Vol. 7 No. 1, 2026, pp. 46-58 **Jurnal Pepadun**

Comparison of ARIMA and LSTM Models for Regional Export Values in Lampung Province

[*1]Rian Kurnia, [2]Vina Nurmadani, and [3]Indah Suciati

[1,2,3]Department of Data Science, Insitut Teknologi Sumatera, Jalan Terusan Ryacudu, Lampung Selatan, Indonesia
e-mail: [*1]**rian.kurnia@sd.itera.ac.id**, [2]vina.nurmadani@sd.itera.ac.id , [3]indah.suciati@sd.itera.ac.id

Abstract *– Accurate forecasting of regional export values is critical for effective macroeconomic planning. However, these indicators often exhibit complex volatility and structural shocks that challenge traditional frameworks. This study compares the forecasting performance of Autoregressive Integrated Moving Average (ARIMA) method against the machine learning architecture Long Short-Term Memory (LSTM), utilizing the monthly export values of Lampung Province. Data from January 2015 to December 2024 were partitioned into a training set (2015-2022) and testing set (2023-2024). For the linear approach, following Box-Cox transformation and first-order differencing, an ARIMA(1,1,0)(1,0,1)[12] model was fitted to the data based on Akaike Information Criterion (AIC) with comparison to other models of ARIMA. Simultaneously, an LSTM network was constructed using a 12-month lookback window and Min-Max scaling. The results indicate that the optimized ARIMA model achieved a lower Root Mean Squared Error (RMSE) of 94,030,344 compared to the LSTM network of 395,566,847 during the 24 months testing window. The ARIMA model effectively captured the underlying linear trends and stable annual seasonality without overfitting the training data. The study concludes that for moderately sized time series, ARIMA remains highly robust and can outperform complex machine learning architectures. Consequently, while neural networks offer advanced capabilities, classical frameworks should remain a primary tool for establishing baseline indicators in regional forecasting.*

Keywords: *Time Series Forecasting; ARIMA; LSTM; Export Value.*

1. INTRODUCTION

International trade, particularly export activity, serves as a fundamental pillar for regional and national economic growth. The foundational theories of regional economic development have long established that the growth of the export sector stimulates domestic industries, generates employment, and significantly increases a region's gross domestic product [1]. This remains a critical sector of modern development economics, demonstrating that consistent export activities provide the important foreign exchange necessary to finance infrastructure and technological advancement in developing nations [2]. Consequently, understanding and anticipating the trajectory of export values is very important for policymakers to maintain economic stability.

In the context of Indonesia, Lampung Province represents a critical region in the national export chain. Historically, Lampung has established itself as a primary center for natural resource commodities such as pepper, coffee, and rubber [3]. Regional economic theories suggest that infrastructure development and robust export structure in areas like Lampung are essential for national development [4]. Therefore, accurately forecasting Lampung's monthly export values is mandatory requirement for regional macroeconomic planning, resource allocation, and market strategies. Forecasting macroeconomic indicators presents significant analytical challenges. Time series data for regional exports are inherently volatile, characterized by seasonal cycles, long-term trends, and sudden changes driven by global economic shocks [5]. Specifically, Lampung's export values show a strong linear trend and stable annual seasonality, yet they remain susceptible to non-linear behavior caused by global market shocks. Identifying the underlying process requires robust statistical techniques that can separate true economic signals from random market noise.

The Autoregressive Integrated Moving Average (ARIMA) model has served as the gold standard for univariate time series forecasting. The methodology excels at capturing linear dependencies, stabilizing non-stationary data through differencing, and mapping seasonal fluctuations using historical lags [6]. ARIMA has been

© 2026 The Authors | doi: 10.23960/pepadun.v7i3.342

extensively utilized in applied econometrics to model economic indicators, including gross domestic product, inflation, and trade volumes [7]. Despite its robustness, classical ARIMA models often struggle to capture highly complex, non-linear relationships and are particularly vulnerable to sudden, extreme shocks.

To address the limitations of ARIMA, deep learning architectures have recently emerged as powerful alternatives in economic forecasting. Specifically, the Long Short-Term Memory (LSTM) network, a specialized variant of Recurrent Neural Networks (RNN) developed by Hochreiter and Schmidhuber [8], was designed explicitly to process sequential data. By utilizing a system of internal gate mechanisms, LSTMs can retain long-term memory of temporal dependencies while actively forgetting irrelevant noise [9]. This capability has made LSTM highly effective in forecasting complex, non-linear financial and economic time series that do not satisfy traditional assumptions [10], [11].

Several empirical studies have demonstrated that LSTMs significantly outperform ARIMA by drastically reducing error rates when modeling highly complex, high-frequency datasets [12], [13], [14]. However, deep learning architectures are inherently data-hungry and prone to overfitting, often resulting in traditional statistical models like ARIMA outperforming sophisticated neural networks when applied to smaller, low-frequency datasets [15], [16], [17]. In the context of Indonesian economic forecasting, researchers have begun exploring both methods with mixed results. While some studies advocate for hybrid LSTM approaches to capture non-linear export components[18], others have found that classical ARIMA models continue to yield lower forecasting errors than LSTM when predicting regional supply chain and commercial activities limited by sample size [19].

This study aims to comparatively evaluate the forecasting performance of the traditional ARIMA model against the deep learning LSTM architecture using the monthly export values of Lampung Province from 2015 to 2024. By employing recursive forecasting, this research will determine whether the LSTM or ARIMA provides superior predictive accuracy for moderately sized regional economic datasets. The performance of both models will be objectively evaluated using the Root Mean Squared Error (RMSE) metric, providing regional policymakers with recommendation for future macroeconomic modeling.

While alternative methods such as SARIMAX [20], Prophet [21], GRU [14], or Hybrid models exist [18], this study intentionally focused on a direct comparison between a classical baseline model of ARIMA and fundamental deep learning architecture of LSTM. This boundary is essential to evaluate the capabilities of both frameworks without the bias introduced by exogeneous variables or complexity of hybrid frameworks. Furthermore, this research implements a recursive forecasting strategy. Unlike direct forecasting, which models each future data point independently, recursive forecasting uses the predicted value from the previous step as an input for the subsequent step. This strategy is scientifically chosen as it more accurately represents real-world condition where future actual data is unavailable. However, this approach is highly susceptible to error accumulation at each step.

Although numerous studies have evaluated ARIMA and LSTM, a research gap remains: previous literature has not addressed how the accumulation of error in recursive forecasting differs between ARIMA compared to LSTM in the case of Lampung's export data. Therefore, this study aims to fill this gap by investigating and comparing the ARIMA and LSTM performance during the recursive forecasting process for export values in Lampung Province.

2. RESEARCH METHODOLOGY

2.1. Data

The analysis in this study utilizes secondary time series data obtained from official database of Badan Pusat Statistik Indonesia [22]. The dataset comprises the total monthly export values originating from Lampung Province, measured in US Dollar (US$). The observation period spans exactly ten years, from January 2015 to December 2024, yielding a total of 120 consecutive monthly data points.

 | doi: 10.23960/pepadun.v7i3.342

Table 1. Data description.

Data	**Description**	**Timeframe**	**Observations (n)**
Dependent variable (Y_t)	Total monthly export value	25.00	30
Training set	Sample data for model estimation	75.15	10
Testing set	Data for forecasting validation	44.75	50

As shown in Table 1, for the purpose of model estimation and validation, the dataset is strictly partitioned into two distinct subsets: a training set and a testing set. The training set accounts for 80% of the data (96 months) and is utilized exclusively for identifying the ARIMA parameters and optimizing the LSTM network weights. The remaining 20% (24 months) serves as the benchmark for evaluating the predictive accuracy of both forecasting architectures.

2.2. Stationarity

A fundamental prerequisite for building ARIMA model is the assumption of stationarity. A time series is considered strictly stationary if its mean and variance remain constant over time. Because macroeconomic indicators like export values inherently exhibit growth, increasing volatility, and trends, the raw data must be mathematically transformed and tested prior to model estimation.

The first step in data preprocessing for ARIMA modeling is to address the non-stationarity in variance (heteroscedasticity). When the variance of a time series changes over time it violates the foundational assumptions of classical forecasting. To stabilize the variance, this study employs the Box-Cox transformation [23]. The Box-Cox method evaluates a range of power transformations governed by the parameter λ. The goal of Box-Cox transformation is to find the optimal value of λ that normalizes the data and stabilizes the variance across the entire series. If the calculated value of λ is approximately 1, it indicates that the original data is already stable in variance. Conversely, if $\lambda = 0$, a natural logarithmic transformation is optimal, while other fractional values dictate specific power transformations as in Equation 1:

$$y_i(\lambda) = \frac{y_i^{\lambda} - 1}{\lambda}, \lambda \neq 0. \tag{1}$$

Once the variance is stabilized, the series must be evaluated for stationarity in the mean. A series with a non-constant mean will produce autoregressive property. To objectively determine the stationarity, the Augmented Dickey-Fuller (ADF) test is applied [24]. The ADF test operates under the following hypotheses: (1) Null Hypothesis (H_0): The time series data is non-stationary in the mean; and (2) Alternative Hypothesis (H_1): The time series data is stationary.

2.3. Autocorrelation Analysis

Following the stabilization of variance and mean, the next critical phase is the identification of the structural parameters for the ARIMA model. This is achieved through the diagnostic examination of the Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF) plots. These functions quantify the linear dependence of the time series across different temporal lags, allowing researchers to isolate the specific autoregressive (AR) and moving average (MA) terms.

The Autocorrelation Function measures the gross linear correlation between the current observation of a time series and its historical values at prior lags. The ACF plot is primarily utilized to identify the order of the non-seasonal moving average component and the seasonal moving average component. If the ACF plot exhibits a significant spike at specific lags before sharply cutting off to zero, it indicates that the series is influenced by unobserved shocks from those corresponding past periods. Furthermore, significant spikes at regular, repeating intervals provide visual confirmation of annual seasonality within the variable.

While the ACF measures the total correlation, the Partial Autocorrelation Function isolates the direct correlation between current observation and a specific lagged value by mathematically removing the influence of all intermediate lags. The PACF is the primary diagnostic tool for determining the non-seasonal autoregressive order (p) and the seasonal autoregressive order (P) [25].

2.4. ARIMA

The ARIMA architecture mathematically maps the future value of a variable as a linear combination of its own past values (AR terms), past forecast errors (MA terms), and the differencing required to achieve stationarity. Differencing is required to achieve stationarity, capturing both the non-seasonal and seasonal dynamics. To concisely express the mathematical formulation of the ARIMA model, the backshift operator B is utilized, where $BY_t = Y_{t-1}$ and $B^sY_t = Y_{t-s}$. The general equation $ARIMA(p,d,q)(P,D,Q)[s]$ model is defined as:

$$\Phi_P(B^s)\phi_p(B)(1-B)^d(1-B^s)^D Y_t = \Theta_Q(B^s)\theta_q(B)\epsilon_t \tag{2}$$

Where:

1. Y_t represents the Box-Cox transformed value at time t.
2. ϵ_t represents the error term at time t.
3. s represents the seasonal length.
4. d and D are the orders of non-seasonal and seasonal differencing.
5. $\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$ is the non-seasonal autoregressive polynomial.
6. $\theta_q(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q$ is the non-seasonal moving average polynomial.
7. $\Phi_P(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \cdots - \Phi_P B^{Ps}$ is the seasonal autoregressive polynomial.
8. $\Theta_Q(B^s) = 1 + \Theta_1 B^s + \Theta_2 B^{2s} + \cdots + \Theta_Q B^{Qs}$ is the seasonal moving average polynomial.

Following the identification of the candidate $p, q, P,$ and Q orders via the ACF and PACF plots, the model coefficients are estimated using the Maximum Likelihood Estimation (MLE) method. MLE determines the coefficients that maximize the probability of obtaining the observed data [26].

2.5. Akaike Information Criterion

To objectively select the single most optimal configuration among identified candidate models of ARIMA, the Akaike Information Criterion (AIC) is employed. The AIC evaluates the trade-off between the maximum likelihood and the complexity of the model (number of estimated parameters). The model yielding the lowest AIC score is mathematically selected as the most optimal model. The AIC values are calculated as follows:

$$AIC = -2(\ln L - K). \tag{3}$$

Where $\ln L$ is the maximum likelihood of the model and K is the number of parameters for the model [27].

2.6. Ljung-Box Test

In ARIMA modeling, a model is appropriate if the forecasting errors must closely resemble a white noise process. The residuals must be independently and identically distributed with a mean of zero and a constant variance. Most importantly, there must be no correlation between the residuals across different time lags.

To objectively test the white noise assumption, this study employs the Ljung-Box test. The Ljung-Box test comprehensively examines the overall randomness of the residuals based on a number of cumulative lags [28]. The formal hypotheses for the Ljung-Box test are defined as follows: (1) Null Hypothesis (H_0): The residuals are independently distributed; and (2) Alternative Hypothesis (H_1): The residuals are not independently distributed.

2.7. Data Standardization

Neural networks, particularly Long Short-Term Memory (LSTM) models, are sensitive to the absolute scale of their input features. If extreme raw values are introduced to the functions inside LSTM, this can trigger the vanishing gradient problem, where the neural network algorithm fails to update internal weights. This can lead to the model not being able to learn from the temporal dynamics of the time series.

© 2026 The Authors | doi: 10.23960/pepadun.v7i3.342

To guarantee algorithmic convergence and computational efficiency, the data must be normalized before it fed into the neural network. This study employs Min-Max scaling to transform all observations into bounded range of [0,1]. The standardization is mathematically defined by the following Equation 4:

$$X'_t = \frac{X_t - X_{\min}}{X_{\max} - X_{\min}}. \tag{4}$$

Where X_t is the original value, $X_{\min}$ is the minimum recorded observation, and $X_{\max}$ is the maximum recorded observation [29].

2.8. Long Short-Term Memory (LSTM)

This study employs a deep learning approach using Long Short-Term Memory (LSTM) networks. The fundamental innovation is the introduction of a memory cell state (c_t) capable of carrying relevant information through long temporal sequences. The diagram in Figure 1 provides a visual representation of LSTM internal structure.

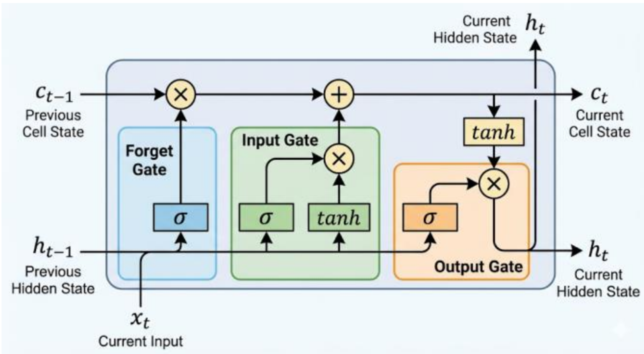

Figure 1. LSTM network structure.

The forget gate determines which information from the previous cell state is no longer relevant and should be discarded. It analyzes h_{t-1} and x_t, outputting a value between 0 and 1 for each component in the previous cell state c_{t-1}. The formulation is given by the following Equation 5:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{5}$$

Where W_f is the weight matrix, b_f is the bias vector, and $[h_{t-1}, x_t]$ is the concatenation of the previous hidden state and the current input.

The input gate decides what new information will be stored in the cell state. It involves two steps: the input gate (i_t) decides which values to update using a sigmoid function and tangent hyperbolic function creates a vector of new candidate values ($\hat{c}_t$). The formulation is given below:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{6}$$
$$\hat{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Where W_i, W_c are weight matrices and b_i, b_c are bias vectors. The old state c_{t-1} is now updated to the new cell state c_t using the below formula:

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t. \tag{8}$$

The output gate controls what information from the current cell state will be output to the hidden state. This gate generates the hidden state (h_t).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o \tag{9}$$
$$h_t = o_t * \tanh(c_t).$$

Where W_o and b_o are the weight matrix and bias vector for the output gate [30].

2.9. Cross-Validation and Hyperparameter Optimization

To ensure the LSTM network was optimized and to justify the selected parameters, a hyperparameter grid search was conducted. In standard machine learning, hyperparameters are typically optimized using k-fold cross-validation. However, applying standard k-fold cross-validation to time series data is fundamentally flawed since randomly shuffling observations destroys the temporal sequence [31].

To preserve the temporal sequence property, this study employed specific Time Series Cross-Validation, namely Expanding Window (or walk-forward) approach. Under this methodology, the 2015-2022 training partition was divided into three sequential folds. For the first fold, the network was trained on an 2015-2018 data and validated on the subsequent 2019 data. The second fold used 2015-2019 as training data and the year 2020 as validation data. The final fold incorporates 2015-2020 data as training data and 2021 data as validation data. Using this expanding window framework, a grid search was executed to evaluate combinations of three critical LSTM hyperparameters: (1) Number of layers between 1 or 2 layers of LSTM network; (2) Number of neurons for each layer can be 10,20,30, and 50 neurons; (3) Training duration (epochs) with fixed training durations of 20, 50, or 100 epochs.

2.10. RMSE

To objectively determine the superior forecasting architecture between the SARIMA model and LSTM network, a standardized evaluation metric must be applied to testing data. This study relies on the Root Mean Squared Error (RMSE) as the benchmark. The formulation is mathematically defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(\hat{Y}_t - Y_t\right)^2} \ . \tag{10}$$

Where n represents the total number of observations, Y_t represents the actual value at time t, and $\hat{Y}_t$ represents the model's forecasted value at time t [32].

3. RESULTS AND DISCUSSION

3.1. Descriptive Analysis

A visual and descriptive analysis of the raw time series data is conducted to identify underlying patterns. The dataset comprises 120 monthly observations of Lampung Province's total export values spanning a complete decade from January 2015 to December 2024. The time series is illustrated in Figure 2.

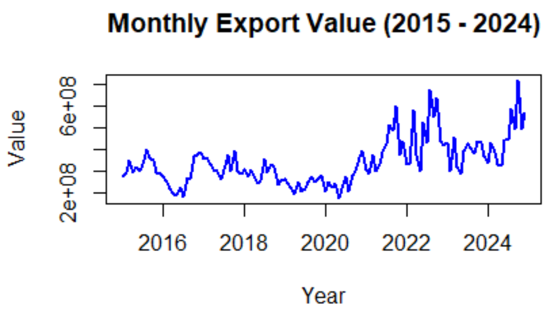

Figure 2. Monthly export value of Lampung Province.

A visual inspection of the complete time series plot reveals distinct macroeconomic phases over the ten-year period. From 2015 through 2022, the export values were relatively stable and repeating seasonal fluctuations. However, the values shift significantly from 2022 to 2024. During this period, export values violently fluctuate and show macroeconomic shocks.

 | doi: 10.23960/pepadun.v7i3.342

To properly evaluate the forecasting under SARIMA and LSTM models, the data is partitioned into two distinct temporal parts: the training set and testing set. The training data contains 96 months from January 2015 through December 2022, accounted for 80% of the whole data. The testing data isolates the final 24-months period from January 2023 to December 2024. The plot of the training and testing data is given in Figure 3.

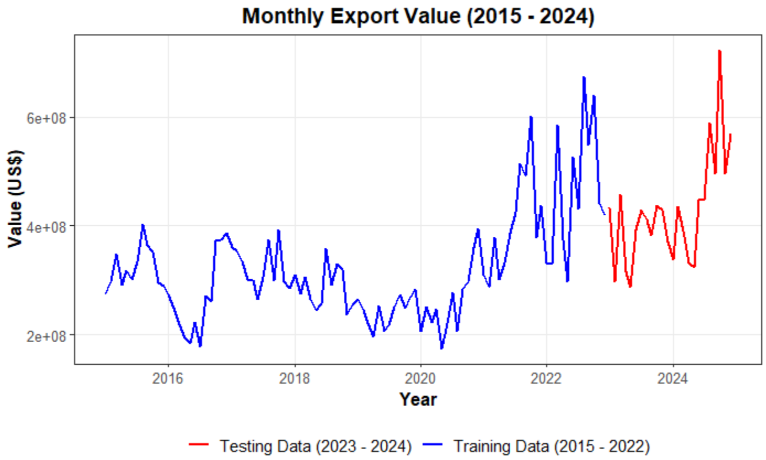

Figure 3. Training and testing data for monthly export value of Lampung Province.

3.2. Stationarity Testing and Variance Stabilization

An initial visual inspection of the raw data suggested the presence of heteroscedasticity, where the variance increased proportionally with the values of exports. The results of the Box-Cox log-likelihood plot is given in Figure 4.

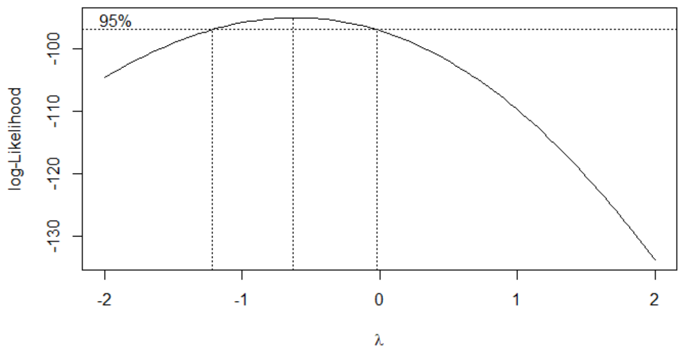

Figure 4. Box-Cox log-likelihood plot.

The plot illustrates a curve, with the maximum likelihood peak identifying the optimal transformation parameter at $\lambda = -0.28$. It confirms that the original export data suffered from variance instability. By applying the Cox-Box transformation, the variance of the new time series data remains constant across the entire training set. Following the stabilization of the variance, the transformed series was tested for stationarity in the mean using the Augmented Dickey-Fuller (ADF) test. The following Table 2 shows the result of the ADF test.

Table 2. ADF test result.

Test	p-value	Description
ADF	0.2805	Non-stationary

The ADF test yielded a p-value of 0.2805 strictly greater than the standard $\alpha = 0.05$ significance threshold. Consequently, this confirms that the time series is non-stationary in the mean. To fix this non-stationarity, first-order differencing was applied to the transformed data. The differenced data is plotted in Figure 5.

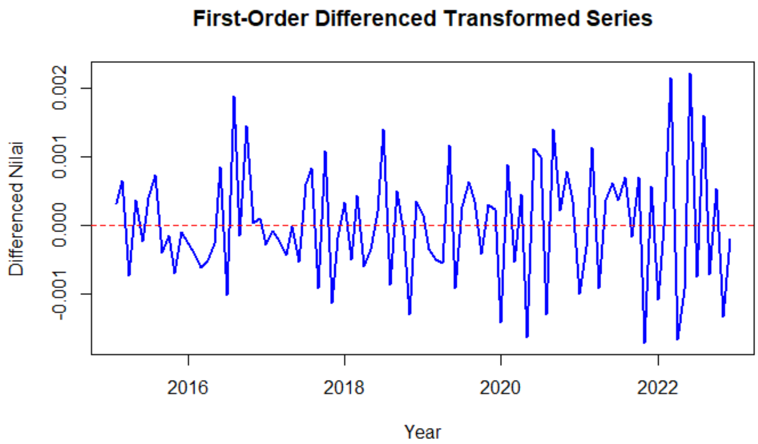

Figure 5. First-order difference data of Lampung's export values.

3.3. Autocorrelation and Partial Autocorrelation Plots

Following the implementation of the Box-Cox transformation and first-order differencing, the next phase involves identifying the appropriate autoregressive and moving average parameters by analyzing the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plot of the differenced data. Figure 6 visualizes the ACF and PACF plots for the training data. The horizontal dashed blue lines represent the 95% confidence intervals. Spikes extending beyond the blue lines indicate statistically significant linear dependencies at those specific lags.

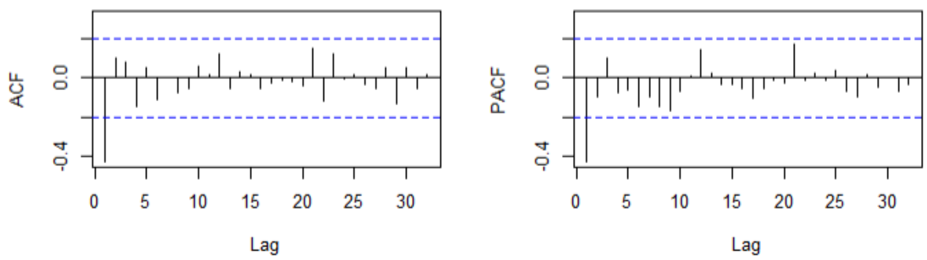

Figure 6. ACF and PACF plots.

An examination of the PACF plot reveals a highly significant negative spike at lag 1. This is primary indicator of a first-order autoregressive process, suggesting an AR parameter of $p = 1$. Furthermore, lag 12 and near 24 are more significant than others, indicating a seasonal component of $s = 12$ months and a first-order seasonal autoregressive term $P = 1$. Combining these visual inspection yields model $ARIMA(1,1,0)(1,0,0)[12]$ as SARIMA base model.

Because visual identification is subjective, it is necessary to specify other candidate models surrounding the initial base model. The following candidate models were constructed for comparative estimation:

1. Model 1: $ARIMA(1,1,0)(1,0,0)[12]$
2. Model 2: $ARIMA(0,1,1)(1,0,0)[12]$
3. Model 3: $ARIMA(1,1,1)(1,0,0)[12]$
4. Model 4: $ARIMA(1,1,0)(0,0,1)[12]$
5. Model 5: $ARIMA(1,1,0)(1,0,1)[12]$.

3.4. SARIMA Model Selection

To objectively select the single most robust model among the candidates, the Akaike Information Criterion (AIC) was employed. A better model will have a lower value of AIC. The estimated model and the corresponding AIC values for all five candidate models are summarized in Table 3.

© 2026 The Authors | doi: 10.23960/pepadun.v7i3.342

Table 3. Candidate ARIMA models and AIC values.

Model	Parameters	AIC
1	$ARIMA(1,1,0)(1,0,0)[12]$	-1092.747
2	$ARIMA(0,1,1)(1,0,0)[12]$	-1092.488
3	$ARIMA(1,1,1)(1,0,0)[12]$	-1091.919
4	$ARIMA(1,1,0)(0,0,1)[12]$	-1092.321
5	$\boldsymbol{ARIMA(1,1,0)(1,0,1)[12}$	**-1094.685**

Based on the evaluation of AIC, the model $ARIMA(1,1,0)(1,0,1)[12]$ yielded the lowest AIC score of -1094.685**.** Consequently, this model selected as the optimal ARIMA model for this study.

3.5. Residual Analysis

In ARIMA modeling, an estimated model is only appropriate if it has successfully extracted all information from the historical data. When a model is fully optimized, the residuals must exhibit the properties of a white noise process, meaning they are independently distributed, normally distributed around a mean of zero, and display no significant autocorrelation.

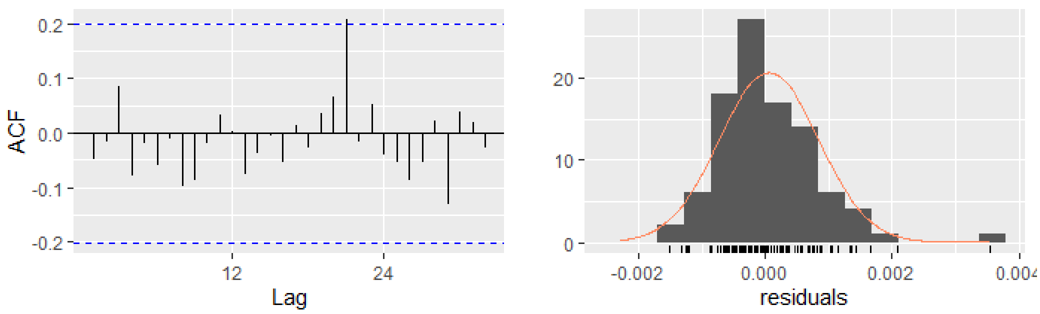

Figure 7. ACF plot and histogram of residuals.

Figure 7 visualizes the diagnostic plots for the model's residuals. The histogram on the right demonstrates that the residual distribution is highly symmetrical and well-centered around zero, closely similar to the normal distribution curve. The ACF plot on the left reveals that all correlation spikes fall safely within the 95% confidence intervals. This strongly suggests that there is no structural information left behind by the ARIMA model.

Table 4. Ljung-Box test result.

Test	p-value	Description
Ljung-Box	0.9933	White noise residual

To provide objective, statistical confirmation of these findings, the Ljung-Box test was applied. The result of this test is summarized in Table 4. This result formally confirms that the residuals are strictly independent and similar to white noise. Consequently, the $ARIMA(1,1,0)(1,0,1)[12]$ is statistically fit for our training data.

3.6. LSTM Network

To empirically determine the optimal LSTM architecture, aa hyperparameter grid search was executed within the 3-fold expanding window. The optimization algorithm evaluated 24 distinct architectural combinations, iterating through varians in network depth (1 or 2 layers), cell state size (10, 20, 30, and 50 neurons per layer), and training duration (20, 50, and 100 epochs). For each configuration, the network was trained and validated across the three folds. The performance of each candidate architecture was calculated using RMSE. Table 5 lists the top five models of LSTM network with each parameter and its RMSE value.

Table 5. Top 5 models of LSTM network specification.

Rank	Layers	Neurons	Epochs	Mean CV RSME
1	**2**	**50**	**100**	**73129098**
2	1	50	100	73601481
3	1	20	50	78079600
4	1	50	20	78879253
5	1	50	50	79845035

The optimal architecture, achieving the lowest Mean Cross-Validation RMSE of 73129098 utilized a stacked design of 2 layers with the maximum tested number of neurons of 50 neurons and the longest training duration of 100 epochs. This validates the need of two-layer architecture to separately process seasonal fluctuations and trends. Furthermore, the 50-neuron capacity provided the necessary dimension to capture complex interactions. Consequently, the 2-layer, 50 neuron, 100 epoch configuration was selected as the deep learning specification for our data.

3.7. Model Selection

With the parameters of both the classical SARIMA model and the deep learning LSTM network mathematically optimized on the 2015-2022 historical data, the study proceeded to the final phase: 24-month recursive forecast for the 2023-2024 period. This testing phase is particularly challenging due to the volatile nature observe in late 2024 data. To objectively quantify predictive accuracy, the forecasts of both models were evaluated against the actual testing data using the Root Mean Squared Error (RMSE). The final RMSE metrics for the 24-month period are presented in Table 6.

Table 6. Forecasting RMSE for ARIMA and LSTM models.

Model	RMSE
$ARIMA(1,1,0)(1,0,1)[12]$	94030344
LSTM network	395566847

The mathematical results demonstrate that the classical ARIMA architecture vastly outperformed the LSTM network. The LSTM's forecasting error was much larger than the forecasting error of the ARIMA model. A visual representation of the export value forecasting results using the ARIMA and LSTM models is presented in Figure 8.

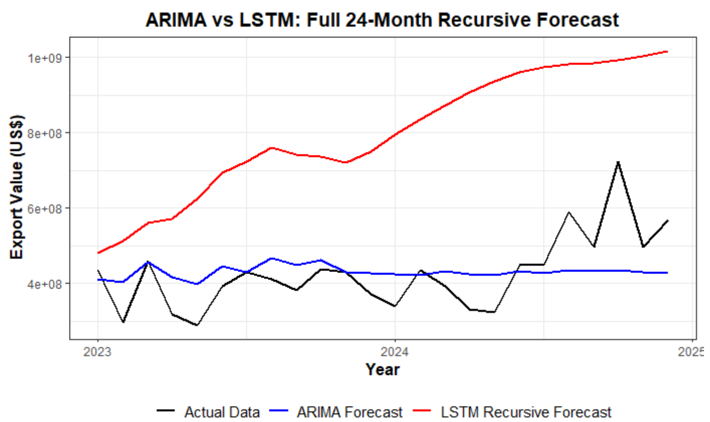

Figure 8. Forecasting plot of ARIMA and LSTM.

In the figure above, the ARIMA model's forecast establishes a relatively flat line prediction between 400 million and 500 million US dollars. Because the ARIMA model relies on explicit mathematical rule, specifically the rigid seasonal parameters and trend, it follows the trajectory of historical data. In contrast, the LSTM network exhibits a massive upward trend that immediately deviate from the actual data in early 2023 and accelerating quickly. Because the neural network has no explicit mathematical model to follow, it relies entirely on its own prior prediction of a small number of data. When the LSTM made a minor overestimation in one month, that creates a compounding feedback loop and the error accumulated.

4. CONCLUSIONS

This study empirically evaluated the forecasting performance of ARIMA and LSTM network using the monthly export values of Lampung Province from 2015 to 2024. While deep learning models are frequently praised for their ability to map complex pattern in data, the forecasting results demonstrated that the classical $ARIMA(1,1,0)(1,0,1)[12]$ outperformed the LSTM network. The LSTMs are relatively poor when executing long-term recursive forecasts on moderately sized, highly volatile regional time series macroeconomic data. The forecasting RMSE of LSTM with value of 395566847 is much larger than RMSE of ARIMA with value 94030344. Under conditions of volatile dataset, mathematical formulation of an ARIMA model provide a safer and more accurate forecasting baseline for economic data with moderate data point size.

LITERATURE

[1] H. Bal, A. H. Mamun, S. Basher, M. R. Uddin, and M. Mowla, "Export-Led Growth Hypothesis in Developing Countries: Econometric Evidence from Bangladesh," *Ömer Halisdemir Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, vol. 12, no. 4, pp. 454–465, Oct. 2019, doi: 10.25287/ohuiibf.507759.

[2] N. M. Odhiambo, "Is export-led growth hypothesis still valid for sub-Saharan African countries? New evidence from panel data analysis," *EJMBE*, vol. 31, no. 1, pp. 77–93, Feb. 2022, doi: 10.1108/EJMBE-06-2020-0156.

[3] G. A. Ariwibowo, "Aktivitas Ekonomi dan Perdagangan di Karesidenan Lampung pada Periode 1856 hingga 1930," *PATANJALA*, vol. 10, no. 2, p. 431, Sep. 2018, doi: 10.30959/patanjala.v10i2.361.

[4] S. Hodijah and G. P. Angelina, "Analisis Pengaruh Ekspor Dan Impor Terhadap Pertumbuhan Ekonomi di Indonesia," *Mankeu*, vol. 10, no. 01, pp. 53–62, Apr. 2021, doi: 10.22437/jmk.v10i01.12512.

[5] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*, Third edition. Melbourne, Australia: OTexts, 2021.

[6] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*, Fifth edition. in Wiley series in probability and statistics. Hoboken, New Jersey: Wiley, 2016.

[7] D. Asteriou and S. G. Hall, *Applied econometrics*, 3. ed. London: Palgrave Macmillan, 2016.

[8] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[9] F. Chollet, *Deep learning with Python*, Second edition. Shelter Island: Manning Publications, 2021.

[10] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PLoS ONE*, vol. 12, no. 7, p. e0180944, Jul. 2017, doi: 10.1371/journal.pone.0180944.

[11] D. M. Ahmed, M. M. Hassan, and R. J. Mstafa, "A Review on Deep Sequential Models for Forecasting Time Series Data," *Applied Computational Intelligence and Soft Computing*, vol. 2022, pp. 1–19, Jun. 2022, doi: 10.1155/2022/6596397.

[12] S. Siami-Namini, N. Tavakoli, and A. Siami Namin, "A Comparison of ARIMA and LSTM in Forecasting Time Series," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Orlando, FL: IEEE, Dec. 2018, pp. 1394–1401. doi: 10.1109/ICMLA.2018.00227.

[13] I. Magfirrah, M. Ilma, K. A. Notodiputro, Y. Angraini, and L. N. A. Mualifah, "Comparative Analysis of ARIMA and LSTM for Forecasting Maximum Wind Speed in Kupang City, East Nusa Tenggara," *Jambura J. Math*, vol. 6, no. 2, pp. 169–175, Aug. 2024, doi: 10.37905/jjom.v6i2.25834.

[14] N. Sunendar and Y. Rianto, “Comparison of ARIMA, LSTM, and GRU Models for Forecasting Sales of HIT Aerosol Products,” *pilar*, vol. 21, no. 2, pp. 153–159, Sep. 2025, doi: 10.33480/pilar.v21i2.6412.

[15] W. A. Pratiwi, I. M. Sumertajaya, and K. A. Notodiputro, “Comparison of ARIMA, LSTM, and Ensemble Averaging Models for Short-Term and Long- Term Forecasting of Non-Stationary Time Series Data,” *Inferensi*, vol. 8, no. 3, p. 231, Nov. 2025, doi: 10.12962/j27213862.v8i3.22643.

[16] F. A. Miranda, K. D. Tania, and R. D. Kurnia, “Comparative Performance Evaluation of ARIMA, SARIMA, and LSTM for Daily Shallot Price Forecasting in Palembang City,” *Electronic. J. Edu. Soc. Econ. Tech.*, vol. 6, no. 2, p. 1323, Dec. 2025, doi: 10.33122/ejeset.v6i2.1323.

[17] Y. N. Hilal, G. D. A. Nainggolan, S. H. Syahputri, and F. Kartiasih, “Comparison of ARIMA and LSTM Methods in Predicting Jakarta Sea Level,” *J. Ilmu dan Teknologi Kelautan Tropis*, vol. 16, no. 2, pp. 163–178, Oct. 2024, doi: 10.29244/jitkt.v16i2.52818.

[18] A. I. E. Nensi, M. Al Maida, K. Anwar Notodiputro, Y. Angraini, and L. N. A. Mualifah, “Performance Analysis of ARIMA, LSTM, and Hybrid ARIMA-LSTM in Forecasting the Composite Stock Price Index,” *CAUCHY*, vol. 10, no. 2, pp. 588–604, Jun. 2025, doi: 10.18860/cauchy.v10i2.33379.

[19] E. S. Putri and M. Sadikin, “Prediksi Penjualan Produk Untuk Mengestimasi Kebutuhan Bahan Baku Menggunakan Perbandingan Algoritma LSTM dan ARIMA,” *FORMAT*, vol. 10, no. 2, p. 162, Aug. 2021, doi: 10.22441/format.2021.v10.i2.007.

[20] D. N. P. Pakaya, N. Achmad, I. K. Hasan, D. Wungguli, and S. N. Abdussamad, “Prediksi Wisatawan Mancanegara di Indonesia Menggunakan Metode SARIMAX dengan Efek Variasi Kalender Libur Nasional,” *JRMM*, vol. 4, no. 6, pp. 287–300, Jul. 2025, doi: 10.18860/jrmm.v4i6.34937.

[21] S. Kwarteng and P. Andreevich, “Comparative Analysis of ARIMA, SARIMA and Prophet Model in Forecasting,” *RD*, vol. 5, no. 4, pp. 110–120, Oct. 2024, doi: 10.11648/j.rd.20240504.13.

[22] Badan Pusat Statistik Provinsi Lampung, “Volume dan Nilai Ekspor.” Nov. 03, 2025. Accessed: Mar. 16, 2026. [Online]. Available: https://lampung.bps.go.id

[23] G. E. P. Box and D. R. Cox, “An Analysis of Transformations,” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 26, no. 2, pp. 211–243, Jul. 1964, doi: 10.1111/j.2517-6161.1964.tb00553.x.

[24] D. A. Dickey and W. A. Fuller, “Distribution of the Estimators for Autoregressive Time Series With a Unit Root,” *Journal of the American Statistical Association*, vol. 74, no. 366, p. 427, Jun. 1979, doi: 10.2307/2286348.

[25] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications: With R Examples*, 4th ed. 2017. in Springer Texts in Statistics. Cham: Springer, 2017. doi: 10.1007/978-3-319-52452-8.

[26] P. J. Brockwell and R. A. Davis, *Introduction to time series and forecasting*, 3rd ed. in Springer texts in statistics. Switzerland: Springer, 2016.

[27] M. A. Haris and P. R. Arum, “Negative Binomial Regression and Generalized Poisson Regression Models on the Number of Traffic Accidents in Central Java,” *BAREKENG: J. Il. Mat. & Ter.*, vol. 16, no. 2, pp. 471–482, Jun. 2022, doi: 10.30598/barekengvol16iss2pp471-482.

[28] D. C. Montgomery, C. L. Jennings, and M. Kulahci, *Introduction to time series analysis and forecasting*, Second edition. in Wiley series in probability and statistics. Hoboken, New Jersey: Wiley, 2015.

[29] I. Goodfellow, A. Courville, and Y. Bengio, *Deep learning*. in Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016.

[30] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, “Deep Learning with Long Short-Term Memory for Time Series Prediction,” *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 114–119, Jun. 2019, doi: 10.1109/MCOM.2019.1800155.

 | doi: 10.23960/pepadun.v7i3.342

[31] V. Cerqueira, L. Torgo, and I. Mozetič, "Evaluating time series forecasting models: an empirical study on performance estimation methods," *Mach Learn*, vol. 109, no. 11, pp. 1997–2028, Nov. 2020, doi: 10.1007/s10994-020-05910-7.

[32] D. K. Sharma, M. Chatterjee, G. Kaur, and S. Vavilala, "Deep learning applications for disease diagnosis," in *Deep Learning for Medical Applications with Unique Data*, Elsevier, 2022, pp. 31–51. doi: 10.1016/B978-0-12-824145-5.00005-8.