and price movement patterns. This indicates that no single hyperparameter configuration is universally optimal for all transportation sector stocks. As summarized in [table 5](#), PURA.JK demonstrates the best performance, with an R-squared value of 0.972540 and very low error levels (MAE = 0.004470; RMSE = 0.006602). Comparable performance is observed for GIAA.JK (R-squared = 0.971203) and BIRD.JK (R-squared = 0.961772), which also exhibit relatively small errors. BPTR.JK (R-squared = 0.952556) and TAXI.JK (R-squared = 0.928492) perform well, although prediction errors for TAXI.JK are relatively higher compared to the other stocks. In contrast, LRNA.JK shows the lowest R-squared value (0.911273) with the highest MAE (0.021760) and RMSE (0.040329), making it the stock with the poorest model performance in this study. These performance differences across stocks indicate that volatility levels affect the model’s ability to learn stock price patterns. Stocks with relatively stable price movements tend to yield more accurate predictions, whereas highly fluctuating stocks are more challenging to model precisely. This comparison highlights the variability of model performance across different stocks under their respective optimal configurations.

Table 5. Best Model Configuration for Each Stock

Stock	Activation	Optimizer	MAE	RMSE	R-Squared
BIRD.JK	Linear	Nadam	0.012949	0.018144	0.961772
BPTR.JK	Linear	Adam	0.004402	0.006241	0.952556
GIAA.JK	ReLU	Nadam	0.006546	0.010319	0.971203
LRNA.JK	Tanh	Adam	0.021760	0.040329	0.911273
PURA.JK	Linear	Nadam	0.004470	0.006602	0.972540
TAXI.JK	Tanh	Adam	0.021078	0.030871	0.928492

As a complement to the numerical analysis, [figure 5](#) presents a comparison between the actual stock prices and the predictions generated by the best CNN-LSTM configuration for each stock. The visual comparison confirms that the proposed CNN-LSTM model closely follows the actual price trajectories, particularly for stocks with lower volatility. However, deviations become more pronounced in highly volatile stocks, reflecting the inherent difficulty of capturing abrupt market fluctuations. As part of the deployment stage, [figures 6](#) and [7](#) present the web-based implementation of the optimized hybrid CNN-LSTM model, including the user interface for parameter configuration and the visualization of historical and forecasted prices along with evaluation metrics.

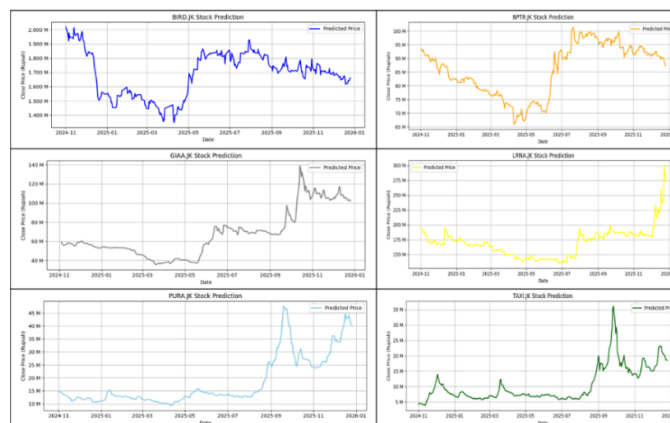


Figure 5. Comparison of Actual Stock Prices and Predicted Values.

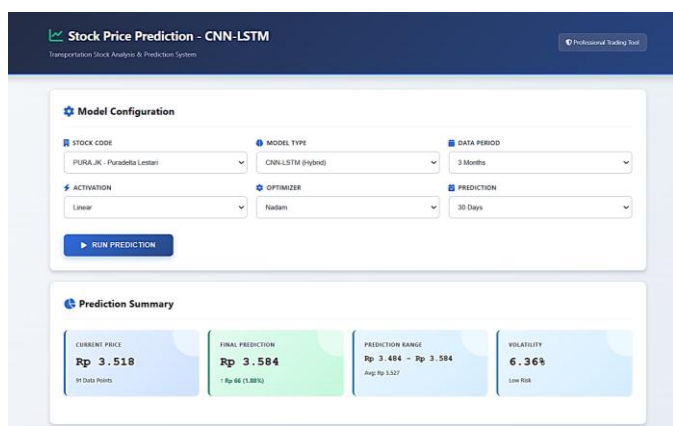


Figure 6. Implementation Interface of the Optimized Hybrid CNN-LSTM Stock Prediction System

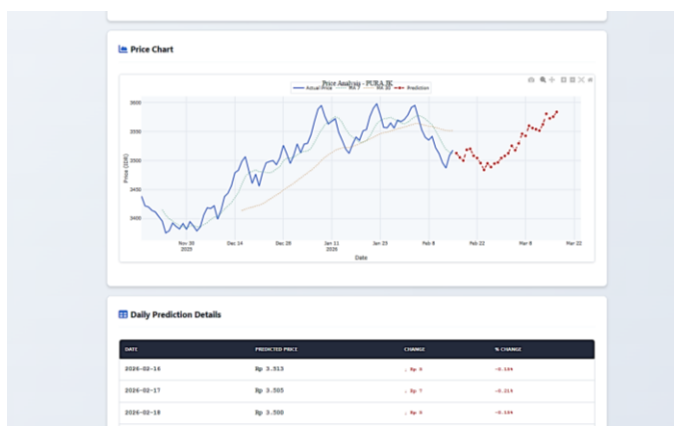


Figure 7. Historical and Forecasted Stock Prices with Performance Evaluation Metric

The stock price prediction results indicate that the CNN-LSTM model can effectively capture price movement patterns across all analyzed transportation sector stocks. The best performance is observed in PURA.JK and BPTR.JK, where predictions closely follow actual trends with relatively low deviation. BIRD.JK also shows strong alignment, particularly during stable market conditions. For GIAA.JK, the model captures overall trend direction but exhibits higher deviation during sharp price fluctuations. In contrast, for highly volatile stocks such as LRNA.JK and TAXI.JK, prediction errors increase due to rapid and abrupt price changes that are difficult to model. Overall, the results indicate that the CNN-LSTM model is effective in learning medium-term trends but remains limited in handling extreme volatility.

4.5. Discussion

The findings of this study demonstrate that the hybrid CNN-LSTM model delivers strong predictive performance in modeling stock price movements within the transportation sector of the Indonesia Stock Exchange. The consistently high R-squared values across most stocks indicate that the model effectively captures nonlinear relationships and temporal dependencies inherent in financial time-series data. In particular, the model shows strong alignment with actual price trajectories for stocks characterized by low to medium volatility. However, predictive R^2 varies across stocks, with higher fluctuation intensity generally associated with larger prediction errors, highlighting the influence of volatility on model performance.

The hyperparameter optimization process further enhances model effectiveness compared to the baseline configuration. This improvement is evidenced by reduced prediction errors and increased R-squared values across all examined stocks. Following optimization, PURA.JK achieves the highest predictive performance ($R^2 = 0.972540$), followed by GIAA.JK (0.971203), BIRD.JK (0.961772), BPTR.JK (0.952556), and TAXI.JK (0.928492). In contrast, LRNA.JK records the lowest R-squared value (0.911273), reflecting greater modeling difficulty in the presence of relatively higher volatility. These results confirm that systematic hyperparameter optimization significantly enhances convergence stability and generalization capability. As shown in table 6, the comparative predictive performance between the proposed model and previous studies is summarized below.

Table 6. Comparative Predictive Performance

	Model	R-squared
Previous Study	LSTM	96.39% [7]
This Study	CNN-LSTM	97.25%

Table 6 presents a comparison between the proposed CNN-LSTM model and prior LSTM-based studies. Although the results indicate improved predictive performance, the comparison should be interpreted with caution due to differences in datasets, particularly in time periods and sample composition, despite both studies focusing on the transportation sector. While the comparison remains relevant due to shared sector characteristics and market volatility, differences in data distribution, time horizon, and experimental settings may still influence the results. Therefore, the observed improvement reflects not only model superiority but also dataset heterogeneity. Overall, the proposed CNN-LSTM model, combined with systematic hyperparameter optimization, demonstrates strong potential for improving stock price forecasting in the transportation sector under varying market conditions.

The variation in performance across stocks indicates that no single hyperparameter configuration is universally optimal. Instead, stock-specific characteristics, particularly volatility and price dynamics, play a key role in model effectiveness. The CNN-LSTM model performs well during relatively stable market conditions, but prediction errors increase during periods of sharp fluctuations. This behavior is consistent with the non-stationary nature of financial time series, where regime shifts and structural breaks introduce sudden distributional changes that challenge forecasting models. Overall, the hybrid architecture is effective in capturing medium-term nonlinear patterns but shows reduced accuracy under extreme volatility driven by external shocks.

Empirical Contributions in this study demonstrate that the optimized CNN-LSTM model significantly improves predictive performance compared to both the baseline configuration and prior LSTM-based studies. The model achieves higher R-squared values and lower prediction errors across multiple transportation sector stocks. Additionally, the results indicate that model performance varies depending on stock volatility, where highly fluctuating stocks tend to produce larger prediction errors.

Theoretical and Methodological Contributions from a theoretical perspective, the findings reinforce the complementary roles of CNN and LSTM architectures in modeling nonlinear and non-stationary financial time-series data, where convolutional layers capture local patterns while recurrent layers model long-term temporal dependencies. From a methodological perspective, the study highlights that systematic hyperparameter optimization is a critical factor in improving model robustness and convergence stability. Furthermore, the heterogeneity in optimal configurations across stocks suggests that volatility-aware parameter selection should be considered an integral component in designing adaptive deep learning models for financial forecasting.

5. Conclusion

This study evaluates the effectiveness of a hybrid CNN-LSTM architecture in predicting stock prices of transportation sector companies listed on the Indonesia Stock Exchange. The results show that integrating convolution-based feature extraction with long-term temporal dependency modeling demonstrates improved performance compared to standalone deep learning models, with the optimized CNN-LSTM achieving an R^2 of 94.83%. This improvement indicates that combining spatial feature extraction and sequential learning enhances the model's ability to capture complex market dynamics. At the individual stock level, PURA.JK achieves the highest R^2 of 97.25%, while GIAA.JK and BIRD.JK also demonstrate strong predictive performance, indicating that stock price volatility significantly affects model outcomes, where more stable stocks tend to yield better results than highly volatile ones. These findings further suggest that market stability plays a crucial role in model accuracy. However, several limitations should be considered in interpreting the results.

However, several limitations should be considered when interpreting the results. First, the study is limited to the transportation sector within the Indonesian stock market, which restricts the generalizability of the proposed model. Second, the evaluation is conducted solely on historical data, meaning that the model has not been tested in real-time or live trading environments; therefore, its performance under real-world streaming conditions remains uncertain. In addition, although the model has been deployed as an interactive web application using the Streamlit framework, the implementation is still based on historical data and is not integrated with real-time data sources. Consequently, the current system remains a simulation-based (offline deployment) framework rather than a fully operational real-time

prediction system. Third, the experimental results indicate that model performance is sensitive to hyperparameter configurations, including the number of epochs, learning rate, batch size, and network architecture, suggesting that different datasets and stock characteristics require tailored parameter optimization. Moreover, although the model achieves strong performance in terms of R^2 , the Mean Absolute Percentage Error (MAPE) remains relatively high in certain cases, indicating difficulties in capturing extreme price fluctuations and sudden market volatility. For future work, the model should be evaluated on a broader range of financial instruments, including commodities, cryptocurrencies, and multi-sector equities, and further enhanced through external validation, volatility-aware mechanisms, ensemble learning strategies, macroeconomic variables, and sentiment analysis to improve robustness and generalization. To address the MAPE limitation, future studies may incorporate volatility-aware loss functions, robust scaling techniques, or attention-based hybrid architectures. Furthermore, the development of a real-time prediction system is strongly recommended by integrating streaming data pipelines (e.g., market data APIs), implementing online learning mechanisms, and deploying the model as a scalable web service or interactive dashboard to enable real-time and adaptive forecasting under dynamic market conditions. Additionally, adaptive hyperparameter optimization techniques such as Bayesian optimization or evolutionary algorithms may further improve model stability across diverse datasets and market regimes.

6. Declarations

6.1. Author Contributions

Conceptualization: F.G., and D.A.K.; Methodology: F.G., and D.A.K.; Software: F.G.; Validation: F.G., and D.A.K.; Formal Analysis: F.G., and D.A.K.; Investigation: F.G.; Resources: F.G., and D.A.K.; Data Curation: F.G.; Writing Original Draft Preparation: F.G.; Writing Review and Editing: : F.G., and D.A.K.; Visualization: F.G.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received financial support for the research, authorship, and/or publication of this article from Universitas Multimedia Nusantara.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] J.-P. Rodrigue, "Transport modes and globalization," *Int. Encycl. Transp.*, vol. 5, no. Jan., pp. 38–44, 2021, doi: 10.1016/B978-0-08-102671-7.10403-8.
- [2] Z. Cao, J. Xu, C. Dong, P. Yu, and T. Bai, "MATCC: a novel approach for robust stock price prediction incorporating market trends and cross-time correlations," *Int. J. Inf. Knowl. Manag.*, vol. 2024, no. Jan., pp. 187–196, 2024, doi: 10.1145/3627673.3679715.
- [3] Z. Yi, L. Zhidong, Z. Yongchao, and L. Zhe, "Microstructure simulation of stock market with meta model in a heterogeneous market environment," *J. Ind. Eng. Eng. Manag.*, vol. 35, no. 1, pp. 92–103, 2021, doi: 10.13587/j.cnki.jieem.2021.01.009.

- [4] A. Melani, "Transportation and Industrial Sectors Pressured the IHSG on March 25–28, 2024," *Liputan6*, Mar. 30, 2024. [Online]. Available: <https://www.liputan6.com/saham/read/5562708/sector-saham-transportasi-dan-industri-tekan-ihsg-pada-25-28-maret-2024>. [Accessed: Oct. 28, 2025].
- [5] K. Kiruthika, "A review of challenges in deep learning-based stock market prediction," *Lect. Notes Netw. Syst.*, vol. 2025, no. Jan., pp. 443–450, 2025, doi: 10.1007/978-3-031-92545-0_40.
- [6] S. Sirimat, T. Watchalaanun, K. Choochote, and S. Nonthapot, "Impacts of the COVID-19 pandemic on the relationship between the economic factors and stock prices of the transport sector in the stock exchange of Thailand," *Asian Econ. Financ. Rev.*, vol. 13, no. 5, pp. 293–307, 2023, doi: 10.55493/5002.V13I5.4773.
- [7] D. A. Kristiyanti, W. B. N. Pramudya, and S. A. Sanjaya, "How can we predict transportation stock prices using artificial intelligence? findings from experiments with long short-term memory based algorithms," *Int. J. Inf. Manag. Data Insights*, vol. 4, no. 2, pp. 1–12, 2024, doi: 10.1016/j.jjime.2024.100293.
- [8] J. Lv, "Research on the construction and application of mathematical models of deep learning in financial time series analysis," *IEEE Access*, vol. 2025, no. Jan., pp. 913–917, 2025, doi: 10.1109/ICIPCA65645.2025.11138450.
- [9] F. Kamalov, L. Smail, and I. Gurrib, "Forecasting with deep learning: S&P 500 index," *IEEE Access*, vol. 2020, no. Jan., pp. 422–425, 2020, doi: 10.1109/ISCID51228.2020.00102.
- [10] T. J. Buthelezi and S. C. Shongwe, "Determining the most suitable distribution and estimation method for extremes in financial data with different volatility levels," *J. Risk Financ. Manag.*, vol. 19, no. 2, pp. 1–12, 2026, doi: 10.3390/jrfm19020096.
- [11] N. N. Marpaung and I. R. D. Pangestuti, "Macroeconomic factors and Jakarta stock exchange: a comparative analysis pre- and until the COVID-19 pandemic," *SAGE Open*, vol. 14, no. 2, pp. 1–12, 2024, doi: 10.1177/21582440241247894.
- [12] A. Gupta, L. K. Tyagi, and V. S. Sisodia, "Long short-term memory (LSTM) networks for stock market prediction," *IEEE Access*, vol. 2025, no. Jan., pp. 565–570, 2025, doi: 10.1109/ICPCT64145.2025.10939100.
- [13] Y. Wang and T. Zhang, "Research on deep learning feature extraction algorithm for big data analysis," *IEEE Access*, vol. 2025, no. Jan., pp. 1265–1269, 2025, doi: 10.1109/ACCTCS66275.2025.00222.
- [14] C. Ni, "Comparative analysis of a hybrid CNN-LSTM-attention model and traditional time series approaches for financial forecasting," *IEEE Access*, vol. 2025, no. Jan., pp. 774–780, 2025, doi: 10.1109/CIPAE66821.2025.00137.
- [15] K. Mae and D. A. Kristiyanti, "Grid search vs Bayesian optimization for intensity scoring classification and channel recommendation prediction," *TELKOMNIKA*, vol. 23, no. 4, pp. 1020–1031, 2025, doi: 10.12928/TELKOMNIKA.v23i4.26341.
- [16] P. Vispute, J. Sujata, and N. A. Natraj, "Predicting stock prices using machine learning techniques: an analysis of historical market data," *Int. J. Innov. Mech. Ind. Appl.*, vol. 2023, no. Jan., pp. 725–733, 2023, doi: 10.1109/ICIMIA60377.2023.10426336.
- [17] L. Liu and Y.-W. Si, "1D convolutional neural networks for chart pattern classification in financial time series," *J. Supercomput.*, vol. 78, no. 12, pp. 14191–14214, 2022, doi: 10.1007/s11227-022-04431-5.
- [18] D. R. Macherla, U. Obinwanne, and W. Feng, "Intraday stock price prediction using machine learning: a case study on YFinance stock data," *Int. J. Softw. Eng. Artif. Intell.*, vol. 2025, no. Jan., pp. 1001–1006, 2025, doi: 10.1109/SNPD65828.2025.11254075.
- [19] R. H. Bemthuis, R. R. Govers, and A. Asadi, "A CRISP-DM-based methodology for assessing agent-based simulation models using process mining," *J. Simul.*, vol. 2025, no. Jan., pp. 1–12, 2025, doi: 10.1080/17477778.2025.2508245.

-
- [20] K. Müller, P. Kröckel, and F. Bodendorf, “Sustainable CRISP-DM extension for energy-aware AI development,” *Int. J. Inf. Syst.*, vol. 2023, no. Jan., pp. 1–10, 2023.
- [21] N. O. Romanov, D. A. Skvortsova, N. A. Filin, B. A. Shvaiko, and A. A. Kuznetsov, “Studying data normalization methods in the sustainable development indicators analysis,” *Int. J. Electr. Power Eng.*, vol. 2025, no. Jan., pp. 1–10, 2025, doi: 10.1109/REEPE63962.2025.10970921.
- [22] A. Khalifeh and S. Almeqdadi, “Machine learning-based remaining useful life predictions and its application on predictive maintenance,” *Int. J. Mechatronics Eng.*, vol. 2024, no. Jan., pp. 381–386, 2024, doi: 10.1109/REM63063.2024.10735638.
- [23] S. Kaur and R. Agrawal, “Convolutional networks,” *J. Cloud Secur. Deep Learn.*, vol. 2021, no. Jan., pp. 109–122, 2021, doi: 10.1002/9781119760542.ch8.
- [24] I. Ito, “Convolution using discrete cosine transforms for improving performance of convolutional neural networks,” *EURASIP J. Signal Process.*, vol. 2022, no. Jan., pp. 1462–1466, 2022, doi: 10.23919/eusipco55093.2022.9909831.
- [25] S. M. Al-Selwi, “RNN-LSTM: from applications to modeling techniques and beyond—systematic review,” *J. King Saud Univ. Comput. Inf. Sci.*, vol. 36, no. 5, pp. 1–12, 2024, doi: 10.1016/j.jksuci.2024.102068.
- [26] X. Xie, “A hybrid CNN-LSTM architecture for enhanced music genre classification,” *Int. J. Mach. Learn. Comput. Appl.*, vol. 2024, no. Jan., pp. 72–75, 2024, doi: 10.1109/ICMLCA63499.2024.10753693.
- [27] A. Liu, J. Li, and H. Ye, “A prediction model combining convolutional neural network and LSTM neural network,” *Int. J. Artif. Intell. Robot. Syst.*, vol. 2023, no. Jan., pp. 318–321, 2023, doi: 10.1109/AIARS59518.2023.00071.
- [28] G. M. Rajeshwari, N. Manasa, K. P. Rajimol, K. Naidu, A. Samal, and N. Garg, “Evaluation of machine and deep learning models for utility mining-based stock market price predictions,” *AIP Conf. Proc.*, vol. 2024, no. Jan., pp. 1–10, 2024, doi: 10.1063/5.0239965.