

Article

A Deep Learning Approach for Short-Term Electricity Demand Forecasting: Analysis of Thailand Data

Ranju Kumari Shiwakoti ¹, Chalie Charoenlarnnopparut ^{1,*}  and Kamal Chapagain ² 

¹ School of Information, Computer, and Communication, Technology Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12120, Thailand; d6422300076@g.siiit.tu.ac.th

² Department of Electrical and Electronics Engineering, Kathmandu University, Dhulikhel P.O. Box 6250, Nepal; kamal.chapagain@ku.edu.np

* Correspondence: chalie@siit.tu.ac.th

Abstract: Accurate electricity demand forecasting serves as a vital planning tool, enhancing the reliability of management decisions. Apart from that, achieving these aims, particularly in managing peak demand, faces challenges due to the industry's volatility and the ongoing increase in residential energy use. Our research suggests that employing deep learning algorithms, such as recurrent neural networks (RNN), long short-term memory (LSTM), and gated recurrent units (GRU), holds promise for the accurate forecasting of electrical energy demand in time series data. This paper presents the construction and testing of three deep learning models across three separate scenarios. Scenario 1 involves utilizing data from all-day demand. In Scenario 2, only weekday data are considered. Scenario 3 uses data from non-working days (Saturdays, Sundays, and holidays). The models underwent training and testing across a wide range of alternative hyperparameters to determine the optimal configuration. The proposed model's validation involved utilizing a dataset comprising half-hourly electrical energy demand data spanning seven years from the Electricity Generating Authority of Thailand (EGAT). In terms of model performance, we determined that the RNN-GRU model performed better when the dataset was substantial, especially in scenarios 1 and 2. On the other hand, the RNN-LSTM model is excellent in Scenario 3. Specifically, the RNN-GRU model achieved an MAE (mean absolute error) of 214.79 MW and an MAPE (mean absolute percentage error) of 2.08% for Scenario 1, and an MAE of 181.63 MW and MAPE of 1.89% for Scenario 2. Conversely, the RNN-LSTM model obtained an MAE of 226.76 MW and an MAPE of 2.13% for Scenario 3. Furthermore, given the expanded dataset in Scenario 3, we can anticipate even higher precision in the results.

Keywords: deep learning; gated recurrent unit; hyperparameter tuning; long short-term memory; recurrent neural network; short-term demand forecasting



Citation: Shiwakoti, R.K.; Charoenlarnnopparut, C.; Chapagain, K. A Deep Learning Approach for Short-Term Electricity Demand Forecasting: Analysis of Thailand Data. *Appl. Sci.* **2024**, *14*, 3971. <https://doi.org/10.3390/app14103971>

Academic Editors: Antonio Gabaldón, María Carmen Ruiz-Abellón and Luis Alfredo Fernández-Jiménez

Received: 28 March 2024

Revised: 1 May 2024

Accepted: 1 May 2024

Published: 7 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background

The electricity consumption within a society serves as a significant indicator of its sustainable development, encompassing both economic and environmental aspects. Consequently, ongoing research aimed at addressing electricity consumption issues is crucial for supporting various domains [1]. Accurate forecasting is a critical need in electricity markets. Additionally, it plays a significant role in ensuring the security of the power system [2]. There are typically three types of electricity load forecasting based on the period of forecasting; these are short-term load forecasting (covering minutes to one day ahead), medium-term load forecasting (spanning weeks to months), and long-term load forecasting (projecting years ahead) [3]. Short-term load forecasting (STLF) is useful for utilities in predicting energy demand during peak hours and preparing for potential supply shortages. Additionally, it assists electricity market participants in managing energy trading activities

and mitigating risks arising from sudden fluctuations in demand or supply [4]. It was observed that 80% of the electrical energy demand forecasting concerned STLF, while MTLF and LTLF accounted for 15% and 5%, respectively. STLF has the highest priority because of its critical role in the daily operational scheduling of power systems, guaranteeing grid stability and maintaining supply and demand in real time. While MTLF and LTLF are critical for strategic planning and infrastructure development, they may not be as urgent in addressing immediate operational challenges, which justifies the more limited research focus [5]. Thailand's electrical supply comprises three main entities: the Metropolitan Electricity Authority (MEA), the Electricity Generating Authority of Thailand (EGAT), and the Provincial Electricity Authority (PEA), where EGAT manages power generation. Our paper focuses specifically on the metropolitan regions within Thailand [2].

1.2. Challenges

Short-term load forecasting plays a vital role in the energy industry, as precise predictions of future electricity demand are essential for maintaining the reliable and efficient operation of power systems [6]. Weather conditions, economic trends, consumer behavior, and unforeseen events such as natural disasters all have an impact on load data, leading to fluctuations that make accurate forecasting difficult, especially over long periods [7]. Deep learning models can independently learn detailed patterns and relationships while adjusting to variations in the data-generating process. However, the training and deployment of these models demand substantial amounts of data and computational resources [8]. Energy systems are constantly evolving, influenced by policy changes, technological advancements, and market fluctuations. Researchers are consistently attempting to enhance existing forecasting models to mitigate the occurrences of both under and over forecasting, depending upon the effects of these influential parameters [9]. In this context, researchers face significant challenges in minimizing prediction errors. Their task involves developing sophisticated models and thoroughly testing them across diverse datasets, including commercial, residential, and collective loads [10].

1.3. Load Forecasting Models

Conventional statistical models include regression models, such as linear regression models, multiple regression models, stepwise regression models, logistic regression models, and polynomial regression models [11], where moving average (MA), autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), autoregressive integrative moving average exogenous (ARIMAX), and seasonal autoregressive integrated moving average (SARIMA) are popular in statistical forecasting models [12]. For non-real-time forecasting, these models are useful, but they have no accuracy with respect to actual time load and no ability to regulate the consumption of non-linear loads [13].

Short-term electrical demand forecasting powered by AI is crucial for optimizing energy generation and distribution, allocating resources effectively and maintaining power network stability [14]. AI-based models, including traditional machine learning (ML) and deep learning (DL) models, were introduced to handle non-linear relationships in load consumption. Traditional machine learning methods, such as support vector machines (SVM), random forests (RF), and decision trees (DT), can capture complex correlations between several factors affecting energy consumption. ML models consist of artificial neural networks (ANN), particle swarm optimization algorithms (PSO), genetic algorithms (GA), fuzzy logic, and expert systems [15]. Feed-forward neural networks (FNNs) simulate complex relationships between inputs and outputs by adjusting parameters. However, they can suffer from overfitting and may not always find the best solution, becoming stuck in a local optimal state. To overcome this, backward propagation learning is often used in FNN training [8]. To overcome the complex ML problems, DL models were introduced, such as the deep neural network (DNN), the recurrent neural network (RNN), long short-term memory (LSTM), the convolutional neural network (CNN), gated recurrent units (GRU), and the deep belief network (DBN), which have been used for STLF [16]. A modification of

RNNs, the LSTM network was created by Hochreiter and Schmidhuber in 1997 to address the vanishing and exploding difficulties in RNNs [17]; it solves time series data problems by memorizing all input information. GRU were introduced to address vanishing and exploding gradients using the concepts of gated cells, like in LSTM, which exists in RNN due to long-term dependencies. So, this is the model for improvement based on RNNs.

This research is related to short-term electrical demand forecasting by using a half-hourly recorded demand dataset from Thailand. There is a lot of previous work related to demand data from EGAT, applying different models by using either all or part of a 5-year period (from 1 March 2009 to 31 December 2013), which is mentioned in table in Section 2. We used a dataset from EGAT, covering about 7 years (from 1 January 2012 to 31 December 2018), with the same half-hourly pattern of demand. Identifying the optimal DNN model based on forecasting accuracy is the primary goal of this work. This paper makes the following significant contributions:

- Based on testing and validation accuracy, a comparative analysis of deep neural networks for simple RNN, LSTM, and GRU is discussed.
- Splitting the complete 7-year demand dataset into three distinct scenarios, encompassing working days, non-working days, and the entirety of the dataset, based on shared demand characteristics, enhances prediction accuracy compared to prior studies using Thai datasets.
- Combining hyperparameter tuning (dropout, epoch, lookback period, number of neurons per layer, number of layers, etc.) to determine which model performs best.
- This paper addresses a gap in the literature by utilizing deep learning techniques to model the EGAT dataset spanning from 2012 to 2018. Previous studies have tested the EGAT dataset from 2009 to 2013 using various models, as outlined in table in Section 2.

The remainder of this paper unfolds as follows: Section 2 explores related literature; our modeling approach, including the principles supporting DNNs and the methodology for model estimation, is described in Section 3; Section 4 states the features of electricity demand data and examines pertinent variables; Section 5 determines the model framework, details the expansive experimental setup, and closely examines the forecast accuracy and model fit quality using the Thai dataset; after this, Section 6 furnishes the findings along with comprehensive discussions, while Section 7 as the concluding section.

2. Related Works

The nature of the data, the accessibility of previous observations, and the forecasting goal have an impact on the decision to use a time series model. Building efficient time series models requires careful consideration and analysis of trends and seasonality, feature engineering, and model selection [15]. Kernel-based multitask learning has been shown to be effective in addressing electricity demand forecasting difficulties. Fiot et al. developed kernels that consider various seasonal patterns to specifically address the complicated task of forecasting individual customer demand [18]. Dudek et al. presented the principles of the pattern-similarity-based methods for short-term load forecasting and similar-pattern-based local linear regression models using Polish power system data [19]. The proposed stepwise and lasso regression methods have been suggested to decrease the number of predictors and have shown superior performance when compared to other models such as ARIMA, ES, ANN, and the Nadaraya–Watson estimator. Ismail et al. [20] used an MLR model with a day-ahead forecast MAPE of 1.71% to examine the effects of temperature and holiday types. In addition to artificial intelligence, linear regression was contrasted with other conventional and adaptable techniques for selecting weather variables and identical energy prediction, such as seasonal ARIMA (SARIMA) and regression ARIMA (RegARIMA) [20]. Lusi et al. evaluated calendar effects and forecast granularity of STL using regression trees, neural networks, and SVR with root mean square error (RMSE) and normalized RMSE as forecast error metrics [10]. Zhang et al. [21] proposed improved adaptive rules for the genetic algorithm, which demonstrated superior performance compared to the traditional genetic algorithm (GA) in optimizing support vector regression (SVR) and enhancing

forecasting accuracy for hourly electricity demand. Additionally, it outperformed the extreme learning machine (ELM) model and various artificial neural network models [21]. Certain scholars have utilized machine learning models to improve their performance by enhancing training models, iterations, hyperparameters tuning, and other different techniques. Khwaja et al. proposed the boosted neural network (BooNN) model, which combines a set of ANNs iteratively, and then compared it with simple ANN and bagged neural networks (BagNN) [22]. Similarly, they proposed the joint bagged-boosted ANNs that train an ensemble of multiple ANN models in parallel, creating individual ANNs and combining the individual ANNs. Then, the generated results of the proposed model were compared with those of single ANN, BoostNN, and BagNN [23].

The attraction of the RNN-based LSTM and GRU networks among researchers is due to their capacity to manage sequential data and dependencies over time and to discover complicated patterns in the data [24,25]. Recent improvements in computation and algorithms have resulted in the development of DNN as the primary approach for forecasting electricity demand. This is due to their ability to enhance the feature abstraction capability of the model. Chen et al. presented a modified deep residual network with a two-stage ensemble strategy to enhance the generalization capability [26]. When compared to other models, deep learning (DL) experiments possess outstanding abilities in managing complex non-linear relationships, model complexity, and computational efficiency. Din et al. investigate and evaluate the effectiveness of feed-forward DNN and recurrent deep neural network (R-DNN) models in terms of accuracy and computational efficiency, and the generated results show that the use of transfer function feature analysis with DNN enables higher accuracy [27]. Li et al. [28] assessed the efficacy of LSTM and FNN models for electricity demand prediction, scrutinizing their preciseness and resilience. Their work underscores LSTM's expertise in capturing complicated long-term patterns in electricity demand data [28]. Hybrid DNN models have been used in electricity demand prediction in addition to LSTM and FNN. To improve the reliability of short-term demand for electricity forecasting, Stosovic et al. recommended five distinct recurrent neural network architectures. In terms of accuracy, this study revealed that the GRU and bidirectional LSTM model surpassed the conventional FNN and RNN models, indicating the possibility of combining several machine-learning techniques for enhanced forecasting performance [29]. Xiao et al. used a hybrid forecasting model for electrical power prediction that incorporates ANN, including BPNN, the generalized regression neural network (GRNN), the Elman neural network, and an optimized model (GA-BP) neural network, in to half-hourly electrical power data of the State of Victoria and New South Wales of Australia [30]. The ensemble approach has lately gained popularity among researchers because of the improved prediction accuracy that has been demonstrated. For the management of the power system, a survey of demand forecasting models was conducted [31]. Gao et al. [32] introduced an adaptive deep-learning load forecasting framework by integrating transformer and domain knowledge. Adaptive-TgDLF effectively combines the transformer model with human insights to enhance short-term electrical load forecasting. It introduces adaptive learning to tackle data scarcity and changing load patterns, marking a significant advancement in this area of research [32]. The researchers explored model predictive control for adjusting power production and distribution strategies in response to predicted demand, thus preventing delays and optimizing resource use. By precisely predicting short-term load demand, services can enhance grid stability, reduce costs, and boost efficiency [33,34].

The SLTF works for Thailand appear to have only lately surfaced. Hence, numerous studies have been conducted in Thailand to forecast power consumption using various approaches and strategies. Several studies [2,35–39] utilizing data sourced from the Bangkok metropolitan region have been published, recorded by the Electricity Generating Authority of Thailand (EGAT) between 2009 and 2013.

Chapagain and Kittipiyakul implemented calendar elements, including the year, month, day, and hour, as well as seasonal elements like holidays and special days, which were considered to be the factors driving demand in [2], and used them, together with other

crucial variables, to construct numerous predictive models. These elements are typically referred to as deterministic variables. When compared to demand on weekends, weekday demand exhibits dramatically distinct patterns. Due to the existence of industrial sector activity on weekdays, weekday demand is specifically much stronger and more consistent than weekend demand. Weekend and weekday holidays both have significantly lower energy demand than weekday demand. In order that the various weekday patterns would be identified, we needed to create dummy variables that represented each day of a week as either weekdays, weekends, or holidays (whether on weekdays or weekends) [40]. Various researchers have generated noteworthy findings based on the identical EGAT dataset spanning from 2009 to 2013. These findings are summarized in Table 1.

Table 1. List of articles that were published using the EGAT dataset [41].

Dataset	Methods	Results	MAPE	References
2009–2013	OLS	The model’s prediction accuracy has increased with the inclusion of interaction variables.	>4%	[37]
2009–2013	OLS and Bayesian estimation	The forecast accuracy has increased by 20% because of the model’s adding a temperature variable.	2% to 3%	[35]
2009–2013	OLS, GLSAR, FF-ANN	Compared to FF-ANN, OLS, and GLSAR, models exhibited greater forecasting accuracy.	1.74% to 2.95%	[2]
2009–2013	MLR with AR (2)	In comparison to OLS estimation, Bayesian estimation offers better and more reliable results.	1% to 5%	[39]
2009–2013	Ensemble for regression and ML	It reduces the test MAPE when implementing the blocked cross validation method.	2.6%	[41]
2013	PSO with ANN	To train the ANN for STLE, PSO outperformed the backpropagation training technique.	3.44%	[19]
2013	PSO + GA with ANN	The backpropagation and PSO training algorithms are outperformed by PSO + GA.	>3%	[42]
2013	ANN	The model’s prediction accuracy has increased with the addition of a temperature variable.	3.27% to 4.61%	[36]

3. Theoretical Background

This chapter presents the theoretical background of all terminologies and an explanation of the proposed model. Deep learning techniques, such as RNN, RNN-LSTM, and RNN-GRU, are utilized to forecast future short-term electricity demand. Forecasting electrical load presents distinct challenges, including seasonality, periodicity, non-linearity, and sequential interdependence within consumption data sequence, which deep learning architectures are especially well-suited to address. By using non-linear modifications and extracting high-level abstractions, deep learning models, as opposed to shallow ANN architectures, can automatically learn complicated temporal patterns [43].

3.1. Deep Neural Network

The DNN is one of the neural networks with an input layer, an output layer, and many hidden layers (more than two). It includes an input layer (x_i), an output layer (y_i), and more than two hidden layers (h_i), as shown in Figure 1. It trains as a feed-forward network to generate the corresponding output values through all hidden layers and neuron nodes in forward propagation. The non-linear activation function f takes a weighted sum of input X values and returns an output Y value, as indicated in Equation (1), where W_{jk}^l = the weight between node j in the layer $(l-1)$ and node k in layer l .

$$Y_i = f\left(\sum_{i=1}^t W_{jk}^l X_i\right) \quad (1)$$

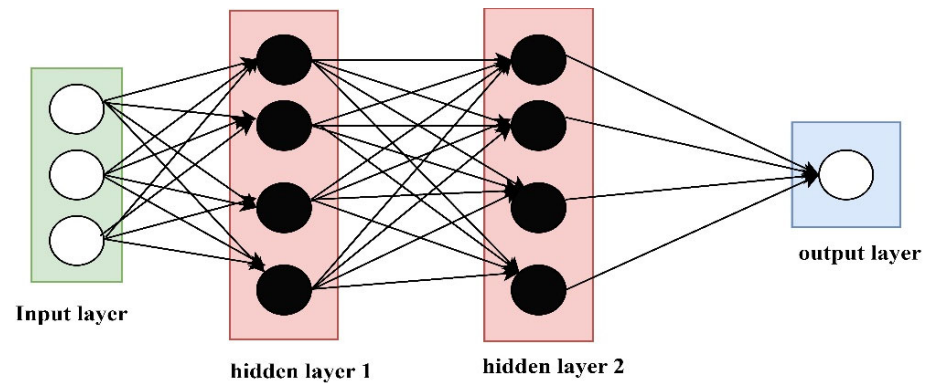


Figure 1. Structure of the deep neural network.

3.2. Recurrent Neural Network

The RNN is defined as an extension of a feed-forward neural network (FNN), with feedback connections that can detect short-term non-linear time relationships. Williams and Zipser created the first RNN in the late 1980s, a time of rapid advancement in neural network construction research that resulted in multiple essential discoveries [44]. RNN computes new states by recursively applying transfer functions to previous states and inputs. This process determines how information is passed between neurons and is critical for creating a learning system that can perform well under different conditions [45].

In terms of prediction, an RNN is trained using input data $x(t)$ to generate a desired output $y(t)$. Figure 2 illustrates the unrolled architecture of an RNN across the t -time dimension with L_N layers. Each time step t features neurons in the hidden layer, which can be viewed as a single layer within an FNN. At time step t , the parameters of the network neurons at the l th layer update their shared states using the following equations [46]:

$$\alpha_1(t) = b_1 + W_1 h_1(t-1) + U_1 x(t), \quad (2)$$

$$h_l(t) = \text{Activation function } \alpha_l(t) \text{ for } l = 1, 2, \dots, N, \quad (3)$$

$$\alpha_l(t) = b_l + W_l h_l(t-1) + U_l h_{l-1}(t) \text{ for } l = 1, \dots, N, \quad (4)$$

$$y(t) = b_N + W_N h_N(t-1) + U_N h_N(t), \quad (5)$$

$$L = \text{loss function } (y(t), y^{\text{target}}(t)),$$

where $x(t)$ denotes the input data at the t^{th} time step; $y(t)$ represents the corresponding prediction; $y^{\text{target}}(t)$ signifies the true values of output targets; $h_l(t)$ indicates the shared states of l^{th} network layer at time step t ; and $\alpha_l(t)$ depicts the input value of the l^{th} layer at time step t , comprising the t^{th} time step input $x(t)$ or the shared state $h_{l-1}(t)$ at time t from the $(l-1)^{\text{th}}$ layer, bias b , and shared states $h_l(t-1)$ at current network layer l from the previous time step $(t-1)$.

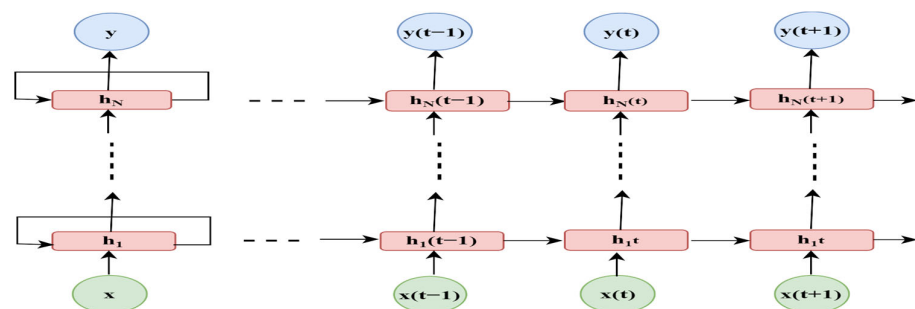


Figure 2. Unfold architecture of N-layer RNN [46].

3.3. RNN-Based Long Short-Term Memory

Despite extensive research, the non-stationary demand data and the long-term forecasting horizon of the STLFF make it difficult to obtain an accurate estimate. The capacity of LSTM to identify long-term dependencies in a time series is its key benefit. The LSTM representation was chosen as one input because it contains methods that account for the sequential character of time series. Therefore, we utilized the LSTM, a specialized form of RNN architecture, to tackle the short-term load forecasting challenge [47,48].

The memory cell, which acts as an accumulation of the state information, is the primary innovation of the LSTM. Three functions of gate controllers are demonstrated as follows [49]:

- Forget gate $f_{(t)}$ decides which part of the long-term state $c_{(t)}$ should be omitted.
- Input gate $i_{(t)}$ controls which part of $\tilde{c}_t$ should be added to the long-term state $c_{(t)}$.
- Output gate $o_{(t)}$ determines which part of c_t should be read and outputs to $h_{(t)}$ and $y_{(t)}$.

The forget gate is used to determine which information to remove from the cell state, as seen in Figure 3.

$$f_{(t)} = \sigma \left(W_f \cdot [h_{(t-1)}, x_{(t)}] + b_f \right) \quad (6)$$

$$i_{(t)} = \sigma \left(W_i \cdot [h_{(t-1)}, x_{(t)}] + b_i \right) \quad (7)$$

$$\tilde{c}_{(t)} = \tanh \left(W_c \cdot [h_{(t-1)}, x_{(t)}] \right) + b_c \quad (8)$$

$$c_{(t)} = f_{(t)} * c_{(t-1)} + i_{(t)} * \tilde{c}_{(t)} \quad (9)$$

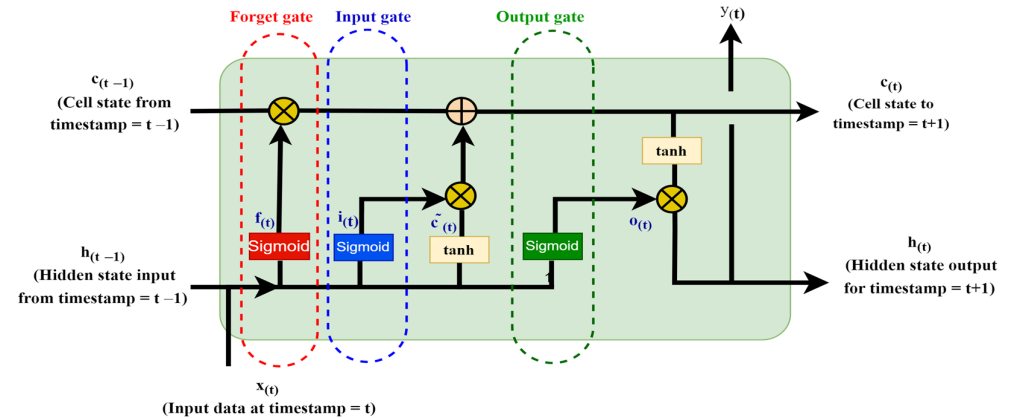


Figure 3. The architecture of LSTM memory block [50].

In conclusion, it is necessary to decide on the output, which involves two steps. First, a sigmoid layer is utilized as an output gate to selectively filter the cell state. Subsequently, the cell state is passed through a $\tanh$ function and multiplied by the output o_t to obtain the desired information.

$$o_{(t)} = \sigma \left(W_o \cdot [h_{(t-1)}, x_{(t)}] \right) + b_o \quad (10)$$

$$h_{(t)} = o_{(t)} * \tanh(c_{(t)}) \quad (11)$$

W_i , W_f , W_c , and W_o represent the appropriate weight matrices. The vectors b_i , b_f , b_c , and b_o denote the corresponding bias vectors. The most significant difficulty associated with the LSTM recurrent neural network lies in the process of adjusting its hyperparameters, such as determining the appropriate number of hidden layers, the nodes within each layer, the batch size, the number of epochs, and the learning rate, and optimizing the connection weights, biases of the network, etc. [51].

3.4. RNN Based Gated Recurrent Unit

A variation type of LSTM is the GRU, which was proposed by Cho et al. (2014) [52] to make each recurrent unit adaptively capture the dependencies of different time scales. Similarly to the LSTM unit, the GRU has gating units that modulate the flow of information inside the unit, albeit without having separate memory cells [52]. Figure 4 shows the details of the GRU system.

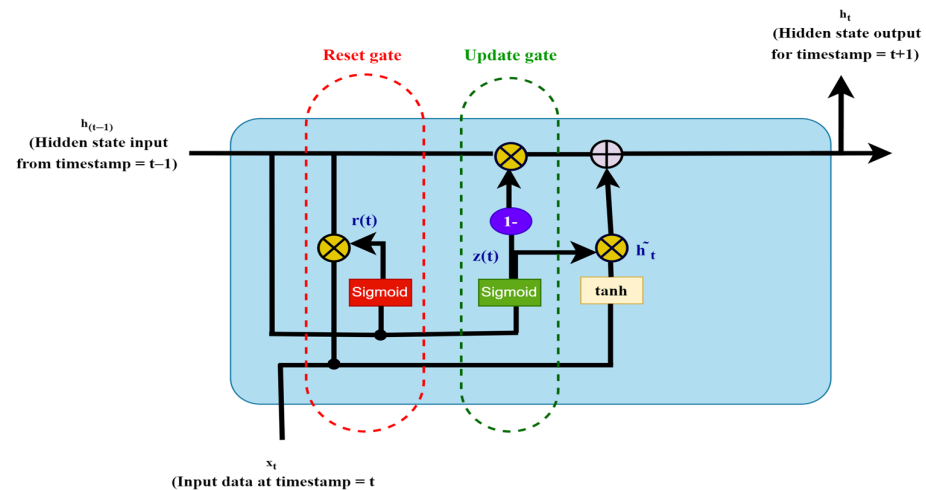


Figure 4. The architecture of the GRU memory block [53].

The diagram illustrates a GRU in which each line represents a complete vector, connecting the output of one node to the input of another node. The light pink circles in the diagram represent mathematical operations, while the yellow boxes denote the neural network layers. The direction of the lines indicates the flow of the contents. It should be noted that the GRU layer is based on the LSTM layer and has a similar equation [54]:

$$z_{(t)} = \sigma \left(W_z \cdot \left[h_{(t-1)}, x_{(t)} \right] \right) + b_z, \quad (12)$$

$$r_{(t)} = \sigma \left(W_r \cdot \left[h_{(t-1)}, x_{(t)} \right] \right) + b_r, \quad (13)$$

$$\tilde{h}_{(t)} = \tanh \left(W_h \cdot \left[r_{(t)} * h_{(t-1)}, x_{(t)} \right] \right) + b_h, \quad (14)$$

$$h_{(t)} = \left(1 - z_{(t)} \right) * h_{(t-1)} + z_{(t)} * \tilde{h}_{(t)}. \quad (15)$$

The function of the reset gate is to decide on the combination of the new input and the memory, while the update gate determines the proportion of the memory that should be retained. This gating mechanism is analogous to that used in LSTM as it aims to capture long-term dependencies [55].

4. Electricity Demand Pattern of Dataset

Our study's scope includes the Thailand metropolitan area, which comprises Pathum Thani, Nonthaburi, and Nakhon, as well as Bangkok and the neighboring provinces. These provinces are linked to an enormous number of enterprises, offices, universities, and industrial parks, all of which increase the overall energy demand. For the period of 1 January 2012 to 31 December 2018, we used half-hourly electrical energy demand data from EGAT (same as [56]) in this paper. Based on meteorological information for the Bangkok metropolitan region, the period of cooling typically spans from March through September, with the highest temperatures and humidity occurring between April and August. Throughout this timeframe, there is a notable rise in electricity usage, primarily for cooling needs such as air conditioning. There is still a noticeable demand for space heating

during the cooler period from December to February. Additionally, there is a consistent need for hot water heating throughout the year in both residential and commercial settings to provide warm water [57]. According to electricity usage data, buildings consume roughly 70% of Bangkok's total demand, which includes residential, business, and institutional areas. The rest (30%) of the city's electrical consumption comes from the industrial sector, which includes manufacturing and production [58]. From Figure 5, we can see that for the moving average, a rolling window of 365 samples was used to track the data's consistent fluctuation over time. This plot shows that seasonality influences total demand growth, which follows a linear trend.

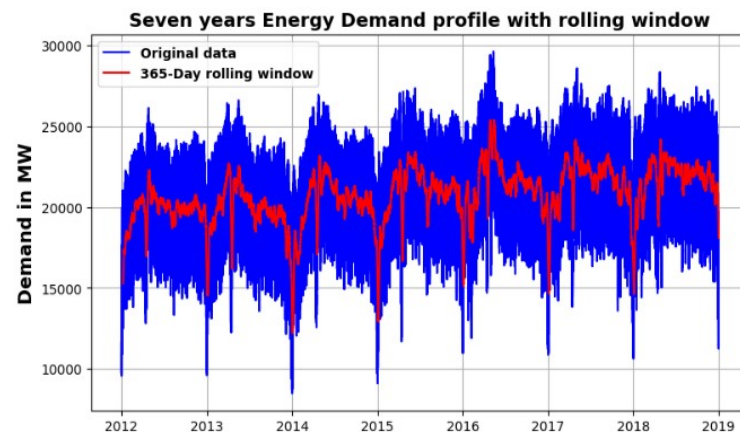


Figure 5. Seven years' overall half-hourly demand profile.

4.1. Weekdays, Weekends, and Holidays Patterns

Every year, during holidays and other special occasions, there is a discernible drop in the electricity demand. Figure 6a,b shows that extended holidays like the Songkran holiday in the second week of April and the New Year's holiday in the first week of January, respectively, have a major effect on demand. But in the case of the New Year holiday, demand has started to decrease from Christmas (December 25th) to the first week of January, as shown in Figure 6b. These effects are important to model and provide a considerable challenge to researchers who want to attain high forecasting accuracy. Furthermore, though they have less of an impact than Songkran and New Year's, holidays like Makha Bucha Day (February), Visakha Bucha Holiday (May), and Father's Day (December) also affect swings in electricity demand.

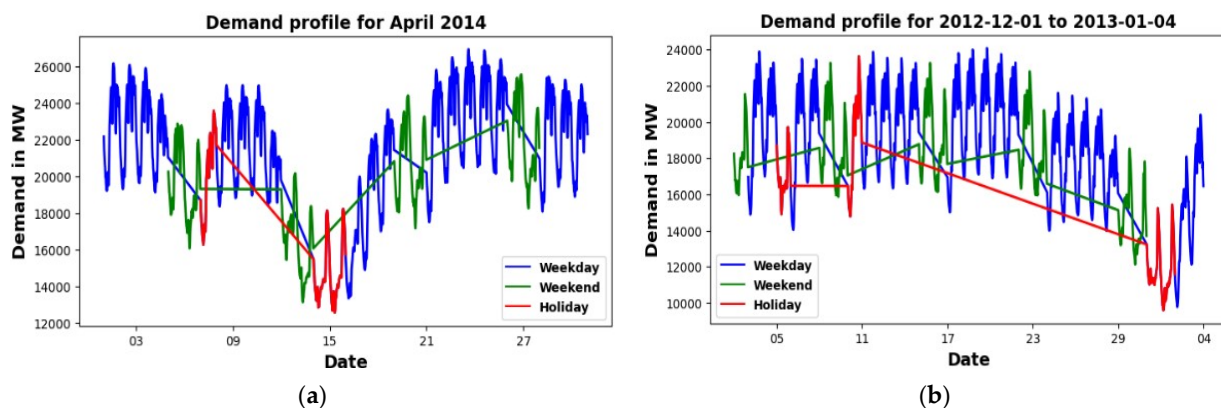


Figure 6. (a) Demand profile of weekdays, weekends, and holidays for April 2014; (b) demand profile of weekdays, weekends, and holidays for December 2012 to 4 January 2013.

4.2. Monthly, Weekly, and Daily Patterns

The demand value varies each month according to the season and holidays observed, especially during these months. Demand was at its peak during the summer, which runs

from March through September, although it was lower in April due to Songkran, a lengthy holiday that fell during the second week of the month. The months of October through February are a little bit cooler than the summer; thus, there is less demand during these months. Figure 7a shows the half-hourly demand pattern for one (the first) week of April 2012, which illustrates that Saturday and Sunday (weekends) have low electricity demand compared to other working days (weekdays) because manufacturing plants, offices, and industrial loads dominate residential demand in our research area. On weekends and holidays, residential demand, driven by human behavior, fluctuates unpredictably, affecting forecasting accuracy. Weekdays show consistent midday and evening demand patterns, unlike weekends and holidays. Late-night and early-morning demand remain consistent across weekdays and weekends (Figure 7b).

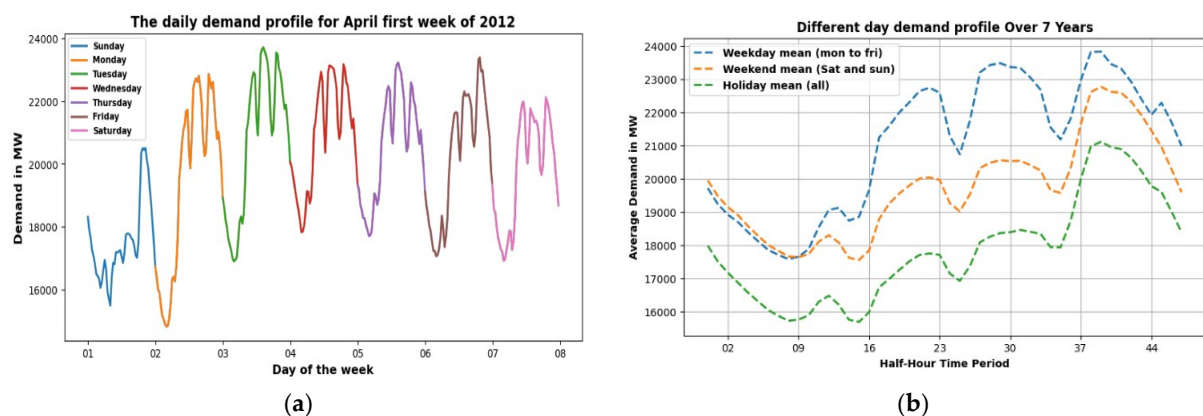


Figure 7. (a) Daily demand for one-week; (b) Average demand profile for weekday, weekend, and holiday patterns for 7 years.

4.3. Temperature vs. Demand

Thailand is a tropical nation with higher average temperatures; as such, temperature is the primary factor influencing electricity consumption in Thailand. Figure 8 illustrates the clear correlation between electricity demand and temperature. In summer, higher AC usage boosts electricity demand for cooling, while in winter, lower temperatures lead to a slight decrease in demand.

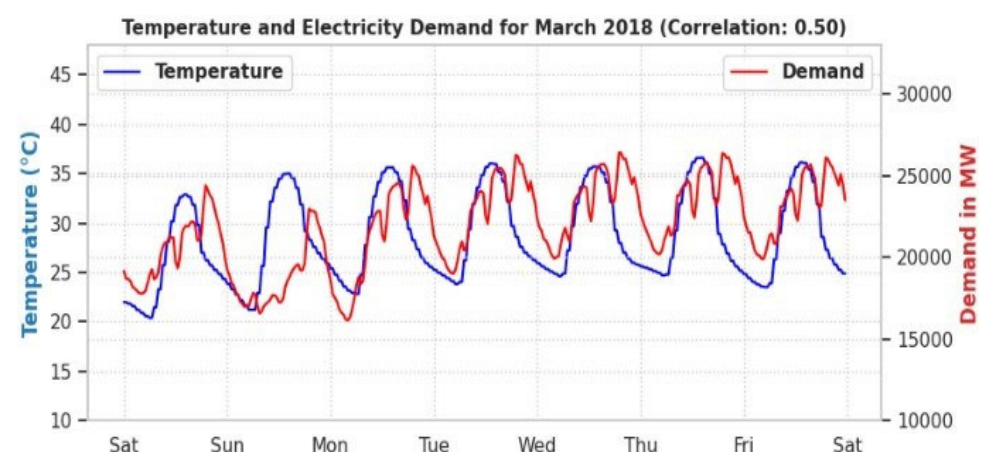


Figure 8. Correlation between temperature and electricity demand for the second week of March 2018.

Temperature's impact on electricity demand on holidays and non-holidays at mid-night and the peak hour are depicted in Figure 9b,c, respectively, where orange dots denote holiday and blue dots denote non-holiday demand. Figure 9c describes how temperature affects demand for both holidays and non-holidays during peak hours (2 pm), where demand varied significantly on holidays during the peak hour period. During working

days, there is a sharp and linear demand for electricity. The influence of temperature on peak demand is negligible. This fact confirms the assertion made in article [59] that temperature has a negligible effect on commercial demand. When the temperature falls below 30 °C or rises above 35 °C, there is a significant fluctuation in demand during holidays. Moreover, Figure 9a shows the features of the electrical power consumption at two distinct hours, 2 pm with blue dots and 11 pm with orange dots, where the relationship between peak electricity demand and temperature during these two time periods is compared. In particular, Figure 9a indicates that in comparison to the demand at 2 pm, the demand at 11 pm shows a more linear relationship with temperature. The same linear relationship is illustrated for both holidays and non-holidays at 11 pm in Figure 9b.

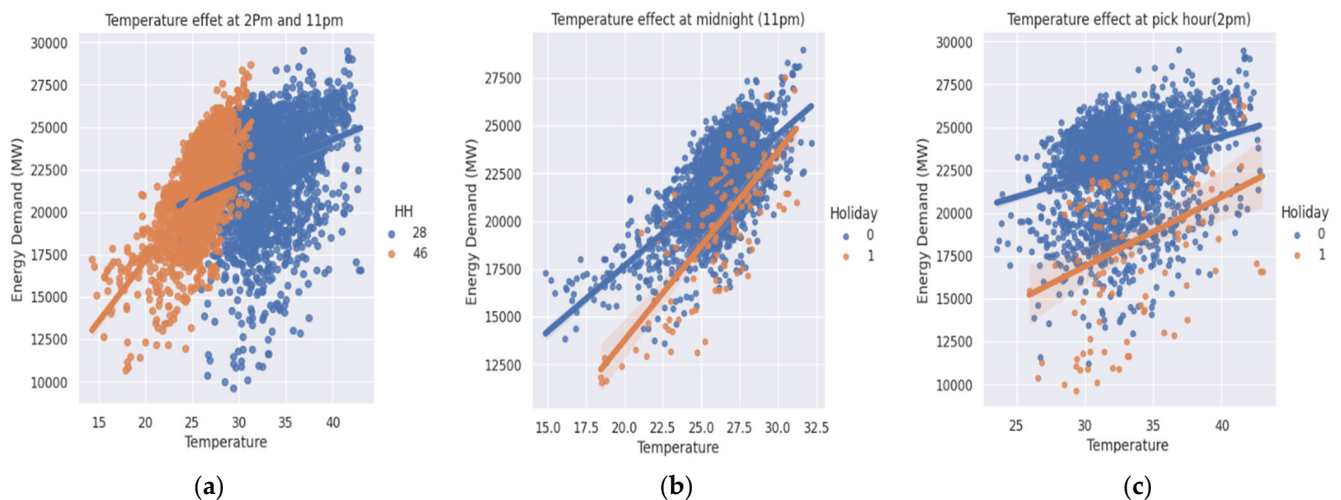


Figure 9. (a) Effect of temperature at two different hours (2 pm and 11 pm); (b) effect of temperature during midnight (11 pm); and (c) effect of temperature during the peak hour (2 pm).

5. System Model

This article provides three types of deep learning models with hyperparameter tuning for short-term electrical demand forecasting. The proposed approaches are described in this section and are shown in Figure 10. There are four frameworks into which the suggested methods are classified: data pre-processing, model design with training data, fine tuning and testing, and performance analysis. Data normalization, structural transformation, and removing outliers were part of the preprocessing process of the raw data.

Large values can hinder the deep neural network's learning and convergence, so feature scaling was used to scale the data in the range of 0 to 1. All three deep learning models are trained using the training dataset. The input data and the associated target values are taught to the model to identify patterns and relationships. To reduce the difference between the predicted and the original target values, the model modifies its internal variables (weights and biases) according to the input data. The validation dataset is used to observe the model's effectiveness during training and to adjust the hyperparameters. It directs the process of determining the model and helps avoid overfitting. After training is finished, the testing dataset is used to assess how well the model performs on unknown data.

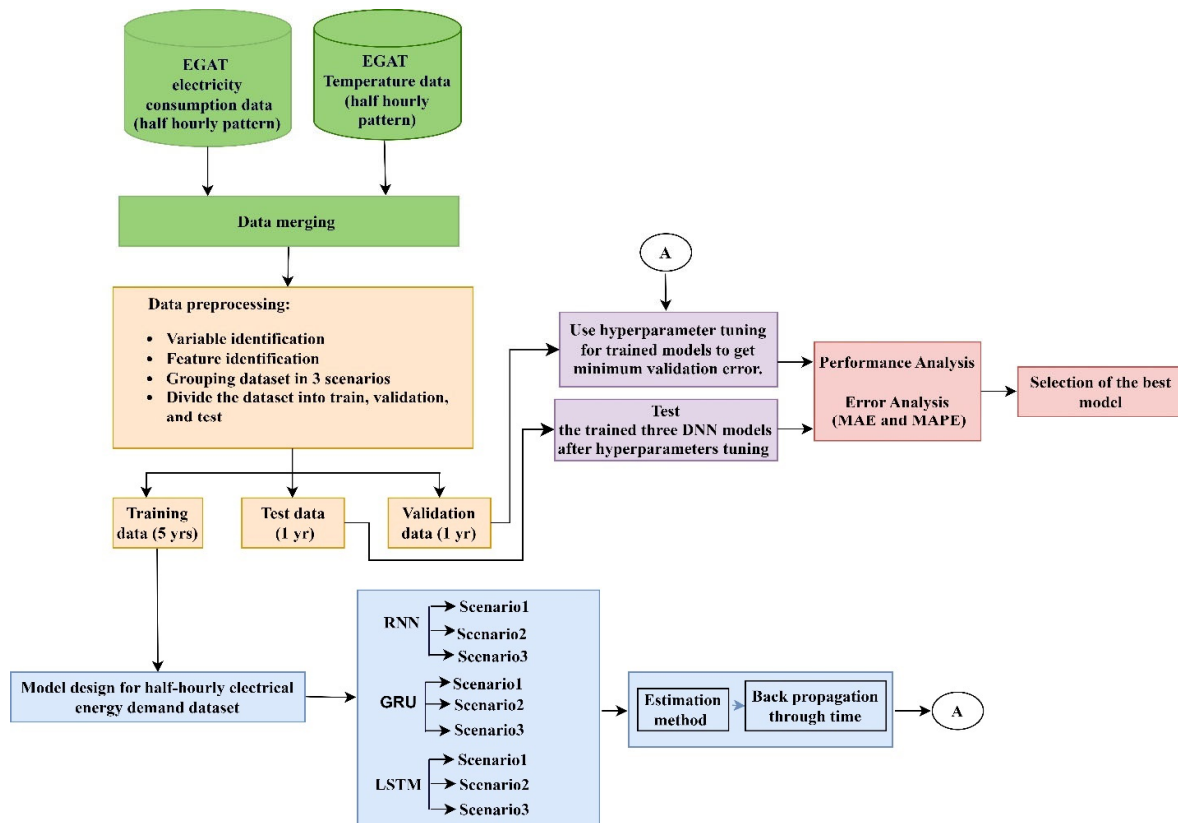


Figure 10. Proposed electrical demand forecasting methodology.

5.1. Data Preparation

EGAT, the largest company in the Thai electricity market, provided the dataset that was applied in this research. EGAT processes 100% of the transmission system and nearly 50% of the generation. The nation is divided into five regions by EGAT, which also maintains separate records of the electric loads [41]. The dataset spans seven years (from 1 January 2012 to 31 December 2018) and contains 122,736 instances of half-hourly electricity demand (in MW). There are additional deterministic, historical load, and interaction variables in the preprocessed data. In the entire dataset, 5 years (from 1 January 2012 to 31 December 2016) were selected as the training dataset, 1 year (from 1 January 2017 to 31 December 2017) as the validation dataset, and the remaining 1 year (from 1 January 2018 to 31 December 2018) as the testing dataset. The dataset was split into three groups, known as scenarios, unlike previous research mentioned in [2] in which [41] they were divided into four groups. Splitting the dataset proved effective in simplifying the modeling and demonstrated a significant accuracy enhancement compared with the related work.

Our available dataset comprised half-hourly (HH) 7-year data with 122,736 instances. Hence, the numbers of training, validation, and test instances for each HH in each scenario were as follows:

- Scenario 1: all the demand from the whole dataset (everyday demand): training = 87,696; validation = 17,520; test = 17,520.
- Scenario 2: only demand for weekdays (demand without Saturday, Sunday, and other holidays): training = 58,848; validation = 11,760; test = 11,808.
- Scenario 3: only demand for non-working days (demand for weekends and holidays): training = 28,848; validation = 5808; test = 5664.

5.2. Forecast Horizon

The EGAT system in Thailand forecasts conditions for 2 pm on the next day, along with 10 to 34 h in advance. Since the EGAT office is not open on weekends, EGAT must forecast up to Monday from Friday. Thailand usually uses data up to 106 h ahead for short-term forecasting, particularly during extended holidays. However, the scope of this study is restricted to forecasting for the following day utilizing data up to 2 pm. Simplifying the model, Figure 11 displays the forecast horizon and lag terms specific to Thailand's practice. To predict demand on a given day, forecasting begins 10 h in advance, at 2 pm. The variable `load1d_cut2pm` now represents the historical demand data from HH = 0 to HH = 28 of d−1 (the preceding day) and HH = 29 to HH = 47 of d−2. Likewise, our model incorporates the lagged demand for the next two days, which is denoted by a parameter called `load2d_cut2pm`.

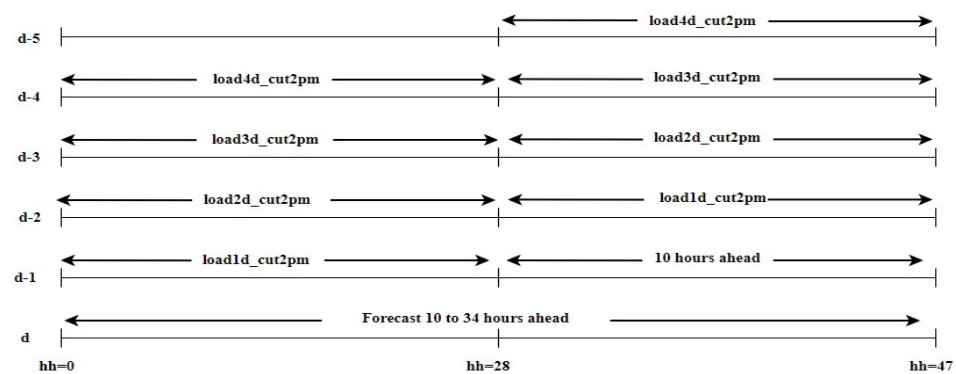


Figure 11. Demand forecast horizon: the practice of EGAT in Thailand [59].

Adding dummy variables and their relationships has been recommended by earlier research to increase forecasting accuracy. Ramanathan et al. [60] suggested the usage of four types of dummy variables—deterministic, temperature, load, and past errors—and their interactions. The term “deterministic” describes predictable variables, like the day of the week, the month, and the year [60]. The variables were grouped into two categories: deterministic and lagged, as presented in Table 2.

Table 2. List of selected input variables [59].

Types	Variables	Description
Deterministic	WD	Weekday dummy (Mon, Tue, ..., Sat, Sun)
	MD	Month dummy (Jan, Feb, ..., Nov, Dec)
	Holiday	Binary 0 or 1
	Day After Holiday	Binary 0 or 1
	Day After long Holiday	Binary 0 or 1
Lagged	Holiday Type	Type of holiday (Songkran, Newyear,, etc.)
	Load1d_cut2pm	1 day ahead until 2 pm and 2 days ahead after 2 pm load
	Load2d_cut2pm	2 days ahead until 2 pm and 3 days ahead after 2 pm load
	Load3d_cut2pmR	3 days ahead until 2 pm and 4 days ahead after 2 pm load
Temperature	Load4d_cut2pmR	4 days ahead until 2 pm and 5 days ahead after 2 pm load
	Temp	Forecasted temperature
	MaxTemp	Maximum forecasted temperature
	Square temperature	Square of the forecasted temperature
Interaction	MA2pmTemp	Moving average of temperature at 2 pm
	WD: Temp	Interaction of weekday dummy to temperature
	MD: Temp	Interaction of month dummy to temperature

5.3. Hyperparameter Tuning

When creating deep learning models, hyperparameter tuning is an essential step. The configuration parameters that a model is trained on and structured with are called hyperparameters. Hyperparameters, in contrast to parameters, which are discovered from the data during training, are preset before training and direct the learning process. The goal of fine-tuning these hyperparameters is to identify the optimal combination that maximizes the model's performance [61]. We independently carried out the hyperparameter optimization procedure for every deep network in our experiment. These were the primary hyperparameters:

- Dropout: This serves as a regularization technique aimed at mitigating overfitting by randomly selecting cells within a layer and nullifying their output based on a specified probability.
- Number of layers: The neural network's depth, which represents its number of layers. Deeper networks have the potential to overfit but can also capture more complex patterns. Achieving the proper depth is critical.
- Epochs: How many times is the neural network used to pass the whole dataset back and forth during training? Epoch determines how frequently the model views the complete dataset. Underfitting may result from a lack of epochs, and overfitting may result from an excess of epochs.
- Number of neurons in each layer: The quantity of neurons in every network layer regulates the model's ability to recognize patterns. Underfitting can occur from having too few neurons, while overfitting can occur from having too many.
- Batch size: The total number of training instances used in a single iteration impacts how quickly and effectively training uses memory. Greater batch sizes may result in quicker training, but they also demand more RAM.
- Learning rate: The step size while attempting to minimize a loss function at each iteration. The model might overshoot the minimum but converge quickly if the learning rate is high. A low learning rate can cause slow convergence.
- Lookback period: The lookback period refers to the number of previous time steps that are taken into consideration by the model when generating predictions. The model becomes more complex when the lookback period is extended, but it also enables the model to capture longer-term dependencies and patterns. If the lookback period is too short, the model might miss important temporal relationships.

5.4. Performance Analysis

An essential step in determining a predictive model's efficacy is performance analysis. In this paper, mean absolute error (MAE) and mean absolute percentage error (MAPE) were used as metrics for performance analysis.

Performance analysis workflow:

- Model Training: Train three deep learning models using the training dataset.
- Hyperparameter Tuning: Fine-tune different hyperparameters, which are mentioned in Section 5.3, based on performance metrics (MAE, MAPE) calculated on the validation dataset.
- Model Evaluation: Evaluate the prediction performance of the completed trained model using the testing dataset.
- Performance Metrics Calculation:
 - Use the trained model to make predictions for energy demand on the testing dataset.
 - Calculate MAE and MAPE metrics using the actual demand value of the test dataset and predicted value from a trained model using testing data.
 - These metrics give us information about the precision and accuracy of the predictions made by our final model.

5.4.1. Mean Absolute Error (MAE)

One useful key performance indicator (KPI) for evaluating forecast accuracy is the mean absolute error (MAE). To ensure that positive and negative residuals never cancel out, we will compute the residual for each data point using only its absolute value. After that, we compute the mean of each of these residuals. The MAE, which merely calculates the absolute difference between the model's predictions and the actual demand data, is the most appropriate metric. The residual's absolute value is utilized, so the under- or overperformance of the model is not displayed. The formal equation of MAE is shown below:

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^n |y_t - F_t|, \quad (16)$$

where

n = total number of data points;

y_t = actual electricity demand at a time " t ";

F_t = predicted electricity demand at a time " t ";

5.4.2. Mean Absolute Percentage Error (MAPE)

In percentage terms, MAE is equivalent to mean absolute percentage error (MAPE). Like MAE, which represents the mean absolute error generated by your model, MAPE indicates the average deviation between the model's predictions and their corresponding outputs. A popular metric for assessing forecast accuracy in time series data is called MAPE, which is computed as the average of the absolute percentage (%) errors (APEs) within the predicted and actual values of time series data. The formal equation of MAE is shown below:

$$\text{MAPE} = \left(\frac{1}{n} \right) * \sum_{t=1}^n \left(\frac{|y_t - F_t|}{|y_t|} \right) * 100\%. \quad (17)$$

6. Results and Analysis

This section provides an outline of the methodology used to empirically determine the hyperparameters for the RNN, RNN-LSTM, and RNN-GRU deep learning models across all three scenarios. We separated the entire dataset into three datasets: training, validation, and testing. This allowed us to fine-tune the model's hyperparameters and increase its efficacy by separating the training dataset into distinct validation datasets. Throughout the training phase, concurrent validation steps were executed using this validation dataset. To conduct the research, the hyperparameter values listed in Table 3 are utilized.

Table 3. Sets of hyperparameters for all three scenarios.

Parameters	Values
Number of hidden layers	[1, 2, 3]
No of nodes	[2, 4, 8, 16, 32, 64, 128]
Look back period	[5 days and 8 days]
Dropout	[0, 0.1, 0.15, 0.2]
Epochs	[up to 1 million]

To test hyperparameters value sets particularly for the RNN, RNN-LSTM, and RNN-GRU models, we have assembled a list for all three scenarios. Initially, we tested the built-in model's validity and functionality using the parameters listed in Table 3. The set of parameters that produced the lowest validation mean absolute error (MAE) was chosen as the ideal or tuned parameter set after its performance was successfully verified. Tables 4, 6, and 8 show the complete set of parameters for all three scenarios.

6.1. Parameters Tuning for Scenario 1

The electricity demand for the year 2018 is forecasted utilizing the refined parameters illustrated in Table 4 with a dataset size of 105,216 and a test size of 17,520. Compared to the other two RNN-LSTM and RNN models, the RNN-GRU model performed better. We achieved the lowest minimum loss of validation (MAE) of 213.36 MW compared to 221.79 MW and 242.35 MW using the RNN-GRU simulation setup.

Table 4. Sets of parameters for lowest validation loss of Scenario 1.

Parameters	LSTM	GRU	RNN
Number of hidden layers	1	1	3
Number of nodes	128	64	64
Epochs	36	42	38
Batch size	48	48	48
Look back period	8 days	5 days	8 days
Dropout	0.15	0.15	0.15
Time period	48	48	48
Validation loss (MAE)	221.79	213.36	242.35

In Scenario 1, as stated in Table 4, we were able to attain the following outcomes:

- The LSTM model with nodes = 128, layers = 1, dropout = 0.15, lookback period = 8 days, and epochs = 36 produced the minimal validation loss of 221.79 MW.
- The GRU model with nodes = 64, layers = 1, dropout = 0.15, lookback period = 5 days, and epochs = 42 obtained the minimal validation loss of 213.36 MW, which was the lowest loss across all three models.
- The RNN model with nodes = 64, layers = 3, dropout = 0.15, lookback period = 8 days, and epochs = 38 achieved the minimal validation loss of 242.35 MW, which was the highest loss across all three models.

When optimized hyperparameters are used, Table 5 shows that the RNN-GRU model achieved its best performance in terms of both performance metrics (MAE and MAPE) with MAE = 214.79 MW and MAPE = 2.08%.

Table 5. Implementing the optimal parameter settings for day-ahead forecasting using the test dataset of Scenario 1.

Model	Nodes	Layers	Dropout	Lookback Period	Epochs	Test MAE	Test MAPE (%)
LSTM	128	1	0.15	8 days	36	224.75	2.12
GRU	64	1	0.15	5 days	42	214.79	2.08
RNN	64	3	0.15	8 days	38	248.64	2.28

6.2. Parameters Tuning for Scenario 2

The electricity demand for the year 2018 is forecasted, utilizing the refined parameters illustrated in Table 6 with a dataset size of 70,608 and a test size of 11,808. Compared to the other two RNN-LSTM and RNN models, the RNN-GRU model performed better. We achieved the lowest minimum loss of validation (MAE) of 179 MW compared to 189.87 MW and 195.11 MW using the RNN-GRU simulation setup.

Table 6. Set of parameters for lowest validation loss of Scenario 2.

Parameters	LSTM	GRU	RNN
Number of hidden layers	1	1	2
Number of nodes	128	64	64
Epochs	57	61	51
Batch size	48	48	48
Look back period	5 days	8 days	8 days
Dropout	0.15	0.15	0.2
Time period	48	48	48
Validation loss (MAE)	189.87	179	195.11

In Scenario 2, as stated in Table 6, we were able to attain the following outcomes:

- The LSTM model with nodes = 128, layers = 1, dropout = 0.15, lookback period = 5 days, and epochs = 57 produced the minimal validation loss of 189.87 MW.
- The GRU model with nodes = 64, layers = 1, dropout = 0.15, lookback period = 8 days, and epochs = 61 obtained the minimal validation loss of 179 MW, which was the lowest loss across all three models.
- The RNN model with nodes = 64, layers = 2, dropout = 0.2, lookback period = 8 days, and epochs = 51 achieved the minimal validation loss of 195.11 MW. which was the highest loss across all three models.

When optimized hyperparameters are used, Table 7 shows that the RNN-GRU model achieved its best performance in terms of both performance metrics (MAE and MAPE) with MAE = 181.63 MW and MAPE = 1.89%.

Table 7. Implementing the optimal parameter settings for day-ahead forecasting using the test dataset of Scenario 2.

Model	Nodes	Layers	Dropout	Lookback Period	Epochs	Test MAE	Test MAPE (%)
LSTM	128	1	0.15	5 days	57	191.57	1.92
GRU	64	1	0.15	8 days	61	181.63	1.89
RNN	64	2	0.2	8 days	51	201.35	1.98

6.3. Parameters Tuning for Scenario 3

The electricity demand for the year 2018 is forecasted utilizing the refined parameters illustrated in Table 8 with a dataset size of 34,656 and a test size of 5664. Compared to the other two RNN-GRU and RNN models, the RNN-LSTM model performed better. We achieved the lowest minimum loss of validation (MAE) of 222.07 MW compared to 229.89 MW and 230.57 MW using the RNN-LSTM simulation setup.

Table 8. Set of parameters for lowest validation loss of Scenario 3.

Parameters	LSTM	GRU	RNN
Number of hidden layers	1	2	1
Number of nodes	128	64	128
Epochs	72	81	69
Batch size	48	48	48
Look back period	8 days	8 days	5 days
Dropout	0.1	0	0.1
Time period	48	48	48
Validation loss (MAE)	222.07	229.89	230.57

In Scenario 3, as stated in Table 8, we were able to attain the following outcomes:

- The LSTM model with nodes = 128, layers = 1, dropout = 0.1, lookback period = 8 days, and epochs = 72 produced the minimal validation loss of 222.07 MW which was the lowest loss across all three models.
- The GRU model with nodes = 64, layers = 2, dropout = 0, lookback period = 8 days, and epochs = 81 obtained the minimal validation loss of 229.89 MW.
- The RNN model with nodes = 128, layers = 1, dropout = 0.1, lookback period = 5 days, and epochs = 69 achieved the minimal validation loss of 230.57 MW, which was the highest loss across all three models.

When optimized hyperparameters are used, Table 9 shows that the RNN-LSTM model achieved its best performance in terms of both performance metrics (MAE and MAPE) with MAE = 226.76 MW and MAPE = 2.13%.

Table 9. Implementing the optimal parameter settings for day-ahead forecasting using the test dataset of Scenario 3.

Model	Nodes	Layers	Dropout	Lookback Period	Epochs	Test MAE	Test MAPE (%)
LSTM	128	1	0.1	8 days	72	226.76	2.13
GRU	64	2	0	8 days	81	231	2.19
RNN	128	1	0.1	5 days	69	236.57	2.22

The results of RNN-LSTM, RNN-GRU, and RNN are presented in Figures 12–14, respectively, for Scenario 2 (result of alternative months). In each plot, the actual demand is represented as “Actual”, while the forecasted results are indicated by their respective model names, such as LSTM, GRU, and RNN. From Figures 12–14, it is evident that the RNN-GRU model performed the best among all three models in terms of prediction accuracy. Specifically, in the case of the RNN-GRU model depicted in Figure 13, there is underestimation in March, May, and July, while January, September, and November exhibit overestimation. Additionally, the results of all three scenarios using all three models are detailed in Appendices A and B.

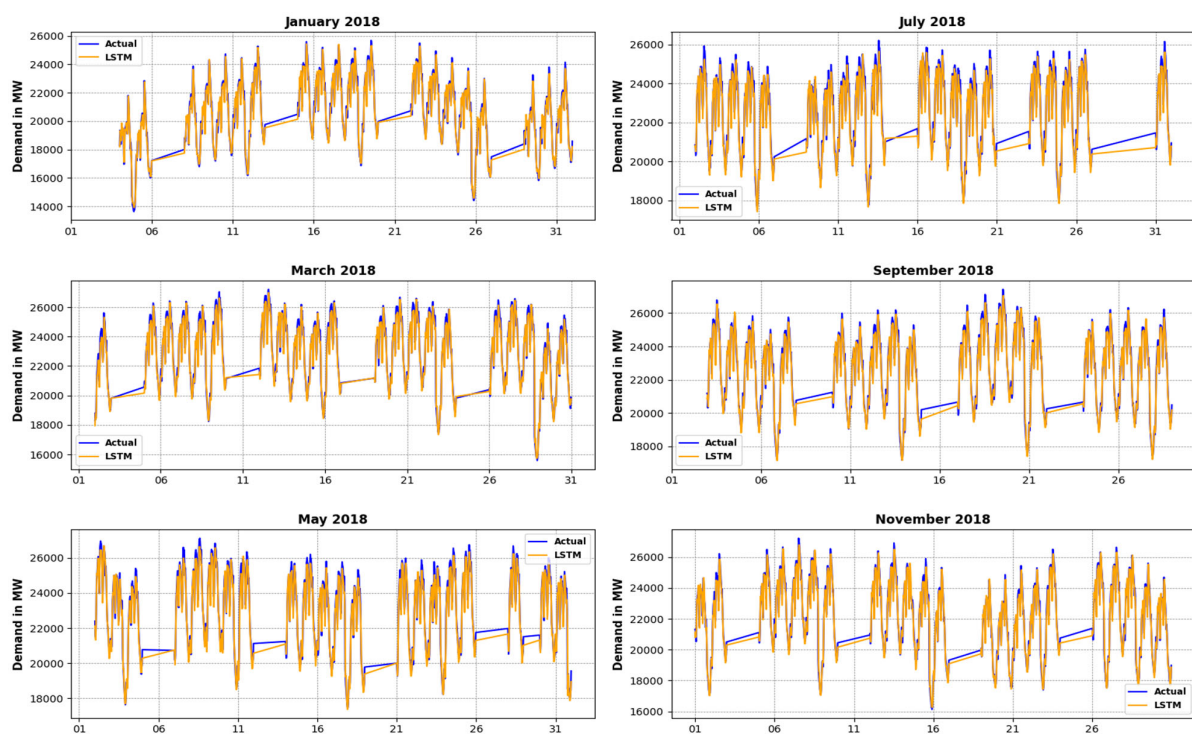


Figure 12. Electricity demand (MW) on the y-axis and days on the x-axis using RNN-LSTM.

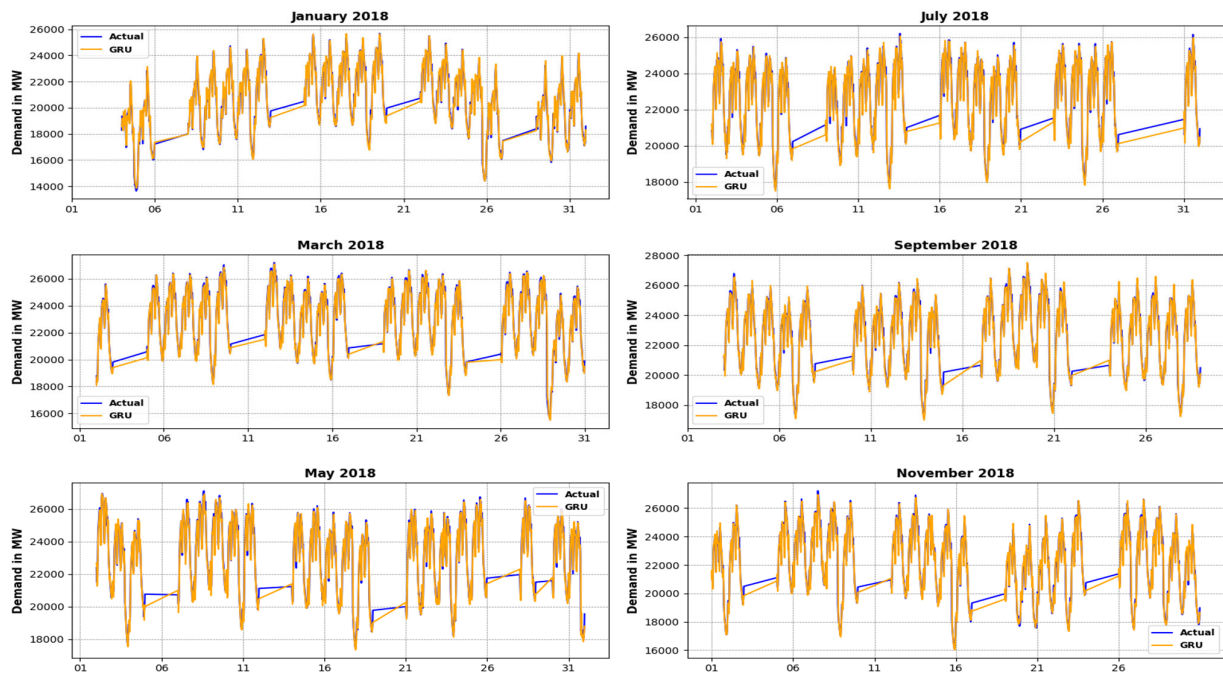


Figure 13. Electricity demand (MW) on the y -axis and days on the x -axis using RNN-GRU.

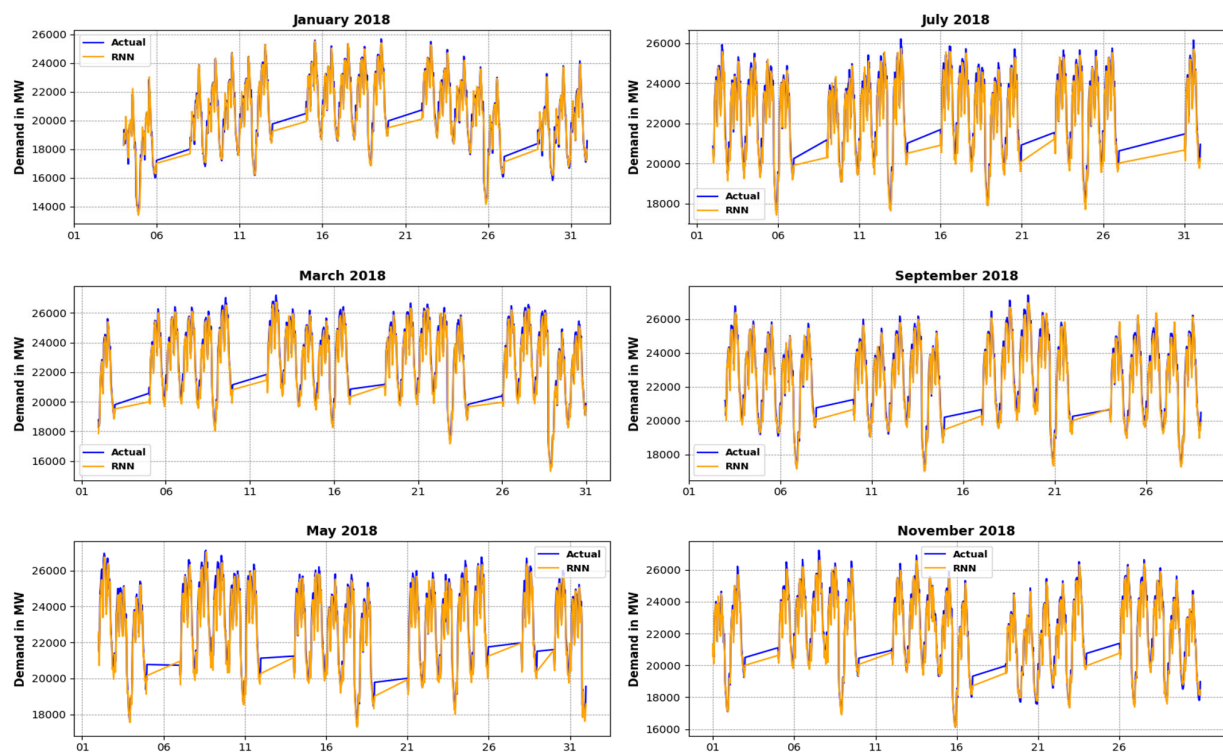


Figure 14. Electricity demand (MW) on the y -axis and days on the x -axis using RNN.

7. Conclusions

Ensuring the equilibrium between electricity production and consumption is crucial for efficient power system operation. Precise load forecasting plays a key role in achieving this balance as it enables the proactive anticipation of fluctuations in both supply and demand, thereby ensuring grid stability. This paper presented three types of deep learning models: RNN-LSTM, RNN-GRU, and RNN. They were each constructed to represent three different scenarios using the same seven-year half-hourly electrical energy demand data

from EGAT for STLF. All three models, along with three scenarios, were validated using one year of data to optimize hyperparameters. These tuned parameters were then utilized to day-ahead forecast the demand for the year 2018.

In Scenario 1, the RNN-GRU model exhibited the lowest validation MAE loss. This model utilized a 5-day lookback period with 128 nodes in a single layer. It achieved the lowest test MAE loss of 214.79 MW and MAPE of 2.08%. For Scenario 2, the RNN-GRU model achieved the lowest validation MAE loss using an 8-day lookback period with 64 nodes in a single layer, resulting in a test MAE loss of 181.63 MW and MAPE of 1.89%. Meanwhile, in Scenario 3, the RNN-LSTM model obtained the lowest-validation MAE with an 8-day lookback period with 128 nodes in a single layer. This model yielded a test MAE loss of 226.76 MW and MAPE of 2.13%. Among the obtained results from three different models using three scenarios, Scenario 2 emerges as the most promising, outperforming the other two scenarios. This superiority can be assigned to its exclusive focus on the weekday demand dataset, which exhibits consistent demand patterns. On the other hand, Scenario 1 encompasses the entire dataset, including weekdays, weekends, and holidays, leading to a higher degree of demand pattern non-linearity, which raises prediction errors. Scenario 3, which comprises only weekend and holidays demand data, shows improved results compared to previous research, likely due to the expanded dataset.

These findings suggest the potential for implementing STLF by categorizing datasets into three distinct scenarios. Scenario 2 can effectively forecast electricity demand for weekdays (Monday to Friday), while Scenario 3 proves valuable for forecasting demand during weekends and holidays (Saturday, Sunday, and public holidays), demonstrating enhanced accuracy. We did not utilize additional variables such as solar insolation, air humidity, cloud cover percentage, and daily temperature amplitude (maximum–minimum difference) in our study. However, we may consider incorporating these variables in future research to assess their impact on prediction accuracy.

Author Contributions: Conceptualization and methodology, R.K.S. and C.C.; software, R.K.S.; validation, R.K.S., K.C. and C.C.; formal analysis, C.C.; investigation, K.C.; writing—original draft preparation, R.K.S. and K.C.; writing—review and editing, C.C., K.C. and R.K.S.; visualization, K.C. and R.K.S.; supervision, C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data utilized in this research are accessible upon request from the corresponding author. However, they are not publicly accessible as the dataset for electricity demand in Thailand is not openly available. Consequently, the author must uphold privacy and ethical considerations, preventing their public release.

Acknowledgments: The recipient acknowledges the EFS Scholarship granted by the Sirindhorn International Institute of Technology, Thammasat University. And we extend our heartfelt gratitude to Chawalit Jeenanunta and the EGAT for their invaluable contribution of the essential data utilized in this study.

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

Appendix A. Visualization of Different Three Scenarios

The demand forecasting for the year 2018 was conducted using RNN-LSTM, RNN-GRU, and RNN models across scenarios 1, 2, and 3. In each plot, the actual demand is represented as “Actual”, while the forecasted results are indicated by their respective model names, such as LSTM, GRU, and RNN, as provided below (for alternative months). In Figure A1, which illustrates the predictions for Scenario 3 across all three models, the RNN-LSTM model outperformed the others. However, it displayed underestimation in

March and May, while overestimation was observed in September and November. From Figures A2 and A3, representing scenarios 2 and 1 for predictions by all three models, it is apparent that the RNN-GRU model outperformed the other two models in both scenarios. There is underestimation in March and May, while September and November exhibited overestimation in Scenario 2. And we observed underestimation in March and May, while overestimation was observed in July, September, and November in Scenario 1.

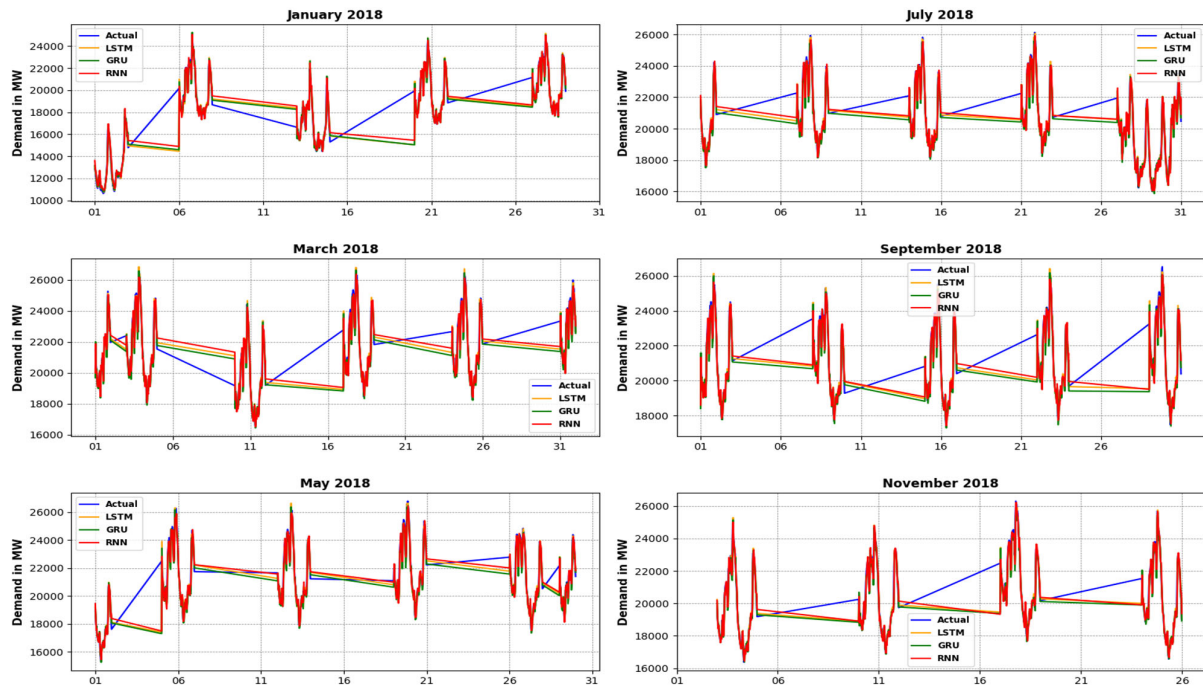


Figure A1. Electricity demand (MW) on the y -axis and days on x -axis, utilizing Scenario 3.

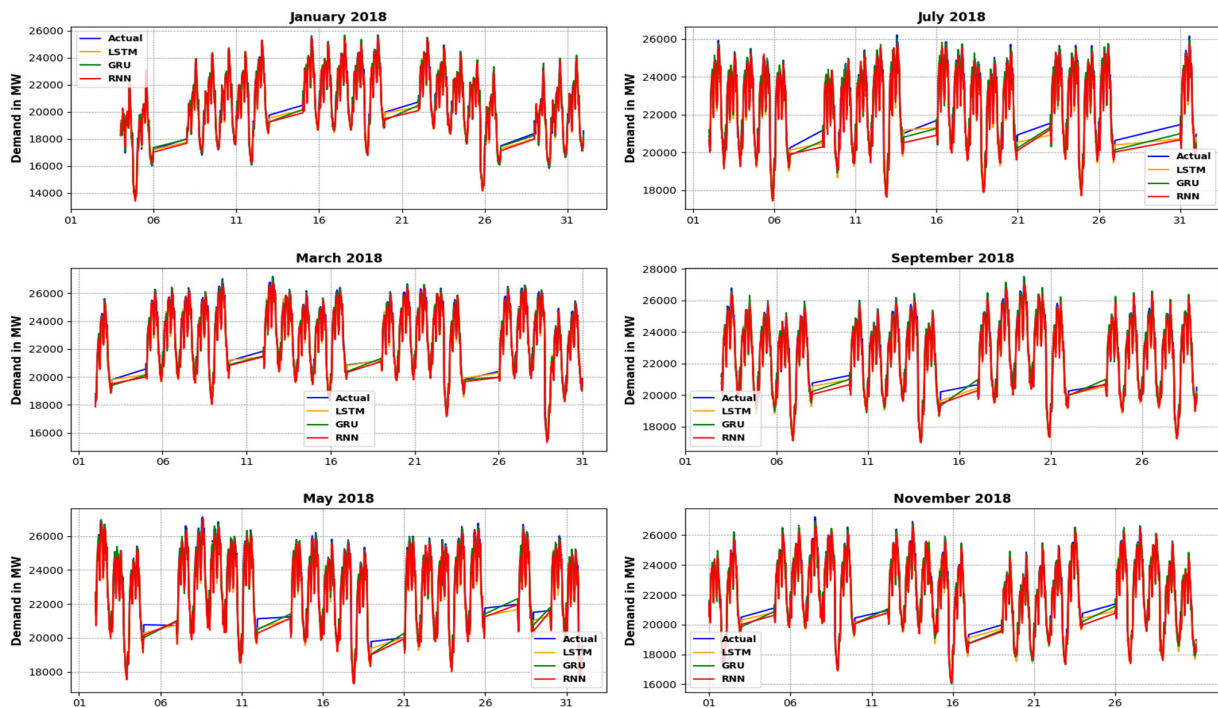


Figure A2. Electricity demand (MW) on the y -axis and days on x -axis, utilizing Scenario 2.

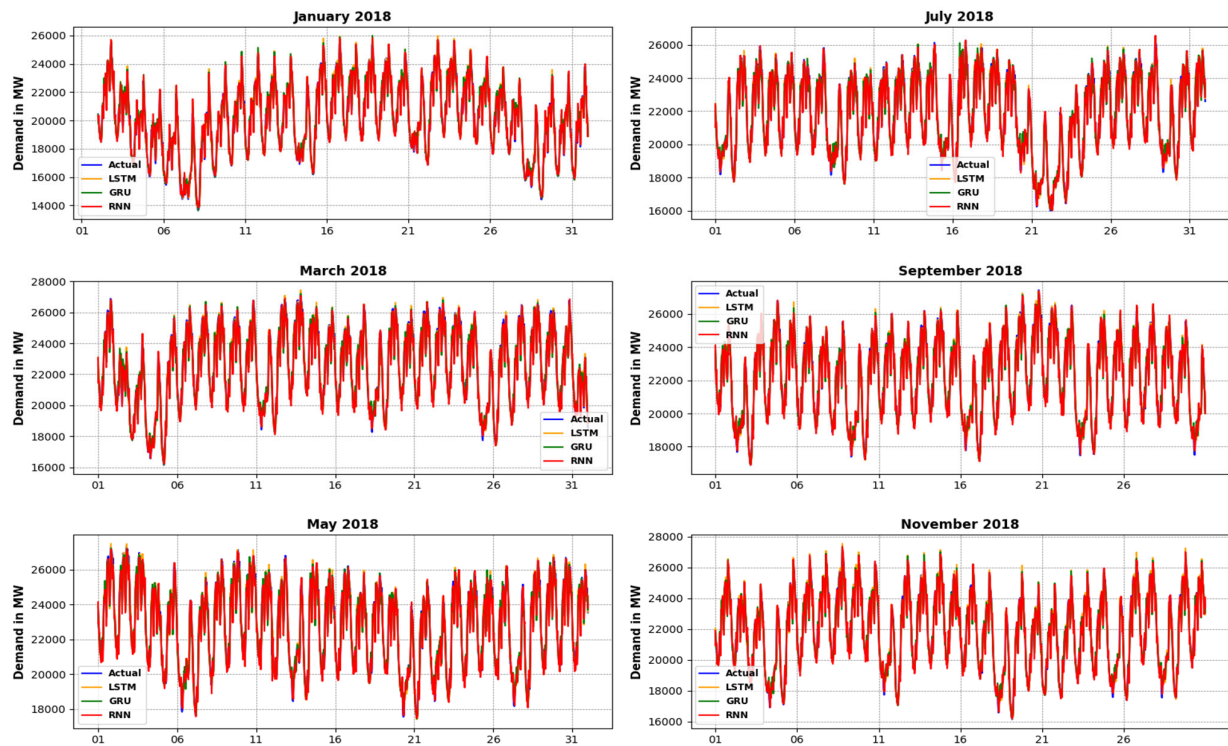


Figure A3. Electricity demand (MW) on the y -axis and days on x -axis, utilizing Scenario 1.

Appendix B. Tuning of Hyperparameters for Different Three Scenarios

The hyperparameter selection process for Scenario 1 is detailed in Table A1, presenting RNN-GRU's superior performance over the other two models. Afterward, a range of parameters was tested for all three models to determine the optimal configuration. The resulting validation and test MAE values for each parameter set are presented in the table below.

Table A1. Parameter variation and corresponding test and validation MAE outcomes for Scenario 1.

Parameters					LSTM Results (MAE)		GRU Results (MAE)		RNN Results (MAE)	
Layers	Nodes	Dropout	Lookback	Epochs	Validation	Test	Validation	Test	Validation	Test
1	128	0.15	5	38	286.71	287.95	273.66	277.59	294.81	298.9
1	64	0.15	5	42	245.44	251.39	213.36	214.79	268.78	273.77
1	32	0.15	5	57	223.69	226.20	284.98	288.17	292.97	295.03
1	128	0.2	5	40	308.23	314.27	312.40	313.16	409.04	414.22
1	64	0.2	5	44	328.21	331.26	361.68	363.98	275.32	277.59
1	32	0.2	5	57	295.88	298.18	301.24	303.36	312.68	314.45
2	64	0.15	5	47	328.18	335.36	250.78	252.21	279.92	281.27
2	64	0.2	5	48	323.48	326.33	293.34	297.81	340.18	343.97
2	32	0.15	5	55	309.48	318.81	249.92	251.27	293.73	294.13
2	32	0.2	5	52	339.40	340.97	306.15	310.35	365.40	368.59
3	64	0.15	5	32	322.84	328.63	268.21	269.81	334.20	341.09
3	64	0.2	5	32	321.49	326.83	269.84	271.46	305.83	306.14
3	32	0.15	5	45	263.27	262.47	238.92	240.83	340.58	345.22
3	32	0.2	5	45	282.52	285.80	251.84	255.24	319.90	323.07

Table A1. Cont.

Parameters					LSTM Results (MAE)		GRU Results (MAE)		RNN Results (MAE)	
Layers	Nodes	Dropout	Lookback	Epochs	Validation	Test	Validation	Test	Validation	Test
1	128	0.15	8	36	221.79	224.75	237.35	238.77	283.90	287.53
1	64	0.15	8	44	236.47	241.07	222.39	221.82	266.77	269.20
1	32	0.15	8	58	341.48	344.46	321.46	324.81	312.57	314.78
1	128	0.2	8	41	231.43	234.45	230.55	234.77	312.21	314.50
1	64	0.2	8	45	277.18	280.36	296.31	298.64	260.88	265.14
1	32	0.2	8	51	273.86	282.31	218.46	221.81	271.51	274.36
2	64	0.15	8	46	396.32	398.83	302.08	304.98	252.26	259.75
2	64	0.2	8	46	280.71	287.92	223.66	226.05	317.17	322.49
2	32	0.15	8	52	313.87	315.12	298.51	304.36	324.10	328.04
2	32	0.2	8	52	335.47	339.83	316.80	319.72	378.45	379.64
3	64	0.15	8	38	251.34	260.03	233.01	238.25	242.35	248.64
3	64	0.2	8	38	224.88	229.52	250.79	256.59	345.59	348.20
3	32	0.15	8	43	323.59	332.42	271.38	276.53	349.90	353.07
3	32	0.2	8	43	292.82	297.20	260.27	261.11	345.96	355.47

The hyperparameter selection process for Scenario 2 is detailed in Table A2, presenting RNN-GRU's superior performance over the other two models. Afterward, a range of parameters was tested for all three models to determine the optimal configuration. The resulting validation and test MAE values for each parameter set are presented in the table below.

Table A2. Parameter variation and corresponding test and validation MAE outcomes for Scenario 2.

Parameters					LSTM Results (MAE)		GRU Results (MAE)		RNN Results (MAE)	
Layers	Nodes	Dropout	Lookback	Epochs	Validation	Test	Validation	Test	Validation	Test
1	128	0.15	5	57	189.87	191.57	203.99	207.72	220.42	224.95
1	64	0.15	5	61	213.33	218.92	284.79	287.91	236.14	240.41
1	32	0.15	5	65	241.23	238.99	251.52	257.98	312.71	323.07
1	128	0.2	5	53	225.65	229.09	221.70	223.70	276.22	281.91
1	64	0.2	5	62	287.76	293.08	333.61	336.98	248.34	250.96
1	32	0.2	5	65	220.51	216.69	290.29	295.29	346.90	354.58
2	64	0.15	5	55	214.95	217.23	223.48	227.45	246.46	249.90
2	64	0.2	5	55	245.09	248.09	264.46	269.72	306.87	310.94
2	32	0.15	5	63	255.89	259.66	237.50	239.69	275.32	277.24
2	32	0.2	5	63	238.06	240.21	241.29	245.60	260.64	263.52
3	64	0.15	5	48	290.62	297.21	262.65	270.85	395.87	397.02
3	64	0.2	5	48	280.20	286.55	262.36	261.49	361.92	366.07
3	32	0.15	5	61	304.10	307.83	365.75	370.82	373.78	377.91
3	32	0.2	5	65	323.02	330.16	319.81	312.70	259.10	261.19
1	128	0.15	8	59	212.71	213.52	225.88	228.92	275.67	278.95
1	64	0.15	8	61	241.63	244.78	179.00	181.63	219.30	223.10
1	32	0.15	8	65	288.75	294.49	315.01	319.41	343.23	346.75
1	128	0.2	8	63	217.95	219.11	189.67	192.42	193.48	198.24
1	64	0.2	8	58	254.38	256.38	264.02	267.69	260.34	263.24
1	32	0.2	8	58	272.24	276.64	290.25	293.63	332.26	335.71

Table A2. Cont.

Parameters					LSTM Results (MAE)		GRU Results (MAE)		RNN Results (MAE)	
Layers	Nodes	Dropout	Lookback	Epochs	Validation	Test	Validation	Test	Validation	Test
2	64	0.15	8	51	212.18	215.14	205.18	209.34	268.26	273.27
2	64	0.2	8	51	264.30	268.52	215.45	220.15	195.11	201.35
2	32	0.15	8	67	279.85	283.45	257.17	261.18	306.63	308.64
2	32	0.2	8	62	251.73	253.98	225.57	229.04	264.94	267.17
3	64	0.15	8	53	255.81	267.71	238.21	245.37	289.13	291.02
3	64	0.2	8	53	305.22	307.92	313.39	315.07	392.00	396.18
3	32	0.15	8	62	359.86	362.78	324.24	328.15	388.14	392.85
3	32	0.2	8	60	202.47	208.63	194.69	197.93	288.14	292.85

The hyperparameter selection process for Scenario 3 is detailed in Table A3, presenting RNN-LSTM's superior performance over the other two models. Afterward, a range of parameters was tested for all three models to determine the optimal configuration. The resulting validation and test MAE values for each parameter set are presented in the table below.

Table A3. Parameter variation and corresponding test and validation MAE outcomes for Scenario 3.

Parameters					LSTM Results (MAE)		GRU Results (MAE)		RNN Results (MAE)	
Layers	Nodes	Dropout	Lookback	Epochs	Validation	Test	Validation	Test	Validation	Test
1	128	0	5	60	249.54	250.75	294.52	297.52	350.44	358.48
1	64	0	5	86	354.42	358.20	305.87	312.25	247.93	255.79
1	32	0	5	75	331.27	334.13	329.71	334.60	248.94	253.74
1	128	0.1	5	69	314.61	322.62	305.65	310.33	230.57	236.57
1	64	0.1	5	85	305.86	309.67	326.13	329.11	264.66	272.89
1	32	0.1	5	70	361.40	368.75	333.36	339.7	317.98	324.56
2	64	0	5	81	342.71	348.37	308.08	311.54	297.63	300.16
2	64	0.1	5	94	314.09	326.60	300.23	308.09	271.50	275.97
2	32	0	5	88	300.57	302.18	263.75	269.30	278.90	283.16
2	32	0.1	5	85	308.54	311.42	281.57	288.34	298.12	302.87
3	64	0	5	68	307.34	311.83	251.97	254.13	309.38	314.75
3	64	0.1	5	68	301.68	304.02	228.18	232.18	290.03	294.37
3	32	0	5	72	298.91	300.49	298.03	305.90	265.13	271.21
3	32	0.1	5	75	270.49	283.11	236.60	242.74	239.19	243.02
1	128	0	8	78	301.39	304.55	282.2	287.95	290.72	301.53
1	64	0	8	75	326.11	332.03	375.61	379.60	350.59	357.21
1	32	0	8	89	303.11	310.78	308.17	314.60	255.50	260.85
1	128	0.1	8	72	222.07	226.76	262.22	267.97	236.05	239.25
1	64	0.1	8	75	260.72	265.18	322.06	325.50	298.49	305.12
1	32	0.1	8	81	305.12	307.46	349.20	352.27	296.19	302.84
2	64	0	8	81	259.75	267.00	229.89	231.00	321.70	323.51
2	64	0.1	8	81	287.52	288.77	267.16	272.22	319.54	322.49
2	32	0	8	87	258.13	263.77	255.08	261.69	280.17	283.86
2	32	0.1	8	79	310.64	312.41	313.43	317.52	253.92	258.11
3	64	0	8	68	271.35	279.64	288.21	293.45	310.36	314.98
3	64	0.1	8	70	370.90	373.40	360.51	362.58	374.57	376.00
3	32	0	8	59	295.83	304.04	319.13	323.66	307.67	312.24
3	32	0.1	8	59	368.19	380.55	360.07	361.88	340.03	343.33

References

1. Ramos, P.V.B.; Villela, S.M.; Silva, W.N.; Dias, B.H. Residential energy consumption forecasting using deep learning models. *Appl. Energy* **2023**, *350*, 121705. [\[CrossRef\]](#)
2. Chapagain, K.; Kittipiyakul, S.; Kulthanavit, P. Short-term electricity demand forecasting: Impact analysis of temperature for Thailand. *Energies* **2020**, *13*, 2498. [\[CrossRef\]](#)
3. Koprinska, I.; Rana, M.; Agelidis, V.G. Correlation and instance based feature selection for electricity load forecasting. *Knowl.-Based Syst.* **2015**, *82*, 29–40. [\[CrossRef\]](#)
4. Shin, S.M.; Rasheed, A.; Kil-Heum, P.; Veluvolu, K.C. Fast and Accurate Short-Term Load Forecasting with a Hybrid Model. *Electronics* **2024**, *13*, 1079. [\[CrossRef\]](#)
5. Nti, I.K.; Teimeh, M.; Nyarko-Boateng, O.; Adekoya, A.F. Electricity load forecasting: A systematic review. *J. Electr. Syst. Inf. Technol.* **2020**, *7*, 13. [\[CrossRef\]](#)
6. Ding, Y.; Zhu, Y.; Feng, J.; Zhang, P.; Cheng, Z. Interpretable spatio-temporal attention LSTM model for flood forecasting. *Neurocomputing* **2020**, *403*, 348–359. [\[CrossRef\]](#)
7. Li, L.-L.; Zhao, X.; Tseng, M.-L.; Tan, R.R. Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. *J. Clean. Prod.* **2020**, *242*, 118447. [\[CrossRef\]](#)
8. Carvalho, T.P.; Soares, F.A.; Vita, R.; Francisco, R.d.P.; Basto, J.P.; Alcalá, S.G. A systematic literature review of machine learning methods applied to predictive maintenance. *Comput. Ind. Eng.* **2019**, *137*, 106024. [\[CrossRef\]](#)
9. Chapagain, K.; Kittipiyakul, S. Performance Analysis of Short-Term Electricity Demand with Atmospheric Variables. *Energies* **2018**, *11*, 818. [\[CrossRef\]](#)
10. Lusi, P.; Khalilpour, K.R.; Andrew, L.; Liebman, A. Short-term residential load forecasting: Impact of calendar effects and forecast granularity. *Appl. Energy* **2017**, *205*, 654–669. [\[CrossRef\]](#)
11. Papalexopoulos, A.D.; Hesterberg, T.C. A regression-based approach to short-term system load forecasting. *IEEE Trans. Power Syst.* **1990**, *5*, 1535–1547. [\[CrossRef\]](#)
12. Hammad, M.A.; Jereb, B.; Rosi, B.; Dragan, D. Methods and models for electric load forecasting: A comprehensive review. *Logist. Sustain. Transp.* **2020**, *11*, 51–76. [\[CrossRef\]](#)
13. Zuo, C.; Wang, J.; Liu, M.; Deng, S.; Wang, Q. An ensemble framework for short-term load forecasting based on timesnet and tcn. *Energies* **2023**, *16*, 5330. [\[CrossRef\]](#)
14. Waheed, W.; Xu, Q. Optimal short term power load forecasting algorithm by using improved artificial intelligence technique. In Proceedings of the 2020 2nd International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 13–15 October 2020; pp. 1–4.
15. Akhtar, S.; Shahzad, S.; Zaheer, A.; Ullah, H.S.; Kilic, H.; Gono, R.; Jasiński, M.; Leonowicz, Z. Short-term load forecasting models: A review of challenges, progress, and the road ahead. *Energies* **2023**, *16*, 4060. [\[CrossRef\]](#)
16. Son, N. Comparison of the deep learning performance for short-term power load forecasting. *Sustainability* **2021**, *13*, 12493. [\[CrossRef\]](#)
17. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Fiot, J.-B.; Dinuzzo, F. Electricity demand forecasting by multi-task learning. *IEEE Trans. Smart Grid* **2016**, *9*, 544–551. [\[CrossRef\]](#)
19. Dudek, G. Pattern-based local linear regression models for short-term load forecasting. *Electr. Power Syst. Res.* **2016**, *130*, 139–147. [\[CrossRef\]](#)
20. Ismail, Z.; Jamaluddin, F.; Jamaludin, F. Time series regression model for forecasting Malaysian electricity load demand. *Asian J. Math. Stat.* **2008**, *1*, 139–149. [\[CrossRef\]](#)
21. Zhang, G.; Guo, J. A novel method for hourly electricity demand forecasting. *IEEE Trans. Power Syst.* **2019**, *35*, 1351–1363. [\[CrossRef\]](#)
22. Khwaja, A.; Zhang, X.; Anpalagan, A.; Venkatesh, B. Boosted neural networks for improved short-term electric load forecasting. *Electr. Power Syst. Res.* **2017**, *143*, 431–437. [\[CrossRef\]](#)
23. Khwaja, A.S.; Anpalagan, A.; Naeem, M.; Venkatesh, B. Joint bagged-boosted artificial neural networks: Using ensemble machine learning to improve short-term electricity load forecasting. *Electr. Power Syst. Res.* **2020**, *179*, 106080. [\[CrossRef\]](#)
24. Bouktif, S.; Fiaz, A.; Ouni, A.; Serhani, M.A. Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting. *Energies* **2020**, *13*, 391. [\[CrossRef\]](#)
25. Bouktif, S.; Fiaz, A.; Ouni, A.; Serhani, M.A. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies* **2018**, *11*, 1636. [\[CrossRef\]](#)
26. Chen, K.; Chen, K.; Wang, Q.; He, Z.; Hu, J.; He, J. Short-term load forecasting with deep residual networks. *IEEE Trans. Smart Grid* **2018**, *10*, 3943–3952. [\[CrossRef\]](#)
27. Din, G.M.U.; Marnerides, A.K. Short term power load forecasting using deep neural networks. In Proceedings of the 2017 International Conference on Computing, Networking and Communications (ICNC), Silicon Valley, CA, USA, 26–29 January 2017; pp. 594–598.
28. Li, Y.; Pizer, W.A.; Wu, L. Climate change and residential electricity consumption in the Yangtze River Delta, China. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 472–477. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Stošović, M.A.; Radivojević, N.; Ivanova, M. Electricity Consumption Prediction in an Electronic System Using Artificial Neural Networks. *Electronics* **2022**, *11*, 3506. [\[CrossRef\]](#)

30. Xiao, L.; Wang, J.; Yang, X.; Xiao, L. A hybrid model based on data preprocessing for electrical power forecasting. *International J. Electr. Power Energy Syst.* **2015**, *64*, 311–327. [\[CrossRef\]](#)
31. Phuangpornpitak, N.; Prommee, W. A study of load demand forecasting models in electric power system operation and planning. *GMSARN Int. J.* **2016**, *10*, 19–24.
32. Gao, J.; Chen, Y.; Hu, W.; Zhang, D. An adaptive deep-learning load forecasting framework by integrating transformer and domain knowledge. *Adv. Appl. Energy* **2023**, *10*, 100142. [\[CrossRef\]](#)
33. Fotopoulou, M.C.; Drosatos, P.; Petridis, S.; Rakopoulos, D.; Stergiopoulos, F.; Nikolopoulos, N. Model predictive control for the energy Management in a District of buildings equipped with building integrated photovoltaic systems and batteries. *Energies* **2021**, *14*, 3369. [\[CrossRef\]](#)
34. Hesselmann, F.; Zhu, C.C.; Köppen-Seliger, B.; Ding, S.X. Model Predictive Control Based Energy Flow Control in the Smart Grid. In Proceedings of the 15th European Workshop on Advanced Control and Diagnosis, Bologna, Italy, 21–22 November 2019; pp. 657–677.
35. Chapagain, K.; Kittipiyakul, S. Short-term electricity load forecasting for Thailand. In Proceedings of the 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Rai, Thailand, 18–21 July 2018; pp. 521–524.
36. Dilhani, M.S.; Jeenanunta, C. Daily electric load forecasting: Case of Thailand. In Proceedings of the 2016 7th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES), Bangkok, Thailand, 20–22 March 2016; pp. 25–29.
37. Chapagain, K.; Kittipiyakul, S. Short-term electricity demand forecasting with seasonal and interactions of variables for thailand. In Proceedings of the 2018 International Electrical Engineering Congress (iEECON), Krabi, Thailand, 7–9 March 2018; pp. 1–4.
38. Jeenanunta, C.; Abeyrathn, K.D. Combine particle swarm optimization with artificial neural networks for short-term load forecasting. *Int. Sci. J. Eng. Technol. (ISJET)* **2017**, *1*, 25–30.
39. Chapagain, K.; Kittipiyakul, S. Short-term electricity load forecasting model and Bayesian estimation for Thailand data. In Proceedings of the MATEC Web of Conferences 2016 Asia Conference on Power and Electrical Engineering (ACPEE 2016), Bangkok, Thailand, 20–22 March 2016; p. 06003.
40. Chapagain, K.; Soto, T.; Kittipiyakul, S. Improvement of performance of short term electricity demand model with meteorological parameters. *Kathford J. Eng. Manag.* **2018**, *1*, 15–22. [\[CrossRef\]](#)
41. Sankalpa, C.; Kittipiyakul, S.; Laitrakun, S. Forecasting Short-Term Electricity Load Using Validated Ensemble Learning. *Energies* **2022**, *15*, 8567. [\[CrossRef\]](#)
42. Abeyrathna, K.D.; Jeenanunta, C. Hybrid particle swarm optimization with genetic algorithm to train artificial neural networks for short-term load forecasting. *Int. J. Swarm Intell. Res. (IJSIR)* **2019**, *10*, 1–14. [\[CrossRef\]](#)
43. Alotaibi, M.A. Machine learning approach for short-term load forecasting using deep neural network. *Energies* **2022**, *15*, 6261. [\[CrossRef\]](#)
44. Williams, R.J.; Zipser, D. A learning algorithm for continually running fully recurrent neural networks. *Neural Comput.* **1989**, *1*, 270–280. [\[CrossRef\]](#)
45. Salem, F.M. A basic recurrent neural network model. *arXiv* **2016**, arXiv:1612.09022.
46. Shi, H.; Xu, M.; Li, R. Deep learning for household load forecasting—A novel pooling deep RNN. *IEEE Trans. Smart Grid* **2017**, *9*, 5271–5280. [\[CrossRef\]](#)
47. Arslan, S. A hybrid forecasting model using LSTM and Prophet for energy consumption with decomposition of time series data. *PeerJ Comput. Sci.* **2022**, *8*, e1001. [\[CrossRef\]](#) [\[PubMed\]](#)
48. Dhake, H.; Kashyap, Y.; Kosmopoulos, P. Algorithms for hyperparameter tuning of lstms for time series forecasting. *Remote Sens.* **2023**, *15*, 2076. [\[CrossRef\]](#)
49. Zheng, H.; Yuan, J.; Chen, L. Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation. *Energies* **2017**, *10*, 1168. [\[CrossRef\]](#)
50. Tang, H.; Ling, X.; Li, L.; Xiong, L.; Yao, Y.; Huang, X. One-shot pruning of gated recurrent unit neural network by sensitivity for time-series prediction. *Neurocomputing* **2022**, *512*, 15–24. [\[CrossRef\]](#)
51. Kumar, G.; Singh, U.P.; Jain, S. An adaptive particle swarm optimization-based hybrid long short-term memory model for stock price time series forecasting. *Soft Comput.* **2022**, *26*, 12115–12135. [\[CrossRef\]](#) [\[PubMed\]](#)
52. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
53. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
54. Li, W.; Logenthiran, T.; Woo, W.L. Multi-GRU prediction system for electricity generation’s planning and operation. *IET Gener. Transm. Distrib.* **2019**, *13*, 1630–1637. [\[CrossRef\]](#)
55. Lynn, H.M.; Pan, S.B.; Kim, P. A deep bidirectional GRU network model for biometric electrocardiogram classification based on recurrent neural networks. *IEEE Access* **2019**, *7*, 145395–145405. [\[CrossRef\]](#)
56. Shiwakoti, R.K.; Charoenlarnpoppa, C.; Chapagain, K. Time Series Analysis of Electricity Demand Forecasting Using Seasonal ARIMA and an Exponential Smoothing Model. In Proceedings of the 2023 International Conference on Power and Renewable Energy Engineering (PREE), Tokyo, Japan, 20–22 October 2023; pp. 131–137.

57. Chungloo, S.; Limmeechokchai, B. Application of passive cooling systems in the hot and humid climate: The case study of solar chimney and wetted roof in Thailand. *Build. Environ.* **2007**, *42*, 3341–3351. [[CrossRef](#)]
58. Nguyen, C.T.; Nguyen, D.T.H.; Phan, D.K. Factors affecting urban electricity consumption: A case study in the Bangkok Metropolitan Area using an integrated approach of earth observation data and data analysis. *Environ. Sci. Pollut. Res.* **2021**, *28*, 12056–12066. [[CrossRef](#)]
59. Chapagain, K.; Gurung, S.; Kulthanavit, P.; Kittipiyakul, S. Short-term electricity demand forecasting using deep neural networks: An analysis for Thai data. *Appl. Syst. Innov.* **2023**, *6*, 100. [[CrossRef](#)]
60. Ramanathan, R.; Engle, R.; Granger, C.W.; Vahid-Araghi, F.; Brace, C. Short-run forecasts of electricity loads and peaks. *Int. J. Forecast.* **1997**, *13*, 161–174. [[CrossRef](#)]
61. Wang, S.; Ma, C.; Xu, Y.; Wang, J.; Wu, W. A hyperparameter optimization algorithm for the LSTM temperature prediction model in data center. *Sci. Program.* **2022**, *2022*, 6519909. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.