Variance: Journal of Statistics and Its Applications
P-ISSN: 2685-8738 E-ISSN: 2685-872X
Volume 7 Issue 1 | April 2025
10.30598/variancevol7iss1page1-10

IMPLEMENTATION OF LONG SHORT-TERM MEMORY (LSTM) IN FORECASTING THE NUMBER OF TRAIN PASSENGERS IN JAVA ISLAND

Naftali Brigitta Gunawan[1*], Wiwik Wiyanti[2]
[1,2]Department of Statistics, Faculty of Science, Computer, and Mathematics, Matana University
Jl. CBD Barat Kav, RW.1, Curug Sangereng, Kelapa Dua, Tangerang Regency, 15810, Banten, Indonesia

Corresponding Author's Email: *naftali.gunawan@student.matanauniversity.ac.id*

Abstract: *For certain Indonesians, trains are a particularly popular form of land transportation. Every year, during specific seasons, this mode of transportation consistently experiences a surge in passenger numbers. Due to this, it is necessary to make accurate predictions to make policies, such as whether additional carriages are needed. The selection of prediction methods will significantly impact policymaking. One of the methods currently being developed for prediction is related to machine learning. This study aims to implement a forecasting method using machine learning that can be used to predict time series data. The machine learning used in this study is the Long Short-Term Memory (LSTM) method. In this study, we used time series data on the number of train passengers. The data used is secondary data from the Statistics Indonesia (BPS). The data analysis process in this study uses Python software. The results of this analysis show that the LSTM model has a high level of accuracy in prediction, indicated by the mean squared error value of 2,941,137.156 and MAPE of 0.07%. Forecasts show a gradual increase in the number of passengers, starting from 32,381 people in the first month to 33,068 people in the third month. These results indicate that the LSTM model is thought to be effective in predicting changes in the number of train passengers, and further research is needed to verify this assumption.*

Keywords: *LSTM, machine learning, predict, railway, science data.*

1. INTRODUCTION

Every aspect of modern life is evolving, and technology is no exception. Technology advancements in land transportation, such as trains, are one example. Previously pulled by horses and oxen, the wagon is now powered by engines and is called a train. This illustrates that as time passed, contrivers had the ambition to create more efficient vehicles for transporting commodities and people. It was a steam train before it evolved into an advanced electrical train. Then came the invention of the steam locomotive, which evolved into a train that was powered by diesel and electricity [1]. Electric-powered trains in Indonesia are continuously being developed by PT Industri Kereta Api Persero (INKA). This can be seen from the many products developed by PT INKA. The government is still developing Indonesia's railway transportation infrastructure to meet modern demands [2]. Trains have been one of the community's primary modes of transportation [3].

To realize this development, Martinus continues to monitor the passenger flow surges using trains as Public Relations of Indonesian Railways (KAI) [3]. This monitoring aims to make policies in transportation services so that the public is comfortable using trains. An example is the policy on whether additional tickets or carriages are required when passengers increase. According to data from 2023, KAI sold 549,612 long-distance tickets, accounting for 24% of all tickets sold over Christmas and New Year from December 21, 2023, to January 7, 2024 [4]. Naturally, it is necessary to forecast this kind of yearly spike in passenger volume to plan policies for more carriages or other measures.

In principle, the prediction of the number of train passengers can be made scientifically using statistical methods for forecasting. The number of train passengers has been predicted using various forecasting techniques, including SARIMA. Research [5] using SARIMA predicts the number of executive train passengers on the island of Java in March, approaching the Eid al-Fitr holiday, namely 4470 people. The forecast results show that the highest number of passengers is in January, namely 5506 people, while the lowest number is in October, namely 2233 people. The Holt-Winters Exponential Smoothing and Fuzzy Time Series Markov Chain methods can also be used for prediction, as research [6] shows that the Holt-Winters method is superior, with a MAPE of 3.0643%

Submitted: 2024-12-30 *Accepted: 2025-04-23*

compared to Fuzzy Time Series Markov Chain with 5.2955%. Research by [7] on predictions using Holt's Weighted Exponential Moving Average (H-WEMA) method is more accurate than the Weighted Exponential Moving Average (WEMA), with a MAPE value from H-WEMA of 3.06% and from WEMA of 7.42%. Holt's method applies time series data with a trend character. As in research [8], predicting time series data with a trend character obtained a minimal forecast error value, namely 0%-0.29%.

Furthermore, minimal forecast error values were found when Holt's forecasting algorithm was used to health time series data that showed trends [9]. Moreover, research by [10] using Google Trends data can predict that the number of train passengers in August 2021 will reach 9.03 million people; the number of passengers has increased because people's mobility during COVID-19 has been expanded and more controlled compared to the July 2021 period, thus increasing public confidence in carrying out mobilization by train. Researchers [11], using the state space method, produced a MAPE of 2.04%. Predictions can make use of technology as science and technology advance. Another approach that makes use of recently developed technology is machine learning.

There are several prediction studies using machine learning. Using machine learning can make it easier to work on data analysis because computers assist it. Long Short-Term Memory (LSTM) is one machine learning prediction technique. Using LSTM for prediction is not the first time this has been done. Several previous studies, such as research [12], used LSTM to predict weather, resulting in an RMSE of 1.7444 and a MAPE of 1.9499%. When [13] used LSTM to anticipate the pricing of staple foods, the RMSE for cooking oil predictions was 0.0313, and for bulk cooking oil predictions, it was 0.0531. Research by [14], to predict the condition of a 10 kV motor in a coal-fired power plant using LSTM produced a percentage error of 3.8%. This research aims to apply and understand the Long Short-Term Memory (LSTM) model in predicting the number of train passengers on Java Island and to evaluate the forecasting results produced by the LSTM model.

2. METHODOLOGY

2.1. Neural Network

Neural networks, also known as artificial neural networks, are machine learning programs or models helpful in making decisions by working like the human brain. It works by adopting a method that imitates the interaction of biological neurons in recognizing phenomena, then evaluating options and formulating conclusions [15]. The goal of simulating the natural brain drove the initial efforts to construct artificial neural networks (ANNs). A very general statistical framework for modeling posterior probabilities given a collection of samples (input data) was eventually found to be ANNs [16]. The neuron is the basic building block of a (artificial) neural network (ANN). The neuron is the fundamental unit of a (artificial) neural network (ANN) [16]. A neuron is a processing unit with a single output and some (often several) inputs. From the Figure 1, x_i is weighted by a factor of w_i and the whole some input is calculated a. Then the activation function f is applied to the result a. The output is taken to be $f(a)$. See figure 1.

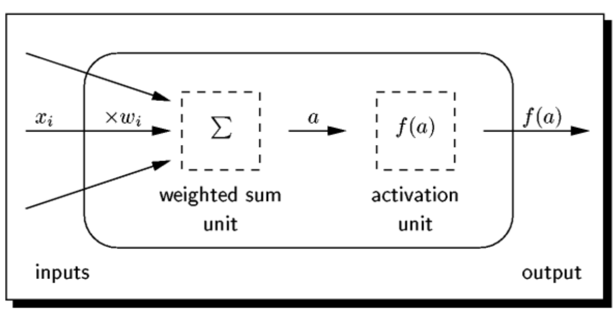

Figure 1. The Neuron [16]

A Recurrent Neural Network (RNN) is a feedback network in which some of its outputs are linked to their inputs [17]. RNN is an artificial neural network designed to process sequential or time-related data. This algorithm effectively deals with problems involving time sequences, including language translation, natural language processing, speech recognition, and captioning images [17]. To develop a machine learning model that can generate sequential predictions or inferences based on sequential input, a recurrent neural network, or RNN, is a deep neural network trained on sequential or time series data [18].

2.2. Long Short-Term Memory (LSTM)

The most widely used Recurrent Neural Network (RNN) type is called an LSTM network. Simple RNNs struggle to learn long-term dependencies in real-world scenarios. Usually trained using backpropagation, RNNs may experience "vanishing" or "exploding" gradient issues. These issues limit the efficacy of learning long-term links by making the network weights either extremely tiny or extremely big, respectively. LSTM networks are a special type of RNN designed to overcome the limitations of standard RNNs. By incorporating additional gates, LSTMs control the flow of information from the hidden state to the output and the next hidden state. This structure enables the model to capture long-term dependencies in the data more effectively. One kind of RNN that is frequently used is LSTMs [19]. Figure 2, illustrate the difference of RNN and LSTM.

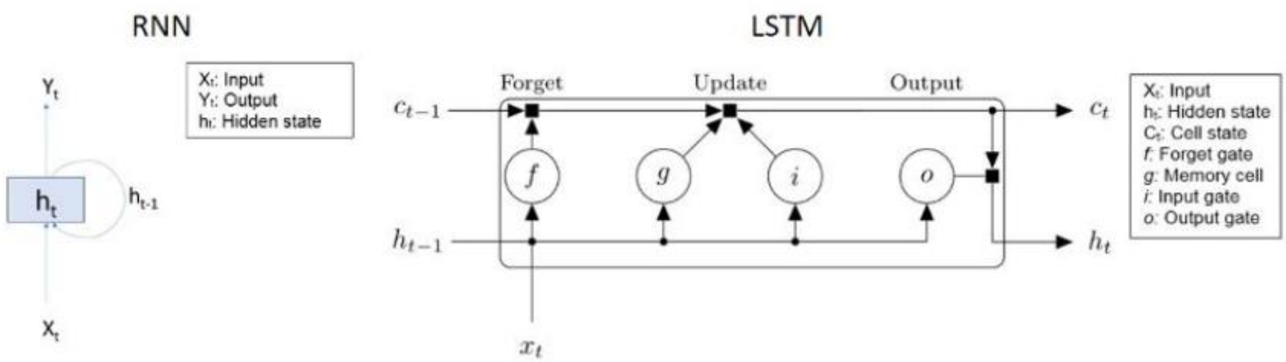

Figure 2. Comparison of an RNN (Left) and LSTM Network (Right) [19]

The hyperbolic tangent function $(\tanh(x))$ is used as an activation function in LSTM to regulate the updating and normalization of the cell states, ensuring that the resulting values remain within a stable range for computational efficiency and accuracy of the LSTM model.

$$tanh(x) = \frac{sinh\ (x)}{cosh\ (x)} = \frac{e^x - e^{-x}}{e^x + e^x} = \frac{e^x - 1}{e^x + 1} \tag{1}$$

$$h_t = tanh\ (W_{xhh}[x_t, h_{t-1}] + b_h) \tag{2}$$

where:

$tanh(x)$: Normalized output (-1 to 1).

e^x : Exponential of x.

h_t : Hidden state.

W_{xhh} : Matrix connected with hidden state.

x_t : Input from time-t.

h_{t-1} : Hidden state from previous time.

b_h : Bias vector output.

2.3. Forget Gate

The first component is the forget gate, which determines which information should be kept or discarded. It processes data from the previous hidden state and the current input through a sigmoid activation function, generating values between 0 and 1—where values near 0 suggest forgetting the information, and values near 1 indicate retaining it [20].

In Figure 3, an illustration of the forget gate, there is a blue circle representing the tanh symbol in function (1) and a red circle representing the sigmoid function. The sigmoid function plays a role in regulating the flow of information through the forget gate.

$$\sigma(x) = \frac{1}{1+e^{x}} = \frac{e^{x}}{1+e^{x}} = 1 - \sigma(-x) \quad (3)$$

where:
$\sigma(x)$: Sigmoid.
e^{x} : Exponential of x.

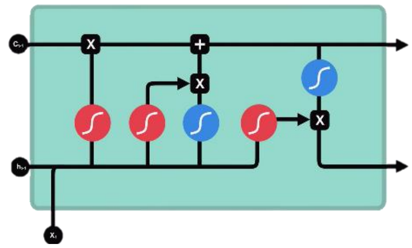

Figure 3. Illustration of Forget Gate [20]

2.4. Input Gate

The input gate processes the previous hidden state and the latest input through a sigmoid function to update the cell state, which sorts out important information (values close to 1) from unimportant information (values close to 0). Next, both inputs are also processed through the tanh function to normalize the values between -1 and 1, which are then multiplied by the sigmoid result to determine what information will be retained in the network [20]. Equation (4) and Equation (5) are two final formulas in the input gate results [20].

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \quad (4)$$

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

where h_{t-1} is the previous hidden state.

2.5 Cell State

At this stage, the cell state is updated using the forget vector by performing element-wise multiplication, allowing certain values to be diminished or removed if the result approaches zero. Then, the outputs from the input gate are added element-wise to introduce new, relevant information selected by the network[20]. Output result in Equation (6).

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

2.6. Output Gate

The final phase in the LSTM process is the output gate, which controls the content of the next hidden state. This hidden state is essential for generating predictions, as it carries information derived from previous inputs. At the gate output result, it will produce,

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

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

where o_t is gate output.

2.7. Python

Python is a programming language designed to improve work efficiency and support seamless system integration. It is developed under an OSI-approved open-source license, allowing unrestricted use and distribution for commercial purposes [21]. In predicting the number of passengers using LSTM with Python, the author imports the necessary libraries for data processing, modeling, and visualization. MinMaxScaler is used for data normalization, Pandas for data manipulation, Numpy for numerical operations, Keras for building and training LSTM models, and Plotly for visualizing results. Data is uploaded from a local file and read using Pandas. The data is filtered to use the 'WAKTU' column as the index and the 'JAWA' column as the data to be analyzed. Then, the data is normalized into a range of 0 to 1 using MinMaxScaler to ensure consistent scaling.

Furthermore, a function generates data sequences of a certain length (sequence length), which is used as input and output in the LSTM model. Next, the sequence length is set, and the input and output data are prepared in a form suitable for the LSTM model. Scikit-Learn's train_test_split function was used to divide the data into training and testing sets [22] .

The LSTM model is constructed with a single LSTM layer followed by a dense output layer. It is compiled using the 'adam' optimization algorithm and the 'mean_squared_error' loss function. The training process is conducted for 20 epochs with a batch size of 1, using training data, while validation is carried out using testing data. After the training is completed, the model is employed to generate predictions on the test dataset. These predictions are then converted to their original scale using the inverse_transform method [23].

Model performance is evaluated by calculating and printing the Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE). Predictions for the next three months are made using the last sequence of testing data as the initial input [24].

The predicted and actual values are visualized using plotly.go.figure(), which is used to create a graph object, and add_trace(), which is used to add lines representing the actual and predicted values. Next, update_layout() adjusts the graph's appearance, clearly comparing actual and predicted values [23].

2.8. Mean Square Error (MSE)

MSE function is a metric that measures how well a model predicts data. It's used in statistics and machine learning. [25]. Mathematically, MSE is defined in Equation (9) according to [25] as follow:

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(Y_i - \widehat{Y}_i)^2 \tag{9}$$

where:

Y_i : Data total

$\widehat{Y}_i$: Time

PE_t : Percentage error.

2.9. Mean Absolute Percentage Error (MAPE)

MAPE is useful to show how big the forecast error is compared to the actual value [25]. Mathematically, MAPE is defined in Equation (10) according to [25] as follow:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}|PE_t| \tag{10}$$

and

$$PE_t = \left(\frac{Y_t - F_t}{Y_t}\right) \times 100 \tag{11}$$

where:

n : Data total

t : Time

PE_t : Percentage error.

3. RESULTS AND DISCUSSION

3.1. Data

This study uses secondary data from the Statistics Indonesia (BPS). The data used is the number of train passengers on Java Island from January 2021 to December 2023. The amount of data used was 36 pieces of data [26]. This research method is a quantitative research design. Prediction data analysis in this study uses the LSTM method with the help of Python.

3.2. Flow Chart of Research

This study utilizes the LSTM model, with a flowchart that begins with data collection and concludes with forecasting for the next three months. The process starts with gathering the required data, which is uploaded and read into the system. After successful data input, normalization is performed to adjust the data to a suitable scale for modeling. Following normalization, the dataset is split into two subsets—training and testing data—to facilitate model training and evaluation processes.

Once the data is ready, the process continues to build the LSTM model. This model is then trained using the prepared training data. After training, the model is evaluated using the Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) metrics to measure the prediction accuracy. You can view the flowchart in Figure 4.

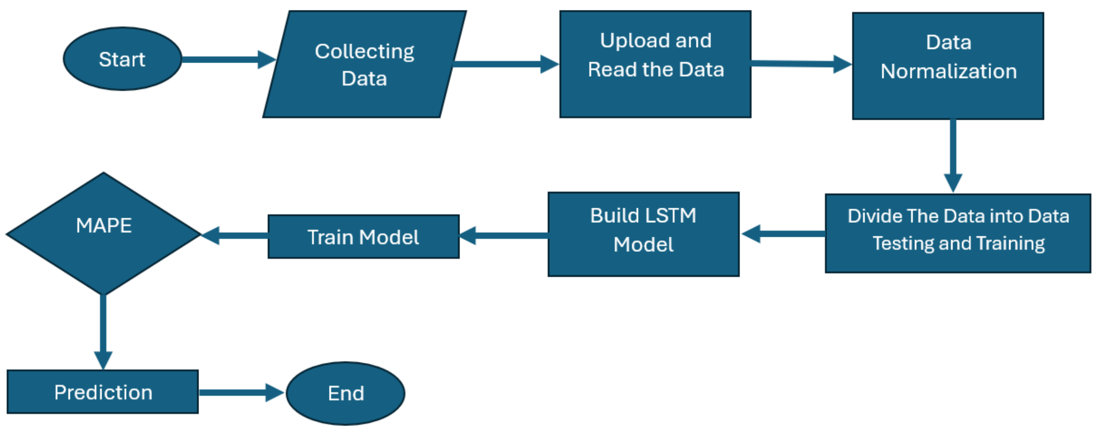

Figure 4. Flowchart of This Study

3.3. Result of Prediction

The data analysis process in this study was carried out through several systematic stages. The stages are carried out according to the flowchart, with the results being as follows:

The historical data on train passenger numbers in chronological order is the first step; this study's data is monthly. Examples of sequential data on the number of train passengers utilized in this investigation are shown in Table 1. The number of train passengers each month in the Java region between January 2021 and December 2023 served as the study's data source.

Table 1. The Data

Time	Count of the Train Passengers
01/01/2021	11,631
01/02/2021	11,223
01/03/2021	13,995
⋮	⋮
01/11/2023	32,424
01/12/2023	34,500

Next, the data that has been collected is then uploaded and read using the programming software used in this study, namely Python, with the "pandas" library to load the dataset into the data structure. An illustration of the process can be seen in Figure 5. When data is entered into Python, it automatically saves the data used in the research.

The next step is to normalize the data read into the range [0,1] using "Min-Max Scaling." This process is important to adjust the data scale to the characteristics of the activation functions in the LSTM network, especially sigmoid and $\tanh(x)$. The Python coding in this process can be seen in Figure 5.

```
# Normalize the data
scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(data)
```

Figure 5. Code of The Data Process Upload by Python

```
# Upload and read the data
uploaded = files.upload()
data = pd.read_excel(io.BytesIO(uploaded['DATA_PAPER.xlsx']))
data = data[['WAKTU', 'Jawa']].set_index('WAKTU')
Choose Files DATA_PAPER.xlsx
• DATA_PAPER.xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 10281 bytes, last modified: 02/05/2024 - 100% done
Saving DATA_PAPER.xlsx to DATA_PAPER.xlsx
```

Figure 6. Normalizing The Data Process

After data normalization is done, the next step is to divide the data into two parts, namely training and testing data. In this study, 80% was used as training data for the first 29 data periods (January 2021 to May 2023). Then, 20% of the test data used the last 7 data points (June 2023 to December 2023).

The next step is to build an LSTM model using several main layers: the input layer, LSTM layer, and dense (output) layer. The internal working mechanism of LSTM refers to the mathematical formulas from Equations (2) to (8), which explain the role of the input gate, forget gate, cell state, output gate, and hidden state components in filtering and forwarding information at each time step. This model is implemented using the Sequential API. The code for building the model can be seen in Figure 7. The output results can be seen in Table 2.

```
# Build the LSTM model
model = Sequential()
model.add(LSTM(units=50, return_sequences=False, input_shape=(seq_length, 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mean_squared_error')
```

Figure 7. Build The LSTM Model

Table 2. Output of the LSTM Model

Layer (type)	Output Shape	Total Parameter
LSTM	(None, 50)	10,400
Dense	(None, 1)	51
Total		10,451
Trainable params		10,451
Non-trainable params		0

Table 2 illustrates that the model architecture consists of a single LSTM layer with 50 memory units designed to capture temporal dependencies, followed by a dense output layer with one neuron responsible for generating predictions. The total number of trainable parameters is 10,451, which reflects a moderate level of complexity—lightweight yet practical for recognizing patterns in time series data.

Subsequently, the model was trained on historical data using the Adam Optimizer, with MSE as the loss function. The training results yielded an MSE value of 0.00027, indicating that the model has effectively learned the underlying data patterns, as the error is close to zero. Additionally, the MAPE was recorded at 0.07%, suggesting that the model's predictions are highly accurate and closely match the actual values.

Lastly, the prediction results are presented in Figure 8. The forecasted number of passengers in the first month was 32,381; in the second month, it rose to 32,883; and in the third month, it reached 33,068 passengers. From the three-results prediction value, it shows that the number of passengers is increasing every month or there is an increasing trend.

Figure 8. Predicted Number of Passengers

4. CONCLUSION

From the research, applying the LSTM method in predicting the number of train passengers on Java Island showed minimal prediction errors. The LSTM method successfully identified temporal patterns from passenger data quite accurately, as seen from the mean squared error value of 2,941,137.156 and MAPE of only 0.07%. The forecast results show an increase in the number of passengers from the first month by 32,381 people, the second month by 32,883 people, and the third month by 33,068 people. These findings prove the effectiveness of the LSTM model in predicting passenger number fluctuations, which can be very useful for capacity planning and resource management in railway services.

For further research, inferentially examine the effectiveness of LSTM in analyzing data patterns that share similar characteristics with the data on the number of train passengers. Besides that, it is recommended to integrate external variables such as weather conditions and special events like national holidays that may affect the number of passengers.

REFERENCES

[1] D. P. Aceh, “Siapa Penemu Kereta Api: Sejarah Kereta Api,” 29 Agustus. Accessed: Sep. 15, 2023. [Online]. Available: https://dishub.acehprov.go.id/siapa-penemu-kereta-api-sejarah-kereta-api/

[2] Biro Informasi dan Komunikasi Publik, “Memacu Pengembangan Infrastruktur Perkeretaapian Indonesia,” 28 Juni. [Online]. Available: https://dephub.go.id/post/read/memacu-pengembangan-infrastruktur-perkeretaapian-indonesia

[3] B. K. dan I. Publik, “KERETA API, MODA TRANSPORTASI ANDALAN MASYARAKAT,” 14 Agustus. Accessed: Sep. 15, 2024. [Online]. Available: https://dephub.go.id/post/read/kereta-api-moda-transportasi-andalan-masyarakat-14311

[4] S. Tobing, “KAI Tambah 34 Kereta Api Tambahan Jelang Nataru, Ini Daftarnya,” 29 November. Accessed: Apr. 19, 2024. [Online]. Available: https://katadata.co.id/berita/industri/65671ee063f5b/kai-tambah-34-kereta-api-tambahan-jelang-nataru-ini-daftarnya

[5] L. L. Ibrahim and E. Kurniati, “Peramalan Jumlah Penumpang Kereta Api Eksekutif di Pulau Jawa Menggunakan Model SARIMA,” *J. Ris. Mat.*, pp. 73–82, 2023, doi: 10.29313/jrm.v3i1.1747.

[6] S. A. Mendila, I. T. Utami, and P. Kartikasari, “Peramalan Jumlah Penumpang Kereta Api Di Pulau Jawa Menggunakan Metode Holt Winters Exponential Smoothing Dan Fuzzy Time Series Markov Chain,” *J. Gaussian*, vol. 12, no. 1, pp. 104–115, 2023, doi: 10.14710/j.gauss.12.1.104-115.

[7] A. O. Imaniar, “Prediksi Jumlah Penumpang Kereta Api menggunakan Perbandingan Metode Holt’s Weighted Exponential Moving Average dan Weighted Exponential Moving Average (Studi Kasus : Data Penumpang PT. Kereta Api Indonesia (Persero) Pulau Jawa dan Sumatera periode April ,” Universitas Islam Indonesia, 2023. [Online]. Available: https://dspace.uii.ac.id/handle/123456789/46701

[8] W. Wiyanti, “Effectiveness of Single and Double Exponential Smoothing: SES, ARRSES and Holt’s Linear for Time Series Data Prediction with Trend and Non-seasonal Characteristic (Covid-19 Vaccinate Case),” *J. Mat. Stat. dan Komputasi*, vol. 20, no. 1, pp. 52–64, 2023, doi: 10.20956/j.v20i1.27193.

[9] C. Della Evania, W. Wiyanti, and C. Author, “Perbandingan Metode Holt’s Linier dan GPT-4 untuk Peramalan Jumlah Kasus Kematian Diabetes Melitus di Indonesia,” *J. Math.*, vol. 7, no. 1, pp. 1–9, 2024.

[10] P. D. W. Oktama, “Nowcasting Jumlah Penumpang Kereta Api di Indonesia Menggunakan Indeks Google Trends,” *Semin. Nas. Off. Stat.*, vol. 2021, no. 1, pp. 958–967, 2021, doi: 10.34123/semnasoffstat.v2021i1.820.

[11] F. A. D. SARASWITA, I. W. SUMARJAYA, and L. P. I. HARINI, “Metode State Space Dalam Meramalkan Jumlah Penumpang Kereta Api Di Pulau Jawa,” *E-Jurnal Mat.*, vol. 9, no. 1, p. 8, 2020, doi: 10.24843/mtk.2020.v09.i01.p272.

[12] N. K. D. R. tita lattifia, Putu Wira Buana, "Model Prediksi Cuaca Menggunakan Metode LSTM," *J. Ilm. Teknoologi dan Komput.*, vol. 3, no. 1, p. 35, 2022, doi: https://doi.org/10.24843/JTRTI.2022.v03.i01.p35.

[13] J. Cahyani, S. Mujahidin, and T. P. Fiqar, "Implementasi Metode Long Short Term Memory (LSTM) untuk Memprediksi Harga Bahan Pokok Nasional," *J. Sist. dan Teknol. Inf.*, vol. 11, no. 2, p. 346, 2023, doi: 10.26418/justin.v11i2.57395.

[14] M. K. Wisyaldin, G. M. Luciana, and H. Pariaman, "Pendekatan LSTM untuk Memprediksi Kondisi Motor 10 kV pada PLTU Batubara," *Kilat*, vol. 9, no. 2, pp. 311–318, 2020, [Online]. Available: http://jurnal.itpln.ac.id/kilat/article/view/997%0Ahttps://jurnal.itpln.ac.id/kilat/article/download/997/775

[15] P. Picton, "What is a Neural Network?," Introduction to Neural Networks. Accessed: Jun. 19, 2024. [Online]. Available: https://www.ibm.com/think/topics/neural-networks#Types+of+neural+networks.

[16] R.M.Hristev, *The ANN Book*, 1st ed. R.M.Hristev, 1998.

[17] M. T. H. Demuth, Mark BealeB., *Neural Network Design*. United States of America: PWS Publishing Company, 1996.

[18] IBM, "Apa itu RNN." Accessed: Jun. 15, 2024. [Online]. Available: https://www.ibm.com/id-id/topics/recurrent-neural-networks#:~:text=RNN dapat digunakan untuk memprediksi,pengenalan suara%2C dan teks gambar.

[19] Mathworks, "Recurrent Neural Network (RNN)." [Online]. Available: https://nl.mathworks.com/discovery/rnn.html

[20] Michael Phi, "Illustrated Guide to LSTM's and GRU's: A step by step explanation," 25 September. Accessed: Jun. 18, 2024. [Online]. Available: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21

[21] "Python is powerful... and fast; plays well with others; runs everywhere; is friendly & easy to learn; is Open." Accessed: Jun. 14, 2024. [Online]. Available: https://www.python.org/about/

[22] sklearn, "Train_Test_Split." Accessed: Apr. 15, 2025. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

[23] Y. Sfaihi, "[LSTM] S&P500 Stocks Time Series Forecasting." Accessed: Apr. 15, 2025. [Online]. Available: https://www.kaggle.com/code/yassinesfaihi/lstm-s-p500-stocks-time-series-forecasting

[24] F. C.Pampel, *Logistic Regression A Primer*. Thousand Oaks, London, New Delhi: Sage Publication, Inc, 2020.

[25] H. R. Makridakis S, Wheelwright SC, "1 / the Forecasting Perspective," *Forecast. methods Appl.*, pp. 1–632, 1997.

[26] B. P. Statistik, "Jumlah Penumpang Kereta Api (Ribu Orang), 2024," 2 Desember. [Online]. Available: https://www.bps.go.id/id/statistics-table/2/NzIjMg==/jumlah-penumpang-kereta-api--ribu-orang-.html