FRONTIER ENERGY SYSTEM AND POWER ENGINEERING
ISSN: 2720-9598

Gated Recurrent Unit (GRU) for Forecasting Hourly Energy Fluctuations

Aji Prasetya Wibawa [1], Alfiansyah Putra Pertama Triono [2], Andien Khansa'a Iffat Paramarta [3], Faradini Usha Setyaputri [4], Ade Kurnia Ganesh Akbari [5], Akhmad Fanny Fadhilla [6], Agung Bella Putra Utama [7], Leonel Hernandez [8]

[1,2,3,4,5,6,7] Department of Electrical Engineering and Informatics, Faculty of Engineering, Universitas Negeri Malang, Jl. Semarang no. 5, Malang 65145, Indonesia
[8] Institución Universitaria de Barranquilla IUB Cra. 45 #48-31, Nte. Centro Historico, Barranquilla 080020, Colombia
Corresponding: aji.prasetya.ft@um.ac.id

Abstract

In the current digital era, energy use undeniably supports economic growth, increases social welfare, and encourages technological progress. Energy-related information is often presented in complex time series data, such as energy consumption data per hour or in seasonal patterns. Deep learning models are used to analyze the data. The right choice of normalization method has great potential to improve the performance of deep learning models significantly. Deep learning models generally use several normalization methods, including min-max and z-score. In this research, the deep learning model chosen is Gated Recurrent Unit (GRU) because the computational load on GRU is lighter, so it doesn't require too much memory. In addition, the GRU data is easier to train, so that it can save training time. This research phase adopts the CRISP-DM methodology in data mining as a solution commonly used in business and research. This methodology involves six stages: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. In this research, the model was obtained using five attribute selection, which applied 2 normalization methods: min-max and z-score. With this normalization, the GRU model produces the best MAPE of 3.9331%, RMSE of 0.9022, and R2 of 0.9022. However, when using z-score normalization, the model performance decreases with MAPE of 10.4332%, RMSE of 0.7602, and R2 of 0.4213. Overall, min-max normalization provides better performance in multivariate time series data analysis.

Keywords

Energy, Forecasting, GRU, Normalization

INTRODUCTION

Energy development has become a crucial issue in the modern era, where energy needs continue to increase, and environmental impacts have become increasingly significant, requiring us to take smart steps in managing energy resources [1]. In this context, time series analysis using deep learning approaches has emerged as a tool with great potential for understanding energy use patterns, forecasting energy demand, and optimizing energy efficiency. However, there are several challenges in applying deep learning in energy time series analysis energy.

Time series analysis is a quantitative forecasting method to determine past data patterns collected based on a time sequence called time series data [2]. The data consists of observations made at different points in time. Multivariate time series expands the data description to include multiple variables, allowing a more detailed understanding of the underlying relationships between different variables over time [3]. Observation of multivariate time series data contains many variables that change from time to time.

Deep learning is one of the models used to analyze time series data. Deep learning can be interpreted as a technique in machine learning that directs a computer or machine system to work like a natural human, namely by studying situations with certain learning or programming [4]. Several models that can be used include Recurrent Neural Network (RNN) [5], Convolutional Neural Network (CNN) [6], Long Short-term Memory (LSTM) [7], Gated Recurrent Unit (GRU) [8], and Bidirectional LSTM (Bi-LSTM) [9].

Multivariate time series data, such as energy forecasting data, requires rearranging the scale so that the attributes have a more standardized or normal distribution of values in the data. Normalization is used to obtain data with a similar range of values. The normalization methods chosen are min-max and z-score. The normalization method is selected to achieve optimal performance in deep learning models.

This research applies various attribute scenarios and normalization techniques to test their influence on GRU performance in time series forecasting [10]. The hope is that with normalization, GRU can provide better forecasting results with smaller evaluation errors. This research uses three general performance measurements, namely MAPE, RMSE, and R2, to measure GRU's performance in forecasting.

ISSN: 2720-9598

METHODS

A. Normalization

Data normalization allows several variables to have the same range of values, none too large or too small, making statistical analysis easier. Data normalization is an important preprocessing step in data mining and machine learning (ML) techniques. The main goal of data normalization is to eliminate data redundancy (repetition) and standardize information for better data workflow. Data normalization is used to scale data for an attribute so that it is in a smaller range, such as -1 to 1 or 0 to 1. Several data normalization methods exist, but this research uses the min-max and z-score normalization methods.

B. Deep Learning

Deep learning is part of artificial intelligence and machine learning, which is a development of neural networks. Deep learning was first developed in 1950, but it was only in 1990 that Deep Learning could be applied successfully. In its application, the learning algorithm used is not much different from the 1990s [11]. Deep Learning itself has several algorithms that can be used, such as RNN, CNN, LSTM, GRU, and Bi-LSTM.GRU

LSTM is a model that can handle vanishing gradient problems in RNN, but it is constrained by the many parameters used, so it requires a long training time. This problem can be handled using the GRU method, a variation of LSTM. GRU has a simpler structure than LSTM [12]. Processing in LSTM goes through 3 gates, namely the output gate, the forget gate, and the input gate, while in GRU, it goes through 2 gates, namely the update gate and the reset gate, as in Figure 1.

The advantage of GRU is its simpler structure than LSTM because it uses fewer parameters. The computational load on GRU is lighter than LSTM, so it doesn't require too much memory. In addition, the GRU data is easier to train, so that it can save training time. Among these advantages, GRU still has some weaknesses. The weakness of GRU is that its performance is worse than that of LSTM in processing large time series data. In addition, GRU does not have memory cells, so it is less effective at remembering long-term and complex patterns.

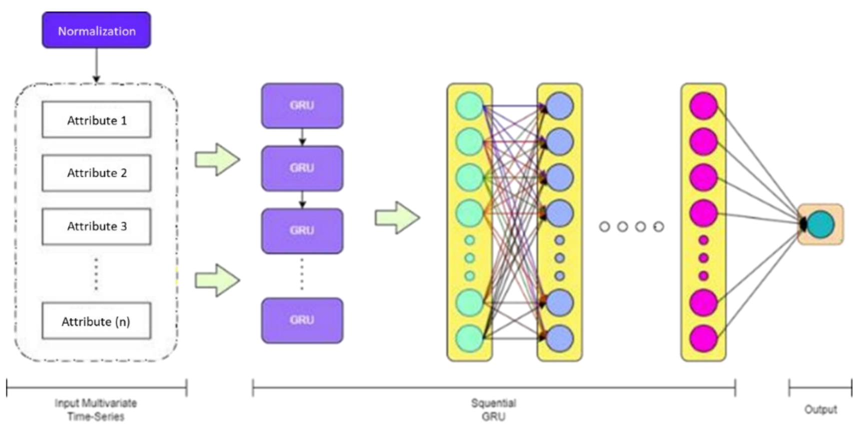

Figure 1. GRU Process Flow

In the context of time series data, the GRU algorithm begins by preparing the data in an appropriate format. The GRU layer consists of one GRU unit, which has two gates: the reset gate and the update gate. This GRU unit can combine information from the past to the present and from the present to the future, enabling the understanding of temporal patterns in time series analysis. The process begins by streaming the data sequence to the GRU unit. Reset gates help regulate the extent to which information from the past should be ignored, while update gates regulate the extent to which information from the present should be integrated. After processing the data sequence, the representation of the results from the GRU unit becomes richer in understanding the time window.

Furthermore, this representation is directed to the output layer, suitable for time series analysis modelling [13]. If the goal is prediction, the output layer may consist of a single neuron that produces predicted values for the next point in time. If the goal is time classification, the output layer will have several neurons according to the number of possible classes, where each neuron represents the probability of a particular class.

In summary, the GRU algorithm in time series analysis utilizes a GRU unit with a reset gate and update gate to combine information from both time directions [14]. The resulting representation of these GRU units is used for purposes such as prediction or time classification, subject to an appropriate output layer structure.

ISSN: 2720-9598

RESEARCH METHODS

This research uses the CRISP-DM (Cross Industry Standard Process for Data Mining) data mining methodology as a common problem solver for business and research. The CRISP-DM methodology consists of six stages: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. The methodological process in this study can be explained as follows and presented in Figure 2.

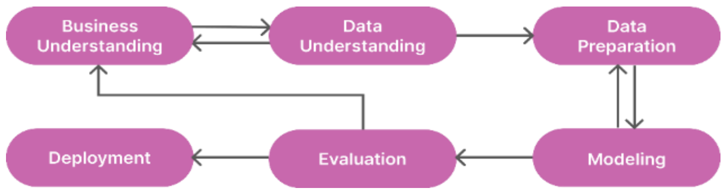

Figure 2. Research Phase

A. Business Understanding

Business Understanding is the initial stage of assessing research needs and objectives [15]. Multivariate time series analysis using deep learning methods involves several important factors. One is the value scale in multivariate time series data, which can differ. This value scale mismatch can affect the optimal performance of deep learning models. A normalization process is needed to reset the value scale in the data. This research aims to find the best normalization method for the GRU method. This will be done by conducting a comparative analysis of the GRU evaluation results using two different normalization methods: Min-Max and Z-Score.

B. Data Understanding

Data understanding contains an understanding analysis of the collected data [16]. The dataset used in this study is sourced from the Kaggle site by downloading the .csv file. There are 29 attributes in the dataset and 35064 instances, which will be used for attribute selection to select features. The attribute used in this research is total load Table 1.

TABLE 1

DATASET ATTRIBUTE DETAILS

Attribute	Data type	Description (min, max)
Generation biomass	Float	(0, 592)
Generation of fossil brown coal/lignite	Float	(0, 999)
Generation of fossil coal/derived gas	Float	(0)
Generation fossil gas	Float	(0, 20034)
Generation fossil hard coal	Float	(0, 8359)
Generation fossil oil	Float	(0, 449)
Generation fossil oil shale	Float	(0)
Generation fossil peat	Float	(0)
Generation geothermal	Float	(0)
Generation hydro-pumped storage aggregated.	Float	NaN
Generation hydro-pumped storage consumption.	Float	(0, 4523)
Generation hydro run-off river and poundage	Float	(0, 2000)
Generation hydro water reservoir	Float	(0, 9728)
Generation marine	Float	(0)
Generation Nuclear	Float	(0, 7117)
Generation other	Float	(0, 106)
Generation other renewable	Float	(0, 119)
Generation solar	Float	(0, 5792)
Generation waste	Float	(0, 357)
Generation wind offshore	Float	(0)
Generation wind onshore	Float	(0, 17436)
Forecast solar day.	Float	(0, 5836)
Forecast wind offshore day ahead	Float	NaN
Forecast wind onshore day ahead	Float	(0, 17430)
Total load forecast	Float	(0, 41390)
Total load actual	Float	(0, 41015)
Price day ahead	Float	(0, 98.69)
Price actual	Float	(0, 99.95)

http://dx.doi.org/10.17977/um049v5i1p%25p

C. Data Preparation

The data preparation stage is carried out by three stages of data preprocessing so that the model can accept the data and the testing process runs more easily. In this study, the technique used is missing value handling. Missing values can cause problems in the case of machine learning and statistical models because the models can produce inaccurate results. This research handles missing values using deletion. The data used contains two columns containing NaN. The two columns will be deleted because they do not correlate with the target attribute.

D. Modeling

Modelling further explains the design of the research model used. The design of the research model is shown in Figure 3.

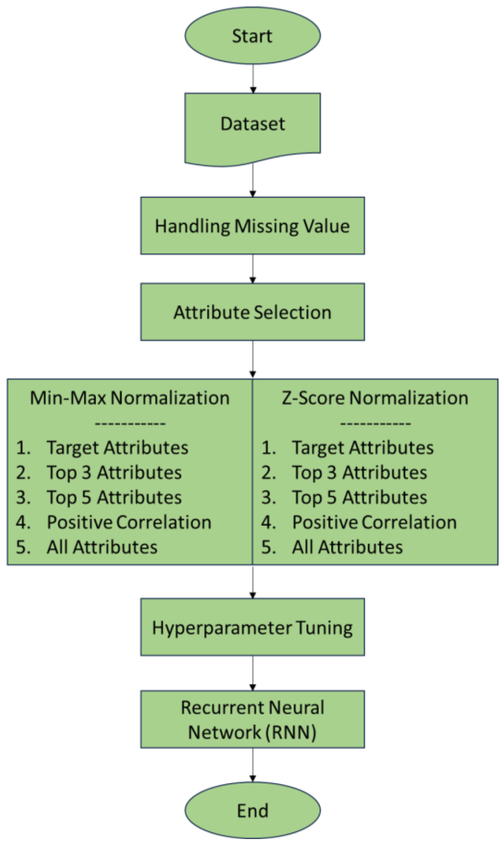

Figure 3. Research Modeling Framework

In this study, modelling uses two kinds of normalization in data preprocessing. Data normalization converts data at different scales into a uniform or comparable scale. By applying the normalization technique, each variable will have the same effect on model performance. In other words, no variable dominates because the value is too large. Min-max normalization transforms features or variables in the dataset to values between 0 and 1 [17]. This transformation process can be carried out on all features in the data so that the features have values from 0 to 1. The Min-max normalization calculation process is as in (1).

$$x' = \frac{(x - x_{min})}{(x_{max} - x_{min})} \tag{1}$$

Z-score normalization, or standardization, is a technique where the attribute values are normalized based on the mean and standard deviation. Z-Score transforms data into values -1 to 1 [18]. The process of calculating the Z-score normalization is shown in (2).

$$x' = \frac{x - \mu}{\sigma} \tag{2}$$

ISSN: 2720-9598

Next is data testing to evaluate the performance of the previously trained model. The forecasting model used is GRU with a multivariate time series dataset [19]. Testing was carried out with two normalization methods and five different attribute selections. The activation function is tanh, and other parameters are determined through the grid search process.

E. Evaluation

Evaluation is done by evaluating the results of forecasting. The evaluation stage is carried out to determine the model's accuracy by calculating the error value. Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Coefficient of Determination (R2) [20]. The MAPE & RMSE functions are used to determine the accuracy of the model performance that has been made. The experiment will be repeated five times in each scenario and then averaged. The following is the equation of the MAPE, RMSE, and R2 functions, as in (3) to (5).

$$MAPE = \frac{1}{m}\sum_{i=1}^{m}\frac{|Y_i - X_i|}{Y_i} \tag{3}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(X_i - Y_i)^2} \tag{4}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(X_i - Y_i)^2}{\sum_{i=1}^{m}(\bar{Y} - Y_i)^2} \tag{5}$$

F. Deployment

The deployment phase is carried out by reporting the model evaluation results. In this step, an evaluation report is carried out, which involves comparing the two normalization methods, namely min-max and z-score, in testing the GRU model [21]. The analysis was carried out by comparing the average values of MAPE, RMSE, and R2. The normalization method that produces the lowest MAPE and RMSE values and the highest R2 will be considered the best normalization method in this study.

RESULT AND DISCUSSION

This chapter discusses the next stages, including Data Preparation, Modelling, Evaluation, and Deployment.

A. Data Preparation

The data preparation stage is missing value handling. Two attributes were removed because they have NaN values: generation hydro pumped storage aggregated and forecast wind offshore day ahead. The purpose of removing these attributes is because the missing value causes sampling bias. In addition to attributes with a NaN value, an attribute compares the predicted value generated by the model, namely the total load forecast. This attribute is not included in the processing because it is a predicted value for the target attribute, namely the total load. These attributes can be used to compare the model's predicted value with that used in the industry. Removing these attributes causes the number of attributes to decrease from 29 to 26.

B. Modelling

Process The attribute selection process is carried out by calculating the correlation of each attribute with the target attribute, namely 'actual total load'. The results of calculating the correlation value are sorted in descending order (large to small) to do attribute selection. The attribute selection process is adapted to the five predetermined scenarios. The next step is to carry out the normalization process using two normalization methods: Min-max and Z-score.

After the data preprocessing stage, the next stage is creating the GRU model [22]. The model created requires parameters that match the characteristics of the data [23]. These parameters are determined using a hyperparameter tuning grid search to get the best combination of hyperparameters and ensure that the model provides optimal performance. Apart from that, other hyperparameters, such as the activation function, are determined using tanh because it is centred at point 0 to achieve faster convergence and a dropout of 0.2 to avoid the risk of overfitting [22]. The results of the hyperparameter tuning grid search are shown in Table 2.

Frontier Energy System and Power Engineering
ISSN: 2720-9598

TABLE 2

HYPERPARAMETER TUNING

Parameter	Search Space	Result
Batch Size	‘100’, ‘1000’	100
Epoch	’50’, ‘100’	50
Hidden Layer	‘2’, ‘5’, ‘10’	2
Loss Function	‘mse’, ‘mae’, ’huberloss’	MSE
Neuron	‘32’, ‘64’	32
Optimizer	‘adam’, ‘rmsprop’	Rmsprop

Tests using parameters obtained from hyperparameter tuning produce GRU models that can reach convergent conditions [24]. The convergence graph of the GRU model with min-max and z-score normalization is shown in Figure 4 and Figure 5.

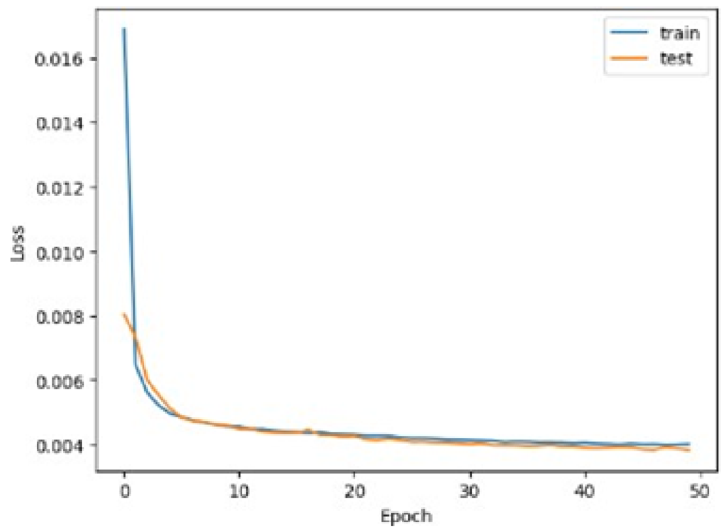

Figure 4. Min-max Convergence Graph

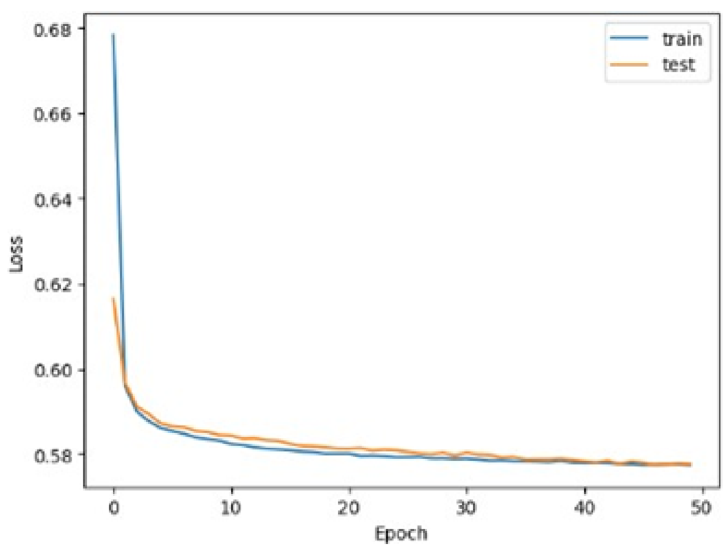

Figure 5. Z-score Convergence Graph

C. Evaluation

The model evaluation methods used are MAPE, RMSE, and R2. The GRU model that has been made is run five times and produces five evaluation values for each scenario [25]. These values are averaged to produce each scenario's final MAPE, RMSE, and R2 values. The evaluation results are shown in Table 3.

FRONTIER ENERGY SYSTEM AND POWER ENGINEERING
ISSN: 2720-9598

TABLE 3

EXPERIMENT RESULTS

Normalization	Scenario	MAPE (%)	RMSE	R^2
Min-max	Target Attributes	4.0756	0.0674	0.8852
	Top 3 Attributes	3.9964	0.0645	0.8949
	Top 5 Attributes	3.9344	0.0629	0.8999
	Positive correlation	3.9331	0.0622	0.9022
	All Attributes	3.9808	0.0625	0.9011
Z-score	Target Attributes	10.5936	0.7648	0.4148
	Top 3 Attributes	10.6485	0.7655	0.4132
	Top 5 Attributes	10.5000	0.7627	0.4176
	Positive correlation	10.4388	0.7605	0.4209
	All Attributes	10.4332	0.7602	0.4213

The results of the GRU evaluation with z-score normalization are shown in Table 3. As with min-max normalization, there is no significant difference in each attribute selection scenario in the MAPE, RMSE, and R2 values. The scenario that shows the best MAPE, RMSE and R2 values is the scenario with all attributes, where the MAPE value obtained is 10.4332%, the RMSE value is 0.7602, and the R2 value is 0.4213. Based on the evaluation results of the two normalizations above, attribute reduction scenarios do not significantly influence model performance. This shows that carrying out attribute selection scenarios in this data analysis is not recommended.

D. Deployment

The Deployment stage is carried out to report the evaluation results between the two normalization methods in testing the GRU model [26]. This step begins by selecting the scenario with the lowest MAPE value for each normalization method. Using the MAPE value as a benchmark is important because it can show the level of prediction error in the form of a percentage that is easier for end users to understand [27].

Figure 6 compares MAPE values in scenarios using min-max and z-score normalization. The graph above shows that the MAPE value produced by min-max normalization is 3.9331% lower than the MAPE value of 10.4332% obtained by z-score normalization.

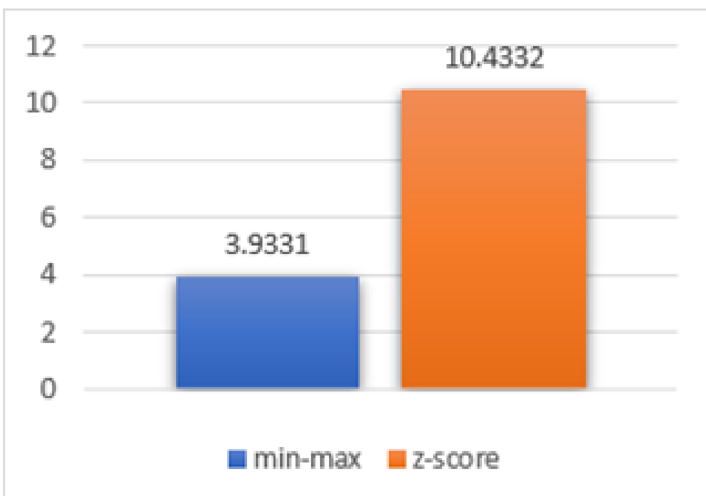

Figure 6. MAPE Comparison Results (%)

Figure 7 shows the comparison of RMSE values between min-max and z-score normalization. It can be seen that the scenario with min-max normalization produces a lower RMSE value, namely 0.0622, compared to z-score normalization, which has an RMSE value of 0.7602. Figure 8 compares the R2 values between min-max and z-score normalization. It can be seen that the scenario with min-max normalization produces a higher R2 value, equal to 0.9022, compared to the normalized z-score, which has an R2 value of 0.4213.

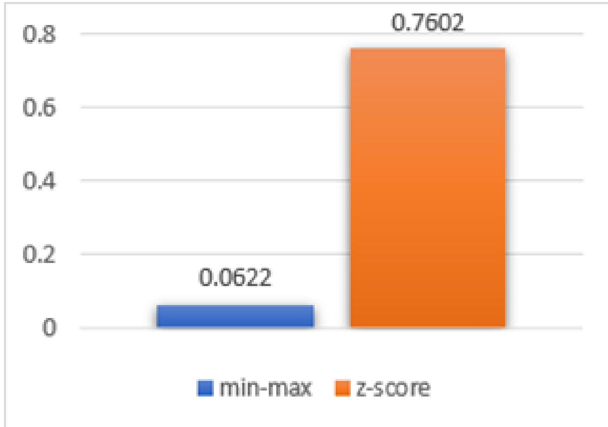

Figure 7. RMSE Comparison Results

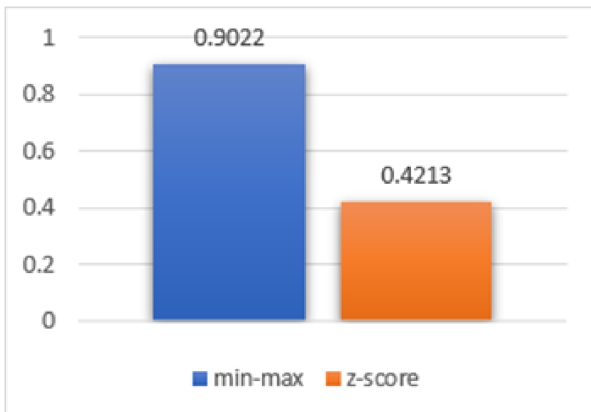

Figure 8. R2 Comparison Results

Based on the results presented in Figure 6 to Figure 8, it can be concluded that min-max normalization produces the best performance. Research comparing min-max and z-score normalization has been carried out in previous research, which produced an RMSE of 9.99 compared to a z-score of 10.50. However, other studies reveal that models using z-score normalization provide higher accuracy than models using min-max normalization. This is due to differences in the characteristics of the data. In addition, the causative factor is caused by using the activation function, namely tanh, which transmits information in the range of -1 to 1 [28]. In other words, the neuron input produced by both types of normalization will be forwarded to the next layer according to that range [29].

The best results from each model using the min-max normalization method are these values. The best evaluation value was obtained with the LSTM model with a MAPE value of 3.9002%, RMSE 0.0621 and R2. 0.9027. The proposed GRU model cannot outperform LSTM in MAPE, RMSE, and R2 values because GRU has weaknesses when processing large time series data [30]. In future research, it is necessary to carry out in-depth observations of the GRU for energy forecasting data analysis [31].

The advice given for energy use is to choose electronic equipment that is economical and performs well [32]. It is best to limit device use to a duration that is not too long. When the weather is extreme, consider using alternative energy sources to regulate room temperature.

CONCLUSION

This study uses the Gated Recurrent Unit (GRU) deep learning method to analyze energy forecasting data. 1. From the test using min-max normalization, optimal performance results were obtained with the best MAPE value of 3.9331%, RMSE of 0.0622, and R2 of 0.9022. This shows that the GRU model successfully analyzes multivariate time series data using min-max normalization. 2. The test results show that the normalization of the z-score produces a slightly lower evaluation, with the best MAPE of 10.4332%, RMSE of 0.7602, and R2 of 0.4213. Overall, min-max normalization performs better than z-score normalization in analyzing multivariate time series data. The strategy that can be considered for further research is testing with multi-input and multi-output concepts. Use the latest activation

ISSN: 2720-9598

functions, such as the swish and mish activation. Use other hyperparameters such as random search, hyperband, and optuna. Different neurons are used in each input, hidden, and output layer.

REFERENCES

[1] Q. Al-Shetwi, "Sustainable development of renewable energy integrated power sector: Trends, environmental impacts, and recent challenges," *Sci. Total Environ.*, vol. 822, p. 153645, May 2022, doi: 10.1016/j.scitotenv.2022.153645.

[2] Lim and S. Zohren, "Time-series forecasting with deep learning: A survey," *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 379, no. 2194, 2021, doi: 10.1098/rsta.2020.0209.

[3] M. Khan, B. Jan, and H. Farman, *Deep Learning: Convergence to Big Data Analytics*. Singapore: Springer Singapore, 2019.

[4] R. Gupta, D. Srivastava, M. Sahu, S. Tiwari, R. K. Ambasta, and P. Kumar, "Artificial intelligence to deep learning: machine intelligence approach for drug discovery," *Mol. Divers.*, vol. 25, no. 3, pp. 1315–1360, Aug. 2021, doi: 10.1007/s11030-021-10217-3.

[5] Kadri and K. Abdennbi, "RNN-based deep-learning approach to forecasting hospital system demands: application to an emergency department," *Int. J. Data Sci.*, vol. 5, no. 1, p. 1, 2020, doi: 10.1504/IJDS.2020.109497.

[6] Raju and S. Das, "CNN-Based Deep Learning Model for Solar Wind Forecasting," *Sol. Phys.*, vol. 296, no. 9, p. 134, Sep. 2021, doi: 10.1007/s11207-021-01874-6.

[7] Jahangir, H. Tayarani, S. S. Gougheri, M. A. Golkar, A. Ahmadian, and A. Elkamel, "Deep Learning-Based Forecasting Approach in Smart Grids With Microclustering and Bidirectional LSTM Network," *IEEE Trans. Ind. Electron.*, vol. 68, no. 9, pp. 8298–8309, Sep. 2021, doi: 10.1109/TIE.2020.3009604.

[8] A. J. Almalki and P. Wocjan, "Forecasting Method based upon GRU-based Deep Learning Model," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, Dec. 2020, pp. 534–538, doi: 10.1109/CSCI51800.2020.00096.

[9] S. S. Subbiah, S. K. Paramasivan, K. Arockiasamy, S. Senthivel, and M. Thangavel, "Deep Learning for Wind Speed Forecasting Using Bi-LSTM with Selected Features," *Intell. Autom. Soft Comput.*, vol. 35, no. 3, pp. 3829–3844, 2023, doi: 10.32604/iasc.2023.030480.

[10] R. Hanocka and H.-T. D. Liu, "An introduction to deep learning on meshes," in *ACM SIGGRAPH 2021 Courses*, Aug. 2021, pp. 1–27, doi: 10.1145/3450508.3464569.

[11] Choi, J. Yi, C. Park, and S. Yoon, "Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines," *IEEE Access*, vol. 9, pp. 120043–120065, 2021, doi: 10.1109/ACCESS.2021.3107975.

[12] S. Yang, X. Yu, and Y. Zhou, "LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example," in *2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI)*, Jun. 2020, pp. 98–101, doi: 10.1109/IWECAI50956.2020.00027.

[13] Biele, J. Kacprzyk, W. Kopeć, J. W. Owsiński, A. Romanowski, and M. Sikorski, Eds., *Digital Interaction and Machine Intelligence*, vol. 440. Cham: Springer International Publishing, 2022.

[14] A. Sethia and P. Raut, "Application of LSTM, GRU and ICA for stock price prediction," in *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2018, Volume 2*, 2019, pp. 479–487.

[15] S. M. H. Rizvi, "Time Series Deep learning for Robust Steady-State Load Parameter Estimation using 1D-CNN," *Arab. J. Sci. Eng.*, vol. 47, no. 3, pp. 2731–2744, Mar. 2022, doi: 10.1007/s13369-021-05782-6.

[16] S. G. K. Patro and K. K. sahu, "Normalization: A Preprocessing Stage," *Iarjset*, no. April, pp. 20–22, 2015, doi: 10.17148/iarjset.2015.2305.

[17] S. Sinsomboonthong, "Performance Comparison of New Adjusted Min-Max with Decimal Scaling and Statistical Column Normalization Methods for Artificial Neural Network Classification," *Int. J. Math. Math. Sci.*, vol. 2022, pp. 1–9, Apr. 2022, doi: 10.1155/2022/3584406.

[18] N. Fei, Y. Gao, Z. Lu, and T. Xiang, "Z-Score Normalization, Hubness, and Few-Shot Learning," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 142–151, doi: 10.1109/ICCV48922.2021.00021.

[19] N. Singh and P. Singh, "Exploring the effect of normalization on medical data classification," in *2021 International Conference on Artificial Intelligence and Machine Vision (AIMV)*, Sep. 2021, pp. 1–5, doi: 10.1109/AIMV53313.2021.9670938.

[20] U. I. Arfianti, D. C. R. Novitasari, N. Widodo, M. Hafiyusholeh, and W. D. Utami, "Sunspot Number Prediction Using Gated Recurrent Unit (GRU) Algorithm," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 15, no. 2, p. 141, Apr. 2021, doi: 10.22146/ijccs.63676.

[21] S. Jeong, W. Ko, A. W. Mulyadi, and H.-I. Suk, "Deep Efficient Continuous Manifold Learning for Time Series Modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 171–184, Jan. 2024, doi: 10.1109/TPAMI.2023.3320125.

[22] Borkowski, C. Sorini, and A. Chattopadhyay, "Recurrent neural network-based multiaxial plasticity model with regularization for physics-informed constraints," *Comput. Struct.*, vol. 258, p. 106678, Jan. 2022, doi: 10.1016/j.compstruc.2021.106678.

[23] Ahmadzadeh, H. Kim, O. Jeong, N. Kim, and I. Moon, "A Deep Bidirectional LSTM-GRU Network Model for Automated Ciphertext Classification," *IEEE Access*, vol. 10, pp. 3228–3237, 2022, doi: 10.1109/ACCESS.2022.3140342.

[24] Hussain, M. K. Afzal, S. Ahmad, and A. M. Mostafa, "Intelligent Traffic Flow Prediction Using Optimized GRU Model," *IEEE Access*, vol. 9, pp. 100736–100746, 2021, doi: 10.1109/ACCESS.2021.3097141.

[25] S. Sah, B. Surendiran, R. Dhanalakshmi, S. N. Mohanty, F. Alenezi, and K. Polat, "Forecasting COVID-19 Pandemic Using

ISSN: 2720-9598

Prophet, ARIMA, and Hybrid Stacked LSTM-GRU Models in India," *Comput. Math. Methods Med.*, vol. 2022, pp. 1–19, May 2022, doi: 10.1155/2022/1556025.

[26] A. W. Saputra, A. P. Wibawa, U. Pujianto, A. B. P. Utama, and A. Nafalski, "LSTM-based Multivariate Time-Series Analysis : A Case of Journal Visitors Forecasting," *Ilk. J. Ilm.*, vol. 14, no. 1, pp. 57–62, 2022, doi: 10.33096/ilkom.v14i1.1106.57-62.

[27] Recchia, "R-Squared Measures for Two-Level Hierarchical Linear Models Using SAS," *J. Stat. Softw.*, vol. 32, no. Code Snippet 2, 2010, doi: 10.18637/jss.v032.c02.

[28] J. Becerra-Rico, M. A. Aceves-Fernández, K. Esquivel-Escalante, and J. C. Pedraza-Ortega, "Airborne particle pollution predictive model using Gated Recurrent Unit (GRU) deep neural networks," *Earth Sci. Informatics*, vol. 13, no. 3, pp. 821–834, Sep. 2020, doi: 10.1007/s12145-020-00462-9.

[29] Abo-Tabik, N. Costen, J. Darby, and Y. Benn, "Towards a Smart Smoking Cessation App: A 1D-CNN Model Predicting Smoking Events," *Sensors*, vol. 20, no. 4, p. 1099, Feb. 2020, doi: 10.3390/s20041099.

[30] B. Aditya Satrio, W. Darmawan, B. U. Nadia, and N. Hanafiah, "Time series analysis and forecasting of coronavirus disease in Indonesia using ARIMA model and PROPHET," *Procedia Comput. Sci.*, vol. 179, pp. 524–532, 2021, doi: 10.1016/j.procs.2021.01.036.

[31] H. Liu, R. Ma, D. Li, L. Yan, and Z. Ma, "Machinery Fault Diagnosis Based on Deep Learning for Time Series Analysis and Knowledge Graphs," *J. Signal Process. Syst.*, vol. 93, no. 12, pp. 1433–1455, Dec. 2021, doi: 10.1007/s11265-021-01718-3.

[32] S. Liu, X. Hao, Z. Meng, J. Li, T. Cui, and L. Wei, "Application of SRNN-GRU in Photovoltaic power Forecasting," *E3S Web Conf.*, vol. 256, p. 02001, May 2021, doi: 10.1051/e3sconf/202125602001.