International Journal of Advances in Data and Information Systems
Vol. 7, No. 1, April 2026, pp. 146~159
ISSN: 2721-3056, DOI: 10.59395/ijadis.v7i1.1497

SARIMA-GARCH and LSTM Performance for Broiler Meat Price Forecasting: A Case Study in West Sumatra

Joshua Bryan Wijaya[1], Indah Dzulkharifah[2], Varel Geo Syah Putra[3] , Harits Abdurahman[4], Laily Nissa Atul Mualifah[5], Windi Pangesti[6]
[1,2,3,4,5,6]Study Program of Statistics and Data Science, Bogor Agricultural University, Indonesia

Article Info
Article history:
Received Dec 22, 2025
Revised Jan 28, 2025
Accepted Feb 09, 2026

Keywords:
SARIMA-GARCH
LSTM
Broiler Chicken Price
Deep learning
Hybrid

ABSTRACT
The price of broiler chicken meat in West Sumatra is characterized by strong seasonality and high volatility. As a primary source of animal protein and a key contributor to regional inflation, accurate forecasting of these price fluctuations is essential for economic stability and policymaking. This study aims to compare the forecasting performance of the SARIMA-GARCH hybrid model against the Long Short-Term Memory (LSTM) model. The dataset consists of 1,198 daily observations spanning from 15 July 2022 to 24 October 2025, sourced from the National Food Agency (Badan Pangan Nasional). The results demonstrate that the SARIMA-GARCH model outperforms the LSTM model in terms of point forecast accuracy, as evidenced by lower prediction error metrics. Furthermore, the hybrid model successfully satisfies the statistical diagnostic criteria for volatility modeling by effectively resolving ARCH effects, ensuring the statistical validity of the residuals. While the LSTM model produces smoother long-term forecasts, the SARIMA-GARCH model effectively captures daily price fluctuations and indicates a modest upward trend over the next 28 days. These findings suggest that SARIMA-GARCH provides a more realistic depiction of short-term price movements for this specific regional market, offering a localized framework for stakeholders in West Sumatra to anticipate future market changes and maintain price stability.

This is an open access article under the CC BY-SA license.

Corresponding Author:
Laily Nissa Atul Mualifah,
Study Program of Statistics and Data Science,
Bogor Agricultural University,
Raya Dramaga Street, IPB University Campus, Dramaga, Bogor 16680, West Java, Indonesia.
Email: lailyatul@apps.ipb.ac.id

1. INTRODUCTION

Classical approaches, notably the Box-Jenkins method, have long served as the standard for structured data forecasting. The Seasonal Autoregressive Integrated Moving Average (SARIMA) model is a prime example of this methodology, proving effective for modeling data that exhibits both trend and seasonal patterns[1], [2]. Conversely, recent advancements in computing have ushered in modern approaches, including various machine learning and deep learning techniques. Among these, the Long Short-Term Memory (LSTM) architecture is widely used for sequential data. As a type of Recurrent Neural Network (RNN), LSTM are specifically designed to overcome long-term dependency issues and can effectively capture highly complex non-linear patterns in the data [3].

A major challenge common to both models in practical applications is data volatility. Volatility is defined by rapid and unexpected fluctuations in value, a characteristic frequently observed in economic and commodity data [4]. The Generalized Autoregressive Conditional Heteroscedasticity (GARCH) method is one established technique used to address this issue[5], [6]. GARCH, introduced by Bollerslev in 1986, is a generalization of the Autoregressive Conditional Heteroscedasticity (ARCH) model, which was originally developed by Engle in 1982 [7].

In the context of commodity trading, food prices including those for broiler chicken meat are known to be highly volatile[8]. Price fluctuations for broiler chicken meat across various districts/cities in West Sumatra Province are a significant concern. Steep increases in food prices directly impact people's access to food [9]. Consequently, maintaining food price stability is a primary focus for the government in its efforts to ensure adequate food sufficiency for the population.

Several preceding studies have utilized the GARCH method and the SARIMA-GARCH hybrid model to manage the volatility of food commodity price data [10], [11]. Furthermore, explored an alternative approach using the ARIMAX-LSTM hybrid method to tackle data volatility. However, a clear research gap remains, there are currently no studies that directly compare classical and modern forecasting methods for the price of broiler chicken meat.

This study aims to compare the forecasting performance of the SARIMA-GARCH hybrid model and the LSTM model for predicting broiler chicken meat prices within the regional market of West Sumatra. Our primary focus is to evaluate the point forecast accuracy of both models while simultaneously assessing the hybrid model's capacity to statistically account for conditional heteroscedasticity a volatility characteristic inherent to this province's food commodities. The findings of this research are anticipated to provide regional authorities in West Sumatra. This will facilitate more precise strategic decision-making to maintain the stability and accessibility of community food, especially amidst provincial price fluctuations and specific regional market interventions.

2. RESEARCH METHODS

This research applies two forecasting methods for predicting broiler meat price, SARIMA-GARCH for model linear components and LSTM for capturing nonlinear patterns. The data analysis was conducted using Python version 3.10 through the VS Code development environment and Rstudio 4.5.0. The modelling workflow comprises data preprocessing, model construction, and model evaluation, as illustrated in Figure 1 and further detailed in the subsequent subsections.

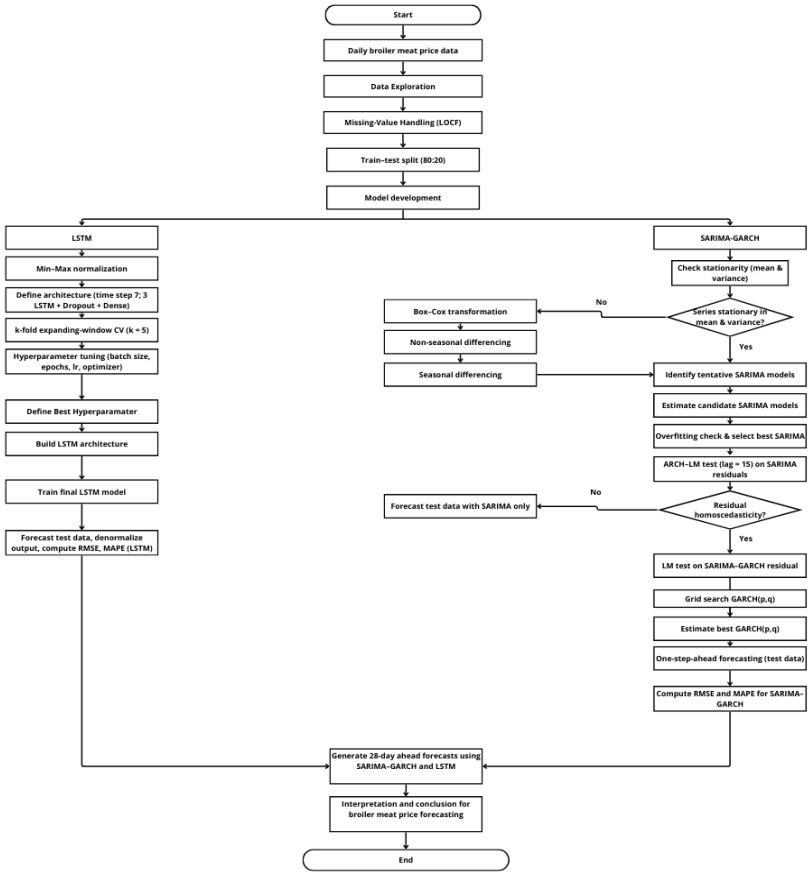


Figure 1. Research flowchart

2.1 Data Preprocessing

The data preprocessing stage is an essential step in model development. The first step performed was an exploratory data analysis to identify the presence of missing values. The dataset was found to contain missing entries. Therefore, a handling procedure was applied using the LOCF (Last Observation Carried Forward) method. LOCF is a simple imputation technique that replaces missing observations with the most recent previously available value in the time series [1]. LOCF is not inherently biased and has been shown to perform reasonably well when the proportion of missing data is small [12].

2.2 SARIMA-GARCH

SARIMA is an extension of the ARIMA model that enables the modelling and forecasting of time series data exhibiting seasonal patterns. This method is effective for predicting data characterized by recurring periodic fluctuations [13]. According to Notodiputro, Angraini, and Mualifah [14], the multiplicative form is appropriate when the seasonal variation increases or decreases in line with the level of the time series, because the seasonal effect operates multiplicatively on the underlying trend. Therefore, it's specification can be carried out using the same procedures as for the non-seasonal ARIMA model.

The development of the SARIMA model begins by examining stationarity in both the mean and variance of the training data using the Augmented Dickey-Fuller (ADF) test, the Autocorrelation Function (ACF) plot, and the Box Cox plot. If non stationarity in the mean is detected, differencing operations are applied to remove trends or seasonal structures. Conversely, when non stationarity appears in the variance, a Box Cox transformation can be implemented to stabilize the variability and improve model performance.

Candidate SARIMA models were identified using the ACF and PACF plots. Model selection was based on AIC, BIC, MAPE, and parameter significance. Model adequacy was then assessed using the Jarque–Bera test for normality, the t-test for zero-mean residuals, and the Box–Ljung test for autocorrelation and homoskedasticity. The model is considered valid when all tests fail to reject the null hypothesis, following the diagnostic approach used in Wulandari & Wahyuningsih [15].

However, SARIMA is not designed to capture time varying volatility (ARCH effects). Therefore, the Lagrange Multiplier (LM) test is applied to detect heteroskedasticity and the presence of ARCH effects identified up to lag 15. When ARCH effects are significant, incorporating ARCH components becomes necessary. To address this limitation, the GARCH model is introduced. The GARCH model is one of the most widely used approaches for handling violations of the homoskedasticity assumption, as it captures time varying variance in a time series [16], [17].

After specifying the tentative GARCH structure, parameters were estimated by maximum likelihood and candidate models were compared using information criteria and forecast metrics. The selected GARCH model was validated with the ARCH-LM test for remaining ARCH effects and inspection of standardized residuals, assumptions are considered met when the LM test is not significant and residuals show no remaining volatility clustering. Forecasts for the test set combine the SARIMA mean forecasts with volatility adjustments from the GARCH model using a one-step-ahead scheme [17].

2.3 LSTM

LSTM is a further development of the RNN architecture and is widely applied in time series forecasting[18]. It was design to overcome the limitations of standard RNNs in capturing long term dependencies through the incorporation of gating mechanisms [14]. Figure 2 illustrates the general structure of an LSTM network, which consists of an input later, a hidden layer with memory cells, and an output layer [19], [20]. Within the hidden layer, each memory cell contains several gates namely the input gate, forget gate, and output gate that regulate the flow of information and enable the model to effectively learn temporal patterns in sequential data [21].

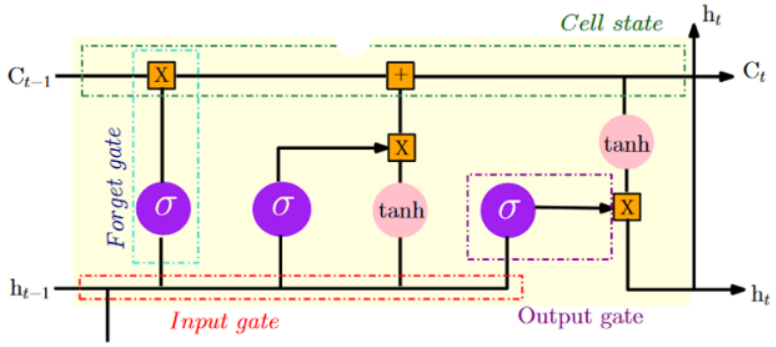


Figure 2. Illustration of the LSTM architecture

The LSTM modelling procedure begins with normalizing the training data using the Min–Max scaling method. This transformation rescales each observation into a fixed range [0,1]. Min-Max scaling method is effectively used on data that does not normally distributed[22]. The normalization is performed using Equation (1), where xi denotes the observed value at period i, min(x) is the minimum value in the dataset, and max(x) is the maximum value[23], [24].

$$z_i = \frac{x_i - min(x)}{max(x) - min(x)} \quad (1)$$

after normalization, model initialization is performed by specifying the hyperparameters used in the training process. The hyperparameters considered in this study are in Table 1.

Table 1. Hyperparamter uses for the k-fold cross validation

Parameter	Nilai
Epoch	100;300
Optimizer	Adam; RMSprop
Batch Size	16;32
Learning Size	0.001;0.01
Time Step	7

Model validation is conducted using k-fold cross validation and the training data are partitioned into five subsets (k=5). The expanding window approach is employed to construct these subsets, gradually increasing the size of the training window while sequentially updating the validation window, as illustrated in Figure 3. This method is well suited for time-series forecasting because it preserves temporal ordering during model evaluation [25].

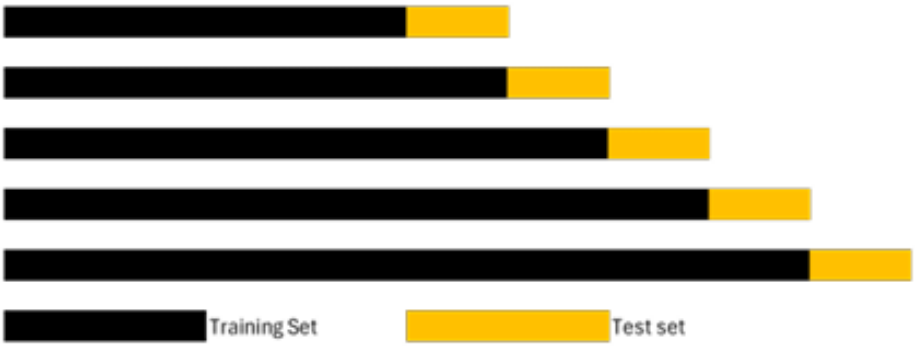


Figure 3. Illustration for the expanding window

After that, the model architecture consists of a time step of 7, three LSTM layers, three dropout layers, and two dense layers. The use of a 7-period time step is based on the finding that broiler chicken meat prices exhibit a seasonal pattern with a weekly cycle. Meanwhile, the layer configuration was designed to resemble the LSTM structure used in the prior study by Sakina [26]. The complete LSTM architecture used for both training and testing is presented in Table 13.

Tabel 2. LSTM architecture

Model LSTM	Struktur
LSTM	128, Return Sequences = True
Dropout	0,2
LSTM	64, Return Sequences = True
Dropout	0,1
LSTM	32, Return Sequences = False
Dropout	0,1
Dense	32
Dense	1

Then, the LSTM model trained to capture temporal dependencies through gating mechanisms that regulate the flow of information into and out of the memory cell. The LSTM cell begins by computing the input gate, which determines the extent to which latest information will be incorporated into the memory cell. The input gate value obtained using:

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

the input gate is followed by the generation of cell candidates, which represent potential new memory content obtained through the hyperbolic tangent activation function:

$$\hat{C}_t = \tanh(W_c x_t + W_{hc} h_{t-1} + b_c) \quad (3)$$

according to Sarangapani [27], the activation functions used in the LSTM cell are expressed as:

$$\tanh(x) = 2\sigma(2x) - 1 \quad (4)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

next, the forget gate determines the proportion of the previous cell state C_{t-1} that should be retained or discarded [28], computed as:

$$f_t = \sigma(W_f x_t + W_{hf} h_{t-1} + b_f) \quad (6)$$

the current cell state is then updated by combining the retained past information and the newly proposed cell candidate:

$$C_t = i_t \hat{C}_t + f_t C_{t-1} \quad (7)$$

the output gate subsequently regulates the information to be passed from the cell state to the hidden state:

$$o_t = W_o x_t + W_{ho} h_{t-1} + W_{co} C_t + b_o) \quad (8)$$

finally, the updated hidden state is computed using:

$$h_t = o_t \times \tanh(C_t) \quad (9)$$

this hidden state serves as the primary output of the LSTM cell and contributes to generating the model's predictions. Once the model trained, forecasting is performed using a one step ahead strategy to predict future values based on prior inputs. After obtaining the predictions, a denormalization process applied to convert the standardized values back to their original scale as given as:

$$x = z \times (max(x) - min(x)) + min(x) \quad (10)$$

2.4 Model Evaluation

Several error-based metrics were applied to measure model performance. The AIC and the Bayesian Information Criterion (BIC) serve as information-based measures that evaluate the goodness-of-fit of a model while simultaneously penalizing excessive complexity. It's defined as $AIC = -2\ln(L) + 2k$ and $BIC = -2\ln(L) + \ln(N)\,k$. Lower AIC and BIC values indicate models that achieve a better balance between accuracy and parsimony.

In addition to information criteria, Root Mean Square Error (RMSE) measures the magnitude of prediction errors based on the differences between actual and predicted values and is computed as $RMSE = \sqrt{\frac{\sum_{i=1}^{n}(D_i - Y_i)^2}{n}}$. Also Mean Absolute Percentage Error (MAPE) was employed as forecasting accuracy in percentage terms. MAPE is calculated as $MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|D_i - Y_i|}{D_i} \times 100\%$ [29]. Table 3 is criteria for interpreting MAPE values.

Table 3. Criteria MAPE

MAPE Values(%)	Criteria
< 10	Excelent
10–20	Good
20–50	Basic
> 50	Bad

2.5 Forecasting

Forecasting is a method used to predict future values based on historical and current data. Once the selected models from the SARIMA-GARCH and LSTM approaches have been identified, the forecasting stage is carried out. This process plays a crucial role in decision-making, particularly because many influencing factors are not directly observable at the time decisions are made. In this study, forecasts of broiler chicken meat prices are generated for the next 28 days.

3. RESULTS AND DISCUSSION

3.1 Exploratory Data Analysis and Data Preprocessing

This study uses secondary data obtained from the official website of Badan Pangan Nasional. The dataset contains daily broiler chicken prices in West Sumatra Province, recorded in Rupiah, from July 15, 2022, to October 24, 2025, totaling 1,198 observations. Missing values identified between December 24 and December 26, 2024. In this study, missing values occur over three consecutive periods, representing only 0.25% of the total observations. Moreover, price changes during the missing intervals are relatively modest. Under these conditions, the use of LOCF is considered appropriate and unlikely to introduce meaningful bias into the analysis. Figure 4 presents the time series before and after imputation.

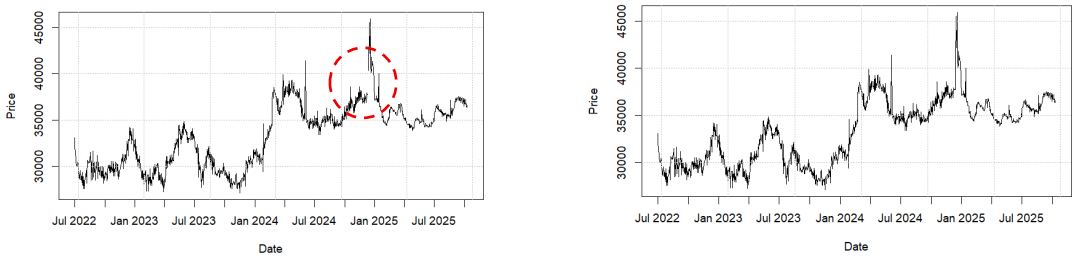


Figure 4. Plot time series,
(a) Before LOCF, (b) After LOCF

Figure 4 illustrates broiler chicken price fluctuations during the observation period, rising from IDR 28,000-33,000 in 2022–2023 to nearly IDR 45,000 in mid-2024 and early 2025, indicating higher variability and a possible seasonal pattern. This is consistent with reports from Badan Pusat Statistik West Sumatra [30], which reported that broiler chicken meat contributed 0.02% to inflation in the volatile food category in February 2024. Figure 5 further supports this seasonal indication through weekly (Monday–Sunday) price dynamics, where overlapping observations show noticeable day-to-day variation. This consistent weekly fluctuation justifies using a seven-day seasonal component (s = 7) in subsequent time series modeling.

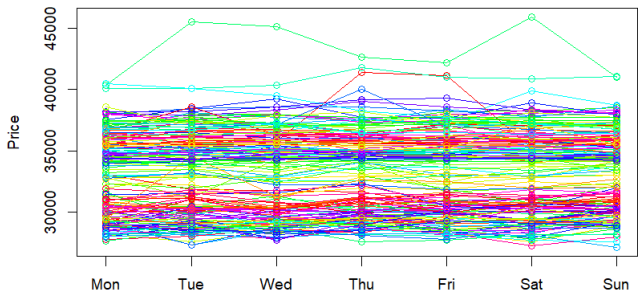


Figure 5. Seasonal plot

3.2 Model SARIMA-GARCH

The dataset was divided into 80% training data and 20% testing data. The training set covers the period from July 15, 2022, to February 26, 2025. While the testing spans February 27, 2025, to October 24, 2025, as shown in Figure 6.

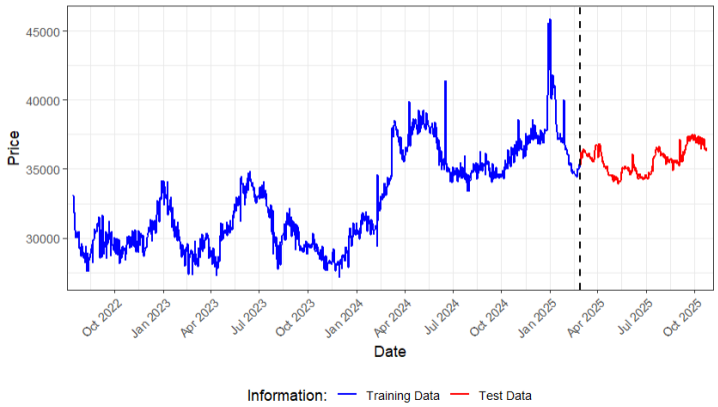


Figure 6. Data split plot

Based on the ACF plot in Figure 7, the slowly decaying autocorrelation pattern indicates that the series is non-stationary in its mean. The ADF test yields a p-value of 0.2343 (>0.05), indicating that the mean is non-stationary. For variance stationarity, Box–Cox parameter shows that the series is also non-stationary in its variance. This is supported by the 95% confidence interval of the estimated λ, which ranges from –1.75 to –0.65 and does not include 1, confirming non-stationary variance.

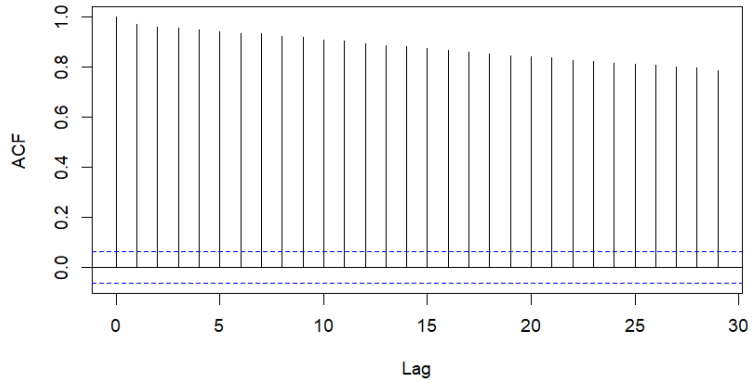


Figure 7. ACF plot before differencing

Based on these diagnostics, a Box–Cox transformation with an optimal λ = -1.19 was applied. The 95% confidence interval of the transformed series includes λ = 1, indicating variance stationarity after transformation. After variance stabilization, first-order differencing was applied to address non-stationary in the mean. Post-differencing, the ACF plot no longer exhibits displays a slowly decaying pattern, and the ADF test yields a p-value of 0.01, indicating that the series is stationary in the mean. Although the series becomes non-seasonally stationary after first differencing, the ACF plot (Figure 8) still shows recurring spikes at lag multiples of 7. This pattern indicates the presence of seasonal non-stationarity. Therefore, a seasonal differencing with period s = 7 is required before SARIMA model identification.

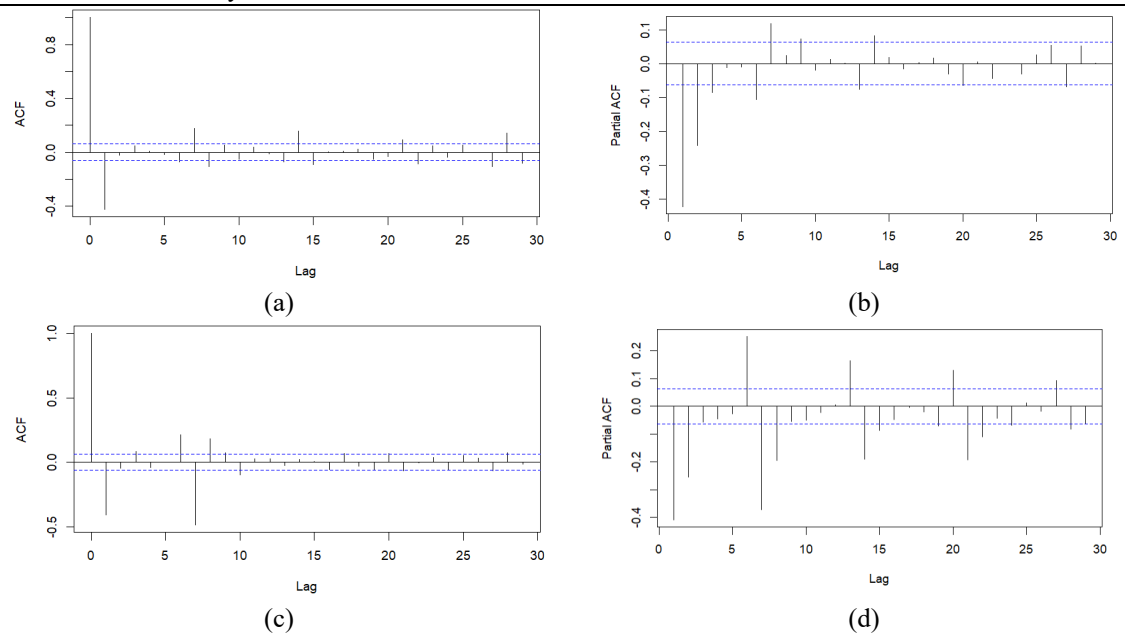


Figure 8. Plot before and after non seasonal and seasonal differencing,
(a) ACF plot before seasonal, (b) PACF before seasonal, (c) ACF after seasonal, (d) PACF after seasonal

ACF and PACF plots of the stationary series reveal ACF cut-offs at lag 1 (non-seasonal) and the first seasonal lag (s=7). Conversely, the PACF displays a non-seasonal cut-off at lag 2 and tails off at seasonal lags. Justifying the tentative SARIMA models in Table 4.

Table 4. Comparison of tentative models

Model	Parameter	Coefficient	P-value	AIC	BIC	MAPE
$SARIMA(0,1,1)(0,1,1)^7$	MA1	-0.5056	< 0.05	-27735.39	-27720.8	1.837
	SMA1	-0.9507	< 0.05			
$SARIMA(2,1,0)(0,1,1)^7$	AR1	-0.5029	< 0.05	-27730.18	-27710.75	1.824
	AR2	-0.2192	< 0.05			
	SMA1	-0.9518	< 0.05			
$SARIMA(2,1,1)(0,1,1)^7$	AR1	-0.1840	> 0.05	-27733.82	-27709.54	1.828
	AR2	-0.0875	> 0.05			
	MA1	-0.3373	< 0.05			
	SMA1	-0.9520	< 0.05			

The best model was selected by comparing the metrics evaluation criteria. Based on these results, the $SARIMA(0,1,1)(0,1,1)^7$ model was initially identified as the best candidate because it had the lowest AIC and BIC, a comparable MAPE to its alternatives, and all parameters were significant. An overfitting procedure was then applied by increasing each of the AR, MA, SAR, and SMA by one. The evaluation showed that the best-performing model was $SARIMA(0,1,1)(1,1,1)^7$, which produced an AIC of -27,740.68, a BIC of -27,721.25, and a MAPE of 1.828. The final model is expressed as:

$$(1 - 0.0919B^7)(1 - B)(1 - B^7)y_t = (1 - 0.5101B)(1 - 0.9596B^7)\varepsilon_t \quad (16)$$

After specifying the model, diagnostic checking was performed to assess the adequacy of the best model. Based on the diagnostic results, the assumptions of residual normality and homoscedasticity are not satisfied. The violation of the homoscedasticity assumption suggests the possible presence of ARCH effects in the residuals. To further investigate this indication, the ARCH LM test is conducted, where a p-value < 0.05 signifies statistically significant ARCH effects and therefore justifies the application of ARCH or GARCH-type volatility modeling. The test is performed across multiple lags to ensure a comprehensive assessment of short-term volatility clustering. The resulting p-values for each lag, summarized in Table 5, clearly demonstrate strong and persistent ARCH effects in the series.

Table 5. ARCH LM test results

Lag	P-value
1	2.6820×10^{-10}
2	3.8346×10^{-11}
3	1.9341×10^{-10}
⋮	⋮
15	3.1078×10^{-9}

Modeling the volatility begins with a grid search over multiple GARCH(p,q) specifications, with p and q ranging from 1 to 5. The results show that the GARCH(1,1) specification provides the best fit, achieving an AIC of –29.3281, a BIC of –29.3078 and a non-significant p-value of 0.7086. Following the selection of this model, the estimated parameter values of its volatility components are presented in Table 6. The selected GARCH(1,1) model was then hybrid with the SARIMA specification to form the integrated SARIMA-GARCH model. After estimating the integrated model, diagnostic using LM test show p-value of 0.06 (>0.05), confirming that the ARCH effect had been effectively resolved.

Table 6. Estimation of GARCH(1,1) component parameters

Parameter	Coefficient	P-value	Information
ω	0.0000	> 0.05	insignificant
α_1	0.1687	< 0.05	significant
β_1	0.8050	< 0.05	significant

The estimated GARCH(1,1) parameters in Table 6 demonstrate high volatility persistence, with $\alpha_1 + \beta_1 = 0.9737$. This value indicates that shocks to the price series have a long-lasting effect on future conditional variance, while satisfying the stationarity constraint ($\alpha_1 + \beta_1 < 1$) to ensure model stability. The statistical insignificance of the constant term (ω) suggests that baseline volatility is negligible, meaning the variance is primarily driven by past shocks and previous volatility. Consequently, the model maintains a stable long-run unconditional variance throughout the observation period.

3.3 Model LSTM

The LSTM model was trained using the optimal hyperparameters selected through time-series k-fold cross-validation with k=5. The expanding-window scheme was used to generate sequential train–test partitions. These partitions are summarized in Table 7.

Table 7. Data train and test period on each fold

Fold	Expanding Windows	
	Train	Test
1	15-July-2022 to 02-Feb-2023 (203)	03-Feb-2023 to 20-Aug-2023 (199)
2	15-July-2022 to 20-Aug-2023 (402)	21-Aug-2023 to 06-Mar-2024 (199)
3	15-July-2022 to 06-Mar-2024 (601)	07-Mar-2024 to 21-Sep-2024 (199)
4	15-July-2022 to 21-Sep-2024 (800)	22-Sep-2024 to 08-Apr-2025 (199)
5	15-July-2022 to 08-Apr-2025 (999)	09-Apr-2025 to 24-Oct-2025 (199)

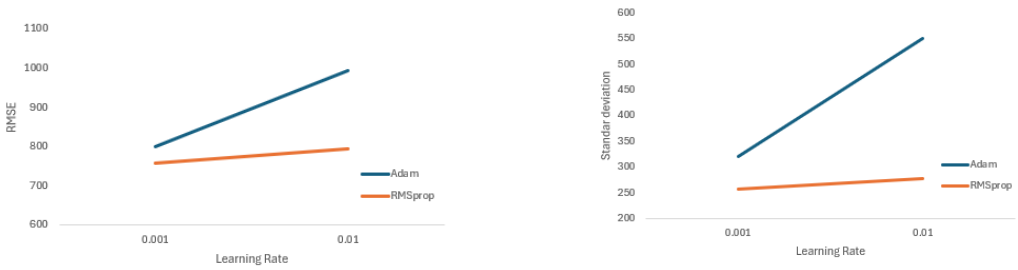
Based on the hyperparameter tuning results, the performance comparison across all scenarios is presented in Table 8. The evaluation indicates that each hyperparameter combination produces varying RMSE values. Through the time-series cross-validation using the expanding-window scheme, the best configuration was obtained with batch size = 16, epochs = 300, learning rate = 0.001, and the RMSprop optimizer, which yielded the lowest average test RMSE as shown in Table 8.

Table 8. Comparison of hyperparameter combinations across expanding-window scenarios

Hyperparameter				RMSE	
Batch Size	Epochs	Learning Rate	Optimizer	Mean	Standart Deviation
16	100	0.001	Adam	773.2052	291.7804
			RMSprop	768.0423	257.4640
		0.01	Adam	831.9612	217.5097
			RMSprop	783.2450	283.7912
	300	0.001	Adam	825.6473	360.2215
			RMSprop	731.0930	265.5601 *
		0.01	Adam	1364.8270	1111.5568
			RMSprop	787.9340	286.4250
32	100	0.001	Adam	817.5436	327.1750
			RMSprop	784.4407	246.6184
		0.01	Adam	867.9338	429.1328
			RMSprop	796.4368	251.3294
	300	0.001	Adam	787.4637	304.2678
			RMSprop	750.1505	256.5075
		0.01	Adam	910.7941	443.2753
			RMSprop	811.5617	287.3308

*The best hyperparameter combination

These findings are further supported by the visualization in Figure 9. Overall, RMSprop consistently achieves lower average RMSE and smaller variance compared to Adam, indicating more stable estimation performance. This observation aligns with Nurhambali, who reported that RMSprop offers greater stability in handling fluctuating gradients in time-series learning.



(a) (b)

Figure 9. Learning Rate Adam vs RMSprop,
(a) Mean RMSE, (b) Standart deviation for each parameter combination

Moreover, the patterns in Figure 9 reveal that larger *learning rate* values generally lead to higher RMSE and greater standard deviations, resulting in less accurate and less stable predictions. This is consistent with Cynthia[31] ,who found that excessively large learning rates can trigger extreme loss fluctuations and potentially cause divergence during training.

The training process of the LSTM model begins with the initialization of weight parameters for each gate, which are randomly generated by the TensorFlow library. Table 9 presents the total number of parameters involved in the model. The total of 128417 parameters is obtained by summing the contributions from the input dimensions, the number of LSTM units, the number of gates within each LSTM cell, and the output dimensions. A detailed breakdown of the parameter count for each layer is provided in Table 9.

Table 9. Total parameters in the LSTM model

Model LSTM	Jumlah Parameter
LSTM	66560
Dropout	0
LSTM	49408
Dropout	0
LSTM	12416
Dropout	0
Dense	32
Dense	1

Comparison between the training loss and validation loss across all epochs provided in Figure 10. The training loss demonstrates a consistent and stable downward trend, indicating that

the model effectively learns the underlying patterns from the training data. In contrast, the validation loss exhibits some fluctuations during the initial epochs; however, it progressively aligns with the decreasing trend of the training loss, albeit with slightly higher variability. The relatively small and non-diverging gap between the two curves suggests that the model does not suffer from overfitting or underfitting.

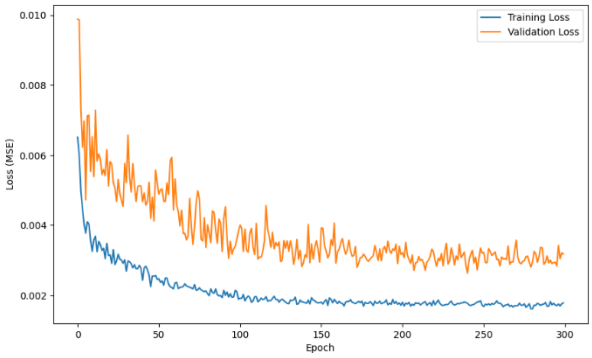


Figure 10. Comparison of training and validation loss

3.4 Comparison of SARIMA-GARCH and LSTM in the prediction of training data

Data partitioning was conducted during the initial stage of the analysis. Both optimized models were trained using the same training subset and subsequently evaluated on the identical test subset. The forecasting results produced by the SARIMA–GARCH and LSTM models for the test data are presented in Figure 11.

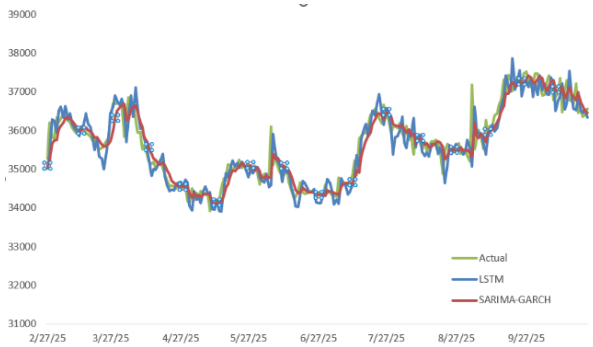


Figure 11. Comparison of test data predictions using SARIMA–GARCH and LSTM

As shown in Figure 11, the predictions generated by the SARIMA–GARCH model exhibit greater fluctuations compared to LSTM model. This is attributed to the GARCH component, which is specifically designed to model volatility in time-series data. Furthermore, SARIMA–GARCH produces forecasts based on conditional mean estimates, whereas LSTM performs point-to-point predictions.

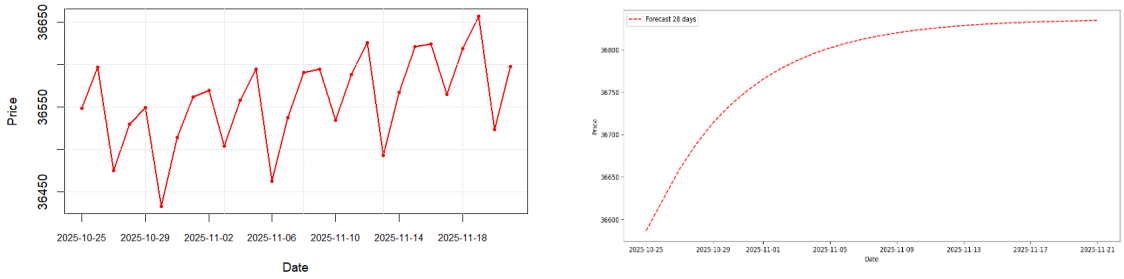
The performance evaluation of both models on the test data is summarized in Table 10. The results indicate that SARIMA–GARCH achieves lower RMSE and MAPE values compared to LSTM. This implies that the hybrid SARIMA–GARCH model yields smaller prediction errors on average both in absolute scale and percentage terms than the deep-learning-based LSTM model. Overall, the hybrid approach demonstrates higher accuracy and effectiveness in handling time-series data characterized by strong seasonality and pronounced volatility.

Table 10. Comparison of forecasting accuracy between SARIMA–GARCH and LSTM

Model	RMSE	MAPE
Hybrid SARIMA-GARCH	301.8914	0.59% *
Deep learning LSTM	366.6246	0.74%

3.5 Forecast results from the SARIMA-GARCH and LSTM models for the next 28 days

The resulting models were used to forecast chicken meat prices in West Sumatra Province for the next 28 days. SARIMA-GARCH model (Figure 12a) produces forecasts that fluctuate from day to day, reflecting its sensitivity to short-term volatility and its ability to incorporate conditional variance through the GARCH component. This results in a prediction pattern that remains dynamic around recent price levels. In contrast, the LSTM model (Figure 12b) generates a much smoother forecast trajectory, showing a gradual increase before stabilizing toward the end of the forecast horizon. This smoother curve illustrates LSTM's tendency to dampening short-run volatility. Overall, SARIMA-GARCH better represents short-term volatility.



(a) (b)

Figure 12. Broiler Meat price forecast for the next 28 days,
(a) SARIMA-GARCH (b) LSTM

4. CONCLUSION

The SARIMA–GARCH hybrid model outperforms the LSTM model, as indicated by its lower MAPE and RMSE values on the test data. Also, when producing a 28-day forecast beyond the training and test samples, the SARIMA-GARCH model appeared more responsive to price volatility. Therefore, SARIMA–GARCH is the most suitable method for forecasting chicken meat prices. Moreover, the SARIMA-GARCH results show that while short-term volatility persists, broiler chicken meat prices are expected to gradually increase throughout the 28-day forecast period.

REFERENCES

[1] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. Melbourne, Australia: Otexts, 2021. Accessed: Nov. 29, 2025. [Online]. Available: https://otexts.com/fpp3/

[2] T. Bunnag, "Forecasting PM10 Caused by Bangkok's Leading Greenhouse Gas Emission Using the SARIMA and SARIMA-GARCH Model," *International Journal of Energy Economics and Policy*, vol. 14, no. 1, pp. 418–426, Jan. 2024, doi: 10.32479/ijeep.15275.

[3] S. Kamolthip, "Macroeconomic forecasting with LSTM and mixed frequency time series data," Sep. 2021, [Online]. Available: http://arxiv.org/abs/2109.13777

[4] L. N. A. Mualifah, A. M. Soleh, and K. A. Notodiputro, "Comparison of GARCH, LSTM, and Hybrid GARCH-LSTM Models for Analyzing Data Volatility," *International Journal of Advances in Soft Computing and its Applications*, vol. 16, no. 2, pp. 150–165, 2024, doi: 10.15849/IJASCA.240730.10.

[5] J. K. Mutinda and A. K. Langat, "Stock price prediction using combined GARCH-AI models," *Sci. Afr.*, vol. 26, Dec. 2024, doi: 10.1016/j.sciaf.2024.e02374.

[6] L. Budianti, M. Yasyfi Avicenna, A. Kusuma Putri, and G. Darmawan, "Pemodelan SARIMA dengan Pendekatan ARCH/GARCH untuk Meramalkan Penjualan Ritel Barang Elektronik," *INNOVATIVE: Journal Of Social Science Research*, vol. 4, pp. 1037–1051, 2024.

[7] T. Bollerslev, "The Story of GARCH: A Personal Odyssey," 2022.

[8] K. S. Sai, K. Suhasisni, and M. A. Baba, "Comparative Analysis of Hybrid SARIMA-GARCH and Neural Networks for Chilli Price Forecasting," *Indian Journal of Agricultural Economics*, vol. 80, no. 2, pp. 370–385, 2025, doi: 10.63040/25827510.2025.02.007.

[9] M. Kalkuhi, J. von Braun, and M. Torero, *MatthiassKalkuhll· JoachimmvonnBraun MaximooTorero Editors Food Price Volatility and Its Implications for Food Security and Policy*. 2016. doi: DOI 10.1007/978-3-319-28201-5.

[10] P. Ghiyal and J. Kumar, "International Journal of Statistics and Applied Mathematics 2024; 9(2): 101-107 Use of hybrid SARIMA-GARCH model for predicting the prices of agricultural product in Haryana," *Maths*, vol. 9, no. 2, pp. 101–107, 2024, [Online]. Available: https://www.mathsjournal.com

[11] H. Hasmon, F. Firmansyah, N. Idris, and F. Hoesni, "Volatilitas Harga Daging Ayam Broiler di Tingkat Pedagang Pengecer pada Berbagai Kabupaten/Kota di Provinsi Jambi," *Ekonomis: Journal of Economics and Business*, vol. 8, no. 2, p. 1434, Sep. 2024, doi: 10.33087/ekonomis.v8i2.1824.

[12] N. Niako, J. D. Melgarejo, G. E. Maestre, and K. P. Vatcheva, "Effects of missing data imputation methods on univariate blood pressure time series data analysis and forecasting with ARIMA and LSTM," *BMC Med. Res. Methodol.*, vol. 24, no. 1, Dec. 2024, doi: 10.1186/s12874-024-02448-3.

[13] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis-Forecasting and Control*, 3rd ed. Upper Saddle River, New Jersey, USA: Prentice-Hall, Inc, 1994.

[14] K. A. Notodiputro, Y. Angraini, and L. N. A. Mualifah, *Analisis Data Deret Waktu dengan Python Pendekatan Box-Jenkins dan Machine Learning*, 1st ed. Bogor: PT Penerbit IPB Press, 2025.

[15] S. Wulandari and T. Wahyuningsih, "PERAMALAN JUMLAH TAMU THE AMRANI SYARIAH HOTEL MENGGUNAKAN MODEL ARIMA," *Majamath: Jurnal Matematika dan Pendidikan Matematika*, vol. 7, Mar. 2024.

[16] F. Merabet, H. Zeghdoudi, R. H. Yahia, and I. Saba, "MODELLING OF OIL PRICE VOLATILITY USING ARIMA-GARCH MODELS F. Merabet, H. Zeghdoudi, R. H Yahia, and I. Saba," *Advances in Mathematics: Scientific Journal*, vol. 10, no. 5, pp. 2361–2380, May 2021, doi: 10.37418/amsj.10.5.6.

[17] R. S. Chadha, Jugesh, S. Parveen, and J. Singh, "Fuel Sales Forecasting with SARIMA-GARCH and Rolling Window," *Journal of Soft Computing Paradigm*, vol. 5, no. 3, pp. 310–326, Sep. 2023, doi: 10.36548/jscp.2023.3.007.

[18] M. R. Nurhambali, Y. Angraini, and A. Fitrianto, "Implementation of Long Short-Term Memory for Gold Prices Forecasting," *Malaysian Journal of Mathematical Sciences*, vol. 18, no. 2, pp. 399–422, 2024, doi: 10.47836/mjms.18.2.11.

[19] N. Kalchbrenner, I. Danihelka, and A. Graves, "Grid Long Short-Term Memory," Jan. 2016, [Online]. Available: http://arxiv.org/abs/1507.01526

[20] W. A. Pratiwi, I. M. Sumertajaya, and K. A. Notodiputro, "Stacking Ensemble RNN-LSTM Models for Forecasting the IDR/USD Exchange Rate with Nonlinear Volatility," *Jurnal Teknik Informatika (Jutif)*, vol. 6, no. 4, pp. 2331–2347, Aug. 2025, doi: 10.52436/1.jutif.2025.6.4.5057.

[21] X. Song, L. Deng, H. Wang, Y. Zhang, Y. He, and W. Cao, "Deep learning-based time series forecasting," *Artif. Intell. Rev.*, vol. 58, no. 1, Jan. 2025, doi: 10.1007/s10462-024-10989-8.

[22] S. Sinsomboonthong, "Performance Comparison of New Adjusted Min-Max with Decimal Scaling and Statistical Column Normalization Methods for Artificial Neural Network Classification," *Int. J. Math. Math. Sci.*, vol. 2022, 2022, doi: 10.1155/2022/3584406.

[23] Y. S. Kim, M. K. Kim, N. Fu, J. Liu, J. Wang, and J. Srebric, "Investigating the impact of data normalization methods on predicting electricity consumption in a building using different artificial neural network models," *Sustain. Cities Soc.*, vol. 118, Jan. 2025, doi: 10.1016/j.scs.2024.105570.

[24] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Deep Adaptive Input Normalization for Time Series Forecasting," Sep. 2019, [Online]. Available: http://arxiv.org/abs/1902.07892

[25] A. Vamsikrishna and E. V. Gijo, "New Techniques to Perform Cross-Validation for Time Series Models," *Operations Research Forum*, vol. 5, no. 2, p. 51, Jun. 2024, doi: 10.1007/s43069-024-00334-8.

[26] D. Febri Sakina, A. Muhaimin, J. Rungkut Madya, and E. Java, "Application of ARIMAX-LSTM Model in Forecasting the Price of Broiler Chicken in Central Java Universitas Pembangunan 'Veteran' Jawa Timur, Indonesia," 2025.

[27] J. Sarangapani, *Neural Network Control of Nonlinear Discrete-Time Systems*. Manchester, United Kingdom: CRC Press, 2006.

[28] P. Goyal, S. Pandey, and K. Jain, *Deep Learning for Natural Language Processing*. Berkeley, CA: Apress, 2018. doi: 10.1007/978-1-4842-3685-7.

[29] C. Tofallis, "A better measure of relative prediction accuracy for model selection and model estimation," *Journal of the Operational Research Society*, vol. 66, no. 8, pp. 1352–1362, Aug. 2015, doi: 10.1057/jors.2014.103.

[30] Badan Pusat Statistik, "Berita Resmi Statistik," Jakarta, 2024. Accessed: Dec. 04, 2025. [Online]. Available: https://bps.go.id/pressrelease.html

[31] E. P. Cynthia, A. H. Saeed, M. Eka, and F. Nursalisah, "Pengaruh Parameter Learning Rate terhadap Konvergensi Model Neural Network dalam Proses Pelatihan," 2025. [Online]. Available: https://journals.raskhamedia.or.id/index.php/juiktiDOI:https://doi.org/99.9999/juikti.vxix.xxxx