JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v10i4.6429

DEEP GATED RECURRENT UNITS PARAMETER TRANSFORMATION FOR OPTIMIZING ELECTRIC VEHICLE POPULATION PREDICTION ACCURACY

Jeni Sugiandi[1*]; Solikhun[2]; Anjar Wanto[3]

Informatics Engineering Study Program[1,2,3]
STIKOM Tunas Bangsa, Pematangsiantar, Indonesia[1,2,3]
jenisugiandi@gmail.com[1*], solikhun@amiktunasbangsa.ac.id[2], anjarwanto@amiktunasbangsa.ac.id[3]

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract— *The development of electric vehicles is an important innovation in reducing greenhouse gas emissions while reducing dependence on fossil fuels. The main problem in developing electric vehicles is the lack of adequate infrastructure. Inaccurate predictions regarding the number of electric vehicles hinder adequate infrastructure planning and development. This research proposes the use of the Gated Recurrent Units (GRU) algorithm to improve the accuracy of electric vehicle population predictions by carrying out GRU parameter transformations. This parameter transformation involves searching and adjusting the parameters of the GRU model in more depth to increase its ability to handle uncertainty in electric vehicle population data. After carrying out the training and testing process, the result was that the hyperparameter model using RandomizedSearchCV was the best model because it had the highest accuracy compared to other models tested with a combination of GRU_unit 64 and 128, dropout 0.5 and 0.6, batch size 64 and the number of epochs was 100 which had MAE results: 257.94, MSE: 66655.087, RMSE: 258.176, and Accuracy of 100%.*

Keywords: *deep learning, electric vehicles, gated recurrent units, optimation, prediction.*

Intisari— *Pengembangan kendaraan listrik merupakan inovasi penting dalam mengurangi emisi gas rumah kaca sekaligus mengurangi ketergantungan terhadap bahan bakar fosil. Permasalahan utama dalam pengembangan kendaraan listrik adalah kurangnya infrastruktur yang memadai. Prediksi yang tidak akurat mengenai jumlah kendaraan listrik menghambat perencanaan dan pembangunan infrastruktur yang memadai. Penelitian ini mengusulkan penggunaan algoritma Gated Recurrent Units (GRU) untuk meningkatkan akurasi prediksi populasi kendaraan listrik dengan melakukan transformasi parameter GRU. Transformasi parameter ini melibatkan pencarian dan penyesuaian parameter model GRU secara lebih mendalam untuk meningkatkan kemampuannya dalam menangani ketidakpastian data populasi kendaraan listrik. Setelah dilakukan proses pelatihan dan pengujian, diperoleh hasil bahwa model hyperparameter dengan menggunakan RandomizedSearchCV merupakan model terbaik karena mempunyai akurasi paling tinggi dibandingkan model lain yang diuji dengan kombinasi GRU_unit 64 dan 128, dropout 0.5 dan 0.6, batch size 64 dan jumlah epoch 100 yang mempunyai hasil MAE : 257.94, MSE : 66655.087, RMSE : 258.176, R2-Score: -10514769.256, dan Akurasi 100%.*

Kata Kunci: *deep learning, gated recurrent units, kendaraan listrik, optimasi, prediksi.*

INTRODUCTION

The development of electric vehicles is a significant innovation aimed at reducing greenhouse gas emissions and decreasing dependence on fossil fuels [1][2][3]. Government policies and a global movement toward environmentally friendly energy sources have begun to promote the use of electric vehicles in various regions. In alignment with these global trends, interest in EVs is on the rise. This increased interest is driven by several factors, including greater awareness of environmental impacts, government subsidies for electric vehicles, and lower long-term costs and efficiency. Electric vehicles not only provide an environmentally friendly solution but are also expected to be a more efficient and economical alternative in the long run [4][5]. Electric vehicles not only offer an environmentally friendly solution, but are also expected to be a more efficient and economical alternative in the long term [6][7][8]. While the market share of electric vehicles is still small compared to fossil fuel vehicles, their adoption is anticipated to continue growing.

The primary challenge in the development of electric vehicles is the insufficient infrastructure, particularly the lack of charging stations. The availability of charging stations is crucial for supporting the growth of electric vehicles [9][10][11]. However, inaccurate forecasts regarding the number of electric vehicles hinder effective planning and infrastructure development. Conventional prediction algorithms often fail to deliver accurate projections, which prevents optimal results [12]. This ultimately affects the ability to deploy infrastructure effectively encounter. The solution proposed in this research is to transform the GRU parameters. This parameter transformation includes searching and adjusting the parameters of the GRU model in more depth, such as the size of hidden layers, learning rate, and activation function, to improve its ability to handle uncertainty in electric vehicle population data [13]. Gated Recurrent Unit (GRU) is an artificial neural network based on Recurrent Neural Network (RNN) which is suitable for sequential data more efficiently than traditional prediction methods [14].

This algorithm is designed to handle the problem of "long-term dependencies", which means the GRU can retain information from previous data when processing subsequent data in a time sequence [15][16]. This algorithm is very useful for time series data. GRU works by utilizing historical data to produce predictions. This algorithm has a "gating" mechanism that allows the model to learn time sequence information, retaining important information and eliminating less relevant information to produce accurate predictions [17]. GRU can overcome the vanishing gradient problem and learn complex patterns from non-linear data, so it is expected to produce more accurate predictions [18][19][20]. This research aims to specifically improve the accuracy of electric vehicle population predictions. Accurate results from these predictions will support infrastructure planning to be carried out more effectively, ensuring adequate availability to support the growth of electric vehicles.

Research that applies backpropagation to predict vehicle growth [21]. This research uses the Central Statistics Agency. The data used is vehicle growth data for North Sumatra Province from 2017-2019. There are three architectural models used, namely 2-2-1, 2-3-1, 2-4-1, 2-5-1, 2-7-1. The results of trials using MATLAB R2011b showed that the 2-2-1 model was the best model with an accuracy level of 94%, MSE 0.000208514, and an epoch value of 789. Research that applies the Conjuget Gradient Polak-Ribiere Backpropagation algorithm in predicting sales of electric motorbikes [22]. The data used is monthly electric motorbike sales data in 2021-2023 obtained from: https://www.aisi.or.id/statistic/. By comparing 4 different architectural models, namely: 2-3-4-1 produces predictions with an accuracy level of 83%, 2-361=92%, 2-3-7-1=92%, and 2-3-8-1 =100%. The best architecture of these 4 models is 2-3-8-1 with an accuracy level of 100% and an MSE value of 0.0000001059. The research that is the main reference in this research is research that applies the Deep Learning Gated Recurrent Units (GRU) algorithm in predicting Bank Central Asia share prices [23]. The data used is Bank Central Asia share price data from 2019 to 2024 obtained using pyfinance python. The dataset is divided into training data and testing data with a ratio of 80:20 and 60:40, the results show that the 80:20 model is the best with lookback 15, timestep 15, and epoch 50, which has an RMSE value of 1.039, MSE 1.079, MAE 0.842, RSquared 0.983, MGD 0.0037 and MPD 0.0197 with accuracy results of 54.87%, recall 59.23%, f1-square 58.11%, precision 57.03%. GRU has been used in many ways in applications, deep parameter transformations and their application to data Electric vehicles are a relatively new and innovative approach. The study also introduces the utilization of parameter optimization techniques in the context prediction of the electric vehicle population to improve prediction accuracy which has not been widely explored in the literature which exists.

In this research, we apply the parameter transformation method in the GRU model to optimize the accuracy of electric vehicle population predictions. Although GRU has been used in many applications, deep parameter transformation and its application to EV data is a relatively new and innovative approach. This research also introduces the use of parameter optimization techniques in the context of electric vehicle population prediction, which has not been widely explored in the existing literature. The GRU model, which is a variant of Recurrent Neural Networks (RNN), is superior in overcoming problems such as vanishing gradients, which are often encountered in conventional RNN models [24][25]. GRU is also able to learn the

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v10i4.6429

dynamics of time series data better, resulting in more accurate predictions. The focus of this research is on the use of GRU to improve prediction accuracy. The main objective of this research is to develop and optimize the GRU model with parameter transformation to increase prediction accuracy to increase prediction accuracy of the electric vehicle population. It is hoped that this research can make a significant contribution to the development of deep learning-based prediction models that are more effective and efficient in predicting the population of electric vehicles in the future. It is also hoped that the results of this research can support the formulation of appropriate policies regarding the adoption of electric vehicles and the development of adequate and optimal infrastructure so that these policies can be more effective in encouraging the transition towards environmentally friendly transportation.

MATERIALS AND METHODS

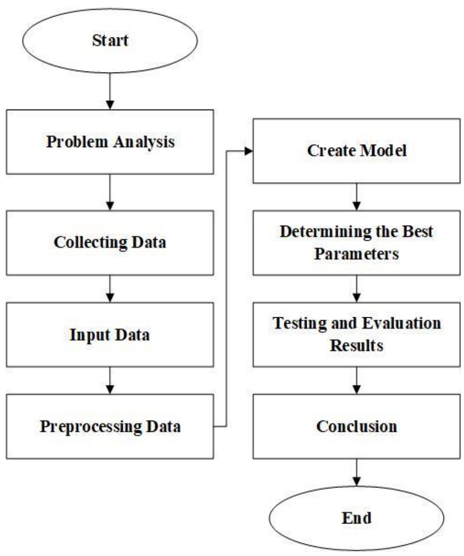
The dataset used in this research is electric vehicle population data obtained from open sources. The dataset was obtained from the open-source platform Kaggle. The data in the dataset ranges from January 31 2017 to February 29, 2024 (2585 days) with a total of 20733 data records. The dataset contains 20733 data records consisting of 10 attributes, namely Date, County, State, Vehicle Primary Use, Battery Electric Vehicles (BEVs), Plug-In Hybrid Electric Vehicles (PHEVs), Electric Vehicle (EV) Total, Non-Electric Vehicle Total, Total Vehicles, Percent Electric Vehicles. The attributes used to train the model are the "Date" and "Electric Vehicle (EV) Total" attributes. The Total Electric Vehicle (EV) attribute was chosen because it is the total number of electric vehicles collected based on the registered areas that have electric vehicles.

Table 1. Dataset

No	Date	County	State	Vehicle Primary Use	Battery Electric Vehicles	...	Percent Electric Vehicles
1	September 30 2020	Riverside	CA	Passenger	7	...	1.50
2	December 31 2022	Prince William	VA	Passenger	1	...	1.57
3	January 31 2020	Dakota	MN	Passenger	0	...	3.03
4	January 31 2020	Ferry	WA	Truck	0	...	0.00
5	June 30 2022	Douglas	CO	Passenger	0	...	1.19
6	July 31 2021	Maui	HI	Passenger	1	...	1.67
7	May 31 2018	Northampton	PA	Passenger	0	...	1.49
8	November 30 2017	Nassau	NY	Passenger	1	...	2.63
9	March 31 2018	DeKalb	IN	Passenger	1	...	50.00
10	March 31 2020	Riverside	CA	Passenger	7	...	1.50
...	...	...	...	...	...	...	...
20733	November 30 2019	Goochland	VA	Passenger	3	...	1.45

Source : (Research Results, 2025)

The research process is further explained through the steps presented in Figure 1.



Source : (Research Result, 2025)
Figure 1. Research Stages

Preprocessing Data

The data preprocessing stage is the process of processing raw data into data that is clean, structured, and ready to be used for analysis or modeling [26]. At this stage, the process that will be carried out is to look at the information in the dataset, then remove null data, change the data type "Date" to datetime, see if there is duplicate data, and after that drop variables that are not used. In the dataset used, there are 10 variables, namely *Date, County, State, Vehicle Primary Use, Battery Electric Vehicles (BEVs), Plug-In Hybrid Electric Vehicles (PHEVs), Electric Vehicle (EV) Total, Non-Electric Vehicle Total, Total Vehicles, Percent Electric Vehicles*. Because in this research we will only use the variables *"Date"* and *"Electric Vehicle (EV) Total"* we will drop or not use the other variables. At this stage, feature scaling is also carried out on the data using MinMax Scaler, with the aim of reducing redundancy preventing anomalies in the data, and ensuring the data has a uniform scale so that the model can learn efficiently. Data that has been

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v10i4.6429

previously processed is divided into 80% for training data and 20% for test data. Training data is used to train the model, and test data is used in the prediction stage.

Create Model

The next stage is to determine the model parameters that will be used in Gated Recurrent Units (GRU) deep learning. The study uses a dense layer with 128 units and an output layer with 1 unit, along with the ReLU activation function. To prevent overfitting and save computational time, the Callbacks EarlyStopping is used, set with patience=5. This function will monitor the validation loss during training, and if the validation loss does not decrease for 5 consecutive epochs, the training will be stopped. The goal is to produce the best performance, such as increasing accuracy and other aspects that will be evaluated at the model testing stage.

Table 2. Parameter Model

Hyperparameter	Parameter	Input
GridSearchCV	Epoch	50
	Batch Size	16
	GRU_Unit	16, 32
	Drop Out	0.3, 0.4
	Optimizer	Adam
RandomizedSearchCV	Epoch	100
	Batch Size	64
	GRU_Unit	64, 128
	Drop Out	0.5, 0.6
	Optimizer	Adam

Source : (Research Result, 2025)

Determining the Best Parameters

Hyperparameter tuning was carried out using GridSearchCV and RandomizedSearchCV to determine the best parameters that are useful for improving model performance [27][28]. In this study, GridSearchCV is applied to a smaller hyperparameter set, which allows and thorough search within a limited parameter space. RandomizedSearchCV is applied to a larger hyperparameter set to explore more possibilities within a broader search space with a limited number of evaluations. Therefore, the rationale for using two separate strategies is to optimize computational time and create a balance between exploring a wider parameter space and exploiting a more refined parameter space. This approach provides greater flexibility in hyperparameter search, maximizing the results obtained without overburdening limited resources. This process involves testing combinations of unit GRU values, dropout probability, batch size, and number of epochs to identify the combination that provides the best accuracy. By systematically exploring these parameters, the model is optimized to be more effective on a particular dataset.

Evaluation Result

At this stage, predictions are made on training data and testing data that have been determined previously. Then evaluate the model performance on train data and test data using MAE, MSE, RMSE, R2 Score, and Accuracy of prediction results.

RESULTS AND DISCUSSION

Test results using the GRU method were carried out with an analysis tool that uses the Python programming language based on Google Collab. The first step in the analysis process with Python is to prepare the necessary libraries. Some of the libraries used include numpy, pandas, seaborn, matplotlib, tensorflow, and keras, as well as GridSearchCV, RandomizedSearchCV and MinMaxScaler from sklearn. How to call these libraries is illustrated in Figure 2.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set_theme(style="whitegrid")
import warnings
warnings.filterwarnings("ignore")
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, GRU
from sklearn.model_selection import GridSearchCV, RandomizedSearchCV
from scikeras.wrappers import KerasClassifier, KerasRegressor
from keras.callbacks import EarlyStopping
```

Source : (Research Result, 2025)
Figure 2. Library

The next step is to check the information in the dataset, namely the number of rows and columns contained in the dataset. Then remove null data, change the data type "Date" to datetime, and check for duplicate data and empty values. Then drop the variables that are not used, because in this research we will only use the variables "Date" and "Electric Vehicle (EV) Total". The results of the drop variables are as follows.

Table 3. Variable Drop Result

No.	Date	Electric Vehicle (EV) Total
1.	2017-01-31	1
2.	2017-01-31	1
3.	2017-01-31	0
4.	2017-01-31	258
5.	2017-01-31	1
...	...	...
20728	2024-02-29	1
20729	2024-02-29	4
20730	2024-02-29	1
20731	2024-02-29	1
20772	2024-02-29	7
20733	2024-02-29	947

Source : (Research Result, 2025)

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v10i4.6429

The next step is to perform feature scaling on the data using MinMax Scaler. This process aims to reduce redundancy prevent anomalies in the data and ensure the data has a uniform scale so that the model can learn efficiently [29]. The processed data is divided into 80% for training data and 20% for test data. Training data to train the model, and test data to evaluate the model at the prediction stage.

Table 4. Scaling Results

No	Date	Electric Vehicle (EV) Total	Scaled
1.	2017-01-31	1	0.000017
2.	2017-01-31	1	0.000017
3.	2017-01-31	0	0.000000
4.	2017-01-31	258	0.004507
5.	2017-01-31	1	0.000017

Source : (Research Result, 2025)

The next step is to create a sliding window function with window size and numpy array input with the resulting variable (X) as input and variable (Y) as target. The sliding window function aims to create a dataset that is ready to be used for time series models using the sliding window technique. This technique takes some data in a certain time range as input and the value at the next time as the target. At this stage, use a Window Size of 30 because the data is collected every month and applied to the training data and test data that has been scaled. With a window size of 30, the model can learn patterns that occur in about one month. A window size of 30 will make the model training process more computationally efficient. This can be useful for identifying short-term trends or fluctuations that may contribute to annual forecasting.

Table 5. Shape Results

Train	Test
16556	4117

Source : (Research Result, 2025)

The next step is to create a forecasting model and hypertuning the parameters using GridSearchCV and RandomizedSearchCV to determine the best parameters, by trying a combination of GRU unit values, dropout probability, batch size, and number of epochs. In this hyperparameter tuning process, 5-fold cross validation is used, which means the dataset is divided into 5 parts, and the model will be trained and tested 5 time, each time with a different combination of training and testing data splits. The results will be averaged to obtain a more stable evaluation of the model, ensuring the selection of optimal parameters. From the training results using GridSearchCV, it was found that the best parameters were GRU unit = 16, and dropout = 0.3, with the average Loss Function from the cross-validation results being -0.002468. Meanwhile, the training results using RandomizedSearchCV showed that the best parameters were GRU_unit = 64, and dropout = 0.6, with the average Loss Function from the Cross Validation results being -0.001869003917696066.

Tabel 6. Best Parameters with GridSearchCV

GRU_Unit	Dropout	Loss Function
16	0.3	-0.002468
16	0.4	-0.002883
32	0.3	-0.007502
32	0.4	-0.005844

Source : (Research Result, 2025)

Tabel 7. Best Parameters with RandomizedSearchCV

Parameters	Result
GRU_Unit	64
Dropout	0.6
Loss Function	-0.001869003917696066

Source : (Research Result, 2025)

The next step is to make predictions on the train data and test data that have been determined previously.

```
# Prediksi data train
predict_train = scaler.inverse_transform(grid_result.best_estimator_.model_.predict(X_train))
true_train = scaler.inverse_transform(y_train)

# Prediksi data test
predict_test = scaler.inverse_transform(grid_result.best_estimator_.model_.predict(X_test))
true_test = scaler.inverse_transform(y_test)
true_test = true_test[:len(predict_test)]
test['predict'] = test['predict'].fillna(0)
```

Source : (Research Result, 2025)
Figure 3. Prediction process

The code makes predictions on the train and test data using the best model obtained. First, predictions are made on train data (X_train) and the results are stored in train_prediction. Next, the same process is repeated for the test data (X_test), with the prediction results stored in prediction_test and the actual target value in true_test. Finally, the 'predict' column in the test DataFrame is filled with predicted values, and missing values (NaN) are filled with 0. The goal is to get model predictions on train and test data, as well as prepare data for evaluating model performance. The model evaluation uses MAE, MSE, RMSE, and Accuracy metrics. The next step measures the model performance on train and test data by calculating several evaluation metrics. The results of these metrics can be used to assess how well the model is at predicting target values and to compare the model's performance with other models. The MAE, MSE, and RMSE metrics measure prediction error,

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v10i4.6429

while the R2 score measures how well the model explains the variability of the data. Accuracy measures how accurately the model predicts the direction of change in the target value.

```
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score, accuracy_score

# Evaluasi performa model pada data train
mae_train = mean_absolute_error(true_train, predict_train)
mse_train = mean_squared_error(true_train, predict_train)
rmse_train = np.sqrt(mse_train)
r2_train = r2_score(true_train, predict_train)
accuracy_train = accuracy_score(true_train_direction, predict_train_direction)

# Evaluasi Performa model pada data test
mae_test = mean_absolute_error(true_test, predict_test)
mse_test = mean_squared_error(true_test, predict_test)
rmse_test = np.sqrt(mse_test)
r2_test = r2_score(true_test, predict_test)
accuracy_test = accuracy_score(true_test_direction, predict_test_direction)
```

Source : (Research Result, 2025)

Figure 4. Model evaluation

The evaluation results obtained using GridSearchCV are MAE: 699.65, MSE: 20815455.48, RMSE: 4562.40, and an accuracy of 60%. Meanwhile, the evaluation results obtained using RandomizedSearchCV were MAE: 257.94, MSE: 66655.087, RMSE: 258.176, and accuracy of 100%.

CONCLUSION

Based on the results of research that has been carried out, the model that has been created can work quite well in predicting the population of electric vehicles. The results show that the hyperparameter model using RandomizedSearchCV is the best model because it has the highest accuracy with a combination of GRU_unit 64 and 128, dropout 0.5 and 0.6, batch size 64 and number of epochs 100 which has MAE: 257.94, MSE: 66655.087, RMSE: 258.176, and 100% Accuracy. It is recommended that further research compare more hyperparameter tuning methods to find other, more efficient alternatives.

REFERENCE

[1] K. E. ArunKumar, D. V. Kalaga, C. M. S. Kumar, M. Kawaji, and T. M. Brenza, "Forecasting of COVID-19 using deep layer Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs) and Long Short-Term Memory (LSTM) cells," *Chaos, Solitons and Fractals*, vol. 146, p. 110861, 2021, doi: 10.1016/j.chaos.2021.110861.

[2] S. A. Almohaimeed, "Electric Vehicle Deployment and Integration in the Saudi Electric Power System," *World Electric Vehicle Journal*, vol. 13, no. 5. p. 84, 2022. doi: 10.3390/wevj13050084.

[3] S. Gurumoorthy, A. K. Kokku, P. Falkowski-Gilski, and P. B. Divakarachari, "Effective Air Quality Prediction Using Reinforced Swarm Optimization and Bi-Directional Gated Recurrent Unit," *Sustainability*, vol. 15, no. 14, pp. 1–19, 2023, doi: 10.3390/su151411454.

[4] W. Wen, S. Yang, P. Zhou, and S. Z. Gao, "Impacts of COVID-19 On The Electric Vehicle Industry: Evidence From China," *Renewable and Sustainable Energy Reviews*, vol. 144, 2020.

[5] M. K. Hasan, M. Mahmud, A. K. M. Ahasan Habib, S. M. A. Motakabber, and S. Islam, "Review of Electric Vehicle Energy Storage and Management System: Standards, Issues, and Challenges," *Journal of Energy Storage*, vol. 41, no. July, 2021, doi: 10.1016/j.est.2021.102940.

[6] Z. Chen, H. Zhao, Y. Zhang, S. Shen, J. Shen, and Y. Liu, "State of Health Estimation for Lithium-Ion Batteries Based on Temperature Prediction and Gated Recurrent Unit Neural Network," *Journal of Power Sources*, vol. 521, pp. 0–27, 2022, doi: 10.1016/j.jpowsour.2021.230892.

[7] X. Liu, Y. Wang, X. Wang, H. Xu, C. Li, and X. Xin, "Bi-Directional Gated Recurrent Unit Neural Network Based Nonlinear Equalizer for Coherent Optical Communication System," *Optics Express*, vol. 29, no. 4, p. 5923, 2021, doi: 10.1364/oe.416672.

[8] J. Gidseg, "The Impact of Electric Vehicle Charging Stations on Consumers," *Journal of Student Research*, vol. 12, no. 3. 2023. doi: 10.47611/jsrhs.v12i3.4996.

[9] R. Anastasya and S. B. Putri, "SDGs 7: Efektivitas Program Penggunaan Bus Listrik Guna Mendorong Transportasi Publik Ramah Lingkungan," *Journal of Environmental Economics and Sustainability*, vol. 1, no. 3, pp. 1–13, 2024, doi: 10.47134/jees.v1i3.343.

[10] G. Rajendran, C. A. Vaithilingam, N. Misron, K. Naidu, and M. R. Ahmed, "A Comprehensive Review on System Architecture and International Standards for Electric Vehicle Charging Stations," *Journal of Energy Storage*, vol. 42, 2021, doi: 10.1016/j.est.2021.103099.

[11] I. Miri, A. Fotouhi, and N. Ewin, "Electric Vehicle Energy Consumption Modelling and Estimation—A Case Study," *International Journal of Energy Research*, vol. 45, no. 1, pp. 501–520, 2021, doi: 10.1002/er.5700.

[12] J. Abramson *et al.*, "Accurate structure prediction of biomolecular interactions with

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 10. NO. 4 MAY 2025
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v10i4.6429

AlphaFold 3," *Nature*, vol. 630, no. 8016, pp. 493–500, 2024, doi: 10.1038/s41586-024-07487-w.

[13] S. Eide, D. S. Leslie, and A. Frigessi, "Dynamic slate recommendation with gated recurrent units and Thompson sampling," *Data Mining and Knowledge Discovery*, vol. 36, no. 5. pp. 1756–1786, 2022. doi: 10.1007/s10618-022-00849-w.

[14] N. Antulov-Fantulin, A. Cauderan, and P. N. Kolm, "A Dynamic Regime-Switching Model Using Gated Recurrent Straight-Through Units," *SSRN Electronic Journal*. 2024. doi: 10.2139/ssrn.4810879.

[15] A. Almutairi, A. Keshavarz-Haddad, and E. Aryafar, "A deep learning framework for blockage mitigation and duration prediction in mmWave wireless networks," *Ad Hoc Networks*, vol. 162. 2024. doi: 10.1016/j.adhoc.2024.103562.

[16] C. H. Wang and W. Q. Li, "A hybrid model of modal decomposition and gated recurrent units for short-term load forecasting," *PeerJ Computer Science*, vol. 9. 2023. doi: 10.7717/peerj-cs.1514.

[17] R. Feng, "Gated Recurrent Units for Lithofacies Classification Based on Seismic Inversion," *Artificial Intelligent Approaches in Petroleum Geosciences*. pp. 97–113, 2024. doi: 10.1007/978-3-031-52715-9_3.

[18] S. A. Kellerhals, F. De Leeuw, and C. Rodriguez Rivero, "Cloud Nowcasting with Structure-Preserving Convolutional Gated Recurrent Units," *Atmosphere*, vol. 13, no. 10. p. 1632, 2022. doi: 10.3390/atmos13101632.

[19] J. Pardede and M. F. Raspati, "Gated Recurrent Units dalam Mendeteksi Obstructive Sleep Apnea," *MIND Journal*, vol. 6, no. 2. pp. 221–235, 2021. doi: 10.26760/mindjournal.v6i2.221-235.

[20] I. Carolina and T. Haryanto, "Modeling Of Hyperparameter Tuned RNN-LSTM and Deep Learning For Garlic Price Forecasting In Indonesia," *Journal of Informatics and Telecommunication Engineering*, vol. 7, no. 2, pp. 502–513, 2024, doi: 10.31289/jite.v7i2.10714.

[21] B. Supranda, Solikhun, and Z. A. Siregar, "Penerapan Jaringan Syaraf Tiruan Backpropagation Dalam Memprediksi Jumlah Pertumbuhan Kendaraan Di Provinsi Sumatera Utara," *Resolusi : Rekayasa Teknik Informatika dan Informasi*, vol. 2, no. 4, pp. 152–160, 2022, doi: 10.30865/resolusi.v2i4.333.

[22] H. Khodijah, "Implementasi Jaringan Saraf Tiruan Untuk Memprediksi Penjualan Sepeda Motor Listrik," *JPILKOM (Jurnal Penelitian Ilmu Komputer)*, vol. 2, no. 3, pp. 103–108, 2024.

[23] K. Prayogi, W. Gata, and D. P. Kussanti, "Prediksi Harga Saham Bank Central Asia Menggunakan Algoritma Deep Learning GRU," *Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 13, no. 1, pp. 647–658, 2024, doi: 10.35889/jutisi.v13i1.1910.

[24] J. M. Lopez, M. Rivera, and A. Mendez, "Timescale Predictions Using Gated Recurrent Units," *Proceedings of the XXXVth URSI General Assembly and Scientific Symposium – GASS 2023*. 2023. doi: 10.46620/ursigass.2023.1371.iztl5570.

[25] S. Park, Y. Choi, and C. K. Kim, "Precipitation Nowcasting with Graph Neural Network and Gated Recurrent Units," *International Conference on Information Networking*. pp. 102–107, 2024. doi: 10.1109/ICOIN59985.2024.10572062.

[26] D. Varma, A. Nehansh, and P. Swathy, "Data Preprocessing Toolkit : An Approach to Automate Data Preprocessing," *Interantional Journal of Scientific Research in Engineering and Management*, vol. 07, no. 03. 2023. doi: 10.55041/ijsrem18270.

[27] M. Açikkar, "Fast grid search: A grid search-inspired algorithm for optimizing hyperparameters of support vector regression," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 32, no. 1. pp. 68–92, 2024. doi: 10.55730/1300-0632.4056.

[28] A. Tunçel, "Comparison of earthquake location parameters determined using grid search and manta ray foraging optimization," *Acta Geophysica*, vol. 72, no. 4. pp. 2581–2596, 2024. doi: 10.1007/s11600-024-01359-7.

[29] H. Lin *et al.*, "Time Series-Based Groundwater Level Forecasting Using Gated Recurrent Unit Deep Neural Networks," *Engineering Applications of Computational Fluid Mechanics*, vol. 16, no. 1, pp. 1655–1672, 2022, doi: 10.1080/19942060.2022.2104928.

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri