BAREKENG: Journal of Mathematics and Its Applications
March 2026 Volume 20 Issue 1 Page 0055–0068
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss1pp0055-0068

LSTM AND GRU IN RICE PREDICTION FOR FOOD SECURITY IN INDONESIA

Triyani Hendrawati [1*], **Kennedy Marthendra** [2],
Brian Riski Jayama Simanjuntak [3], **Anindya Aprilianti Pravitasari** [4]

[1,2,3,4] *Statistics Department, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran*
Jl. Raya Bandung Sumedang KM 21, Jatinangor, Sumedang, 45363, Indonesia

*Corresponding author's e-mail: * **triyani.hendrawati@unpad.ac.id***

Article Info

Article History:
Received: 6th February 2025
Revised: 8th April 2025
Accepted: 20th July 2025
Available online: 24th November 2025

Keywords:
Forecasting;
GRU;
Leadership;
LSTM;
Machine learning;
Rice price.

ABSTRACT

Hunger in Indonesia remains a serious challenge, especially in the face of food price instability, particularly rice as the main staple food. In order to achieve SDG 2 "Zero Hunger" by 2030, policies that support price stability and more effective food distribution are needed. This study aims to assess the predictive power of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models for Indonesian rice prices. The dataset, consisting of 1,424 observations from early 2021 to late 2024, was collected from official sources and preprocessed using normalization techniques. The data was then divided into training, validation, and testing sets. Each model was trained and evaluated using Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) metrics. LSTM, a type of Recurrent Neural Network (RNN), uses three gates and cell memory to identify long-term patterns in time series data. GRU, with a simpler structure involving only two gates, is more efficient in modeling temporal relationships. The results show that the LSTM model achieved MAPE 3.49%, while the GRU model outperformed it with MAPE 1.08%. Overall, the GRU model demonstrated higher accuracy in forecasting rice prices.

How to cite this article:

T. Hendrawati, K. Marthendra, B. R. Jayama Simanjuntak and A. A. Pravitasari., "LSTM AND GRU IN RICE PREDICTION FOR FOOD SECURITY IN INDONESIA," *BAREKENG: J. Math. & App.*, vol. 20, iss. 1, pp. 0055-0068, Mar, 2026.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

One of Indonesia's biggest issues within the ASEAN region is the extent of poverty in the nation. According to the Global Hunger Index (GHI) and reports from the World Food and Agriculture Organization (FAO), Indonesia still suffers from hunger, especially in areas that are vulnerable to economic instability and food price fluctuations. Indonesia, as the country with the largest population in ASEAN, faces great challenges in ensuring sufficient food availability for its entire population, especially rice, which is the main staple food. Significant fluctuations in rice prices can affect food security and exacerbate hunger levels in communities [1], [2].

The Sustainable Development Goals (SDGs) of the UN are important standards for global efforts to combat poverty. SDG 2, "Zero Hunger", specifically targets eliminating hunger, achieving food security and improved nutrition, and supporting sustainable agriculture by 2030 [3], [4]. To achieve this target, Indonesia must address several key barriers, such as food price instability, increased sustainable agricultural production, and more effective management of food distribution.

One of the main factors affecting hunger levels in Indonesia is the price of rice. The volatility of rice prices, which can be influenced by weather, harvest patterns, distribution and market factors, has a direct impact on people's purchasing power, especially in the most vulnerable groups. For example, in 2022, national rice prices experienced sharp fluctuations, rising from around IDR 9,824 per kg in January to over IDR 12,300 per kg by December, driven by disruptions in distribution and harvest failures in several regions. Similar volatility was observed in 2023, with prices peaking during the dry season and falling abruptly after government interventions. Unpredictable price fluctuations can lead to food crises, reduce people's access to food, and increase poverty. Therefore, the development of accurate rice price forecasting models is essential to support policies aimed at maintaining price stability and preventing hunger [5].

In this effort, the utilization of machine learning technology is becoming an increasingly relevant solution. Techniques such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models, which are types of Recurrent Neural Networks (RNNs), are well-suited for time series forecasting tasks. These models are designed to capture sequential dependencies in data, making them effective for predicting future rice prices based on historical trends. In time series data, artificial neural networks like LSTM and GRU excel in identifying complex patterns and temporal relationships. The LSTM model has the advantage of maintaining long-term memory through its gated architecture, making it suitable for data with longer temporal dependencies. However, it requires more computational resources and training time. In contrast, the GRU model offers simpler architecture with fewer gates, allowing for faster training and lower computational cost, although it may be less effective in capturing very long-term dependencies. By using these machine learning methods, rice price prediction can be done more effectively, assisting policymakers and economic actors in designing more appropriate strategies for price stabilization.

Previous research has shown that the LSTM model is effective for rice price forecasting. Study [6] used LSTM with variables such as weather, crop yield, and land area, achieving a low Root Mean Square Error (RMSE) of 0.054 using daily data from 2015 to 2023. These results demonstrate LSTM's ability to model complex price patterns. Study [7] compared LSTM with Bidirectional LSTM (BILSTM) in predicting medium- and high-quality rice prices from 2017 to 2022. The LSTM model achieved lower RMSE values (0.986 and 0.989), showing better stability and accuracy, making it more suitable for supporting price control policies. Although fewer studies focus on GRU in rice price forecasting, studies [8], [9] show that GRU performs well in time series tasks with faster training and comparable accuracy to LSTM, making it a promising alternative.

This study aims to assess how well LSTM and GRU perform in forecasting future rice prices in Indonesia. This study evaluates both models. LSTM and GRU are suitable for this task because they can capture sequential patterns and temporal dependencies in time series data, which are key characteristics of price fluctuations over time. Their ability to learn from historical data enables accurate prediction of future price movements, making them effective tools for economic and agricultural forecasting. The contribution of this research is to provide more comprehensive rice price forecasting using the latest data. Unlike prior studies that focused solely on LSTM or applied GRU to unrelated domains, this study offers a direct comparison of both models using recent national rice price data from 2021 to 2024. This comparative approach with updated data provides practical insights into which model is more reliable for real-world rice price prediction. Long-term support for national food security, the creation of more effective and sustainable food policies, and the achievement of SDG 2 in Indonesia are the expected outcomes of applying this technology. The findings provide valuable insights for policymakers in designing data-driven strategies to stabilize rice prices and ensure food security in Indonesia.

2. RESEARCH METHODS

2.1 Data Collection

The data source used in this study comes from the National Food Agency Website (https://newpanelharga.badanpangan.go.id/beranda) with 1,424 observations from the beginning of 2021 to the end of 2024. In this study, the variables used are the date and national rice price to predict the price of rice to overcome hunger in Indonesia. The original data is collected daily, however for model training and forecasting, the data is resampled into 3-day intervals to reduce noise and improve trend detection.

2.2 Recurrent Neural Network

RNN is part of the Neural Network family designed to process sequential data by utilizing its internal memory [10]. This memory allows the RNN to remember previously acquired information, so that the decision or outcome of an input is influenced by previous information [11]. RNN stores information from the past through a looping mechanism in its architecture, which automatically retains and processes data from previous sequences, making it ideal for processing continuous data [12].

2.3 Long Short-Term Memory

LSTM was first postulated in 1997 by Jürgen Schmid Huber and Sepp Hochreiter. The LSTM method was originally used to learn long-term dependencies in data by overcoming the vanishing gradient problem [13]-[15] LSTM is a special type of RNN that works better in practice, due to the updating of the equations and the backpropagation dynamics involved [16]. LSTM can detect data that should be kept and data that should be discarded, because it has four layers of neurons commonly called gates to manage memory in each neuron. LSTM has gates that function to delete or add information, namely forget gate, output gate, and input gate [17]-[19].

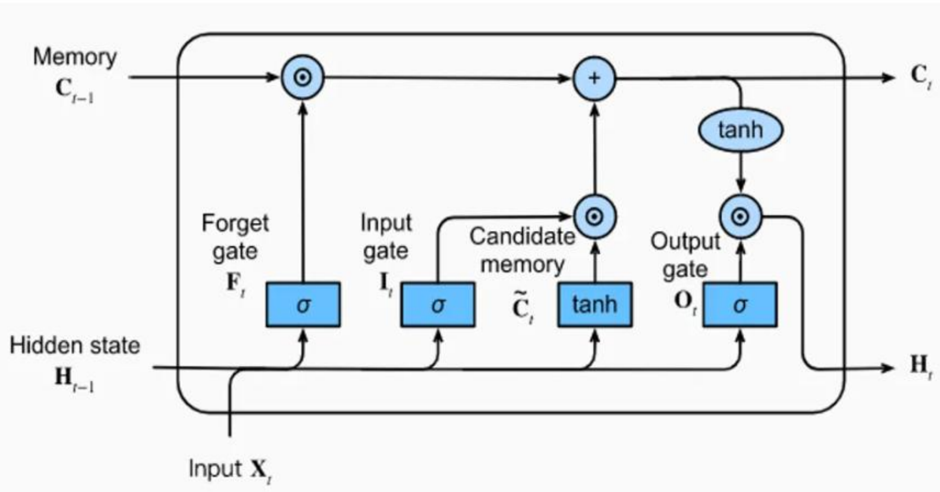

Figure 1. LSTM Architecture

The equations formed from forget gate, input gate, output gate, tanh layer creating a new candidate value vector, cell state, and hidden layer are [17], [20], [21].

2.3.1 Forget Gate

$$f_t = \sigma\left(W_{fx}x_t + W_{fh}h_t + W_{fc}c_{t-1} + b_f\right), \tag{1}$$

where:

f_t : Forget gate at t;
σ : Sigmoid activation function;
x_t : Input at t;

h_t : Hidden state at t;
c_t : Cell state at t;
W_{fx} : Weights to determine x_t;
W_{fh} : Weights to determine h_t;
W_{fc} : Weights to determine c_t;
b_f : Bias for forget gate.

2.3.2 Input Gate

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_t + W_{ic}c_{t-1} + b_i), \tag{2}$$

where:
i_t : Gate input at t;
σ : Sigmoid activation function;
x_t : Input at t;
h_t : Hidden state at t;
c_t : Cell state at t;
W_{ix} : Weights to determine x_t;
W_{ih} : Weights to determine h_t;
W_{ic} : Weights to determine c_t;
b_i : Bias for input gate.

2.3.3 Memory Update

$$c_t = f_t \text{ o } c_{t-1} + i_t \text{ o } \phi(W_{cx}x_t + W_{ch}h_{t-1} + b_c), \tag{3}$$

where:
c_t : Cell state at t;
f_t : Forget gate at t;
c_{t-1} : Cell state at $t-1$;
i_t : Input gate at t;
x_t : Input at t;
h_{t-1} : Hidden state at $t-1$;
W_{cx} : Weights to determine x_t;
W_{ch} : Weights to determine h_t;
b_c : Bias for cell state.

2.3.4 Output Gate

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_t + W_{oc}c_{t-1} + b_o), \tag{4}$$
$$h_t = o_t \text{ o } \phi(c_t), \tag{5}$$

where:
o_t : Gate output at t;
σ : Sigmoid activation function;
W_{ox} : Weights to determine x_t;
W_{oh} : Weights to determine h_t;
W_{oc} : Weights to determine c_t;
b_o : Bias for output gate.

2.4 Gate Recurrent Unit

The main purpose of creating a GRU is to make each recurrent unit capable of capturing dependencies in different time scales adaptively. GRU is a type of RNN. This model introduces a gating structure to read remote information. GRU is similar to LSTM, except it has fewer parameters because it does not have an output gate [8]. GRU only introduces two gates, namely update gate (z_t) and reset gate (r_t) [9].

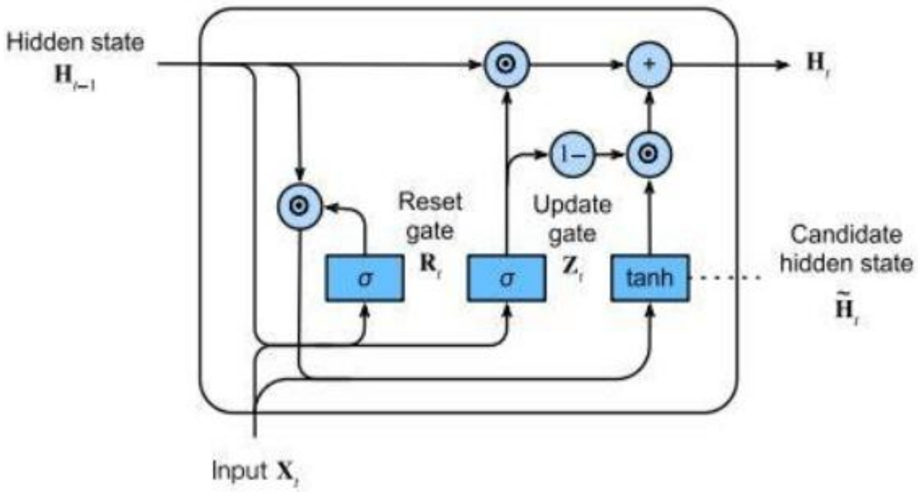

Figure 2. GRU Architecture

Update gate is useful for channeling information from previous inputs and outputs to the next cell, while reset gate is useful for determining which past information should be forgotten [9]. The way GRU works can be seen through the following formula [22].

$$z_t = \sigma\left(W_z * \left[h_{t-1}, x_t\right]\right), \tag{6}$$
$$r_t = \sigma\left(W_r * \left[h_{t-1}, x_t\right]\right). \tag{7}$$

After resetting the gate and updating the gate, the status of the GRU candidate value is h_t and the final output is h_t, where h is the (hidden state) at (time step) t, with the following formula [22]-[24].

$$\underline{h_t} = tanh(w_{\underline{h}} * [r_t * h_{t-1}, x_t]\,, \tag{8}$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \underline{h_t}. \tag{9}$$

where:

z_t : Update gate
r_t : Reset gate
$\underline{h_t}$: Candidate hidden gate
h_t : Candidate gate
σ : Sigmoid activation function
W_z : Weight for update gate
W_r : Weight for reset gate
$w_{\underline{h}}$: Weight for candidate hidden gate
h_{t-1} : Weight for candidate gate
x_t : Input at step t

2.5 Model Evaluation

A prediction accuracy number is obtained by evaluating the predictive model using test data after it has been produced during the training process. For comparing forecasting methods, we used RMSE. RMSE is calculated using the following equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(x_t - \hat{x}_t)^2} \tag{10}$$

If a model yields a lower RMSE value, it is deemed good. Here, n stands for the number of observations, x_t for the observed values, and $\hat{x}_t$ for the predicted values. The RMSE value ranges from $[0, \infty)$ [25]. Furthermore, the average absolute percentage error is computed using Mean Absolute Percent Error (MAPE). MAPE provides a measure of the difference between the series' actual value and the forecasting error. The MAPE equation can be seen as follows [22], [25], [26].

$$MAPE = \frac{\sum_{t=1}^{n} \left| \left(\frac{x_t - F_t}{x_t} \right) \times 100\% \right|}{n} \tag{11}$$

3. RESULTS AND DISCUSSION

3.1 Data Preprocessing

Data preprocessing is an important stage in data processing before it is used in analysis or machine learning model development. This stage includes various processes such as feature scaling, data cleaning, data transformation, data reduction, and data encoding, which aim to prepare raw data to suit the needs of analysis or models. One of the main steps in feature scaling is data normalization so that it has a uniform range of values. A commonly used method is min-max normalization, which changes the data scale so that it is within the range of 0 to 1. This normalization is very important to prevent the dominance of certain features due to differences in value scales, especially in the context of the LSTM and GRU models, which are very sensitive to input scales. By performing normalization, the training process becomes more stable and converges faster, thereby improving the overall performance and accuracy of the model.

3.2 Training, Testing, and Validation Data

To assess the effectiveness of the methods used, the data was divided into three parts, 80% for training, 10% for validation, and 10% for testing. The data was divided sequentially based on time (chronological split) to reflect a realistic time series forecasting scenario. After the preprocessing stage was completed, the training data was used to train the LSTM and GRU models to learn historical rice price patterns. During the training process, the model is run for a few epochs, and model parameters such as the number of neuron units, learning rate, and activation function are gradually adjusted through a trial-and-error approach to obtain the best results. Validation data is used in parallel during the training process to evaluate the model's performance on data not involved in weight updates, to prevent overfitting and determine the optimal point for stopping training (early stopping). After the model is trained, the test data is used as truly unseen data to objectively measure the model's prediction accuracy. The model's performance is measured using two primary evaluation metrics, RMSE and MAPE, which provide an indication of how close the prediction results are to the actual values. To find the best parameters for model creation, trial-and-error testing is employed.

Table 1. The Parameter Settings of LSTM and GRU

Parameter	Value
Learning Rate	0.0005
Epoch	100
Optimizer	Adam
Loss	Huber

Table 1 describes the parameter settings used for the LSTM model. The learning rate is set to 0.0005, which determines the step size for weight updates during training. The model is trained for 100 epochs, meaning it iterates over the dataset 100 times. The Adam optimizer is used to adjust model parameters efficiently, and the Huber loss function is selected to minimize the error while being robust to outliers. The Parameter settings for the GRU model as shown in Table 1. The batch size is set to 64, meaning the model processes 64 data samples at a time during training. The training process runs for 100 epochs, significantly more than the LSTM model. Similar to LSTM, the Adam optimizer is used for weight updates. Additionally, the "Verbose" parameter is set to 1, indicating that training progress and performance metrics are displayed during the process.

3.3 Long Short-Term Memory

In this method, there are several steps involved in building an LSTM model for data prediction. First, the data is imported and prepared using Python libraries such as NumPy, Pandas, Matplotlib, Seaborn, and TensorFlow. After the data is read, a normalization process is performed to ensure that the data scale is uniform, so that the model training process becomes more stable. To prevent overfitting, several regulation techniques are applied, such as the use of dropout and early stopping. Next, the data is divided into three parts: 80% for training, 10% for validation, and 10% for testing (Fig. 3).

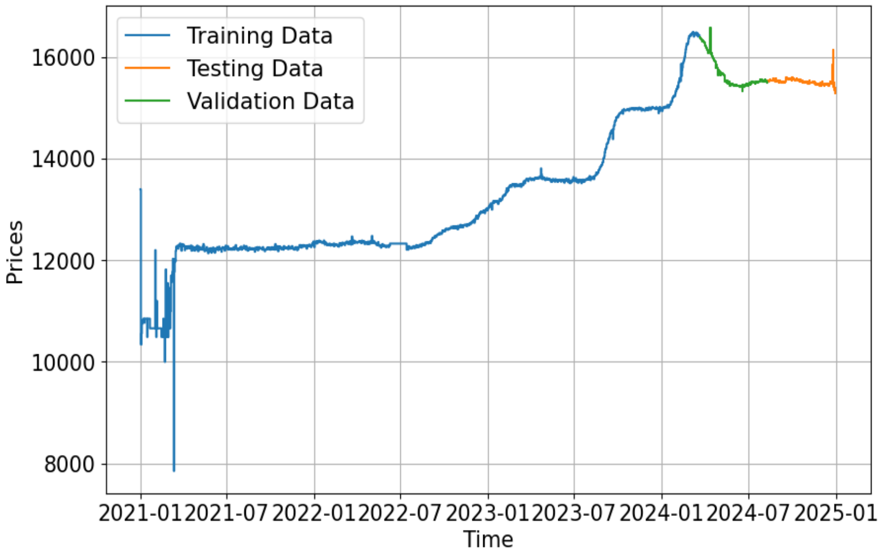

Figure 3. Plot of Data Sharing

The validation data is used during the training process to evaluate the model's performance on data not involved in the weight update process. The purpose is to monitor whether the model is overfitting to the training data and to assist in the optimal adjustment of parameters (such as the number of epochs or architecture configuration). Validation is performed periodically at each epoch. Meanwhile, testing data is used after the training process is complete, as data that the model has never seen before. The goal is to evaluate the model's final generalization ability on new data. Thus, the separation between validation data and testing data is done to ensure that model performance testing is conducted objectively and is not biased toward parameters adjusted during training. The following plots were formed.

The normalized training data is used to train the LSTM model following the data separation procedure. In this model, two LSTM layers are used, each consisting of 64 units, followed by a dropout layer to prevent overfitting, and a Dense (fully connected) layer to generate prediction outputs, dropouts to avoid overfitting, and using appropriate activation functions. During the training process, the model is optimized using Adam's algorithm and Huber's loss function to better capture the data patterns. Once training is complete, the model is tested using validation data to assess its ability to generalize data that has never been seen before. The evaluation was done using the RMSE and MAPE metrics, which showed that the model could capture the data patterns and trends quite well. There is still an opportunity for improvement, though, as there is a little discrepancy between the actual and expected values. The model was then used to perform value forecasting for a specific period by comparing the predicted results against the actual data. To make it easier to analyze trend patterns and assess how well the model can capture past data, the forecast results are displayed as graphs. Model evaluation is done by comparing actual with predicted data.

The RMSE value of 518.1 and the MAPE value of 3.49% during the training evaluation stage demonstrate that the predicted model and the actual data accord well. During the validation phase, the model is evaluated using previously unseen data to gauge its capacity for generalization. With an RMSE value of 489.8 and a MAPE value of 2.81%, the validation results demonstrate that the model can predict very well, despite a little discrepancy between the actual and projected data. The comparison plot between actual data and predicted data at both training and validation stages shows that the LSTM model performs quite well in capturing patterns and trends in the data. The plot of comparison between actual and prediction data in Fig. 4.

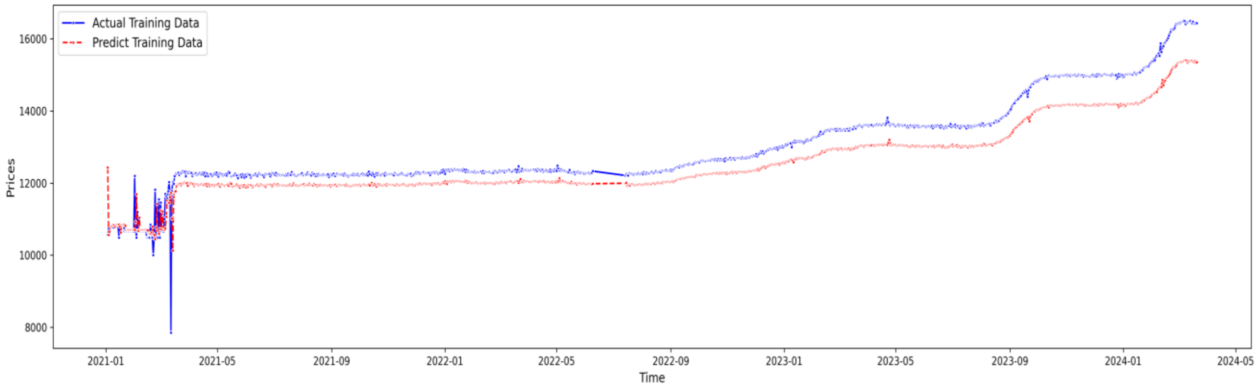

Figure 4. The Actual and Prediction of Training Data

Fig. 5 illustrates the comparison between actual and predicted validation data over time. The x-axis represents the timeline, while the y-axis shows the price values. The actual validation data (depicted in an orange dashed line) demonstrates a fluctuating downward trend, with a sharp peak around April 2024 before continuing its decline until mid-2024, followed by a slight upward trend. In contrast, the predicted validation data (depicted in a dark dashed line) follows a smoother downward trajectory with minimal fluctuations, failing to capture some of the sharp variations observed in the actual data. This suggests that the predictive model may not fully capture sudden market changes but maintains a relatively stable approximation of the trend. The trained LSTM model is used to forecast the national rice price for the period January 2025 to February 2025. To illustrate the predictions made by the model, the forecasting results are displayed as tables and charts.

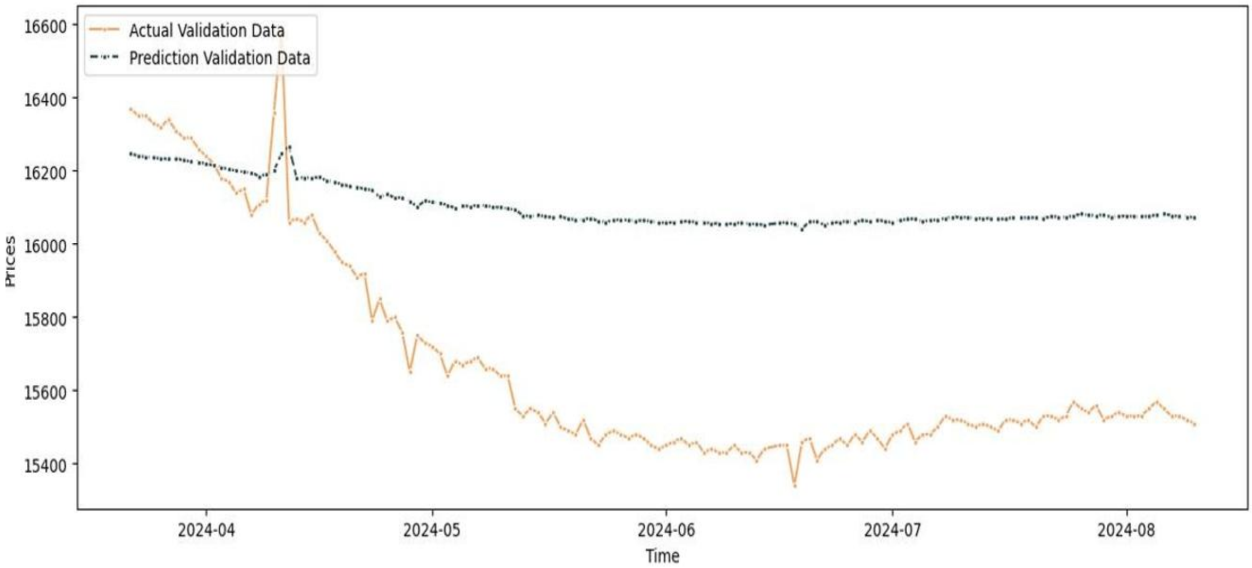

Figure 5. The Actual and Prediction of Validation Data

Table 2 presents the forecasting results of the LSTM model for rice prices over a series of dates in January and February 2025. It consists of three columns: the sequential number (No), the forecast date (Date), and the predicted price value (Forecast). The results show a gradual increase in predicted prices over time, starting from 15,319.29 on January 1, 2025, and reaching 15,810.96 on February 21, 2025. This indicates an upward trend in rice prices based on the LSTM model's predictions.

Table 2. The LSTM Forecasting Results

No	Date	Forecast	No	Date	Forecast
1	01-01-2025	15319.294922	10	28-01-2025	15576.994141
2	04-01-2025	15399.757812	11	31-01-2025	15603.788086
3	07-01-2025	15390.323242	12	03-02-2025	15631.656250
4	10-01-2025	15431.978516	13	06-02-2025	15659.846680
5	13-01-2025	15445.738281	14	09-02-2025	15688.808594
6	16-01-2025	15475.891602	15	12-02-2025	15718.319336

No	Date	Forecast	No	Date	Forecast
7	19-01-2025	15497.634766	16	15-02-2025	15748.532227
8	22-01-2025	15524.894531	17	18-02-2025	15779.392578
9	25-01-2025	15549.883789	18	21-02-2025	15810.961914

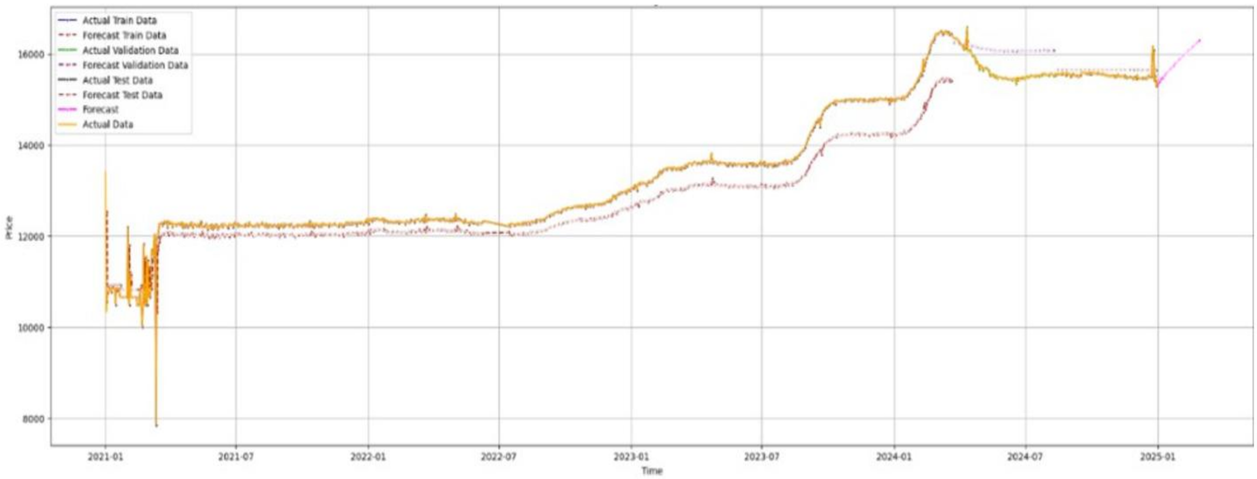

Figure 6. Graph of Actual Data and Forecasts from the LSTM Model

The actual data and the forecasted data are compared to assess the forecasting's accuracy. This plot includes training, validation, and forecast data. In this plot (Fig. 6), the actual data from the training and validation periods are compared with the predictions from the LSTM model. The plot shows that the model's forecast results follow the pattern and trend of the actual data quite well.

Overall, the model can capture the key characteristics of national rice price fluctuations, despite occasional discrepancies between the actual and predicted data outputs. Performance of the model is then clearly visualized by integrating the actual and forecast data into a plot (Fig. 7).

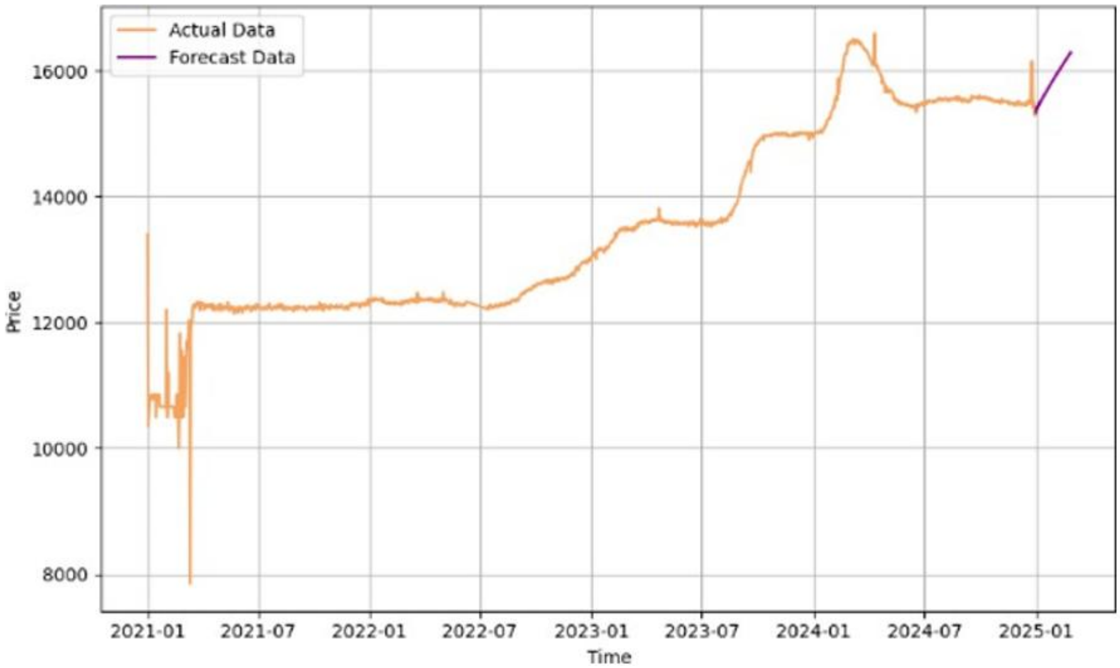

Figure 7. Graph Forecasts Using the LSTM Model

3.4 Gate Recurrent Unit

In applying the GRU method, the steps taken are similar to the process in the LSTM model. First, the data is read and normalized to ensure scale uniformity. The model is then built using a GRU architecture with a number of hidden layers and dropout to prevent overfitting. The data is divided into three parts: 80% for training, 10% for validation, and 10% for testing, with the division done chronologically to maintain the time sequence in the data series. During the training process, the model is trained using the training data and validated at each epoch using the validation data to observe performance and prevent overfitting. This validation helps determine when the model begins to experience a decline in performance on untrained data, so it can be used for early stopping or parameter adjustment.

After the training process is complete, the model is evaluated using test data that the model has never seen before. The evaluation is carried out using the RMSE and MAPE metrics to measure how accurately the model predicts rice prices on new data. This process ensures that the GRU model is not only accurate on the training data but also has good generalization capabilities on data that has never been evaluated before. The following plot below shows the data that has been divided.

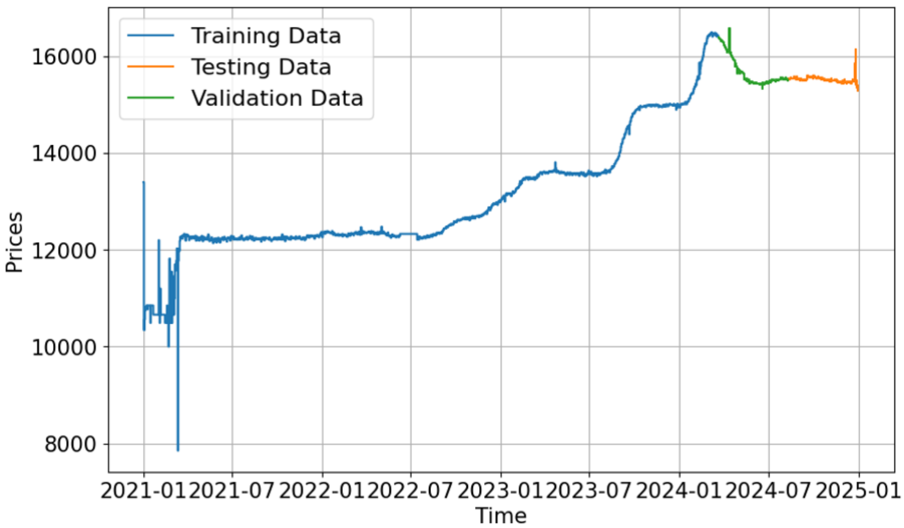

Figure 8. Data Sharing Results

The GRU model is built by adding multiple layers, dropouts to avoid overfitting, and using appropriate activation functions. Adam's approach to the Mean Squared Error (MSE) loss function was used to optimize the model during training in order to enhance its capacity to identify patterns in the data. For the model to converge well, 200 epochs and a batch size of 32 were used during training. After the training is complete, the model is tested using validation data to assess its ability to generalize to data that has never been seen before. The evaluation is done using RMSE and MAPE metrics, where the GRU model shows a MAPE value of 1.08%, which is lower than the LSTM. To make it easier to analyze trend patterns and assess how well the model can reflect past data, the forecast results are then displayed as graphs.

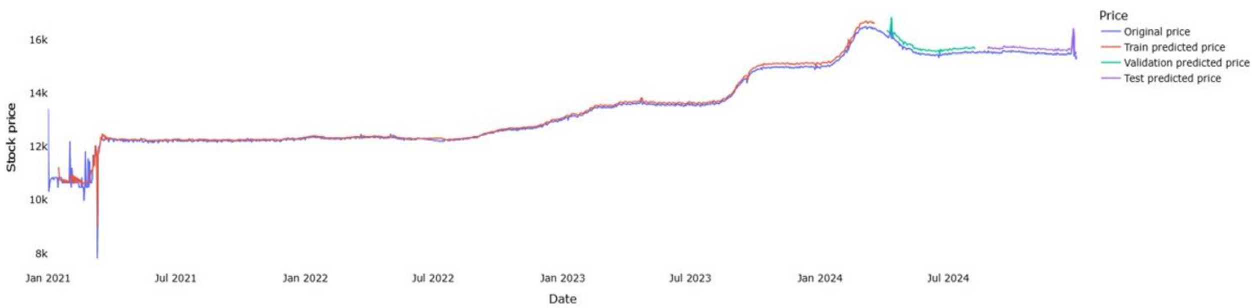

Figure 9. Comparison of Actual and Predicted Price

The national daily rice price is then predicted for the period January 2025 to February 2025 using the GRU model that has been created. The prediction results are then visualized using Table 3. The results show that GRU is superior to LSTM in predicting national rice prices, with a MAPE value of 1.08% for GRU and 3.49% for LSTM. GRU's simplified model architecture explains its superior performance. With just two primary gates—the update gate and the reset gate—GRU enables the model to effectively decide which data should be retained or deleted without the need for an additional state cell. In contrast, LSTM has three main gates and additional state cells, which increases complexity but does not necessarily provide an advantage on data with moderate temporal patterns.

Table 3. The GRU Forecasting Results

No	Date	Forecast	No	Date	Forecast
1	01-01-2025	15394.469727	10	28-01-2025	15585.597656
2	04-01-2025	15369.442383	11	31-01-2025	15613.161133
3	07-01-2025	15419.353516	12	03-02-2025	15640.902344
4	10-01-2025	15427.994141	13	06-02-2025	15669.478516
5	13-01-2025	15460.508789	14	09-02-2025	15698.548828
6	16-01-2025	15480.405273	15	12-02-2025	15728.331055
7	19-01-2025	15508.164062	16	15-02-2025	15758.734375
8	22-01-2025	15532.322266	17	18-02-2025	15789.841797
9	25-01-2025	15559.331055	18	21-02-2025	15821.639648

Table 3 presents the GRU forecasting results, displaying predicted price values for specific dates from January 1, 2025, to February 21, 2025. The table is divided into two sections, each containing forecasted values for different dates. The forecasted prices exhibit an overall increasing trend, starting from approximately 15,394 on January 1, 2025, and reaching around 15,821 by February 21, 2025. This suggests that the GRU model predicts a gradual rise in prices over the given period. The results indicate a relatively smooth progression without significant fluctuations, reflecting the model's capability to capture long-term trends while potentially smoothing out short-term variations.

4. CONCLUSION

The Long Short-Term Memory and Gated Recurrent Unit algorithms are used to predict national rice price fluctuations. Results indicate that GRU outperforms LSTM in terms of accuracy (MAPE 1.08% vs. 3.49%). GRU effectively captures seasonal price patterns without the additional capacity needed by LSTM and helps prevent overfitting, particularly in datasets with short-term temporal dependencies. While LSTM is advantageous for handling longer or more complex temporal patterns, this study highlights that GRU is a more effective and efficient choice for national rice price forecasting. Its implementation can contribute to maintaining price stability, enhancing food security, and supporting sustainable development goals in Indonesia.

Author Contributions

Triyani Hendrawati: Conceptualization, Supervision, Project Administration, Funding Acquisition, Writing - Original Draft, Writing – Review and Editing. Kennedy Marthendra: Data Curation, Formal Analysis, Software, Methodology, and Writing the Original Draft. Brian Riski Jayama Simanjuntak: Data Curation, Formal Analysis, Investigation, Resources and Writing the Original Draft. Anindya Aprilianti Pravitasari: Validation, Visualization, Writing - Review and Editing. All authors discussed the results and contributed to the final manuscript.

Funding Statement

This research was funded by the Internal Matching Funds Research Grant (IMF) at Universitas Padjadjaran, Indonesia, for the project titled "Model Classification of Rice Plant Diseases Based on Deep Learning and Gaussian Copula to Support Sustainable Precision Agriculture," under contract number 4356/UN6.D/PT.00/2025.

Acknowledgment

The authors are grateful to the Research Center for AI and Big Data Universitas Padjadjaran and the Directorate for Research and Community Service (DRPM) Universitas Padjadjaran which supports this research.

Declarations

The authors declare no conflicts of interest to report study.

REFERENCES

[1] W. Arifin, I. S. Rianse, and H. Sudarmo, "ANALISIS FLUKTUASI HARGA BERAS DI TINGKAT PETANI, PEDAGANG, DAN PENGECER DI KOTA KENDARI," *Gabbah: Jurnal Pertanian Dan Perternakan*, vol. 2, no. 1, pp. 36–48, Oct. 2024, doi: https://doi.org/10.62017/gabbah.v2i1.2094.

[2] T. Ulussever, H. M. Ertuğrul, S. Kılıç Depren, M. T. Kartal, and Ö. Depren, "ESTIMATION OF IMPACTS OF GLOBAL FACTORS ON WORLD FOOD PRICES: A COMPARISON OF MACHINE LEARNING ALGORITHMS AND TIME SERIES ECONOMETRIC MODELS," *Foods*, vol. 12, no. 4, p. 873, Feb. 2023, doi: https://doi.org/10.3390/foods12040873.

[3] C. C. Anderson, M. Denich, A. Warchold, J. P. Kropp, and P. Pradhan, "A SYSTEMS MODEL OF SDG TARGET INFLUENCE ON THE 2030 AGENDA FOR SUSTAINABLE DEVELOPMENT," *Sustain Sci*, vol. 17, no. 4, pp. 1459–1472, Jul. 2022, doi: https://doi.org/10.1007/s11625-021-01040-8.

[4] P. Atukunda, W. B. Eide, K. R. Kardel, P. O. Iversen, and A. C. Westerberg, "UNLOCKING THE POTENTIAL FOR ACHIEVEMENT OF THE UN SUSTAINABLE DEVELOPMENT GOAL 2 – 'ZERO HUNGER' – IN AFRICA: TARGETS, STRATEGIES, SYNERGIES AND CHALLENGES," *Food Nutr Res*, vol. 65, May 2021, doi: https://doi.org/10.29219/fnr.v65.7686.

[5] A. W. Putra, J. Supriatna, R. H. Koestoer, and T. E. B. Soesilo, "DIFFERENCES IN LOCAL RICE PRICE VOLATILITY, CLIMATE, AND MACROECONOMIC DETERMINANTS IN THE INDONESIAN MARKET," *Sustainability*, vol. 13, no. 8, p. 4465, Apr. 2021, doi: https://doi.org/10.3390/su13084465.

[6] R. Hidayat and I. Wibisonya, "RICE PRICE PREDICTION WITH LONG SHORT-TERM MEMORY (LSTM) NEURAL NETWORK," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 5, pp. 658–664, Oct. 2024, doi: https://doi.org/10.29207/resti.v8i5.6041.

[7] A. Hanafiah, Y. Arta, H. O. Nasution, and Y. D. Lestari, "PENERAPAN METODE RECURRENT NEURAL NETWORK DENGAN PENDEKATAN LONG SHORT-TERM MEMORY (LSTM) UNTUK PREDIKSI HARGA SAHAM," *Bulletin of Computer Science Research*, vol. 4, no. 1, pp. 27–33, Dec. 2023, doi: https://doi.org/10.47065/bulletincsr.v4i1.321.

[8] M. Abumohsen, A. Y. Owda, and M. Owda, "ELECTRICAL LOAD FORECASTING USING LSTM, GRU, AND RNN ALGORITHMS," *Energies (Basel)*, vol. 16, no. 5, p. 2283, Feb. 2023, doi: https://doi.org/10.3390/en16052283.

[9] Y. Gao, R. Wang, and E. Zhou, "STOCK PREDICTION BASED ON OPTIMIZED LSTM AND GRU MODELS," *Sci Program*, vol. 2021, pp. 1–8, Sep. 2021, doi: https://doi.org/10.1155/2021/4055281.

[10] A. Hanafiah, Y. Arta, H. O. Nasution, and Y. D. Lestari, "PENERAPAN METODE RECURRENT NEURAL NETWORK DENGAN PENDEKATAN LONG SHORT-TERM MEMORY (LSTM) UNTUK PREDIKSI HARGA SAHAM," *Bulletin of Computer Science Research*, vol. 4, no. 1, pp. 27–33, Dec. 2023, doi: https://doi.org/10.47065/bulletincsr.v4i1.321.

[11] S. J. Pipin, R. Purba, and H. Kurniawan, "PREDIKSI SAHAM MENGGUNAKAN RECURRENT NEURAL NETWORK (RNN-LSTM) DENGAN OPTIMASI ADAPTIVE MOMENT ESTIMATION," *Journal of Computer System and Informatics (JoSYC)*, vol. 4, no. 4, pp. 806–815, Aug. 2023, doi: https://doi.org/10.47065/josyc.v4i4.4014.

[12] A. Hanifa, S. A. Fauzan, M. Hikal, and M. B. Ashfiya, "PERBANDINGAN METODE LSTM DAN GRU (RNN) UNTUK KLASIFIKASI BERITA PALSU BERBAHASA INDONESIA," *Dinamika Rekayasa*, vol. 17, no. 1, p. 33, Jan. 2021, doi: https://doi.org/10.20884/1.dr.2021.17.1.436.

[13] M. Yang and J. Wang, "ADAPTABILITY OF FINANCIAL TIME SERIES PREDICTION BASED ON BILSTM," *Procedia Comput Sci*, vol. 199, pp. 18–25, 2022, doi: https://doi.org/10.1016/j.procs.2022.01.003.

[14] D. I. Puteri, "IMPLEMENTASI LONG SHORT TERM MEMORY (LSTM) DAN BIDIRECTIONAL LONG SHORT TERM MEMORY (BILSTM) DALAM PREDIKSI HARGA SAHAM SYARIAH," *Euler : Jurnal Ilmiah Matematika, Sains dan Teknologi*, vol. 11, no. 1, pp. 35–43, May 2023, doi: https://doi.org/10.34312/euler.v11i1.19791.

[15] F. Aksan, Y. Li, V. Suresh, and P. Janik, "CNN-LSTM VS. LSTM-CNN TO PREDICT POWER FLOW DIRECTION: A CASE STUDY OF THE HIGH-VOLTAGE SUBNET OF NORTHEAST GERMANY," *Sensors*, vol. 23, no. 2, p. 901, Jan. 2023, doi: https://doi.org/10.3390/s23020901.

[16] D. I. Mulyana, Y. K. H. Sanur, F. Yadi, Sahroni, and A. S. Sumarsono, "PENERAPAN METODE NEURAL NETWORK DENGAN STRUKTUR BACKPROPAGATION UNTUK MEMPREDIKSI KEBUTUHAN STOK PADA TOKO UMKM PERLENGKAPAN BAYI BABYQU," *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 4, no. 1, pp. 121–128, Jan. 2023, doi: https://doi.org/10.35870/jimik.v4i1.131.

[17] S. A. Jofipasi, A. Salma, D. Vionanda, and D. Fitria, "PREDICTION OF BOGOR CITY RAINFALL PARAMETERS USING LONG SHORT TERM MEMORY (LSTM)," *UNP Journal of Statistics and Data Science*, vol. 1, no. 5, pp. 434–440, Nov. 2023, doi: https://doi.org/10.24036/ujsds/vol1-iss5/110.

[18] J. Cahyani, S. Mujahidin, and T. P. Fiqar, "IMPLEMENTASI METODE LONG SHORT TERM MEMORY (LSTM) UNTUK MEMPREDIKSI HARGA BAHAN POKOK NASIONAL," *Jurnal Sistem dan Teknologi Informasi (JustIN)*, vol. 11, no. 2, p. 346, Jul. 2023, doi: https://doi.org/10.26418/justin.v11i2.57395.

[19] E. Supriyadi, "PREDIKSI PARAMETER CUACA MENGGUNAKAN DEEP LEARNING LONG-SHOOT TERM MEMORY(LSTM)," *Jurnal Meteorologi dan Geofisika*, vol. 21, no. 2, p. 55, Jan. 2021, doi: https://doi.org/10.31172/jmg.v21i2.619.

[20] M. D. A. Carnegie and C. Chairani, "PERBANDINGAN LONG SHORT TERM MEMORY (LSTM) DAN GATED RECURRENT UNIT (GRU) UNTUK MEMPREDIKSI CURAH HUJAN," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 3, p. 1022, Jul. 2023, doi: https://doi.org/10.30865/mib.v7i3.6213.

[21] Y. Karyadi, "PREDIKSI KUALITAS UDARA DENGAN METODA LSTM, BIDIRECTIONAL LSTM, DAN GRU," *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 1, pp. 671–684, Mar. 2022, doi: https://doi.org/10.35957/jatisi.v9i1.1588.

[22] S. N. Haliza and M. A. Rosid, "DEEP LEARNING STACK ENSEMBLE TO DETECT SARCASM IN NEWS HEADLINE DATASET," Apr. 06, 2023. doi: https://doi.org/10.21070/ups.689.

[23] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "EMPIRICAL EVALUATION OF GATED RECURRENT NEURAL NETWORKS ON SEQUENCE MODELING," Dec. 2014.

[24] K. Cho et al., "LEARNING PHRASE REPRESENTATIONS USING RNN ENCODER-DECODER FOR STATISTICAL MACHINE TRANSLATION," Jun. 2014.doi: https://doi.org/10.3115/v1/D14-1179

[25] X. J. Luo et al., "GENETIC ALGORITHM-DETERMINED DEEP FEEDFORWARD NEURAL NETWORK ARCHITECTURE FOR PREDICTING ELECTRICITY CONSUMPTION IN REAL BUILDINGS," *Energy and AI*, vol. 2, p. 100015, Nov. 2020, doi: https://doi.org/10.1016/j.egyai.2020.100015.

[26] R. Bakri, S. Inayati, Y. Yuliana, and A. Hanafiah, "PREDIKSI JUMLAH PESERTA BPJS PENERIMA BANTUAN IURAN (PBI) APBN MENGGUNAKAN FUZZY TIME SERIES," *Barekeng: Jurnal Ilmu Matematika dan Terapan*, vol. 15, no. 2, pp. 373–384, Jun. 2021, doi: https://doi.org/10.30598/barekengvol15iss2pp373-384.