IJRCS
International Journal of Robotics and Control Systems
Vol. 4, No. 2, 2024, pp. 628-654
ISSN 2775-2658
http://pubs2.ascee.org/index.php/ijrcs
ASCEE

Utilize the Prediction Results from the Neural Network Gate Recurrent Unit (GRU) Model to Optimize Reactive Power Usage in High-Rise Buildings

Ahmad Rofii [a,1], Busono Soerowirdjo [b,2], Rudi Irawan [c,3], Wahyu Caesarendra [d,4,*]

[a] Faculty of Engineering and Informatics, Universitas 17 Agustus 1945 Jakarta, Jakarta 14350, Indonesia
[b] Faculty of Computer Science and Information Technology, Universitas Gunadarma, Depok 16424, Indonesia
[c] Faculty of Industrial Technology, Universitas Gunadarma, Depok 16424, Indonesia
[d] Faculty of Integrated Technologies, Universiti Brunei Darussalam, Gadong BE1410, Brunei
[1] rofii.ahmad@uta45jakarta.ac.id; [2] busono@staff.gunadarma.ac.id; [3] rudi_irawan@staff.gunadarma.ac.id;
[4] wahyu.caesarendra@ubd.edu.bn
* Corresponding Author

ARTICLE INFO

Article history
Received March 19, 2024
Revised May 02, 2024
Accepted May 08, 2024

Keywords
Peak Load;
Reactive;
GridSearch;
GRU;
Energy Efficiency;
High-Rise Building

ABSTRACT

The growing urbanization and the construction sector, efficient use of electric energy becomes important, especially the use of reactive power. If excessive use causes decreased efficiency and increased operational costs. Decreased efficiency contributes to increasing exhaust gas volumes and greenhouse emissions. Efficient energy can achieved if planning and predictions are correct. This research applies the GRU neural network method with grid search initialization as a novelty predictive model for energy-use high-rise buildings in form fast training without multiple iterations because optimal hyperparameters are obtained. Experimental show the MAE and RMSE performance metrics of the GRU better than LSTM in predicting energy consumption data peak loads, off-peak loads and reactive power. The accuracy of GRU predictions can optimize the use of energy to contribute to saving the environment from exhaust emissions and the greenhouse effect in urban systems. Experimental results demonstrate the superiority of GRU over LSTM, proof of the much lower MAE and RMSE values. This metric shows the accuracy of GRU in generalizing data both during peak and off-peak hours, as well as in reactive power usage. By Utilizing GRU's capabilities, building management can manage reactive power usage effectively, allocate reactive power resources appropriately, and mitigate peak load times and the power factor within the threshold, thus avoiding additional costs and electrical system efficiency and contributing to reducing the carbon footprint and gas emissions greenhouse. Research on GRU is widely open in the high-rise building sector, including its integration with sensors to automatically control energy use.

1. Introduction

The rapid development of the construction sector and urbanization brings significant implications for electricity consumption, especially in high-rise residential buildings, which are crucial elements in urban structures [1]. The efficient use of energy has become an urgent necessity global climate change and demands for sustainable resources [2]. One crucial aspect of electricity consumption management is the utilization of reactive power in high-rise residential buildings. This factor can significantly influence system efficiency and energy costs, particularly during peak electricity demand periods within an electrical system. It's often during these peak energy demand moments that energy efficiency can either be enhanced or diminished [3], [4].

- Management Challenges: Peak load management is an important aspect of energy system sustainability [5]. When loads reach their peak, energy infrastructure faces significant pressure. Increasing operational costs, the risk of overload, and the potential for disruptions or power outages become issues that need to be addressed [6]. Effective strategies in managing peak loads, such as load scheduling and the use of cloud computing technology, can help optimize energy usage, reduce operational costs, and prevent energy supply failures or disruptions [7].
- Role of Reactive Power in High-Rise Building Electrical Systems: There is a correlation between peak loads and reactive power that highlights the importance of optimizing reactive power settings; the system can effectively support line voltages, especially during peak loads [8]. Additionally, the implementation of time-based electricity tariffs can improve energy usage efficiency by providing price signals that reflect the actual production costs of electricity at specific times. Responsive electricity demand management, understanding reactive power, and the implementation of time-based tariffs are essential steps in managing peak loads and ensuring energy sustainability in the future [4], [9].
- Optimizing Reactive Power Management in High-Rise Residential Buildings: Previous research has highlighted the importance of reactive power management, but there are still shortcomings in understanding and accurately predicting reactive power usage during peak and off-peak conditions [8]. Data analysis approaches and advanced modeling techniques are used to develop predictive models that can identify patterns of reactive power usage in high-rise residential buildings during peak and off-peak periods [10]. By detailing the characteristics and variability of reactive power usage, it is expected to provide in-depth insights to building managers and electricity service providers on how reactive power can be optimized during peak and off-peak hours, thus contributing positively to energy efficiency, operational cost reduction, and the creation of a more sustainable environment [11].
- Significance of the Predict Model: This relevance is not only in the context of managing high-rise residential buildings but also in supporting a broader transformation towards more efficient and sustainable energy systems [12]. By understanding the dynamics of reactive power usage, proactive steps can be taken to improve energy performance and design more environmentally friendly buildings in the future. Accurate predictions of reactive power in the future, especially during low peak load times and off-peak loads, enable efficient resource management [11]. Deep learning techniques such as RNN and its various variants have been proven effective in producing accurate predictions, thus supporting sustainable planning and future technological innovation [13].

Recurrent Neural Network (RNN) is a part of deep learning that processes sequential inputs and stores information from the past [14]. RNN is used in many temporal processing applications and can store information for decision-making. Recurrent network models generally use column input vectors with weight matrices representing the relationship between neurons and features in the network [15]. There are three common types of RNN variants: LSTM, GRU, and Bidirectional. Each variant of RNN has its advantages and disadvantages [16]. LSTM is the solution for large and complex data, while GRU or bidirectional is the solution for smaller data. The architecture of the GRU model is simpler than Bidirectional [17]. GRU is one variant of RNN that allows its use to be more adaptive and efficient for various machine learning tasks and sequential processing that are not too large. GRU can be applied to sequential data or time series data [18]. The time series phenomenon is a phenomenon produced by an activity that has values and time sequences [19]. Time series are also often used in decision-making and planning in various fields because they can provide insights into patterns and trends that may occur over a certain period [20]. The time series phenomenon also occurs in the consumption of electrical energy in a building [21], [22]. Electrical energy consumption is a time series phenomenon, so Deep learning models are relevant to be used as predictive models as previously applied by researchers [23]. So far, the management of electrical energy consumption in a building is based on daily data from a month ago and the present as material for deciding regulations regarding the availability and use of electricity in the coming month, especially in regulating the use of reactive power in the future [24]. The following is the advantages of the GRU model:

- The GRU model can predict the future with not too large data and can be used to overcome the complexity and dynamics of high reactive power consumption data, thus helping estimate power needs at different times [22]. This solution can provide accurate information and assist building managers in designing more efficient energy management strategies [25], [26].
- Improving the Performance of the GRU Model in Predicting Reactive Power Usage GridSearch Initiation Approach.
- The use of GRU with the initiation of GridSearch to predict reactive power usage is an important contribution to the field of electrical energy management.

The following are several gaps in previous research summarized in Table 1.

- The use of hyperparameter optimization with GridSearch initiation to predict the use of reactive power in a high-rise building can still be developed.
- The Neural Network model performance in the previous research can be developed further especially in GridSearch initiation method to optimize hyperparameters GRU model performance for improving prediction accuracy of electrical energy management.
- Previous research has not emphasized comparative analysis between the GRU model and other approaches.
- In the previous research, GridSearch initiation was used and compare with other prediction models, such as LSTM and GRU. The result showed that the GridSearch is better than LSTM.

Table 1. Comparative research

Reference	Model	Novelty	Dataset Type
[27]	GRU AdaBoost	different weightings	Time series failure machine data
[28]	Combine	Ensemble RNN, LSTM, GRU	Economics times series data
[22]	GRU and LSTM	Update gate	Time series data
[29]	GRU and LSTM	GridSearch	Dataset sequential electrical tools
[28]	LSTM and GRU	Ensemble	Sequential Data
[30]	LSTM and GRU	Arrange hyperparameter	Noise dataset
[18]	GRU	Internal parameter	Economic dataset series
[31]	GRU	Bidirect	Text semantics
[17]	LSTM GRU	Increasing layer and cells	Human activation
[16]	GRU and LSTM	Trajectory linearization	different MPC schemes
[32]	GRU and LSTM	Reviews methods for building energy consumption forecasts	building energy consumption

Through a literature review, it was shown that previous research has introduced various models, as displayed in Table 1. However, there remains a gap in enhancing the performance of prediction models. Specifically on reactive power consumption in buildings. The majority of previous studies do not specifically study reactive power in high-rise buildings, Even though reactive power is a crucial aspect of enhancing energy efficiency in urban settings. With gaps occurring, we propose a study focused on GRU predictive models initialized with GridSearch. This study compares GRU and LSTM models to ensure GRU is processing reactive power consumption better.

Based on existing gaps, this research objective develops a neural network prediction model that can generalize across various data domains, to increase the efficiency of electrical energy use in high-rise buildings by using energy consumption data in buildings. Furthermore, the resulting prediction model, especially for the use of reactive power, can be used for building management decision-making to increase energy efficiency and minimize operational costs. This research contribute to overcoming the use of reactive power due to bad planning and avoiding energy waste and greenhouse gas emissions. Therefore, neural network prediction models, especially the GRU model, are the subject of discussion as an option to contribute to overcoming environmental problems caused by energy use.

Without an accurate prediction model, the use of reactive power will be excessive, which can lead to increased efficiency environmental damage and waste of energy. This happens when too much reactive power is used, the electrical system has to produce more energy than it needs, ultimately increasing energy consumption and greenhouse gas emissions. This not only increases operational costs but also negatively impacts the environment by increasing the carbon footprint and exacerbating climate change. Therefore, efficient reactive power management is important to preserve the environment.

2. Method

This research was built GRU predictive model to optimize the use of reactive power in highrise buildings. The research used methodology tools the Python library, with the steps of literature review, modelling and initiation, data processing, training process, experiment setup.

2.1. Electricity Power Consumption Study

The study begins by understanding the significance of reactive power consumption in multi-story buildings and its impact on operational costs and electrical network stability. Different types of power, including reactive power, active power, and apparent power, are defined and their significance in building energy management is highlighted. The consumption of reactive power during peak loads in multi-story residential buildings as the main consumers of electrical energy has a significant impact on the stability of the electrical network and operational costs [8]. This is represented by the following formula. In electricity, there are three types of power i.e., reactive power (Q), active power (P), and apparent power (S).

$$Apparent\ power: (S) = P^2 + Q^2 \tag{1}$$

$$Reactive\ power: (Q)\ = V\ X\ I\ x\ sin\ \emptyset \tag{2}$$

Current (I) is the current in the line, $\emptyset$ is the phase angle between Voltage and Current, Active power (P): Active power is measured in watts (W) and defined as the product of voltage, current, and the cosine of the phase angle between voltage and current.

$$Active\ power: (P)\ =\ V\ X\ I\ x\ cos\ \emptyset \tag{3}$$

$$Power\ factor\ = \frac{P}{S} \tag{4}$$

The fine (D) imposed on electricity users (buildings) for excessive reactive power usage can be calculated based on the difference between the actual reactive power (Q_{actual}) used and the maximum allowable limit (Q_{max}). An illustration of excess reactive power which causes additional costs in the form of fines is shown in Fig. 1.

$$D = (Q_{actual}) - \ (Q_{max}) \tag{5}$$

The more the reactive power used, the smaller the power factor, the smaller the power factor, the lower the efficiency of the electrical system. Increasing reactive power during peak loads can cause significant power losses and require quick corrective action [33]. On the other, during off-peak loads, inefficient use of reactive power can result in energy wastage, and suboptimal reactive power management can lead to inefficient energy usage and increasing operational costs for reactive power load [33]. Reactive power concept is analogous to force in a mechanical system, as illustrated in Fig. 2.

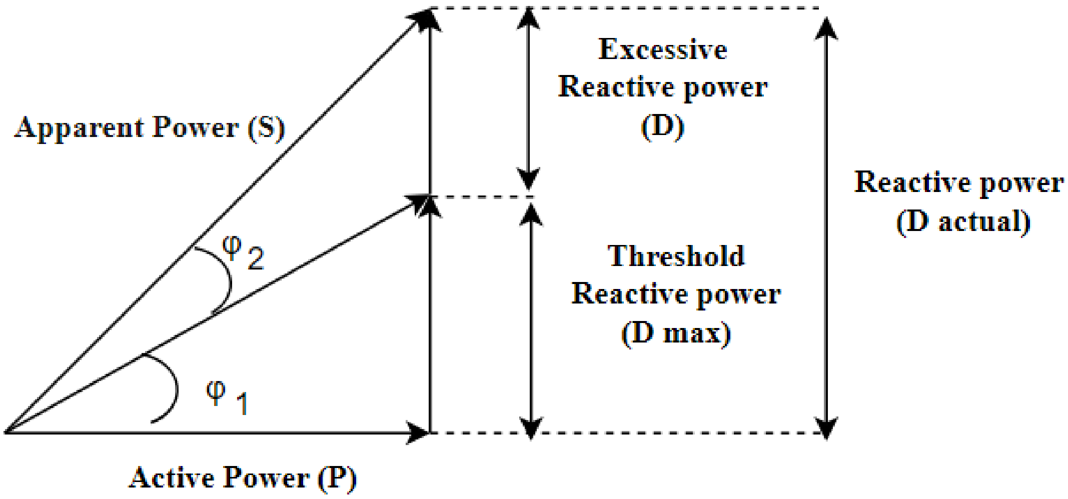

Fig. 1. Relationship between Excessive actual and maximum allowable limit reactive power

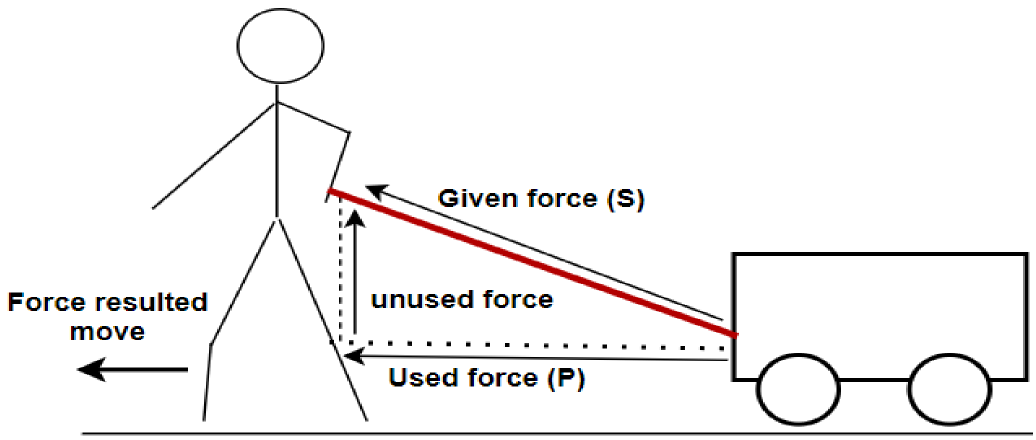

Fig. 2. Analogy of reactive power with mechanical concepts

Reactive power in electrical system, if likened to mechanical force, is illustrated in Fig. 2 this figure elucidates the force exerted (S) by the hand to pull the load, the force applied by the foot to initiate movement, and the upward force symbolizes reactive power (unused force). From this depiction, it becomes apparent that higher reactive power results in greater provided power, yet diminished power utilization renders the system less efficient [34].

2.2. Neural Network Study

Neural Network is a category of Soft Computing science. Neural Networks actually adopt the ability of the human brain which is able to provide stimulation, carry out processes, and provide output. The output is obtained from variations in stimulation and processes that occur in the human brain. A recurrent neural network (RNN) is a type of artificial neural network architecture whose processing is called repeatedly to process input which is usually sequential data [35]. RNN is included in the deep learning category because data is processed through many layers.

The RNN began with the familiar Artificial Neural Network (ANN) concept. The transition from ANN to RNN involves additional terminology in the hidden layer (h), where there is the presence of weighted metrics multiplied by input plus bias, and there is a difference, namely the addition of weight metrics from one hidden layer to another [36]. The RNN concept builds upon the familiar ANN concept. In transitioning from ANN to RNN, additional terms are introduced in the hidden layer (h), where weighted metrics are multiplied by input plus bias. The key difference lies in the addition of weight metrics from one hidden layer to another. In the RNN architecture, the hidden layer (h) is linked to the previous hidden layer (h-1), resulting in an output similar to a regular ANN. The neural network calculates the previous hidden layer (ht-1) and incorporates it into the output. This process continues for each input, with each subsequent calculation incorporating the previous hidden layer. There are several types of RNN (Recurrent Neural Network) architectures used to model data sequences:

$$D = (Q_{actual}) - (Q_{max}) \tag{6}$$

2.3. Get Recurrent Unit

GRU is a type of recurrent neural network that was developed to overcome the difficult training and long-term memory problems in RNN, but still has the ability to handle the problem of long-term dependencies in serial data [32]. The GRU structure is shown in architecture by combining several gates into one "update" gate as presented in Fig. 3. GRU has two main Gates [18], [31], [32], [37].

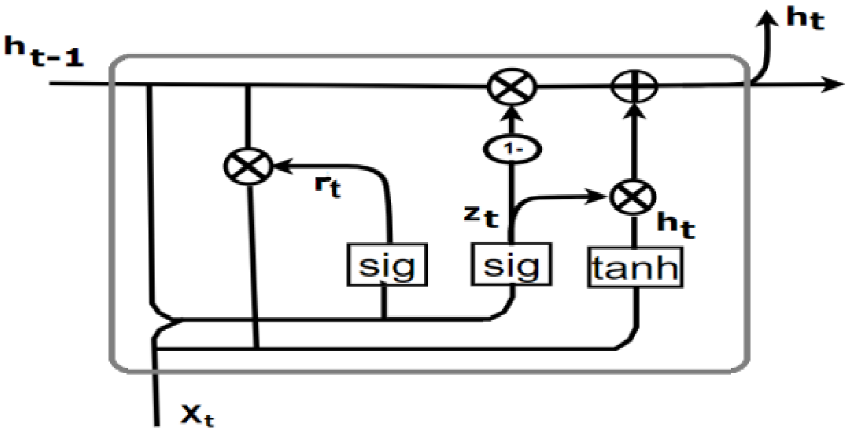

Fig. 3. Architecture of GRU [32]

The formula for determining how much information from the past will be forgotten is Reset Gate (r_t).

$$r_t = \sigma(w_z.[h_{t-1}, x_t]) \tag{7}$$

The reset gate controls forgetting old information, aiding the network to focus on relevant data. Meanwhile, the update gate manages the retention of past information while integrating new data, ensuring the GRU adapts to new insights while retaining valuable past knowledge. The formula Update Gate (z_t) is used.

$$z_t = \sigma(w_z.[h_{t-1}, x_t]) \tag{8}$$

z_t is similar to a gate controlling information flow from the past to the present in the network. It's similar to deciding how much of a previous conversation we retain before accepting new information, if relevant the gate opens widely, if not it remains partially open or shuts completely. The sigma (σ) function acts as an activator, adjusting the weight of past and current inputs. A higher output means more past information is retained or ignore. This analogy clarifies how GRU gates manage information flow within the network. Represents the value proposed to become a new hidden state, namely Candidate Hidden State (h_t)

$$h_t = \tanh(w.[r_t \odot h_{t-1}, x_t]) \tag{9}$$

Vol. 4, No. 2, 2024, pp. 628-654

h_t is a search or exploration process in the cell to find new hidden state candidates. The hyperbolic tangent function (tanh) acts like a filter that regulates how much the candidate is relevant and worthy of becoming a new hidden state (h_t). Sigma (σ) represents exploration activity, while dot product (⊙) indicates the association between the previous hidden state (h_{t-1}) and the current input (x_t) with the new candidate under consideration. The larger the product, the more significant the candidate's contribution to the new hidden state. The next stage is the Hidden state which is determined by information from the candidate's hidden state and updated gate.

$$h_t = (1 - z_t) \odot h_{t-1} \odot \tilde{h}_t \tag{10}$$

2.4. Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) is a type of RNN specifically designed to solve the problem of vanishing or exploding gradients. By using a gate mechanism, LSTM can select relevant information to store or delete from memory cells [32], [38]-[42]. The working principle of LSTM (Long Short-Term Memory) is based on its structure consisting of interconnected memory cell units. Each memory cell has three main gates: the forget gate, the input gate, and the output gate shown Fig. 4.

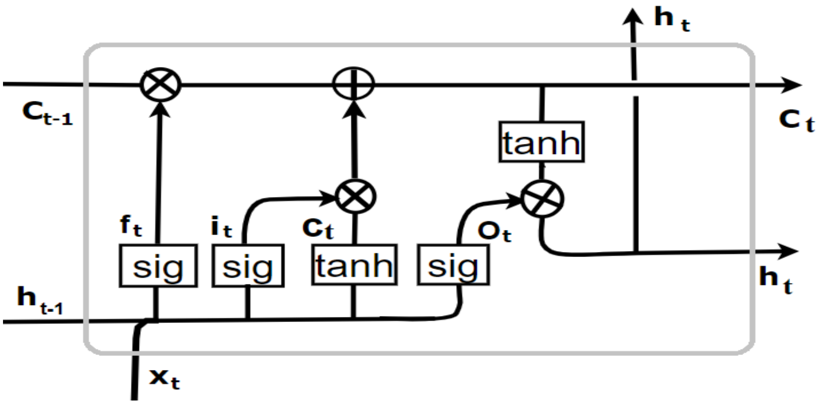

Fig. 4. Architecture of LSTM [15]

Forget Gate (f_t) determines how much information from the previous memory cell will be forgotten or retained. It helps LSTM to "forget" irrelevant information from the past, enabling the model to focus on relevant information. The forget gate formula is:

$$f_t = \sigma\left(w_f . [h_{t-1}, x_t] + b_f\right) \tag{11}$$

Input Gates (i_t) decides how much new information will be stored in the memory cell. The Input Gate formula

$$i_t = \sigma\left(w_f . [h_{t-1}, x_t] + b_i\right) \tag{12}$$

It allows LSTM to incorporate new information into memory based on the current input. Output Gate (O_t) is control gate that is the output produced by the memory cell. It helps LSTM to select which information will be passed to the next layer in the network. The formula Output Gate.

$$O_t = \sigma(w_o . [h_{t-1}, x_t] + b_o) \tag{13}$$

In addition to these gates (C_t), LSTM also has internal memory cells that store both short-term and long-term information. These memory cells aid LSTM in overcoming the vanishing or exploding gradient problem common in traditional recurrent networks. The update state formula.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{14}$$

2.5. Units

Advantages and disadvantages GRU and LSTM shown in Table 2.

Table 2. Advantage and disadvantage of GRU and LSTM

Comparison item	LSTM	GRU
Data series input capability	Capable of capturing long sequences	Less able to capture long sequences
Vanishing gradient	Controlled	Resolved
Over fitting	prone	Prevent
Computation	complicated	Simple
Temporal data	adaptation to temporal patterns, LSTM has been proven to be effective in handling complex temporal data	Suitable for short time series, suitable for text data
Hyper parameter	Necessary setting	Necessary setting
Memory cells	Address long-term dependency issues well and retain relevant information from the past	Managing long-term dependency problem memory cells and remembering relevant information from the past
Data set	Capable of large time series datasets	Fits small data sets
Internal architecture	Necessary setting	Necessary setting

2.6. Data Processing

Data processing is carried out before training, namely by ensuring the data structure whether there are potential problems such as data inconsistencies, imbalances, or incompatible formats that can be resolved which reduces the effectiveness and accuracy of the model being trained. In data processing, validation of the time series data format is also carried out to ensure the readiness of the data for time series analysis regarding the suitability of the date and time of the data which must be arranged based on time. To ensure that the data is consistent over time intervals, the data is visualized with Time Stamps in daily, weekly, and monthly terms. Fig. 5 shows the data processing steps.

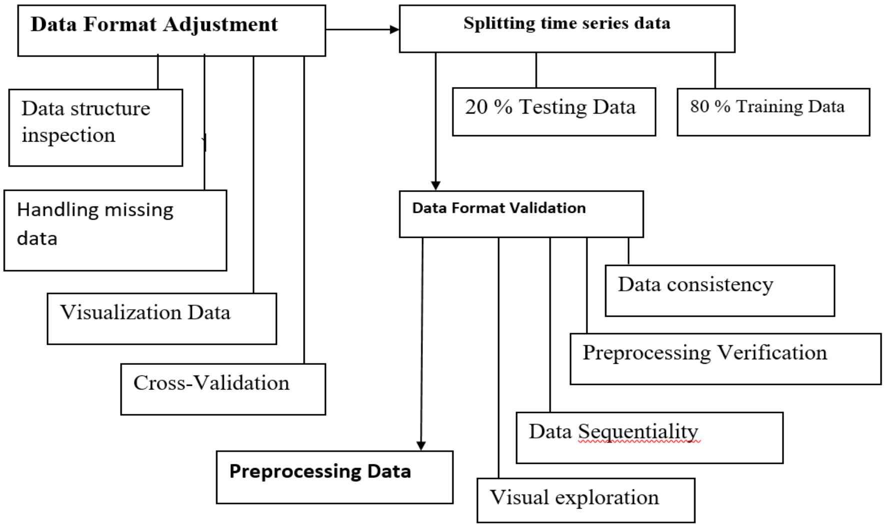

Fig. 5. Steps of data processing

2.6.1. Data Format Adjustment

Data processing is carried out before data training. In data processing, it is ensured that the data is structured by the GRU Model to avoid data inconsistencies, imbalances, or incompatible formats that can be overcome, by dealing with problems of missing data and outliers in the data. Next, the training material data that has been arranged is validated in a time format that is arranged based on time. The data is also split for training materials and testing materials, while to adjust the scale of the 12 existing data variables to be uniform to make the training process easier, the data is converted into a time series and normalized.

2.6.2. Data Format Validation

Adjusting the data format before the training process is adjusted to the structural needs of the GRU Model to avoid data inconsistencies, integration, or incompatible formats which can be overcome, by dealing with problems of missing data and outliers in the data. Furthermore, the training material data that has been prepared is validated in a time format arranged based on time. The data is also separated for training material and test material, to adjust the scale of the 12 existing variable data so that it is uniform so that the training process is easier, the data is converted into a time series and normalized.

2.6.3. Data Splitting

Before training, the data is split into several different subsets: 80% is used for training input-output pairs, and 20% is for testing. The testing subset is utilized to assess the model's performance after training is completed. The testing data subset is to estimate how well the model will perform on new, unseen data.

Adjusting the data format before the training process is adjusted to the structural requirements of the GRU Model to avoid data inconsistencies, integration, or incompatible formats which can be overcome, by dealing with problems of missing data and outliers in the data. Furthermore, the training material data that has been prepared is validated in a time format arranged based on time. The data is also separated for training material and test material, to adjust the scale of the 12 existing variable data so that it is uniform so that the training process is easier, the data is converted into a time series and normalized.

2.6.4. Data Sequentiality Testing

The data is validated for the existence of patterns or trends over time and ensures that the time structure is set correctly in statistical analysis or modeling, sequential tests are carried out on the data that will be used as training data.

The sequential testing process of the GRU prediction model begins by generating sequential data and converting it into a NumPy array in the Python library using np.array(). This data serves as a time series dataset for sequential processing. Next, a function is created to divide the data into different orders. In a loop that iterates over data indices, a long sequence seq_length is created by truncating the corresponding data. This sequence is then added to the list. The sequences are divided into training and test sets. The split_ratio variable determines the proportion of data allocated for training, while the split_index variable computes an index for splitting sequences based on this ratio. The sequence is divided into X_train and y_train, representing the target input and output features for training, respectively. Similarly, X_test and y_test are created for testing purposes [43].

In the next step, the input data is reshaped to meet the requirements of the GRU model, which expects input in three dimensions: (number of samples, number of time steps, and number of features). Since the data only has one feature per sequence, the third dimension is set to 1. Finally, the training and test data forms are printed to ensure that the data has been properly prepared for use in the GRU model.

2.6.5. Data Normalization

Normalization is a method for changing the scale of variables in order to speed up and compress neural network algorithms [44]. The data to be trained or used as training material and to be tested or used testing consists of variables with different scales, such as WBP, LWBP, and other data with varying dynamic ranges. Therefore, in the training process using deep learning, GRU requires uniform scaling (normalization). In this case, the normalization method employed is Min-Max Scaling. With this method, the data values are adjusted to fit within a specified range, typically between 0 and 1. The formula for Min-Max Scaling is as follows:

$$\mathrm{x}_{\mathrm{Norm}} = \frac{\mathrm{x} - \mathrm{x}_{\mathrm{min}}}{\mathrm{x}_{\mathrm{max}} - \mathrm{x}_{\mathrm{min}}} \tag{15}$$

where x is the original data value, $\mathrm{x}_{\mathrm{Norm}}$ is the normalized data value, $\mathrm{x}_{\mathrm{min}}$ is the minimum value in the data set, and $\mathrm{x}_{\mathrm{max}}$ is the maximum value in the dataset.

2.6.6. Metric Evaluation

Mean Absolute Error (MAE) used to measure model performance when testing prediction results on test data or data that was not used during training. The use of MAE in the testing phase is to evaluate the scop to which the model is able to generalize to new data that was never seen during training. Referring to researchers who have used prediction accuracy evaluation, namely Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) [45]. Prediction performance is determined generally using prediction accuracy evaluation, namely Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are used as the prediction accuracy evaluation. MAE is calculated in a similar way to during training, but this time comparing the model predictions to the actual values on the test data [46], [47].

$$MAE = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} |X_i - Y_i| \tag{16}$$

The formula for Root Mean Squared Error (RMSE) is as follows [47], [48]:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n_{test}} (X_i - Y_i)^2} \tag{17}$$

where n_{test} is the number of samples in the test dataset Y_i is the actual value for sample i in the test dataset. Y_i is the value predicted by the model for the i^{th} sample in the test data set and the value predicted by the model for the i^{th} sample. Based on both formula, the minimum value is 0, while there is no limit to the maximum value because they are an absolute measure of prediction error. The smaller they value, the better the model is at predicting the data.

2.7. Training and Testing Process

This research uses a prediction neural network principle, namely Gated Recurrent Unit (GRU) and compare it with the Long Short Term Memory (LSTM) method which are compared with recurrent neural network. Both models undergo an optimization process using the Adam algorithm, ensuring fast and efficient convergence in learning. Next, hyperparameter tuning was carried out to find the best combination of parameters in both models, namely by using the GridSearch method, with a focus on adjusting the number of neurons and layers.

The training process was done in stages using data on energy consumption during peak load (wbp), energy consumption during off-peak hours, and reactive power consumption (kVh) data as prediction output targets. The process of training a GRU model using the grid search method involves several steps. The following is an explanation of this process [32], [39].

Training models

- Data were prepared according to the requirements of the GRU model
- Hyperparameters are determined in advance through gridsearch, including number of layers, learning rate, optimization type, loss function and epoch
- The Grid search was done to find the optimal combination of previously determined hyperparameter values.

- After the optimal hyperparameter combination was found via grid search, the GRU model was trained using the training data. The training process is carried out using an optimal combination of hyperparameters.

Testing Models

- The data was prepared according to the requirements of the GRU model, the same as when preparing training data, including normalization and sequence formation
- After the test data is prepared, the data is fed into the trained GRU model to make predictions. The model will predict.
- After getting the predicted values, the model performance is evaluated using the mean Absolute Error (MAE) and Root Mean Square Error evaluation metric.
- The results of the testing process are analyzed to conclude the effectiveness of the GRU model in making predictions on previously unseen data

The goal of these steps is to produce optimal predictions, improve accuracy, and make a significant contribution to energy efficiency and more effective power management. The following is a diagram of the research steps in Fig. 6.

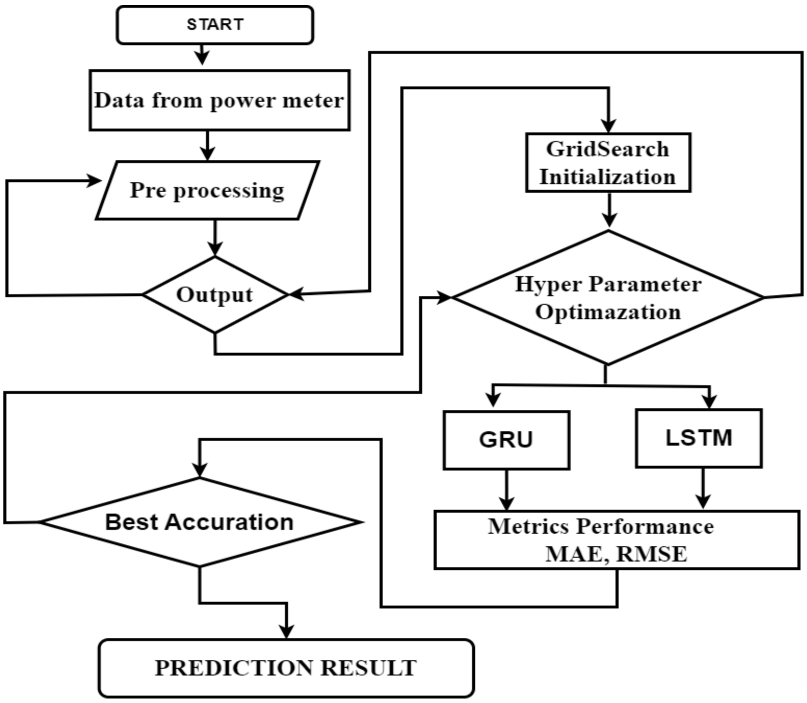

Fig. 6. Step of research

2.7.1. Initialization

A statement in a journal, initialization can produce stable and fast training for networks with weight distribution [49]. The mathematical formulation for GridSearch involves an objective function f(p), where p is a parameter vector to be optimized. The search aims to find the combination of parameter values that minimize or maximize the objective function. Assuming there are n parameters to be optimized [50].

Fig. 7 illustrates the proposed architecture in this study, employing GridSearch initialization on the GRU model to discover the optimal combination of weight and bias parameters for enhanced prediction accuracy. The initiation process for GridSearch is as follows: The GRU model is first defined using the TensorFlow library. GridSearch is then defined using the scikit-learn library. Subsequently, the training and testing processes for produce prediction outputs.

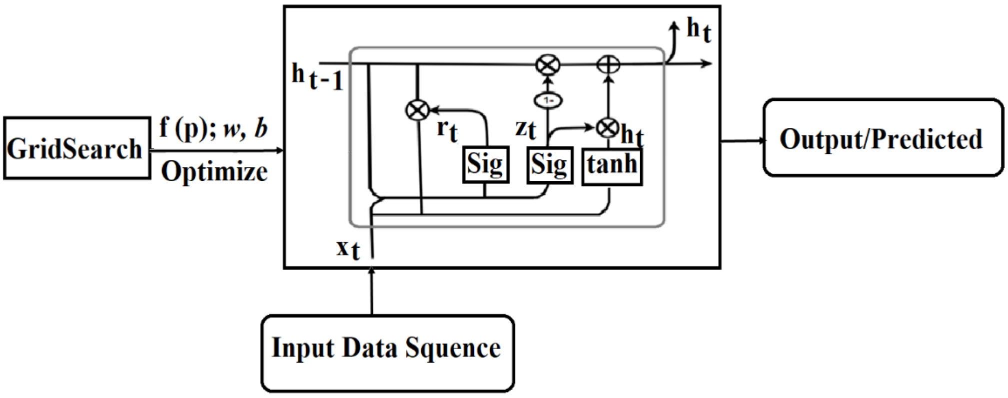

Fig. 7. The architecture of neural network GRU GridSearch

2.7.2. GridSearch Neural Network Architecture

First, determine the set of possible values for each parameter. If M1 possible values for the first parameter, m2 values for the second parameter, and so on up to the nth parameter [51], where mi represents the number of values for the ith parameter.

Then, consider all combinations of these parameter values. The total number of combinations to be tested is m1 × m2 ×⋯×mn. For each parameter combination, evaluate the objective function f(p) and record the resulting value. After evaluating all combinations, determine the parameter combination that yields the minimum or maximum value for the objective function. Thus, mathematically, GridSearch can be represented as follows initialize the set of values for each parameter. For each parameter combination, evaluate the objective function and record the resulting value. Then, select the parameter combination with the minimum and maximum values from the evaluation results [52].

2.7.3. Model Initialization

The process of initiating the search for the most optimal combination of bias values and weight values uses the GridSearch algorithm which has the following process. Define the GRU model defined as a function 〖(f〗 _αb) with parameter α,b as weight and bias of the GRU model if {a, b} 〖{α,b}={(α〗 _1,α_2,α_3,…,α_n),(b_1,b_2,b_3,…b_1)} and n is the number of weight and bias parameters to be optimized.

The GridSearch Initialization process aims to find the optimal combination of weight and bias values in the GRU model using a grid search algorithm [53]. The first step is to define the GRU model. GRU is defined as a function 〖(f〗 _αb) with parameter α,b as weight and bias of the GRU model if {a, b} 〖{α,b}={(α〗 _1,α_2,α_3,…,α_n),(b_1,b_2,b_3,…b_1)} and n is the number of weight and bias parameters to be optimized. Referring to equation (6), if bn = 0, then there will be no contribution from bias to the output, and if W=0, then there will be no contribution from weight to the output. Thus, the output (y) will heavily rely on the input (x) and the previous hidden state (ht−1). This implies that the GRU model may not be able to produce accurate predictions if no patterns are found. Therefore, the range of weight and bias values is greater than 0 (zero).

Before running training with GridSearch initialization, several hyperparameters are determined before. The following are the hyperparameters that influence the neural network model include:

- Defining the param_grid dictionary involves structuring data in the form of a dictionary containing a list of parameters to be tested and the values to be tried for each parameter during the process of finding the best parameters using GridSearch.
- In this study, it is deemed best to evaluate the 'n_estimators' parameter (the number of estimators in the model) with values of 50, 100, and 200. This means that we will search for the best value for 'n_estimators' by trying values of 50, 100, and 200.

- The best value for 'max_depth' is a parameter that controls the maximum depth of the decision tree in the model, with options of None, 10, and 20. A value of 'None' indicates no limitation, while values of 10 and 20 determine the maximum allowed depth. By trying out these values, we can assess the complexity of the model in various ways.
- During each iteration, one part of the data will be used as the testing data, while the rest will be used as the training data. By using this cross-validation method, we can measure the model's performance more reliably and reduce the risk of overfitting or overgeneralization. In this study, cross-validation is set to 5, meaning the data will be divided into 5 equally sized parts, and the training and testing processes will be repeated 5 times. During each iteration, one part of the data will be used as the testing data, while the rest will be used as the training data.

In the article [54], [55], one of the obstacles faced is determining the optimal hyperparameters for the LSTM model used in predicting electrical loads, that finding the optimal number of hidden layers and number of neurons is a difficult and non-deterministic problem.

2.8. Experimental Setup and Data Acquisition

The data to be used for training comes from an apartment where the electricity measurements are taken at the MDP (Main Distribution Panel) on the Medium Voltage side. The energy usage measurement process is depicted in Fig. 6. Measurements are conducted using digital kWh meters, kVARh meters, ammeters, voltmeters, and power factor meters. The electrical parameters read by the measuring devices are in units of kilowatt-hour (kWh), kilovolt-ampere reactive hour (kVARh), amperes, volts, and power factor, respectively, at the Main Distribution Panel (MDP) position. The MDP is located in the Medium Voltage 20kV cubicle area, which represents the customer's workspace at State Electricity Company (PT Perusahaan Listrik Negara). This cubicle area is positioned at the medium Voltage 20 kV (primary transformer) location. Table 3 presents the data, which has been given new identities. These are shown in the form of a table excerpt. The data is recorded every day at 8:00 PM.

Table 3. Terminology dataset acquisition

Data set acquisition	Information	Variable identity
Energy consumption during peak load	Electricity usage during high-demand times; optimizing energy resources.	wbp
Energy consumption during off-peak load	Electricity usage during low-demand periods; minimal energy consumption.	lwbp
Reactive power consumption	Reactive power utilization in electrical systems	kvh
Current line 1	Electrical current flowing electrical cable 1.	I1
Current line 2	Electrical current flowing electrical cable 2.	I2
Current line 3	Electrical current flowing electrical cable 3.	I3
Voltage line 1	Voltage in electrical cable 1	V1
Voltage line 2	Voltage in electrical cable 1	V2
Voltage line 3	Voltage in electrical cable 1	V3
Current reactive 1	Electrical reactive current flowing electrical cable 1	I1h
Current reactive 2	Electrical reactive current flowing electrical cable 2	I2h
Current reactive 3	Electrical reactive current flowing electrical cable 3	I3h

2.9. Data Characteristics Type and Dimensions

The data collected has the characteristics of time series data. This characteristic can be seen from the presence of the date and time column (Date Time) as an index or feature in the dataset.In this case, the data of variables measured in date and time, electrical power consumption (wbp, lwbp, kvh), voltage (v1, v2, v3), and current (I1, I2, I3). The data dimensions are 11 variables (wbp, lwbp, kvh, v1, v2, v3, I1, I2, I3, I1h, I2h). There are 340 different dates as index or time columns. Therefore, the data dimensions are 340 rows and 11 columns.

The data a date or string (Date Time) which represents the date or time. Numerical variable data (wbp, lwbp, kvh, I1, I2, I3, Ih1, Ih2, Ih2, V1, V2, V3) is a decimal number (float) data type. . The data set shows the characteristics of time series data. This can be seen from the date and time column (Date Time) which functions as an index or feature in the dataset.

The dimensions of the given dataset are 11 variables (wbp, lwbp, kvh, v1, v2, v3, I1, I2, I3, I1h, I2h). There are 340 different dates serving as indices or time columns. Therefore, the data dimensions are 340 rows and 11 columns. Each row in the collected dataset represents observations made daily, thus the data frequency is per day.

The data acquisition process is shown in Fig. 8. The data is obtained through a Digital kWh Meter located in the Main Distribution Panel (MDP) of a 20 kV Medium Voltage system in an Apartment Building in Jakarta, Indonesia. The digital kWh data is monitored and recorded every day at 20:00. The data used in this study ranges from January 1, 2022, to December 7, 2022. The recorded sample data is presented in Table 4. This data is a conversion of the log sheet data of the electricity manager's daily records.

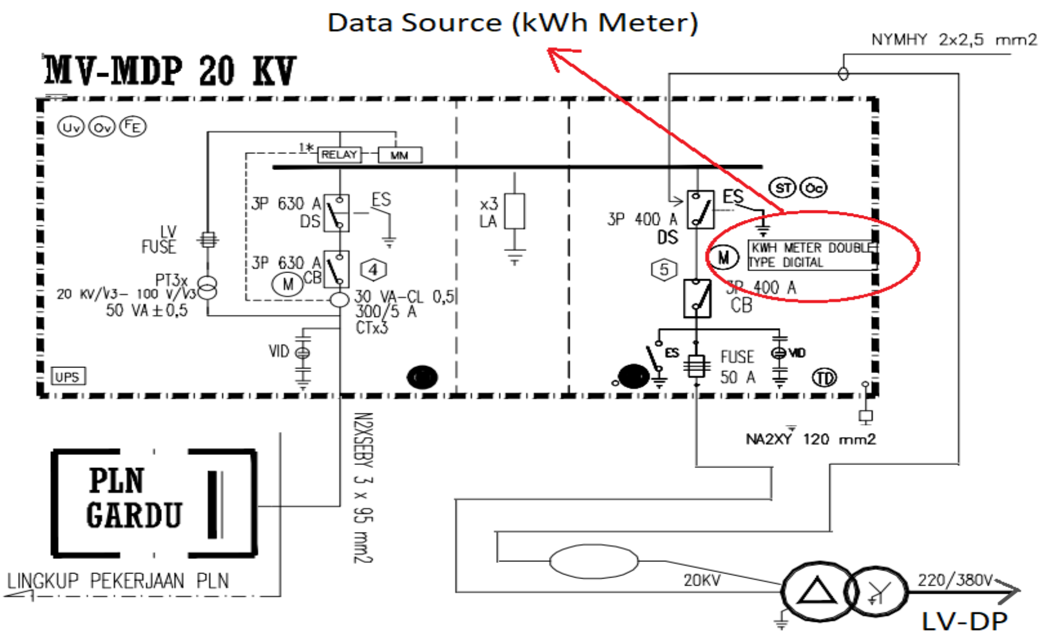

Fig. 8. kWh meter position for data acquisition

Table 4. Actual data

Variable	Date Time					
	01/01/2022	02/01/2022	03/01/2022	Date	06/12/2022	07/12/2022
wbp	0.72	0.74	0.69	…	0.78	0.78
lwbp	3.06	3.06	3.06	…	3.51	3.46
kvh	1.02	1.02	1.06	…	1.07	0.98
v1	58.126	58.567	58.226	…	58.56	58.59
v2	58.125	58.873	58.527	…	58.867	58.82
v3	58.352	58.97	58.62	…	58.965	58.87
I1	0.8272	10.217	0.9044	…	10.286	1.141
I2	0.8327	0.8952	0.7716	…	0.8721	0.993
I3	0.8086	0.9021	0.8965	…	0.9272	1.026
I1h	14.072	60.563	52.556	…	98.296	9.351
I2h	14.54	67.802	60.816	…	10.783	94.177

2.10. Preprocessing Data

From the data source Table 4, there are 12 variables, each of which will be assigned a new identity. Out of these 12 variables, 3 data variables will serve as the main features of interest in this prediction. They are energy consumption during peak load with the identity 'wbp', energy consumption outside peak hours with the identity 'lwbp', and reactive energy consumption with the identity 'kvh'. The other nine data variables function as features to predict the target variables. After ensuring the data segregation, further research steps will be conducted for analysis.

For the GRU model training process, Table 4 data is converted into time series data. Converting time series data into a single-column feature format involves transforming it into three dimensions (number of samples, number of time steps, and number of features). Table 5 shows the results from set features for one input out of 12 inputs. Input prediction at peak load time (wbp), off-peak load time (lwbp) and active power (kVh).

Table 5. Feature input

Time step	Series Value										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
01/01/2022	0.72	3.06	1.02	58.126	58.125	58.352	0.8272	0.8327	0.8086	14.072	14.54
02/01/2022	0.74	3.06	1.02	58.567	58.873	58.97	10.217	0.8952	0.9021	60.563	67.802
03/01/2022	0.69	3.06	1.06	58.226	58.527	58.62	0.9044	0.7716	0.8965	52.556	60.816
…	…	…	…	…	…	…	…	…	…	…	…
06/12/2022	0.78	3.51	1.07	58.56	58.867	58.965	10.286	0.8721	0.9272	98.296	10.783
07/12/2022	0.78	3.46	0.98	58.59	58.82	58.87	1.141	0.993	1.026	9.351	94.177

2.10.1. Setting up Data Subset for Prediction Model

The subset used consists of available columns, 'lwbp', 'kvh', 'v1', 'v2', 'v3', 'I1', 'I2', 'I3', 'I1h', 'I2h', 'I3h'. The values from the dataset are used to convert the Pandas data frame into a numpy array, resulting in variable 'X' containing a numpy array containing the values of the selected features, which will be used as input (independent variables) for the prediction model [23].

Features are observed data used as input for the model (X1,X2,X3).The target variables (Y1,Y2,Y3) or dependent variables that we want to predict are 'wbp', 'lwbp', and 'kvh', which are energy consumption during peak load, energy consumption during off-peak hours, and reactive power usage, in the follow.

X1 = dataset[['lwbp', 'kVh', 'v1', 'v2', 'v3', 'I1', 'I2', 'I3', 'I1h', 'I2h', 'I3h']].values

Y1 = dataset['wbp'].values

X2 = dataset[['lwbp', 'kVh', 'v1', 'v2', 'v3', 'I1', 'I2', 'I3', 'I1h', 'I2h', 'I3h']].values

Y2 = dataset['lwbp'].values

X3 = dataset[['lwbp', 'kVh', 'v1', 'v2', 'v3', 'I1', 'I2', 'I3', 'I1h', 'I2h', 'I3h']].values

Y3 = dataset['kVh'].values

Energy consumption during peak load, off-peak load and reactive power consumption have unique data phenomena and are seasonal and fluctuating so they are used as prediction targets, while voltage tends not to have large fluctuations.

2.10.2. Modeling Data for GRU Model

Converting time series data into a single-column feature format involves transforming it into three dimensions (number of samples, number of time steps, and number of features). It is as follows in Table 3 result from the set features one input out of 12 inputs. Input prediction at peak load time (wbp), off-peak load time (lwbp), and active power (kVh). A flowchart of algorithm implementation in the program and a machine learning pipeline GRU model for time series analysis and prediction is presented in Fig. 9.

According to Fig. 9, the algorithm steps is as follows: the algorithm loads a time series dataset, preprocesses it, and visualizes the data. It then splits the dataset for training and testing, trains linear regression and Random Forest models GridSearch initialization, and normalizes the data. Additionally, it trains a GRU and LSTM model for time series prediction, evaluates model performance, and visualizes predictions. By stopping training when performance on the validation dataset begins to decline, early stopping helps prevent the GRU model from overfitting the GRL LSTM Bidirect neural network model. The final part of Fig. 9 is an evaluation of the performance of the model that has been built, using MAE and RMSE, namely to see the average mean absolute error and how well the model is affected by large errors. The actual data whose features have been adjusted is normalized using the min max formula, this is because the dataset processed is positive numeric data so the range used is 0 to 1.

2.11. Handling Missing and Outlier Data

The handle outliers in this study, data normalization is utilized as a technique to correct scale differences among features, ensuring more consistent and effective utilization of the model. This step not only enhances performance but also directly assists in managing outliers that may affect the analysis outcomes [56].

In this research, the initialization is done with grid search, which utilizes the Random Forest algorithm. Random Forest excels at handling missing data due to its non-parametric nature. It employs ensemble learning to prevent overfitting and maintains performance even with missing data. Additionally, Random Forest effectively handles non-linear relationships and variable interactions, enabling reliable imputation of missing values. The missForest method iteratively imputes missing values using Random Forest models until convergence. Unlike traditional methods, Random Forest does not rely on parametric assumptions, providing flexibility in handling missing data [57].

2.12. Contribution and Research Method Conclusion

Conclusions of this research method begin with a literature review. Then the model design is adjusted to the data on the characteristics of high-rise buildings. Improved performance using the GridSearch model. The data obtained is a collection of relevant data. Ensure generalization of the model with cross-validation (Testing with test data) and three different target data. Performance metrics with MAE and RMSE and comparison with LSTM. Analysis of how the implications of the best model built answer the goal of achieving energy consumption efficiency and contributing to reducing the effects of greenhouse gases in cities. Based on the methodological description above, this research has the following contributions and a novelty in research for neural network for reactive power in high-rise building.

- Technology Contribution: This research contributes to energy management technology in high-rise buildings. By using the GRU prediction results from GridSearch initialization, the results of electricity operational planning are more certain. So that the resulting planning is useful in operating the electricity system efficiently.
- Contribution to science: Utilizing the GRU model and GridSearch initialization contributes to combining two scientific methods which makes it easier to obtain new datasets of prediction results using computer equipment that is not too high-spec.

3. Results and Discussion

After ensuring that the data exhibits seasonality and fluctuations at each time stamp, making it suitable for training and testing according to the methodology's steps, training and testing were conducted on the acquired data using GRU and LSTM models. The testing results on 20% of the acquired data are represented by evaluation metrics such as MAE and RMSE. Additionally, visual representations of the prediction outcomes for each variable are depicted in Fig. 14, Fig. 15, Fig. 16, Fig. 17.

3.1. Authors and Affiliations

All the data to be trained is visualized in a time series with monthly timestamps, and the results can be seen in plot form. Training material data on energy consumption during peak loads is shown in Fig. 10. Fig. 10 shows the data describe daily fluctuations in peak electricity usage, which reflects energy consumption patterns that are influenced by human activities, weather, and habits. This data fluctuation shows that this data can be used as training and testing data for the GRU model as a method for recognizing the use of data patterns. Fig. 11 shows the data to be trained visualized in a time series with monthly timestamps of energy consumption at off-peak load times. Based on this visualization, off-peak load time data can be trained with the GRU model in order to produce accurate predictions.

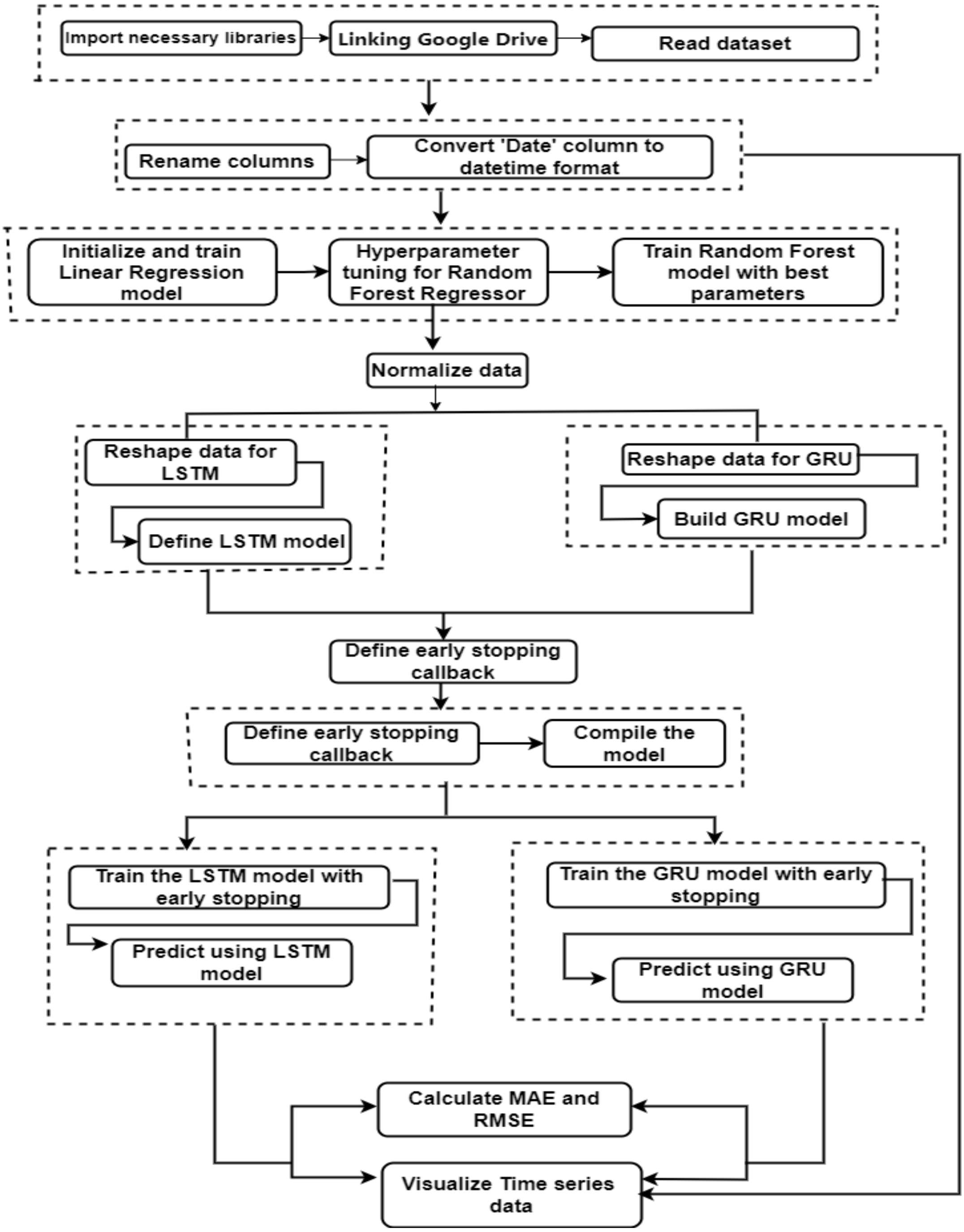

Fig. 9. Pipeline for time series analysis and prediction

Fig. 12 shows the results of the visualization of reactive power consumption data, which will be used for training and testing. In the plot, it looks like there is outlier data, but the data is still on a reasonable scale, namely between 0.5 and 2 kVAR (kilo Volt Ampere Reactive), so it can still be used as prediction material.

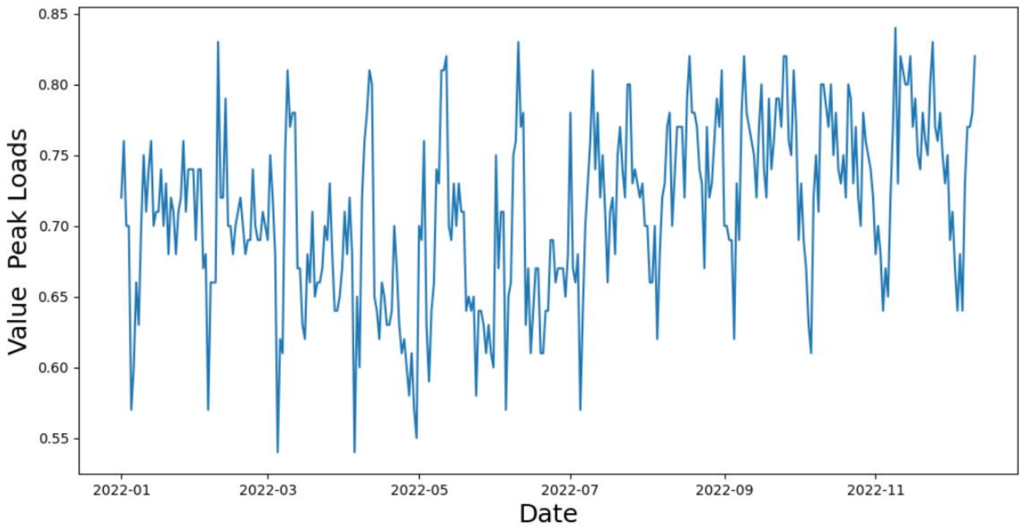

Fig. 10. Visualization of energy consumption during peak load

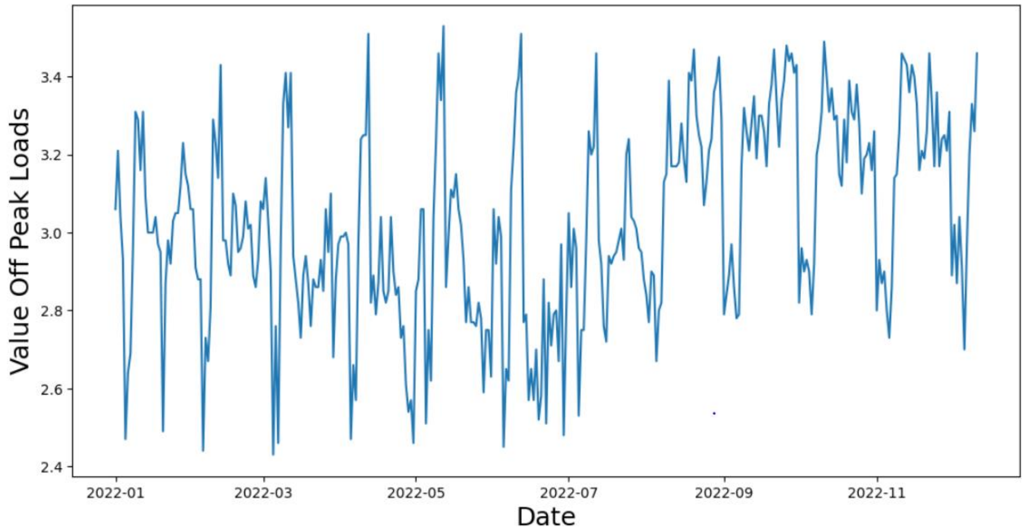

Fig. 11. Visualization of energy consumption during Off-peak load

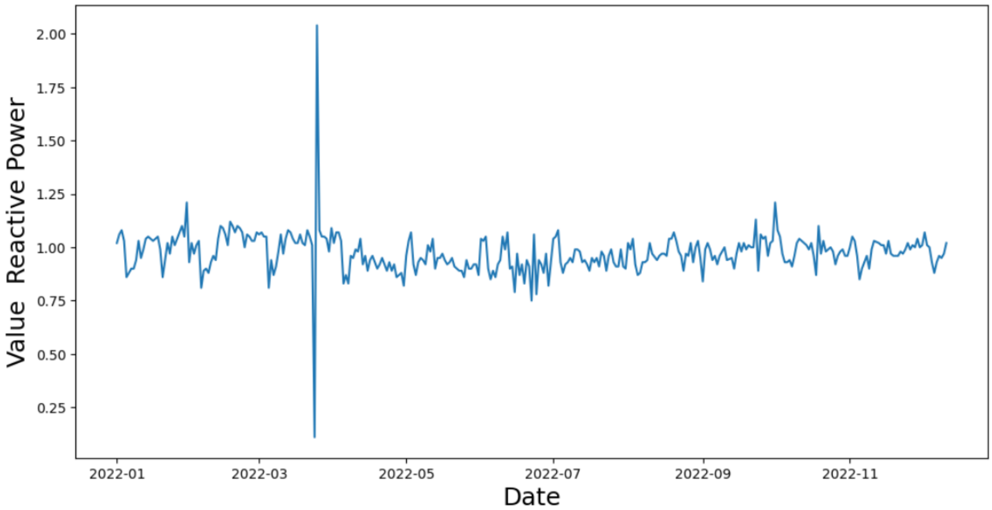

Fig. 12. Energy consumption reactive power data visualization

3.2. Input-Output Relationship to Obtain Targets

Three datasets have different input and output characteristics, yet all share the same 11 features in the overall dataset.

- The first set of input, X1, consists of features such as 'lwbp', 'kvh', 'v1', 'v2', 'v3', 'I1', 'I2', 'I3', 'I1h', 'I2h', and 'I3h'. These features represent variables that may influence energy consumption during peak load. The corresponding target variable (Y1) is 'wbp', which represents energy consumption during peak load data. Therefore, the relationship between input and output is that the model will use these features to data predict energy consumption during peak load.
- The second dataset of input i.e. X2, is identical to X1, but the target variable (Y2) is 'lwbp' represents energy data consumption during off-peak hours. Therefore, the relationship between input and output will using model of these features to data predict energy consumption off-peak load
- The second dataset of input i.e. X3, is identical to X1 and X2, but the target variable (Y3) is 'kvh' represents data reactive power consumption. Therefore, the relationship between input and output will using model of these features to data predict energy consumption reactive power.

The relationship between inputs (features) and outputs (target variables) in this case is that GRU and LSTM models use observed to predict target variables. Model performance is measured using MAE and RMSE metrics, which provide an idea of how accurate the model predictions are in comparing the predicted values with the actual values of the target variables.

3.3. Result in Visualization and Performance Models

After designing the research model, this study delves into utilizing data to train the GRU and LSTM models, subsequently evaluating the generalization capacity of each model with previously unseen data (20% of the dataset). The prediction targets include peak load, off-peak load, and reactive energy data. Each outcome is depicted through a graph that compares predicted results with actual data, while the performance of each prediction is assessed using the MAE and RMSE metrics.

A graph of test results of the GRU prediction model with data that the model has never seen before (20% of the actual data) for the energy consumption variable during off-peak loads is presented in Fig. 13. The x-axis in this figure displays timestamp data per 10 days. In line with the graph in Fig. 14 GRU model test performance metrics produce MAE values of 0.00205, and RMSE: 0.00316. It can be seen that GRU's predicted data and actual data is coincide at all time stamps indicate that GRU's predictions for energy consumption data during off-peak loads is accurate.

The results of predicting energy use at peak load times using the GRU model tested on 20% of the data produce an MAE of 0.00205 and an RMSE of 0.00270. A graphical representation of this condition, along with the time stamp, is depicted in Fig. 15. This figure shows the prediction results reshaped to 10 days on test data showing that the time stamp of the prediction results coincides with the real data even though it does not yet coincide 100 percent.

The energy consumption during peak load is also predicted using the LSTM model for comparison. Fig. 16 depicts the energy usage graph during peak load generated by LSTM, resulting in an MAE of 0.00471 and RMSE of 0.00654. The prediction results between GRU and LSTM for energy consumption during peak load as depicted in Fig. 15 and Fig. 16 indicate that GRU still outperforms LSTM.

The prediction result of GRU for reactive power values represented in MAE of 0.00292 and RMSE of 0.00842. Fig. 17 is the prediction result of GRU compared to actual data. Fig. 18 is the prediction result of LSTM compare to actual data. It indicate that the two graphs show that GRU is more resistant to outlier data compared to LSTM that is timestamp position 54 of the test data.

The prediction results using the LSTM model for the reactive power consumption variable yield an MAE of 0.00325 and RMSE of 0.00898, with fluctuations between the actual data and the prediction results shown in Fig. 18. From this figure, it can be observed that at timestamps 50 and 60, the prediction errors are quite significant, resulting in a larger RMSE for the LSTM model compared to the GRU model, and ultimately leading to a larger MAE as well. In this case, the GRU model performs better in predicting the reactive power consumption in the building compared to LSTM.

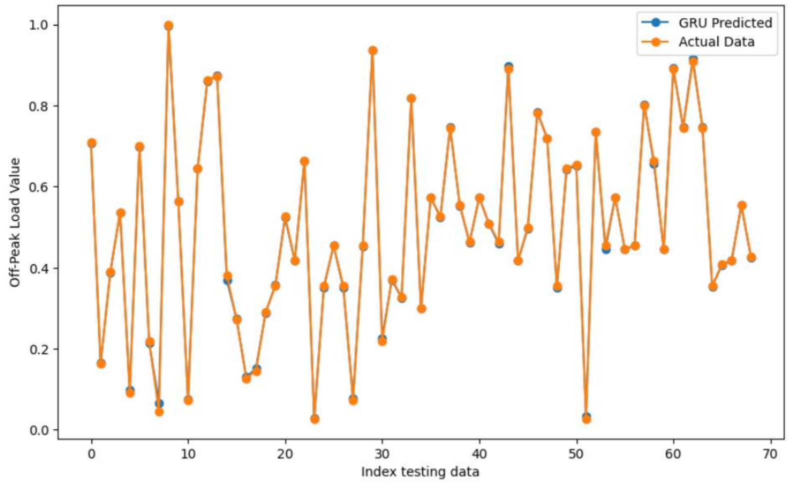

Fig. 13. Comparison of actual off-peak load data and GRU predicted

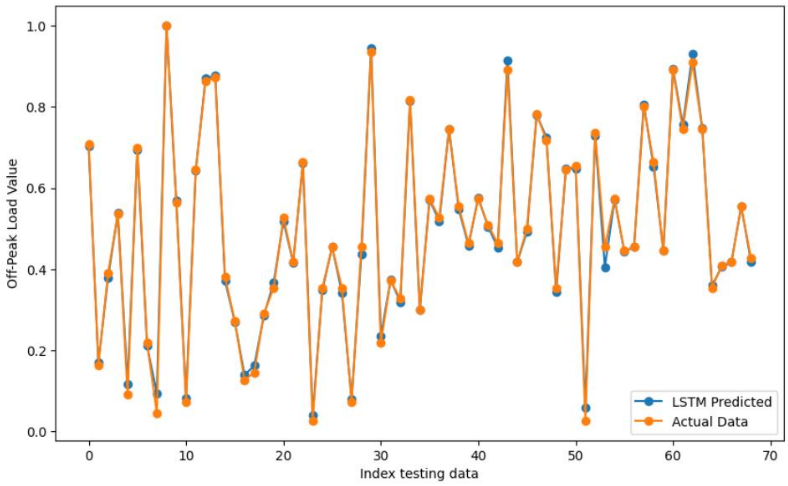

Fig. 14. Comparison of actual Off-peak load data and LSTM predicted

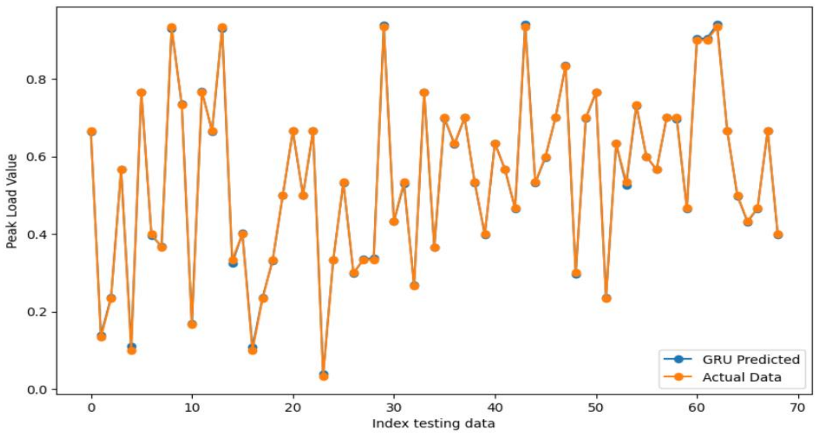

Fig. 15. Comparison of actual peak load and GRU predicted

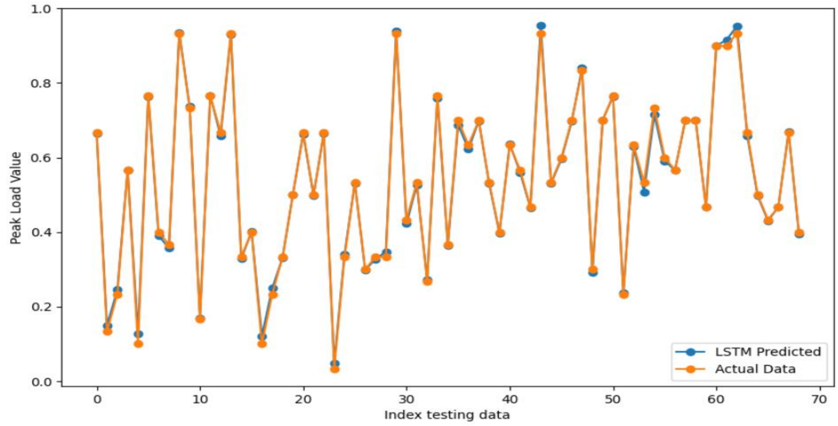

Fig. 16. Comparison of actual peak load data and LSTM predicted

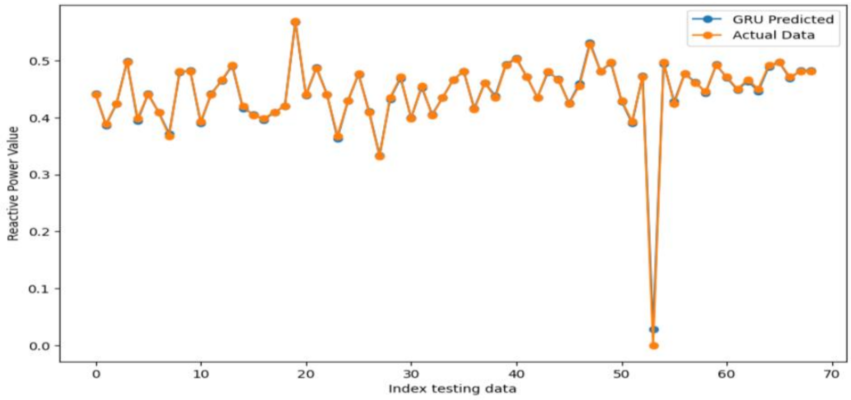

Fig. 17. Comparison of actual Reactive Energy data and GRU predicted

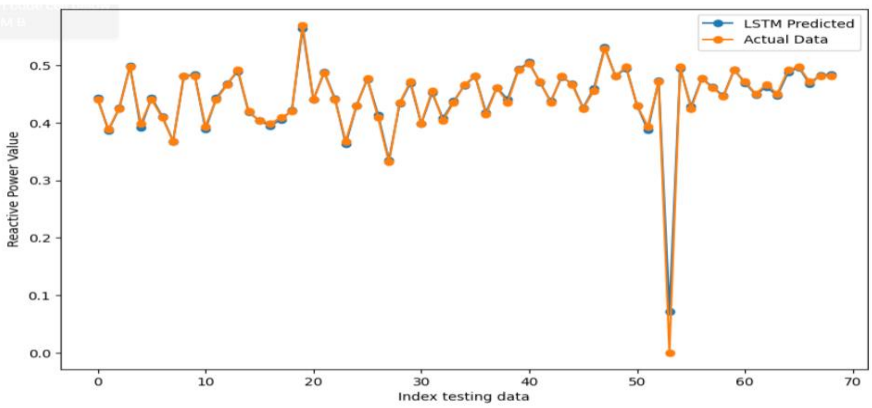

Fig. 18. Comparison of actual reactive Energy and LSTM predicted

3.4. Analysis Performance of the Proposed Model

The analysis of the Gate Recurrent Unit (GRU) Artificial Neural Network prediction model using GridSearch indicates its successful prediction and robust generalization on 20% of the dataset relocated for testing, Covers data variables such as peak load, off-peak load, and reactive power. The GRU model's predictions are accurate over the LSTM model. Table 6 shows a comparison of the performance metrics of the GRU and LSTM models for energy consumption data variables during peak load, off-peak time and reactive power consumption.

The prediction results for reactive power consumption, peak and off-peak energy consumption variables using both GRU and LSTM models demonstrate trends and patterns consistent with the actual data, including outliers. Visualization and evaluation metrics such as MAE and RMSE show that GRU outperforms LSTM for all data variables. The handling of outlier data is addressed by normalizing the data before training and testing, and GRU possesses inherent capabilities in dealing with outlier data. Based on previous research analysis that the GRU model has processing and sequential capabilities, the results of previous research analysis are shown in Table 7.

Table 6. Comparison of metric performance between GRU and LSTM across various variables

Variable Data	Model	Performance Metric	
		MAE	RMSE
Energy consumption during peak load	GRU	0.00204	0.00315
	LSTM	0.00315	0.00204
Energy consumption off-peak load	GRU	0.00205	0.00270
	LSTM	0.00471	0.00654
Reactive power consumption	GRU	0.00292	0.00842
	LSTM	0.00325	0.00898

Table 7. Analysis of previous research

Ref.	Optimization	Performance Matrix	GRU Implication
[27]	Random forest	RMSE and MAE GRU better then LSM	Good generalization ability in emotion recognition from sound signals
[28]	Ensemble GRU LSTM	RMSE and MAE	GRU enhancing operational efficiency in industries oil and gas
[22]	Tree dataset	RMSE and MAE	Better generalization then ARIMA, MLP, RNN, LSTM
[29]	GridSearch optimization	Hyperparameter tuning	GRU contributes to capture long-term dependencies
[30]	Ensemble	F1 Score, computation metric	GRU generalization by efficiently learning from short-term dependencies
[18]	Random forest	RMSE and MAE	GRU generalization reliable for batch production historical data is limited
[31]	Hybrid GRU & Bidirect	CRF (Condition Random Field)	The signification incorporated of GRU in optimizing, efficiency, accuracy, and generalization for feature extraction and analysis NLP
[17]	Arrange hyperparameter	architectural internal combination	GRU enhanced performance in recognizing actions in video stream
[16]	MPC Algorithm	The performance compare with other research	The corporate GRU networks enhance control quality and reduce computational compare to LSTM
[32]	Hyperpara meter tuning	MAE, RMSE, MAPE and R2	The GRU reduce the speed of training, the ability to handle long-term dependencies
Present study	GridSearch	MAE, RMSE	GRU prediction results contribute to environmental mitigation and reduction of the greenhouse effect

3.5. Implication Finding

The uniqueness of this research compared to previous research is in dataset aspects. Busari et al. [27] highlighted the generalization ability of GRU in emotion recognition from speech signals. Kamal et al. [28], predict The Baltic Dry Index (BDI) data used as an indicator of global shipping and trade activity data with ensemble GRU. Fileli et al. [29] use the model GRU and Grid search to predict weekly sales data for the last five years of electrical products. GRU is used to capture long-term dependencies, explore the GRU across various data and attempt to address the challenges associated with focusing on a dataset with limited implications. The uniqueness of this study focuses on energy consumption data sets during peak load times, off-peak load time data sets, and reactive power consumption which have never been discussed in detail by previous researchers. This research and other previous researchers explored the GRU model with various methods to generalize the model for different data domains as evidenced by performance metrics. Performance metrics that are often used in neural networks are the MAE and RMSE methods. Before using the predicted neural network the provision of electrical power in high-rise buildings was based on rough monthly estimates, the implication that there often be excesses and shortages of power in the building. This

Vol. 4, No. 2, 2024, pp. 628-654

research finds that the GRU model initiation by Grid search on three datasets (lwbp, wbp, kvh) provides the implication the model has generalised predictions able on different data domains. This discovery has implications that produce estimates of future conditions such as trends and detailed electricity usage data Accurate GRU predictions make it easier to prepare and distribute electricity supplies optimally. The process of optimizing and planning electricity system operations, especially for the use of reactive power for increasing electricity system efficiency and reducing energy costs. GRU's role in predicting reactive power usage helps reduce the impact of low power factors and optimize reactive power compensation.

4. Conclusion

The research shows a better GRU in predicting reactive power consumption than LSTM, with lower MAE and RMSE values for all variables (Table 5). The GRU's predictive abilities have minimal errors and close alignment between predicted and actual data. Additionally, the GRU model, initialized with Grid Search, offers accurate predictions while simplifying implementation by avoiding repetitive training with different hyperparameters caused by the GridSearch process to optimize hyperparameters, thus improving prediction accuracy and faster data processing and does not require a high-spec computer.

The results of this research show that GRU with Grid search initiation can generalize different data domains. The accurate predictions produced by the GRU model significantly influence the decision-making process for building electricity managers in preparing equipment that consumes reactive power, such as air conditioning systems and electric pump machines. This approach mitigates reactive power surges so that they do not exceed the available reactive power by the reactive power compensator (Capacitor Bank). Predicting the use of reactive power in high-rise buildings can help maintain the power factor below the threshold by scheduling large power-absorbing machines so that they do not work simultaneously. This is important to avoid additional operational costs and contribute to mitigating the impact of greenhouse gases due to the low efficiency of the global electricity system.

Planning with accurate predictions will provide direction for energy management in dealing with conditions of increasing load or conversely decreasing capacity due to the building's capabilities and capacity decreasing in function day by day. Prediction models are needed to prepare operational regulations for the use of electrical energy in high-rise buildings. Use of energy has an impact on the global electricity efficiency system because high-rise buildings are the largest users of electrical energy from state electricity companies. So energy use efficiency in high-rise buildings can reduce fuel use in producing electricity, so high-rise buildings have efficiency implications for the energy use system at large and contribute to environmental sustainability through mitigating greenhouse gases, in this case in urban areas.

Developing research in energy management is vital for addressing emerging challenges in urban energy systems. Based on efforts, the integration of renewable energy sources like solar and wind power is crucial for sustainability, along with implementing smart grid technologies to optimize energy distribution and improve the resilience of energy.

The difficulty of data acquisition is an obstacle to developing recurrent neural network models. Data in certain domains, such as building-related data, may not be readily accessible, but there are solutions. Trusted platforms provide datasets that can serve as alternatives for research development in such scenarios.

Author Contribution: All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] J. Yao, “Electricity consumption and temperature: Evidence from satellite data,” *International Monetary Fund*, 2021, https://doi.org/10.2139/ssrn.3799614.

[2] L. Chang, J. Wang, Z. Xiang, H. Liu, “Impact of Green Financing on Carbon Drifts to Mitigate Climate Change: Mediating Role of Energy Efficiency,” *Frontiers in Energy Research*, vol. 9, p. 785588, 2021, https://doi.org/10.3389/fenrg.2021.785588.

[3] A. Singaravelan, C. Ganesh, P. Sridharan, V. Pooja, R. Shanmugasundaram, B. Gunapriya, “An Overview and Advancement of Electricity Peak Load Saving Methods: A Review,” *Advances in Automation, Signal Processing, Instrumentation, and Control: Select Proceedings of i-CASIC 2020*, pp. 945-958, 2021, https://doi.org/10.1007/978-981-15-8221-9_88.

[4] E. N. Mbinkar, D. A. Asoh, J. N. Mungwe, L. Songyuh, E. Lamfu, “Management of Peak Loads in an Emerging Electricity Market,” *Energy Engineering*, vol. 119, no. 6, pp. 2637–2654, 2022, https://doi.org/10.32604/ee.2022.023419.

[5] Z. Zhang, R. Li and F. Li, "A Novel Peer-to-Peer Local Electricity Market for Joint Trading of Energy and Uncertainty," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1205-1215, 2020, https://doi.org/10.1109/TSG.2019.2933574.

[6] H. M. Salman, J. Pasupuleti, A. H. Sabry, “Review on Causes of Power Outages and Their Occurrence: Mitigation Strategies,” *Sustainability*, vol. 15, no. 20, p. 15001, 2023, https://doi.org/10.3390/su152015001.

[7] A. Maheshwari, K. K. Murari, T. Jayaraman, “Peak electricity demand and global warming in the industrial and residential areas of Pune: An extreme value approach,” *arXiv preprint arXiv:1908.08570*, 2019, https://doi.org/10.48550/arXiv.1908.08570.

[8] F. Kyeremeh, Z. Fang, F. Liu, F. Peprah, “Techno-economic analysis of reactive power management in a solar PV microgrid: A case study of Sunyani to Becheam MV feeder, Ghana,” *Energy Reports*, vol. 11, pp. 83-96, 2024, https://doi.org/10.1016/j.egyr.2023.11.031.

[9] Y. Yang, M. Wang, Y. Liu, L. Zhang, “Peak-off-peak load shifting: Are public willing to accept the peak and off-peak time of use electricity price?,” *Journal of Cleaner Production*, vol. 199, pp. 1066-1071, 2018, https://doi.org/10.1016/j.jclepro.2018.06.181.

[10] G. Besagni, L. P. Vilà, M. Borgarello, “Italian household load profiles: A monitoring campaign,” *Buildings*, vol. 10, no. 12, p. 217, 2020, https://doi.org/10.3390/buildings10120217.

[11] H. C. Anderson, A. Al Hadi, E. S. Jones and D. M. Ionel, "Power Factor and Reactive Power in US Residences – Survey and EnergyPlus Modeling," *2021 10th International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 418-422, 2021, https://doi.org/10.1109/ICRERA52334.2021.9598561.

[12] J. Andruszkiewicz, J. Lorenc and A. Weychan, “Determination of the Optimal Level of Reactive Power Compensation That Minimizes the Costs of Losses in Distribution Networks,” *Energies*, vol. 17, no. 1, p. 150, 2024, https://doi.org/10.3390/en17010150.

[13] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020, https://doi.org/10.1016/j.physd.2019.132306.

[14] B. U. Islam, S. F. Ahmed, “Short-Term Electrical Load Demand Forecasting Based on LSTM and RNN Deep Neural Networks,” *Mathematical Problems in Engineering*, vol. 2022, 2022, https://doi.org/10.1155/2022/2316474.

[15] M. Sajjad *et al*., "A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting," *IEEE Access*, vol. 8, pp. 143759-143768, 2020, https://doi.org/10.1109/ACCESS.2020.3009537.

[16] K. Zarzycki, M. Ławryńczuk, “Advanced predictive control for GRU and LSTM networks,” *Information Sciences*, vol. 616, pp. 229–254, 2022, https://doi.org/10.1016/j.ins.2022.10.078.

[17] A. W. Muhamad, A. A. Mohammed, "a Comparative Study Using Improved Lstm /Gru for Human Action Recognition," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 5, pp. 1863–1879, 2023, https://doi.org/10.21203/rs.3.rs-2380406/v1.

[18] X. Li, X. Ma, F. Xiao, F. Wang, S. Zhang, "Application of gated recurrent unit (GRU) neural network for smart batch production prediction," *Energies*, vol. 13, no. 22, p. 6121, 2020, https://doi.org/10.3390/en13226121.

[19] A. A. Cook, G. Mısırlı and Z. Fan, "Anomaly Detection for IoT Time-Series Data: A Survey," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6481-6494, 2020, https://doi.org/10.1109/JIOT.2019.2958185.

[20] X. Wen and W. Li, "Time Series Prediction Based on LSTM-Attention-LSTM Model," *IEEE Access*, vol. 11, pp. 48322-48331, 2023, https://doi.org/10.1109/ACCESS.2023.3276628.

[21] A. Rahman, V. Srikumar, A. D. Smith, "Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks," *Applied Energy*, vol. 212, pp. 372–385, 2018, https://doi.org/10.1016/j.apenergy.2017.12.051.

[22] Z. Zainuddin, P. Akhir E. A., M. H. Hasan, "Predicting machine failure using recurrent neural network-gated recurrent unit (RNN-GRU) through time series data," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 2, pp. 870–878, 2021, https://doi.org/10.11591/eei.v10i2.2036.

[23] E. O. Edoka, V. K. Abanihi, H. E. Amhenrior, E. M. J. Evbogbai, L. O. Bello, V. Oisamoje, "Time Series Forecasting of Electrical Energy Consumption Using Deep Learning Algorithm," *Nigerian Journal of Technological Development*, vol. 20, no. 3, pp. 163–175, 2023, https://doi.org/10.4314/njtd.v20i3.1424.

[24] K. Sayed, H. A. Gabbar, "Building Energy Management Systems (BEMS)," *John Wiley & Sons*, 2017, https://doi.org/10.1002/9781119422099.ch2.

[25] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv preprint arXiv:1412.3555*, 2014, https://doi.org/10.48550/arXiv.1412.3555.

[26] I. K. Nti, M. Teimeh, O. Nyarko-Boateng, A. F. Adekoya, "Electricity load forecasting: a systematic review," *Journal of Electrical Systems and Information Technology*, vol. 7, no. 13, 2020, https://doi.org/10.1186/s43067-020-00021-8.

[27] G. A. Busari, N. W. Kwak, D. H. Lim, "An Application of AdaBoost-GRU Ensemble Model to Economic Time Series Prediction," *Indian Journal of Science and Technology*, vol. 14, no. 31, pp. 2557–2566, 2021, https://doi.org/10.17485/IJST/v14i31.1204.

[28] I. M. Kamal, H. Bae, S. Sunghyun, H. Yun, "DERN: Deep ensemble learning model for shortand long-term prediction of baltic dry index," *Applied Science*, vol. 10, no. 4, p. 1504, 2020, https://doi.org/10.3390/app10041504.

[29] M. R. Haq and Z. Ni, "A New Hybrid Model for Short-Term Electricity Load Forecasting," *IEEE Access*, vol. 7, pp. 125413-125423, 2019, https://doi.org/10.1109/ACCESS.2019.2937222.

[30] M. R. Ahmed, S. Islam, A. K. M. M. Islam, S. Shatabda, "An ensemble 1D-CNN-LSTM-GRU model with data augmentation for speech emotion recognition," *Expert Systems with Applications*, vol. 218, p. 119633, 2023, https://doi.org/10.1016/j.eswa.2023.119633.

[31] D. Wang, J. Su and H. Yu, "Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language," *IEEE Access*, vol. 8, pp. 46335-46345, 2020, https://doi.org/10.1109/ACCESS.2020.2974101.

[32] M. L. Abdulrahman *et al.*, "A Review on Deep Learning with Focus on Deep Recurrent Neural Network for Electricity Forecasting in Residential Building," *Procedia Computer Science*, vol. 193, pp. 141–154, 2021, https://doi.org/10.1016/j.procs.2021.10.014.

[33] G. Xu, B. Zhang, L. Yang, Y. Wang, "Active and Reactive Power Collaborative Optimization for Active Distribution Networks Considering Bi-Directional V2G Behavior," *Sustainability*, vol. 13, no. 11, p. 6489, 2021, https://doi.org/10.3390/su13116489.

[34] A. Yani, J. Junaidi, M. Irwanto, A. H. Haziah, “Optimum reactive power to improve power factor in industry using genetic algortihm,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 2, pp. 751–757, 2019, http://doi.org/10.11591/ijeecs.v14.i2.pp751-757.

[35] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi and J. A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690-6709, 2019, https://doi.org/10.1109/TGRS.2019.2907932.

[36] I. K. Nti, A. Samuel, A. Michael, “Predicting Monthly Electricity Demand Using Soft-Computing Technique,” *International Research Journal of Engineering and Technology*, vol. 06, no. 06, p. 1967–1973, 2019, https://www.irjet.net/archives/V6/i6/IRJET-V6I6428.pdf.

[37] E. P. Lens Shiang, W. -C. Chien, C. -F. Lai and H. -C. Chao, "Gated Recurrent Unit Network-based Cellular Trafile Prediction," *2020 International Conference on Information Networking (ICOIN)*, pp. 471-476, 2020, https://doi.org/10.1109/ICOIN48656.2020.9016439.

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

[39] J. Hwang, D. Suh, M. O. Otto, “Forecasting electricity consumption in commercial buildings using a machine learning approach,” *Energies*, vol. 13, no. 22, p. 5885, 2020, https://doi.org/10.3390/en13225885.

[40] G. Van Houdt, C. Mosquera, G. Nápoles, “A review on the long short-term memory model,” *Artificial Intelligence Review*, vol. 53, pp. 5929–5955, 2020, https://doi.org/10.1007/s10462-020-09838-1.

[41] D. G. D. Silva, A. A. D. M. Meneses, “Comparing Long Short-Term Memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction,” *Energy Reports*, vol. 10, pp. 3315–3334, 2023, https://doi.org/10.1016/j.egyr.2023.09.175.

[42] A. Gasparin, S. Lukovic, C. Alippi, “Deep learning for time series forecasting: The electric load case,” *CAAI Transactions on Intelligence Technology*, vol. 7, no. 1, pp. 1–25, 2022, https://doi.org/10.1049/cit2.12060.

[43] D. M. Brandão Lent, M. P. Novaes, L. F. Carvalho, J. Lloret, J. J. P. C. Rodrigues and M. L. Proença, "A Gated Recurrent Unit Deep Learning Model to Detect and Mitigate Distributed Denial of Service and Portscan Attacks," *IEEE Access*, vol. 10, pp. 73229-73242, 2022, https://doi.org/10.1109/ACCESS.2022.3190008.

[44] U. S. Shanthamallu, A. Spanias, “Neural Networks and Deep Learning,” *Synthesis Lectures on Signal Processing*, pp. 43-57, 2022, https://doi.org/10.1007/978-3-031-03758-0_5.

[45] M. Faiq *et al*., “Prediction of energy consumption in campus buildings using long short-term memory,” *Alexandria Engineering Journal*, vol. 67, pp. 65-76, 2023, https://doi.org/10.1016/j.aej.2022.12.015.

[46] J. Lederer, “Activation Functions in Artificial Neural Networks: A Systematic Overview,” *arXiv preprint arXiv:2101.09957*, 2021, https://doi.org/10.48550/arXiv.2101.09957.

[47] D. Chicco, M. J. Warrens, G. Jurman, “The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation,” *PeerJ Computer Science*, vol. 7, p. e623, 2021, https://doi.org/10.7717/peerj-cs.623.

[48] H. Dhake, Y. Kashyap, P. Kosmopoulos, “Algorithms for Hyperparameter Tuning of LSTMs for Time Series Forecasting,” *Remote Sensing*, vol. 15, no. 8, p. 2076, 2023, https://doi.org/10.3390/rs15082076.

[49] D. Doshi, T. He, A. Gromov, “Critical Initialization of Wide and Deep Neural Networks through Partial Jacobians: General Theory and Applications,” *arXiv preprint arXiv:2111.12143*, 2021, https://doi.org/10.48550/arXiv.2111.12143.

[50] R. J. Jesus, M. L. Antunes, R. A. D. Costa, S. N. Dorogovtsev, J. F. F. Mendes, R. L. Aguiar, “Effect of initial configuration of weights on training and function of artificial neural networks,” *Mathematics*, vol. 9, no. 18, p. 2246, 2021, https://doi.org/10.3390/math9182246.

[51] D. M. Belete, M. D. Huchaiah, "Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results," *International Journal of Computers and Applications*, vol. 44, no. 9, pp. 875–886, 2022, https://doi.org/10.1080/1206212X.2021.1974663.

[52] C. G. Siji George, B. Sumathi, "Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, pp. 173–178, 2020, https://doi.org/10.14569/IJACSA.2020.0110920.

[53] A. Lambora, K. Gupta and K. Chopra, "Genetic Algorithm- A Literature Review," *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pp. 380-384, 2019, https://doi.org/10.1109/COMITCon.2019.8862255.

[54] S. Bouktif, A. Fiaz, A. Ouni, M. A. Serhani, "Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches," *Energies*, vol. 11, no. 7, p. 1636, 2018, https://doi.org/10.3390/en11071636.

[55] S. Sridevi, G. R. Karpagam, "Genetic algorithm-optimized gated recurrent unit (GRU) network for semantic web services classification," *Malaysian Journal of Computer Science*, vol. 35, no. 1, pp. 70-88, 2022, https://doi.org/10.22452/mjcs.vol35no1.5.

[56] R. Feng, W. Jiang, N. Yu, Y. Wu and J. Yan, "Projected Minimal Gated Recurrent Unit for Speech Recognition," *IEEE Access*, vol. 8, pp. 215192-215201, 2020, https://doi.org/10.1109/ACCESS.2020.3041477.

[57] S. Hong, H. S. Lynn, "Accuracy of random-forest-based imputation of missing data in the presence of non-normality, non-linearity, and interaction," *BMC Medical Research Methodology*, vol. 20, no. 19, pp. 1–12, 2020, https://doi.org/10.1186/s12874-020-01080-1.