Preprints.org

Article

Not peer-reviewed version

Comparison of Sarima and LSTM Model to Forecast the Comprehensive Income (Loss) of Life Insurances in Indonesia

Hiteash Mahajan *

Posted Date: 24 March 2026

doi: 10.20944/preprints202603.1855.v1

Keywords: ARIMA; forecast; insurance; Indonesia; LSTM; total comprehensive income

Article

Comparison of Sarima and LSTM Model to Forecast the Comprehensive Income (Loss) of Life Insurances in Indonesia

Hiteash Mahajan

Department of Computer Science & Engineering, UIET Kathua, India; hiteashgupta1@gmail.com

Abstract

This study compares the performance of the Seasonal Autoregressive Integrated Moving Average (SARIMA) and Long Short-Term Memory (LSTM) models in predicting time series data with seasonal patterns. Using historical data, namely total comprehensive income (loss) data for all insurance companies in Indonesia that contain strong seasonal components, the SARIMA model proved superior in capturing and representing seasonal patterns compared to LSTM. However, the evaluation results showed that the overall prediction accuracy of both models is still at a level that needs improvement. These findings suggest that although SARIMA is effective for seasonal data, the integration of statistical and deep learning-based models has the potential to improve prediction performance. Future research is recommended to explore hybrid approaches, hyperparameter optimization, and the use of external variables to improve the accuracy and generalization of predictive models.

Keywords: ARIMA; forecast; insurance; Indonesia; LSTM; total comprehensive income

---

1. Introduction

The comprehensive income statement is a report that measures a company's performance over a specific period. It is used to assess and predict the amount and timing of future cash flows, subject to uncertainties. The comprehensive income statement describes the sources of income earned by the company in conducting its business, as well as the types of costs incurred by the company in carrying out its activities. By observing or noting the difference between revenues and expenses, it is possible to determine the company's profit or loss during a specific period.

The comprehensive income statement is useful for helping financial statement users predict future cash flows, in order to determine profitability, investment value, and creditworthiness. The comprehensive income statement is often used by investors use information about a company's past earnings as an important input in predicting future earnings and cash flows, which then serve as the basis for predicting the company's future stock price and dividends. Creditors also use past earnings information, creditors can understand a potential borrower's ability to generate future cash flows to cover interest and repay loan principal. Creditors do not primarily desire the liquidation of collateral, but rather the company's success in generating income and cash flow from operations. Beside that, the comprehensive income statement is considered important to investors and creditors, so it is only natural that management should also be interested in the comprehensive income statement. Furthermore, in many companies, bonuses given to managers are determined based on their success in achieving profit targets.

Previous research has shown that traditional statistical models such as ARIMA and GARCH have advantages in capturing linear patterns and volatility in financial data. ARIMA has been shown to be effective in forecasting short-term trends and seasonal patterns in time series data (Box & Jenkins, 2015). Meanwhile, GARCH excels in modeling heteroscedasticity, thus representing the volatility fluctuations common in financial and insurance market data (Bollerslev, 1986; Engle, 2001).

 doi:10.20944/preprints202603.1855.v1

With their relatively simple structures, these two models provide fairly accurate results for data with linear patterns, clear interpretations, and lower computational requirements than modern machine learning approaches.

Insurance financial data is volatile due to the influence of premiums, claims, investment returns, and operational costs. Traditional statistical models such as ARIMA or GARCH are capable of capturing simple linear patterns, but are often less effective when dealing with complex and non-linear interactions. Therefore, modern approaches such as machine learning, specifically LSTM (Long Short-Term Memory), are more suitable because they can learn both short-term and long- term patterns in insurance time series data, thus providing more accurate predictions of comprehensive profit.

The LSTM model significantly improved prediction accuracy of mutual fund performance over traditional statistical approaches. It effectively captured temporal patterns in the data, helping investors anticipate fund behavior and make informed decisions. The study from (Wang, 2024) LSTM can serve as a powerful tool for fund managers and retail investors navigating complex financial markets. It confirms that LSTM models offer a robust, accurate, and practical method for predicting mutual fund performance, making them valuable for data-driven investment strategies (Wang, 2024). The study from Nelson et. al (2017) explores the application of Long Short-Term Memory (LSTM) networks in predicting future stock price trends, utilizing historical price data in conjunction with technical analysis indicators. A predictive model was developed and a series of experiments were conducted. The resulting performance was evaluated using several metrics to determine whether this type of algorithm offers improvements over traditional machine learning methods and existing investment strategies. The results are promising, with the model achieving an average accuracy of up to 55.9% in predicting whether the price of a given stock would rise in the near future.

Most forecasting research in the insurance sector still focuses on premiums, claims, or claim reserves, while the Total Comprehensive Income (Loss) variable, which encompasses both profit and loss and Other Comprehensive Income, is relatively rarely studied. This indicator, however, is a more comprehensive indicator for assessing the financial health of insurance companies. Most ARIMA/GARCH studies have been conducted on the stock market or macroeconomics (e.g., GDP, inflation, oil prices), but their application to financial data of insurance companies in Indonesia is still very limited. Previous research tends to use only one type of model (e.g., ARIMA alone), without comparing it with variations such as seasonal SARIMA or simple benchmarks (naïve/ETS). This results in less comprehensive evaluations of model accuracy.

Some previous research has focused solely on short-term forecasting, while medium- and long-term forecasting is more relevant for risk management and financial planning, but remains rarely studied. This study fills this gap by using Total Comprehensive Income (Loss) as the primary variable, thus providing a more representative forecasting method for assessing the overall performance of insurance companies than simply looking at premiums or claims. This study provides a contextual contribution by testing a forecasting model on insurance financial data in Indonesia, an area where there is still limited literature.

In addition to using ARIMA/SARIMA, this study tests model performance using various evaluation metrics (RMSE, MAE, MAPE), providing a more comprehensive picture of prediction quality. The results offer an analytical basis for planning risk management and investment strategies, as they are able to anticipate future comprehensive profit trends.

2. Literature Review

The Autoregressive Integrated Moving Average (ARIMA) is a family of Box–Jenkins models for time series forecasting that combines autoregressive (AR), differencing (I), and moving average (MA) components.

The ARIMA model consists of three parts:

- Autoregressive (AR): The linear relationship between its current value and its lagged observations.

 doi:10.20944/preprints202603.1855.v1

- Differencing (I): The number of times the series is differenced to achieve stationarity to remove trends and seasonality.
- Moving Average (MA): Relates the current value to past forecast errors using a number of lagged errors.

This model is very effective for data that has a linear pattern and requires the assumption of stationarity before modeling. The following is the general formula for ARIMA.

$$\Delta Yt = \varphi 1\Delta Yt{-}1 + \varphi 2\Delta Yt{-}2 + \varphi p\Delta yt{-}p + et - \theta 1et{-}1 - \theta 2et{-}2 - \theta qet{-}q$$

Where:

- ΔYt is the differenced value of the original time series at time t
- $\varphi 1, \varphi 2, \ldots \varphi p$ are the autoregressive (AR) coefficients
- $\theta 1, \theta 2, \ldots \theta q$ are the moving average (MA) coefficients
- et is the white noise (random error) at time t
- p is the number of autoregressive terms
- q is the number of moving average terms

This equation combines both autoregressive (AR) and moving average (MA) components, applied to a differenced version of the series (indicated by Δ), to model time-dependent patterns in the data.

Standard practice includes identification (stationarity tests, ACF/PACF), parameter estimation, and residual diagnostics; common extensions include SARIMA (seasonal) and ARIMAX/SARIMAX (with exogenous variables). Classical and modern introductory references place ARIMA alongside exponential smoothing as the two most widely used approaches in applied forecasting.

Long Short-Term Memory (LSTM) networks have been shown to be one of the best choices when it comes to handling sequential data. LSTM (Long Short-Term Memory) is a development of the Recurrent Neural Network (RNN) architecture designed to overcome the weakness of RNNs in remembering long-term information (vanishing gradient problem). LSTM has a cell state structure with three main gates: input gate, forget gate, and output gate, which function to regulate the flow of information adaptively. The advantage of LSTM compared to conventional RNNs is its ability to remember patterns over long time spans, manage long-term and short-term dependencies, and capture non-linear relationships in the data. This makes LSTM very suitable for forecasting complex time series, high fluctuations, or patterns that are difficult to capture by linear models such as ARIMA. Their unique architecture comprises memory cells along with three gates:

- Input Gates: Controls which parts of the previous memory are discarded.
- Forget Gates: Determines what new information should be added to the memory.
- Output Gates: Regulates what information is output for the current step.

These components work together to selectively retain, update, and output information to capture both the short-term fluctuations and long-term trends in the dataset. Zou and Qu (2020) implemented LSTM networks to predict stock prices and analyze this strategy against others. Their study showed that the LSTM model outperformed the traditional RNNs and machine learning methods by effectively learning patterns in historical stock price data. Sonkavde et al. (2023) conducted a review comparing machine learning and deep learning methods which included LSTM, Convolutional Neural Networks (CNNs), and traditional RNNs. Their result showed the effectiveness of the LSTM model compared to RNNs and CNNs by handling temporal and spatial features better. Study from Sak et al (2014), LSTM recurrent neural networks (RNNs) outperform both deep neural networks (DNNs) and traditional RNNs in the context of acoustic modeling.

The following are the general equations and architecture of LSTM

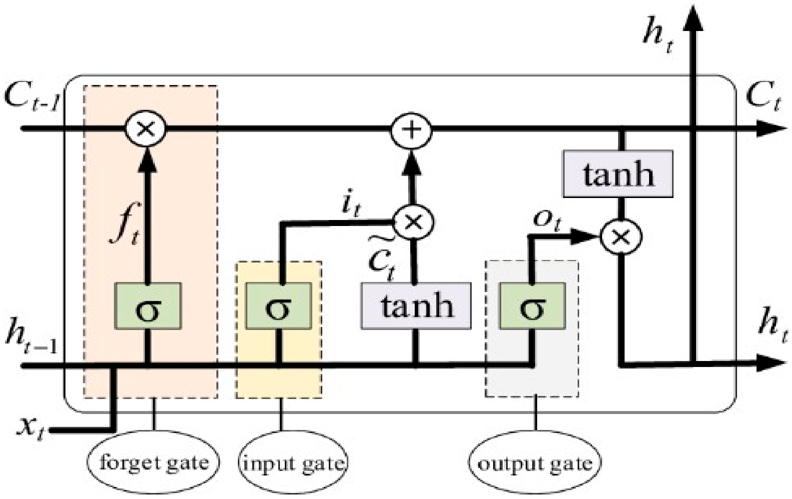

Figure 1. Gambar 1 Arsitektur LSTM.

Forget gate :
Controls what portion of the previous cell state should be retained:

$$ft = \sigma(Wf \times yt + Uf \times ht{-}1 + bf) \quad (1)$$

Input gate :
Determines which new information should be added to the cell state:

$$It = \sigma(Wi \times yt + Ui \times ht{-}1 + bi\,) \quad (2)$$

$$\acute{C}t = \tanh(Wc \times yt + Uc \times ht{-}1 + bc) \quad (3)$$

$$Ct = \tanh\,((ft \times Ct{-}1 + it) \times \acute{C}t) \quad (4)$$

Output gate :
Controls the output of the LSTM cell:

$$Ot = \sigma(Wo \times yt + Uo \times ht{-}1 + bo) \quad (5)$$

$$ht = Ot \times \tanh(Ct) \quad (6)$$

Where:

- σ denotes the sigmoid activation function
- tanh denotes the hyperbolic tangent activation
- Ot indicates element-wise multiplication
- W, U, and b represent weight matrices and bias vectors

These equations enable the LSTM to retain, update, and output relevant features across time steps, making it highly effective for sequence modeling tasks such as language modeling, time- series forecasting, and speech recognition.

3. Research Method

This study uses a quantitative approach with time series forecasting. The Autoregressive Integrated Moving Average (ARIMA/SARIMA) model chosen because it has proven effective in predicting both linear and fluctuating financial data.

3.1. Research Data

Data Type that we used in this study is secondary data in the form of insurance company financial reports. With predicted variable is Total Comprehensive Income (Loss) of Life Insurance in Indonesia. Data Period is from 2017 to 2025.

3.2. Analysis Stages

The analysis process follows the Box–Jenkins framework (Box & Jenkins, 2015; Hyndman & Athanasopoulos, 2021):

- Data Stationarity Test: we utilized the Augmented Dickey-Fuller (ADF) Test and/or the KPSS Test. If non-stationary, we perform a transformation (log/Box-Cox) or differencing.
- Model Identification: The Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots are used to determine the order of p, d, and q. If a seasonal pattern is present, the SARIMA (Seasonal ARIMA) model is used.
- Model Parameter Estimation: ARIMA/SARIMA parameters are estimated using the Maximum Likelihood Estimation (MLE) method.
- Model Diagnostics: The Ljung Box test is used to ensure the residuals are white noise. The best model is selected based on the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC).
- Forecasting: The best model is used to predict Total Comprehensive Income (Loss) in the future period. Accuracy is evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

4. Results and Discussion

Based on Table 1, the company's monthly Total Comprehensive Income (Loss) during the period January 2017 to May 2025, which consists of 101 observation data, has a maximum value of 14,759,160.71 which occurred in December 2017, while the minimum value of -12,191,989.15 occurred in October 2018. Overall, the average Total Comprehensive Income (Loss) during the period was 1,481,665.00, which shows that in general the company recorded positive comprehensive income even though there were significant fluctuations in certain periods.

Table 1. First ten observations of total monthly Comprehensive Income (Loss).

Months	Total Comprehensive Income (Loss)
2017-01-01	1,474,870.8
2017-02-01	2,638,107.4
2017-03-01	5,308,832.8
2017-04-01	6,259,147.3
2017-05-01	8,074,418.8
2017-06-01	9,828,258.4
2017-07-01	10,828,261.0
2017-08-01	6,566,310.7
2017-09-01	7,458,188.1
2017-10-01	11,460,697.5

According to Figure 2, the company's monthly Total Comprehensive Income (Loss) trend from January 2017 to May 2025 shows a fluctuating pattern with several significant changes. At the beginning of the period (2017), there was a sharp increase that peaked in December 2017, followed by a drastic decline until it reached its lowest point in October 2018. Afterward, the Total Comprehensive Income (Loss) fluctuated with several extreme spikes and declines, such as in early 2022. Entering the period from 2023 to 2025, the movement tended to be more stable with an upward trend in early 2025, indicating potential positive growth in the coming period.

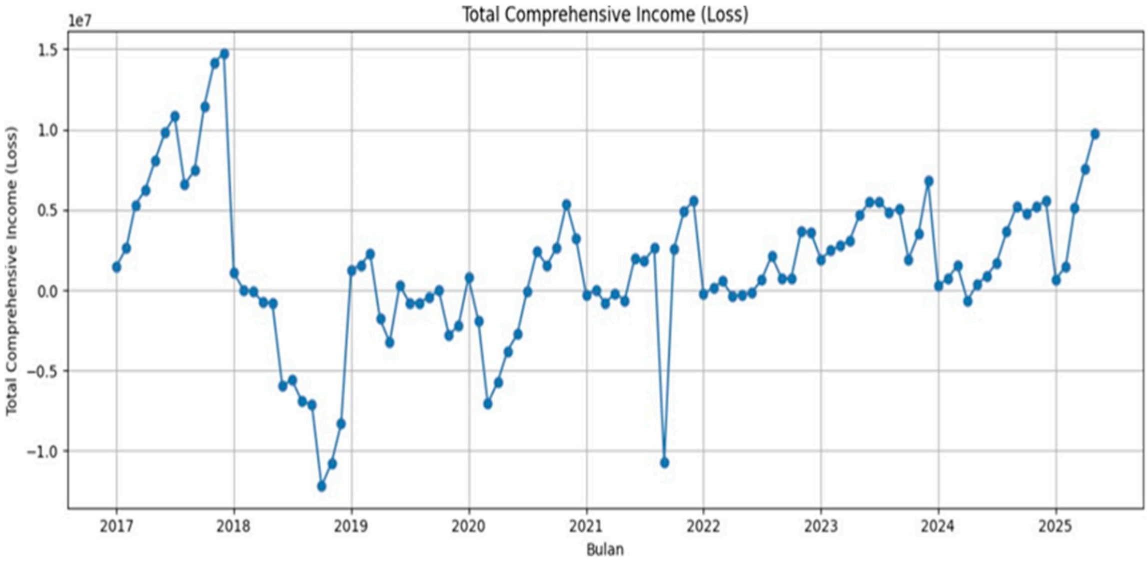

Figure 2. Monthly Comprehensive Income (Loss) Trend.

4.1. Time Series Decomposition

This figure shows the results of decomposing the Total Comprehensive Income (Loss) time series into four main components. The Observed component shows the original data observed from January 2017 to May 2025, while the Trend component reveals a long-term pattern with a sharp decline until 2019, then a plateau with a slight increase towards 2025. The Seasonal component displays a seasonal pattern that repeats annually with significant fluctuations, indicating a periodic cycle in the data. Meanwhile, the Residual component contains random variation that cannot be explained by either trends or seasonal patterns, likely influenced by external factors or unforeseen events. This decomposition is useful for separating the influence of trends and seasonality from random variation, thus facilitating the analysis and forecasting process**.**

4.2. Split Data

Figure 3 shows the division of the Total Comprehensive Income (Loss) data into two parts: training data and testing data in an 80:20 ratio. Training data is used to build predictive models, while test data is used to evaluate the model's performance on data it has never seen before.

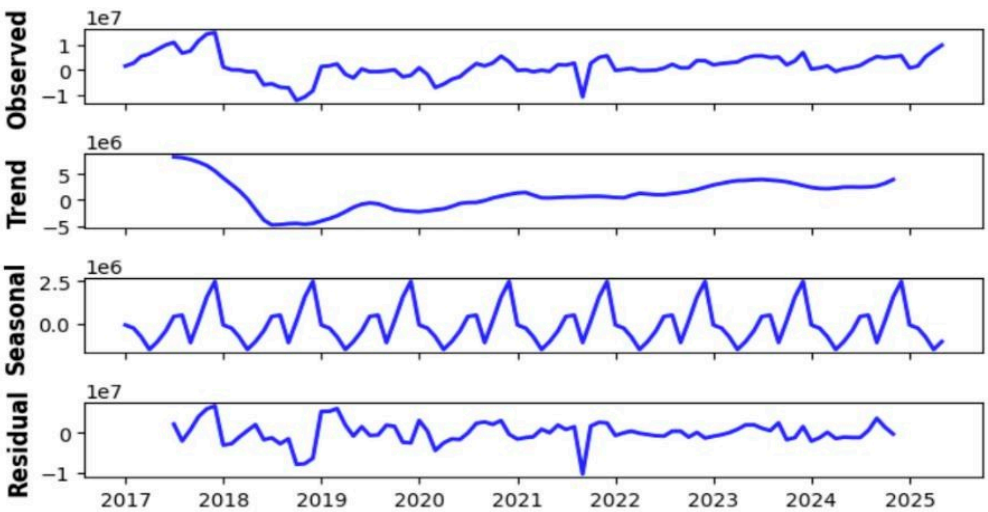

Figure 3. Time Series Decomposition Plot.

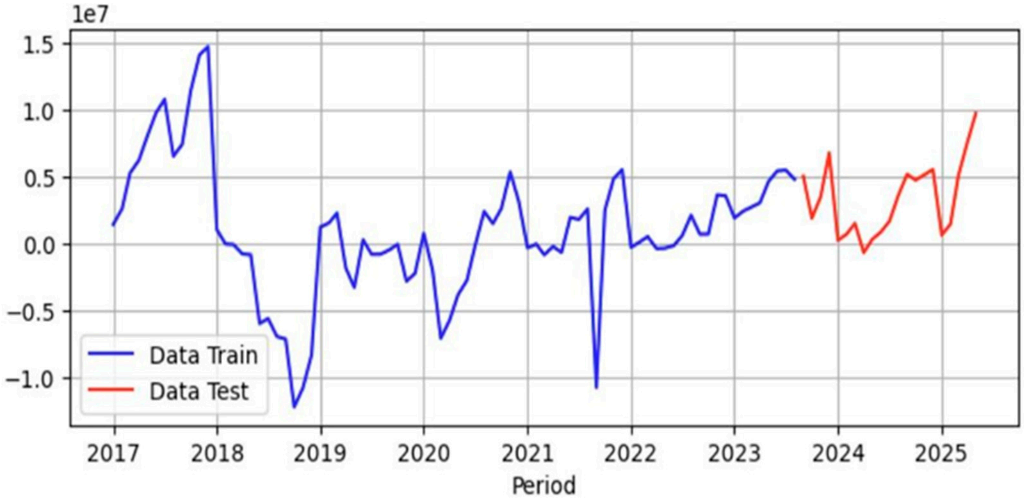

Figure 4. Training data and testing data of the Total Comprehensive Income (Loss) of Life Insurance in Indonesia.

From this visualization, we can see that the training data (blue line) covers the period from January 2017 to approximately mid-2023, while the test data (red line) covers the remaining period until May 2025. This separation is important so the model can learn from historical patterns and test its ability to predict future trends without data leakage.

4.3. Data Stationarity

The following figure presents plots of the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) on the training data set. These plots are used to observe the interrelationships between observations in a time series and help identify potential autoregressive (AR) and moving average (MA) components before building an ARIMA or SARIMA model.

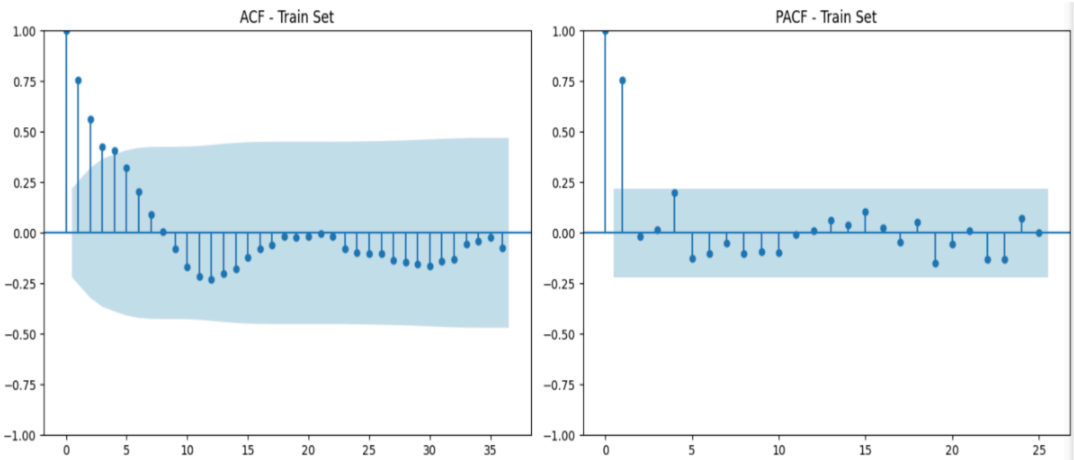

Figure 5. Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) on the training data set.

The ACF and PACF plots in the figure are used to identify the possible orders of the autoregressive (AR) and moving average (MA) components in the time series model. Based on the results of the Augmented Dickey-Fuller (ADF) test, the ADF statistic is -3.2049 with a p-value of 0.0197, which is less than the 0.05 significance threshold. Furthermore, this value is also lower than the critical value at the 5% significance level (-2.8989), thus concluding that the data is stationary. In the next stage, seasonal differencing will be performed to eliminate any remaining seasonal patterns and ensure the data meets the stationary assumption more optimally. This stationary condition is important because it is one of the main prerequisites before building an ARIMA or SARIMA model. SARIMA Model Identification (p,d,q) (P,D,Q)^s

After seasonal differencing to eliminate annual recurring patterns, the SARIMA model identification process begins with an analysis of the ACF and PACF plots on the differencing data to estimate the initial values of the parameters p, q, P, Qp, q, P, Q. The auto-ARIMA method is also used to automatically suggest parameter combinations based on the information criterion (AIC/BIC). The results of these two approaches are then compared and tested through a model overfitting process, which involves trying various parameter combinations around candidate values to ensure the selected model strikes a balance between goodness of fit and simplicity. This process yields three best candidate SARIMA models, which will be further evaluated in depth during the final model selection stage.

Table 2. ARIMA Model Evaluation Based on AIC, BIC, and HQIC.

Model	AIC	BIC	HQIC
ARIMA (1,0,0) (0,1,1) [12]	2265.71	2272.38	2268,36
ARIMA (1,0,1) (0,1,1) [12]	2268.10	2276.98	2271,62
ARIMA (2,0,0) (0,1,1) [12]	2268.73	2277.60	2272,24

Based on the model selection results, three best SARIMA candidates were obtained after going through the initial identification process, auto-ARIMA, and overfitting. The three models are ARIMA (1,0,0) (0,1,1) [12], ARIMA (1,0,1) (0,1,1) [12], and ARIMA (2,0,0) (0,1,1) [12].

The AIC, BIC and HQIC values for each model are relatively close, indicating that they have a similar level of model fit. The ARIMA (1,0,0)(0,1,1) [12] model has the lowest AIC and HQIC, making it slightly superior to the other candidates in terms of information criteria

4.4. Evaluation of the SARIMA Model on the Testing Data

The SARIMA (1,0,0) (0,1,1) [12] model was used to forecast Total Comprehensive Income (Loss) data. In the graph, the training data is shown by the blue line, the test data by the green line, and the model's prediction results by the dashed red line. The model can be seen to follow the general direction of data movement, but tends to produce smoother predictions and is less able to capture sharp fluctuations in the test data.

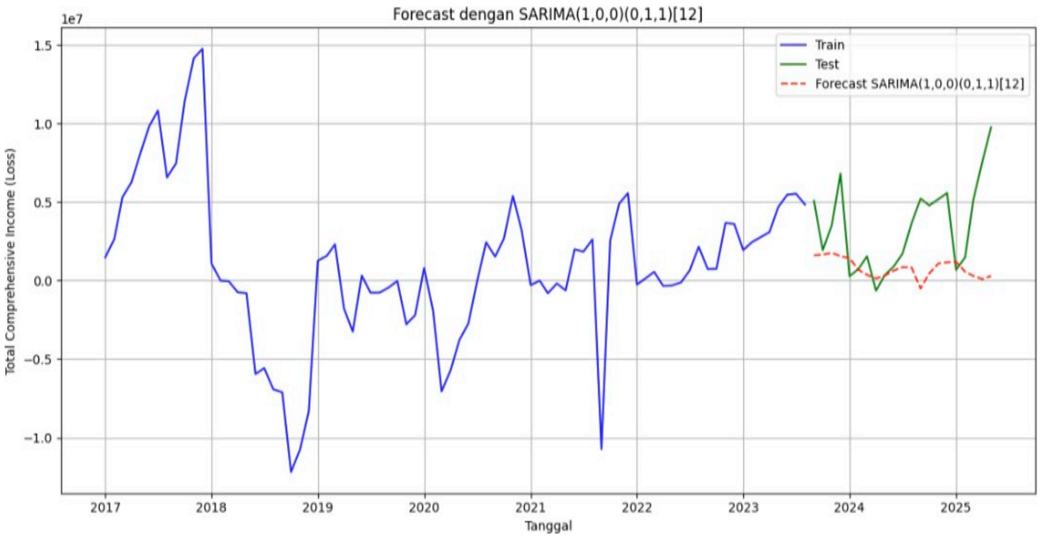

Figure 6. Forecasting of SARIMA model.

Based on the performance evaluation, this model produced an RMSE of 3,858,420.47, an MSE of 1.4887×10^{13}, and a MAPE of 85.00%. The relatively high MAPE value indicates that the relative error of the predictions to the actual data is still large, so the model's accuracy is not optimal even though seasonal patterns can be represented.

 doi:10.20944/preprints202603.1855.v1

4.5. LSTM Model Hyperparameters

The prediction process using the LSTM model begins with designing an architecture tailored to the characteristics of the time series data. The architecture used consists of one LSTM layer connected to a dense layer as the final output. Model training was conducted by testing 144 hyperparameter variations, including different numbers of neurons, epochs, learning rates, and the use of two optimization algorithms: Adam and RMSProp. All combinations were trained on the training data and validated using the validation data to achieve consistent performance.

The batch size was selected by considering the proportion of training and test data, ensuring efficient and stable training. The best combination for each walk-forward validation scenario was selected based on the lowest average RMSE value obtained from all tested configurations.

Table 3. LSTM Model Hyperparameters.

Hyperparameter	Value
Optimizer	Adam, RMSProp, SGD
Neuron	4, 8, 12, 16
Learning Rate	0.001, 0.0001, 0.00001
Epoch	10, 25, 40, 60
Layer	1
Dropout	0.1

The best hyperparameters were selected using a manual grid search: all candidate combinations (number of neurons, number of epochs, learning rate, and optimizer type Adam, RMSprop, or SGD) were arranged as a grid. Each combination was then trained (with dropout and early stopping), evaluated on validation data, and its value recorded. The primary selection criterion was the average RMSE (in a walk-forward scenario, if applicable); the combination with the lowest RMSE was selected as the final configuration. This manual grid search approach ensured a transparent and thorough process, consistently testing every point in the hyperparameter space, allowing the best results (16 neurons, 60 epochs, lr=0.001, Adam optimizer) to be objectively identified.

4.6. LSTM Model Testing with Best Hyperparameters

The Total Comprehensive Income (Loss) time series was divided into training (blue) and test (green) data. The data was prepared for the LSTM through normalization and sequential windowing (input–target) to allow the model to learn temporal patterns. Experiments were performed with various combinations of hyperparameters (number of neurons, learning rate, Adam/RMSProp optimizer, epochs, and batch size) and evaluated on validation to select the most stable configuration.

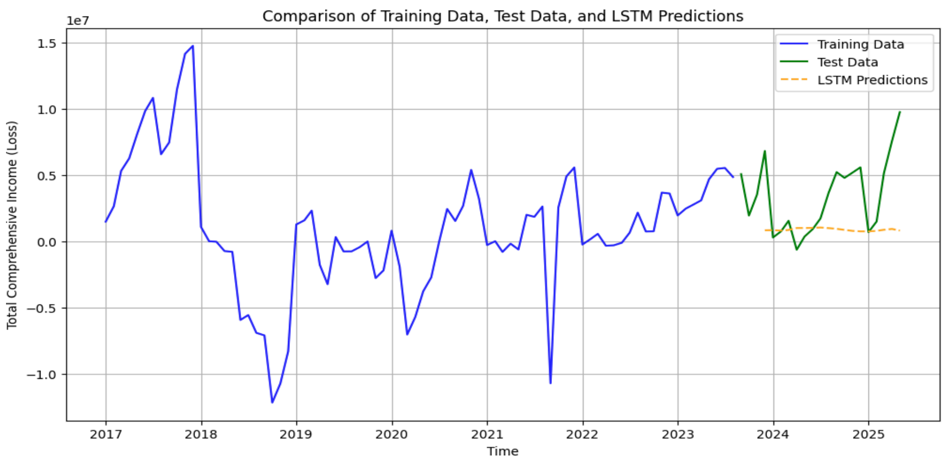

Figure 7. Comparison of Training Data, Testing Data and LSTM Prediction.

The graph shows a comparison of the training data (blue), test data (green), and LSTM predictions (dotted orange). The model is able to follow the general level of the series during the test period, but tends to be unable to capture large spikes in the test data. Overall, the main pattern is captured, but the amplitude of the fluctuations remains muted; improvements could be directed towards adjusting the window length, adding more layers/neurons, tuning the learning rate, or adding exogenous features if available.

Based on the performance evaluation results, the LSTM model produced an RMSE of 3067615×106, an MSE of 9410260×10^{12}, and a MAPE of 165.29% on the test data. The relatively large RMSE and MSE values indicate a significant deviation from the actual predictions on an absolute scale. Meanwhile, a high MAPE indicates a large relative error percentage, so the model is unable to provide predictions with sufficient accuracy. This is consistent with the visualization results, where the model tends to produce predictions that are too smooth (underfitting) and is less able to capture spikes in values during certain periods.

5. Conclusion

SARIMA is capable of capturing annual seasonal patterns and general trends, but its predictions tend to be smooth, making them less responsive to sharp spikes or drops. The relatively high MAPE value indicates that the prediction accuracy is relatively suboptimal. LSTM is more flexible in capturing non-linear patterns, but on this data it tends to underfit, producing predictions that are too smooth and do not capture extreme fluctuations. The high MAPE value reinforces this indication. In general, SARIMA provides better results than LSTM on this dataset, especially in terms of representing seasonal patterns, although overall accuracy still needs improvement. Given that SARIMA outperformed LSTM in capturing seasonal patterns in this dataset, future studies should consider the following directions to enhance overall forecasting accuracy, Combining the strengths of both models through hybrid frameworks (e.g., SARIMA-LSTM or ARIMA-GRU) could leverage SARIMA's ability to handle seasonality and LSTM’s strength in modeling nonlinear patterns and long-term dependencies. Future research could apply more advanced hyperparameter tuning methods (e.g., grid search, Bayesian optimization) to optimize LSTM architecture (e.g., number of layers, neurons, learning rate), which may result in performance improvements.

Acknowledgements: The author would like to express sincere gratitude to all parties who have provided support in the preparation of this article. Special thanks to my parents, colleagues and others who have offered constructive feedback, suggestions and encouragement.

References

1. Bollerslev, T. (1986). “Generalized autoregressive conditional heteroskedasticity”. *Journal of Econometrics, 31*(3), 307–327. https://doi.org/10.1016/0304-4076(86)90063-1
2. Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: Forecasting and control (5th ed.). *Wiley*.
3. Engle, R. F. (2001). GARCH 101: The use of ARCH/GARCH models in applied econometrics. *Journal of Economic Perspectives, 15*(4), 157–168. https://doi.org/10.1257/jep.15.4.157.
4. Hyndman, R. J., & Athanasopoulos, G. (2021). Forecasting: Principles and practice (3rd ed.). *OTexts*. https://otexts.com/fpp3/
5. D. M. Q. Nelson, A. C. M. Pereira and R. A. de Oliveira, (2017). Stock market's price movement prediction with LSTM neural networks, 2017 *International Joint Conference on Neural Networks (IJCNN)*, Anchorage, AK, USA, pp. 1419-1426, doi: 10.1109/IJCNN.2017.7966019.
6. Sak, H., Senior, A.W., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *Interspeech.*
7. Sonkavde, G., Dharrao, D. S., Bongale, A. M., Deokate, S. T., Doreswamy, D., & Bhat, S. K. (2023). Forecasting Stock Market Prices Using Machine Learning and Deep Learning Models: A Systematic

 doi:10.20944/preprints202603.1855.v1

Review, Performance Analysis and Discussion of Implications. *International Journal of Financial Studies*, *11*(3), 94. https://doi.org/10.3390/ijfs11030094

8. Wang, Y. (2024). Predicting Mutual Fund Performance based on LSTM Models. *Highlights in Science, Engineering and Technology*. https://doi.org/10.54097/qezdm154.
9. Zou, Z., & Qu Z. (2020). Using LSTM in Stock prediction and Quantitative Trading. CS230: Deep Learning, *Winter 2020*. https://cs230.stanford.edu/projects_winter_2020/reports/32066186.pdf accessed on 10 July 2025