IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 7, 2021, accepted November 5, 2021, date of publication November 10, 2021,
date of current version November 18, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3127211

Daily Load Forecasting Based on a Combination of Classification and Regression Tree and Deep Belief Network

PYAE PYAE PHYO AND CHAWALIT JEENANUNTA
School of Management Technology, Sirindhorn International Institute of Technology, Thammasat University, Bangkok 10200, Thailand
Corresponding author: Chawalit Jeenanunta (chawalit@siit.tu.ac.th)

This work was supported by the Center of Excellence in Logistics and Supply Chain Systems Engineering and Technology (COE LogEn), Sirindhorn International Institute of Technology (SIIT), Thammasat University (TU), Thailand.

ABSTRACT The next-day load forecasting is complex due to the load pattern variations driven by external factors, such as weather and time. This study proposes a hybrid model that incorporates the Classification and Regression Tree (CART) with pruning conditions and a Deep Belief Network (DBN) to improve forecasting accuracy. The CART can recognize the load patterns by classifying similar groups with low variance, thus reducing the complexity of the forecasting model. The actual 48-period load data from the Electricity Generating Authority of Thailand (EGAT) is used. The proposed model is compared with six widely used standalone forecasting benchmark models and provides better at the minimum 0.46% mean absolute percentage error. Moreover, the forecasting performance of DBN and the other four benchmark models are improved by using our hybrid approach.

INDEX TERMS Classification and regression tree (CART), daily load forecasting, deep belief network (DBN), forecasting accuracy, pruned-CART.

I. INTRODUCTION

Load forecasting plays a vital role in the planning and operation of electric generators as it requires meeting the generating units between supply and demand. It can be divided into short-term, medium-term, and long-term forecasting duration [1]. Forecasting electrical load for annual data is considered as long-term, and monthly forecasting is considered as medium-term. Daily load forecasting is called short-term load forecasting, ranges from hours to weeks. The accuracy of the forecasting model can reduce the operating cost as it could reduce the standby generators for spinning reverse. However, the reliability of the predicted results depends on the features of the input and the model fitting.

Daily load patterns depend upon myriad aspects that change incessantly due to weather effects, seasonal effects, calendar effects, and so on. Weather effects typically include temperature, sunshine, cloudy, and humidity, while seasonal effects comprise spring, summer, fall, and winter. Among these effects, the temperature variable plays a significant impact in examining electric load demand from the consumers [1]. Calendar effects vary from day to day, month to month, and on holidays. Thus, understanding the nonlinear relationship between load patterns and influential variables can enhance electric load forecasting performance. This paper proposes the classification and regression tree (CART) with pruning predefined conditions to classify the pattern of electrical data. The proposed classification depends on the calendar, such as day of the week, the month of the year, holidays, and bridge holidays.

A. RELATED WORKS

Pattern classification was introduced in the late 20^{th} century and later applied in power system load forecasting [2]–[4]. Load pattern clustering for electricity customers was conducted using the defined harmonic-based features grouped by fixing the minimum modified Euclidean distance. The primary purpose of their system is to store electric customers' big data into manufacturer databases in a classified manner. The scholars developed a clustering algorithm to store refined data according to the time-frequency domain. This technique reduced the data by selecting and storing essential features

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Tong.

in the database following the specific cluster. It used the harmonic order in terms of the time zone for clustering the electric customer's data [5]. In addition, Zhong and Tam proposed characteristic attributes in the frequency domain (CAFD), which helps to create a classification tree for load profiles. This CAFD classification method was experimented with different complex parameter values to finetune the number of final leaf nodes of the tree. The illustrated work is the hierarchal classification of load profiles according to the frequency domain [6].

Ferreira *et al.* evidenced that recognizing load patterns could improve the reliability of the electric power sector and support decision-making levels in management. In their research, the whole sample using specific features was clustered at the first iteration. Next, the similarity between medians of each group and the median at the first iteration are verified to cluster some patterns for further iterations [7]. Kotriwala *et al.* combined load patterns and power-zone attributes for clustering power-zones and used the $k-$means algorithm for classifying the load patterns. They used logistic regression (LR) and support vector machine (SVM) for classification and linear regression model for forecasting the power load. They revealed that load classification could optimize the system efficiency of power generation planning [8]. Yin *et al.* proposed the deep forest regression, wherein random forest with Gini index parameter was applied both for classification and regression to improve the forecasting accuracy of the power system. Besides, the random forest (RF) algorithm randomly chooses the sample data and features randomly, which causes sample duplication and a lack of essential features [9].

Classification is one of the significant components of data mining techniques. The prime task is to analyze a set of provided data and generate some essential rules, which can be used to classify future data. There exist many applicable classification models, which include the k-means algorithm, SVM, naive Bayesian classifier, artificial neural network (ANN), and decision tree (DT) [10]. These models are generally helpful for both classification and regression problems. The k-mean algorithm was used for clustering hourly load data to improve the functional linear forecasting model. It optimized the number of clusters, denoted as G, by computing the regression model's determination coefficient and the G-dimensional function. Their approach used a fixed number of groups for classification and used the functional regression technique for forecasting. The method is complex and only works with a parametric system such as parametric regression models [11].

Moreover, load patterns were clustered using a threshold value of different average loads (30MW and 60MW) concerning day types between train and test data. Then the average load is predicted by using the SVM model. However, they used manual classification in terms of day type [12]. Zhang *et al.* combined two models where the CART was used with Gini index parameter to build tree analysis, and the SVM model was used for forecasting one-week test data. They have performed short-term load forecasting (STLF) on smart meter data using big data analytics technologies. However, there was an overfitting problem during the training process of the machine learning (ML) algorithm [13]. Besides, Moon *et al.* proposed a hybrid model for STLF by combining RF and multilayer perceptron (MLP). The RF model selects a subset of variables from the whole original set. To form the optimal tree, this model adjusts the total number of trees to be generated (nTree) as a minimum split and the decision tree-related parameters (mTry) as a split criterion, respectively. Nevertheless, they have tried various ML models with different configurations, but their error accuracies are still high [14].

Many types of research on classifiers or clusters for electricity load forecasting have been investigated since the last decade. Among popular classification models, DT is an up-and-coming model. Many tree-based algorithms such as Iterative Dichotomiser 3 (ID3), C4.5 algorithm, RF, and CART are used to create the DT [15]. Srivastava *et al.* conducted STLF using the DT algorithms, for instance, RF, Bagging, and M5P tree algorithms. The results are compared for each algorithm. In their research, the bagging model trained a DT from each bootstrap training data sample and selected the final bagged model with a minimum mean for prediction. Nonetheless, they applied all models with default configuration for regression performance [16]. Loh also reviewed some comprehensive classification tree algorithms for classification and regression algorithms for prediction, comparing their strengths and weaknesses [17]. Among the above-discussed algorithms, CART is considered one of the most popular models, which can efficiently deal with the problems of both classification and regression [4].

Mori and Kosemura proposed optimizing the optimal CART by the tabu search algorithm at each split node and pruning it by computing the cross-validation and standard errors [18]. Similarly, Mori *et al.* used the simplified fuzzy inference to find the efficient rules from actual data for CART and predicted one-step load by MLP. Their CART model classified the data into two terminal nodes, which can cause impurity of load profile [19]. Hambali *et al.* worked with three decision tree algorithms for forecasting electrical power load: CART, reduced error pruning tree (REPTree), and decision stump (DS). CART and REPTree used the Gini index and regression logic tree in the cited work, while DS measures the dataset's entropy according to each attribute. On the other side, these tree models were applied with the purpose of regression [20]. Besides, Lei *et al.* combined the $k-$means algorithm for clustering, whereas RF and CART for regression. Lei *et al.* implement these algorithms on Apache Spark machine learning library (MLlib). However, the data processing ability of the model needs to be more optimized to improve the forecasting performance [21]. Additionally, Guo used the CART to classify load patterns and the "If…Then" structure to create pattern rules. The author measured the misclassification risk and complexity to prune the CART. The simple ANN model has been used for load forecasting, and it requires more computation power and time for training [22].

Many researchers have cited myriad advantages of tree pruning since the late 1990s. In this regard, Shah and Sastry proposed Alopex, learning automata (LA), and generative algorithm (GA) for pruning the tree handling on two classification problems [23]. Alternatively, pruning the CART algorithm could help the penalized regression model selection by proving the risk bounds [24]. Guo and Niu proposed the CART pruning by calculating the cost-complexity measure, which chooses the minimal sum of squared residual with the complexity parameter to get the final pruned tree. Likewise, the cost complexity function is used for trimming CART of load pattern recognition that is clustered primarily using the fuzzy C-means clustering to increase the forecasting accuracy [25]. The feature selection and reduction can also be done by using a classification model with a predefined set of clusters to assign new electrical loads customers [26]. Hanif *et al.* proposed the optimal constrained pruning of decision trees, applied the proposed model to a large-scale transportation analysis and simulation system (TRANSIMS), and revealed the flexibility and effectiveness of the proposed approach. For these reasons, adding objective functions and constraints could optimally prune the original decision tree [27].

Based on the above-cited works, it is evident as the broad daylight that the forecasting performance can be improved by using an appropriate classification model. Therefore, the CART seems to be a promising approach for grouping the similarity of load profiles, improving the model's forecasting performance. The data preprocessing stage can be eliminated by using the CART algorithm. It can also reduce data imbalance issues. In addition, pruning the decision tree could reduce the size and the overfitting problem, which might slightly increase the training error while decreasing the testing error. As a result, the pruned tree is more efficient, accurate, and understandable than the original unpruned tree.

At the start of the 21$^{\text{st}}$ century, all neural network-based models have been successfully deployed in the field of load forecasting. These models can be divided into traditional ML and deep learning (DL). The popular ML forecasting models are regression trees, gradient boosting model [22], [28], bagged neural network (BNN) [29], ANN, and SVM [30]–[32]. It has also been noticed that the above-cited models applying different data performed better accuracy for STLF. Nevertheless, DL models have become more popular because of exceptional capabilities such as controlling non-linear relationships, model complexity, and computation time over ML models. Schmidhuber proposed the deep ANN, the learning of unsupervised and reinforcement, and evolutionary computation to clarify the use of DL models [33]. Recurrent neural network (RNN) [34], long short-term memory (LSTM) [35], convolutional neural network (CNN) [36], deep neural network (DNN) [37], and deep belief network (DBN) [38] have been investigated in STLF. The forecasting performance of these DL models was satisfactory by comparing them with traditional statistical time series and ML models based on different case studies.

Furthermore, many researchers performed STLF using hybrid models that combine two or more models. The hybrid approach appears to be a viable choice based on the input structure, selected method, and application. Kouhi applied a cascaded neural network that consists of an intelligent two-stage feature selection with the forecasted engine of three cascaded neural network structures [39]. In the same way, GA operations in particle swarm optimization (PSO) and Bayesian optimization (BO) were applied to achieve better forecasting accuracy for ANN and SVM, respectively. However, their proposed ML model costs high computation time [40], [41].

As mentioned above, the hybrid models provide promising results compared to the standalone models in terms of accuracy in most studies. However, these approaches still required manual data preprocessing. In our work, we use CART to group the similarity of load patterns and the variance reduction. Therefore, there is no need for data preprocessing and handling data imbalance issues. Moreover, the CART can automatically manage the variable selection, missing values, outliers, variable interaction, and non-linear relationships against other tree-based models. The additional benefit of our approach is using pruning conditions that reduce overfitting and provide a balanced training set for the forecasting model. Our forecasting DBN model has several advantages: the vanishing gradient descent problem, the pertained and finetuning process, and less computation time than other NNs. Therefore, it also can learn complex features from the information. The summary table of classification models on electricity data is reviewed in the Appendix.

B. CONTRIBUTIONS

A mixture of classification and regression models is proposed to predict short-term electrical loads for the first time. The historical electrical load data is classified using the CART model, which uses mean squared error (MSE) criteria to group similar load data with low variance. Because of this, the proposed hybrid model omits the usage of the data preprocessing module, which is compulsory in conventional electrical load forecasting models. Predefined conditions are created to prune the original CART because of including many irregular load patterns on special days, such as holidays or the day before or after holidays, separately split by CART. This issue could impact training the forecasting model. Keeping in view this issue, we pruned the original CART with predefined algorithms. A different DBN forecasting model for each terminal node of CART is also built to train and test input data. The forecasting performance of the DBN model is better than benchmark models, which use manual classification (MC) data based on day of the week (DoW) as the input load data has almost the same variance each other. Finally, an extensive comparison is carried out between our proposed hybrid model and other baseline models to prove

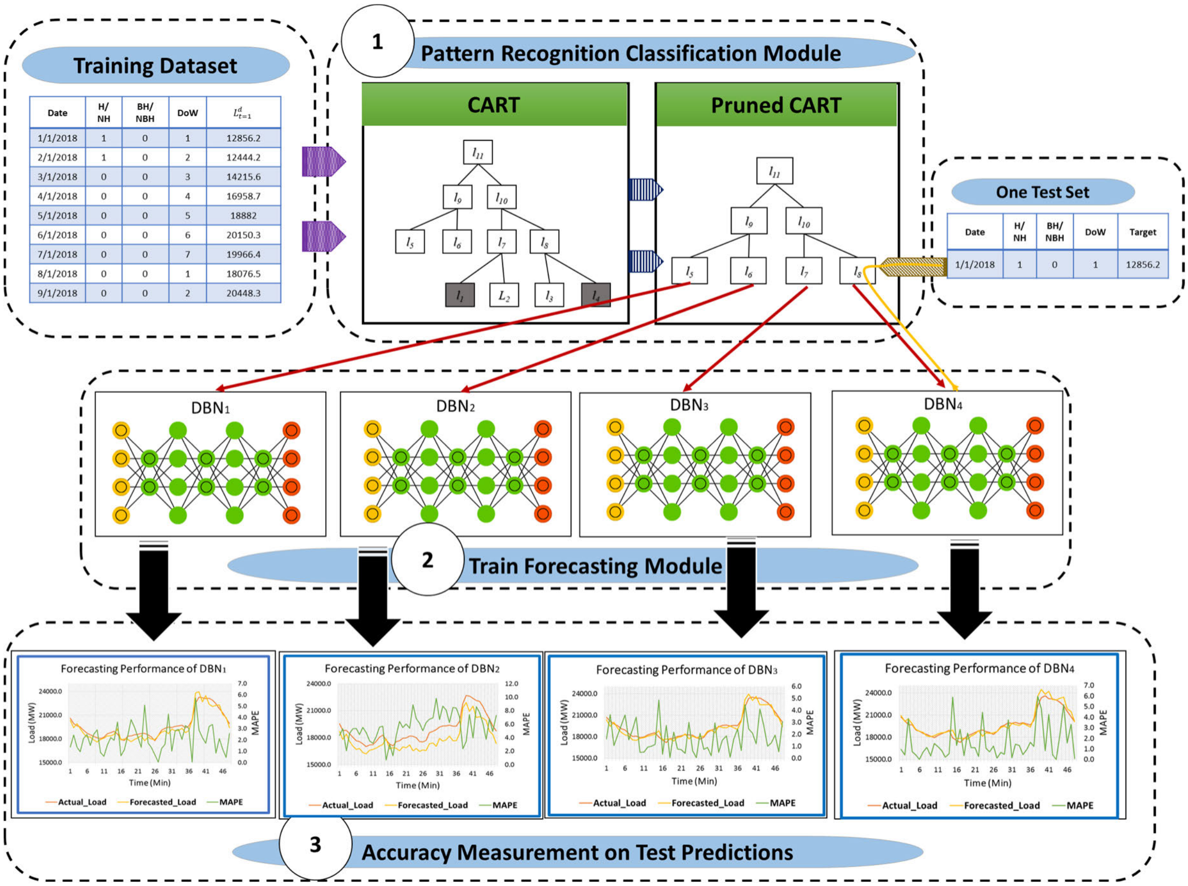

FIGURE 1. The framework of the proposed system.

the validity of our proposed hybrid electrical load forecasting model.

C. PAPER ORGANIZATION

The structure of the article is organized using the following sections. Section II describes the system modeling of the proposed hybrid model. Section III gives the details for the design of experiments. The outcomes of the experiments with comparative analysis are also provided in Section III. The conclusion of this manuscript is provided in Section IV.

II. SYSTEM MODELING

This section mainly describes the whole system of the proposed hybrid model, including a brief description of both CART and DBN models. Following are three steps of the proposed framework, including mainly the classification module, the training forecasting module, and the accuracy measurement using test data, as exhibited in Figure 1. The process of CART, pruned-CART and DBN regression models are further explained in different subsections.

- At first, the classification module arranges the training and testing datasets of independent and dependent variables based on case studies. Next, training data is provided to the classification module to build the training CART. The original CART is executed with pure homogenous leaf nodes of target load data by recognizing independent variables. The original CART is pruned based on predefined constraints. The process of pruned-CART continues until all leaf nodes satisfy the requirements. After creating the final pruned-CART, a test set, including both independent and dependent variables, is given into this CART. The test set is then fallen into the respective terminal node based on its independent variables.
- The second train forecasting module builds different DBN regression models regarding each terminal node generated by the pruned-CART. Afterward, the training set for each forecasting model is trained along with adjusted parameters.
- Finally, predictions of each DBN trained model on the whole test set are performed, and error metrics are calculated on test data to measure the forecasting accuracy.

A. CLASSIFICATION AND REGRESSION TREE (CART)

CART is one of the DT models that build either regression or a classification model in the form of a tree structure [42]. An example of CART in the framework is shown in Figure 2, where CART is a pair, $T = (N, E)$, in which N is a set of

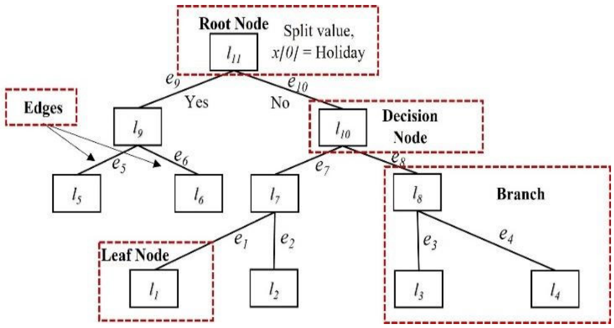

FIGURE 2. An example of CART with six leaf nodes and five decision nodes.

nodes and E are the edges. Suppose that there is a sample CART as shown in Figure 1, where $\{l_1, l_2, \ldots, l_{11}\}$ are the nodes and $\{e_1, e_2, \ldots, e_{10}\}$ are the edges of CART. The leaf nodes are represented by l_1, l_2, l_3, l_4, l_5 and l_6, whereas an edge is between two leaf nodes, for example, $e_1 = (l_7, l_1)$ is an edge going from l_7 to l_1. CART depends on independent variables to split a node into two or more sub-nodes by minimizing the total data variance in sub-nodes.

The minimum calculation of the criteria value of all variables is determined to choose the best split variable. In our case, the reduction in variance is fitted for the continuous data type. Therefore, MSE is minimized to group the similar load patterns along with the feature selection, which is expressed mathematically as,

$$|MSE\ D_l| = MSE\ d_1^l + \ldots + MSE\ d_j^l \tag{1}$$

where D_l be the set of load demand vector in node l, $D_l = \bar{d}_{k_1}^l, \bar{d}_{k_2}^l, \ldots, \bar{d}_{k_i}^l, \bar{d}_{k_i}^l$ is the load demand vector of day k_i in node l. Let d_k be the vector of load demand d on day k, $\bar{d}_k = \{d_{k,1}, d_{k,2}, \ldots, d_{k,48}\}$ where d_{kj} is the load demand on the day at k period j, Day_l be the set of the day in node l, $Day_l = \{k_1, \ldots, k_i\}$, $AVG\ D_l$ be the set of the average load demand for each period, $AVG\ D_l = \{AVG\ d_1^l, \ldots, AVG\ d_{48}^l\}$, where $AVG\ d_i^l$ is the average load demand at period i, $MSE\ D_l = \{MSE\ d_1^l, \ldots, MSE\ d_{48}^l\}$ be the set of mean squared error of load demand for each period. The $MSE\ d_j^l$ of demand at period j in node l is given as,

$$MSE\ d_j^l = \frac{\sum_{k \in Day_l} \left(AVG\ d_j^l - d_{k,j}^l\right)^2}{|D_l|} \tag{2}$$

B. PROCESS OF PRUNED-CART

Pruning methods aim to generally solve overfitting problems and reduce the size of the tree, which might slightly increase the training error while decreasing the testing error. Moreover, the pruned tree is more efficient, accurate, and understandable than the original unpruned tree. Hence, the pruned-CART is proposed to solve the overfitting for training the forecasting model. The original CART is recursively checked on each leaf to see whether it satisfies the predefined constraints. If the leaf node is not associated with them, it is pruned or deleted from the tree. A sample-size-constraint

Algorithm 1 Pruned-Cart Algorithm

Let $e_i = (l_j, l_i)$ be an edge going from node l_j to l_i, $\alpha(e_i) = l_j$ be a parent node, $\beta(e_i) = l_i$ be a child node.
INPUT: (i) The set of nodes of CART, N $= \{l_1, \ldots, l_m, l_{m+1}, \ldots, l_n\}$
(ii) The set of edges of CART, E $= \{e_1, \ldots, e_m, e_{m+1}, \ldots, e_{n-1}\}$, (i.e., $e_i = (l_j, l_i)$)
(iii) The set of leaf nodes of CART, L $= \{l_1, \ldots, l_m\}$ (i.e., $L \subset N$)
(iv) The set of samples of nodes, S $= \{s_{l_1}, \ldots, s_{l_m}, s_{l_{m+1}}, \ldots, s_{l_n}\}$
OUTPUT: N, E, and L
1: **procedure** pruned-CART (N, E, L, S)
2: for each $l \in L$
3: if $s_l < mins_l$
4: for each $e \in E$
5: if $\beta(e) = l$ then $p \leftarrow \beta(e)$ end if
6: end for
7: $R \leftarrow \emptyset$ (i.e., R is a set of removed edges)
8: for each $e \in E$
9: if $\alpha(e) = p$ then $R \leftarrow R \setminus e$ end if
10: end for
11: for each $r \in R$
12: $E \leftarrow E \setminus \{r\}$
13: $L \leftarrow L \setminus \{\beta(r)\}$
14: $N \leftarrow N \setminus \{\beta(r)\}$
15: end for
16: if $|R| > 0$ then $L \leftarrow L \cup \{p\}$ end if
17: end for
18: return N, E, L
19: **end procedure**

Algorithm 2 Sample Size Constraint Algorithm

INPUT: (i) The set of leaf nodes of CART, L $= \{l_1, \ldots, l_m\}$ (i.e., $L \subset N$)
(ii) The set of samples of nodes, S $= \{s_{l_1}, \ldots, s_{l_m}, s_{l_{m+1}}, \ldots, s_{l_n}\}$
OUTPUT: a Boolean value: *flag*
1: **procedure** check-sample (L, S)
2: $flag \leftarrow True$
3: for each $l \in L$
4: if $s_l < s_l$ then $flag \leftarrow False$ end if
5: end for
6: **end procedure**

specifies the minimum number of samples for each leaf in the tree. In this paper, the optimum number of samples in each leaf should have enough depending on the trained DBN model. The step-by-step process of pruned-CART is in algorithms I, II, and III.

Suppose that we have a CART according to Figure 3(a), where grey boxes and yellow boxes represent leaf nodes with and without enough samples, correspondingly. Also, let $N = \{l_1, \ldots, l_{11}\}$, $E = \{e_1, \ldots, e_{10}\}$, (e.g., $e_1 = (l_7, l_1)$ is an edge going from l_7 to l_1), $L = \{l_1, \ldots, l_6\}$ (i.e., $L \subset N$) RT, and $S = \left\{s_{l_1}, \ldots, s_{l_{11}}\right\} = \{3, 4, 15, 4, 3, 15, 7, 19, 18, 26, 44\}$. For leaf node l_1, according to the third step of Algorithm I, $s_{l_1} = 3$ is less than 15, so that l_1 is removed from the tree and l_2 is automatically pruned based on the ninth step of the pruned-CART due to coming from the same parent node l_7. l_1 and l_2 are removed from both sets L and N. The edges

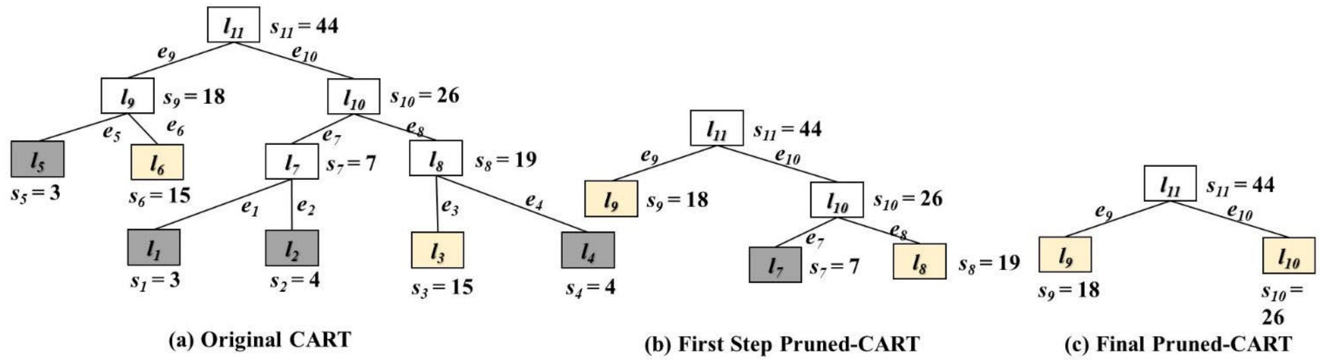

FIGURE 3. Example of the pruned-CART algorithm.

Algorithm 3 Recursive Pruned-Cart Algorithm

INPUT: (i) The set of nodes of CART, N $= \{l_1, \ldots, l_m, l_{m+1}, \ldots, l_n\}$
(ii) The set of edges of CART, E $= \{e_1, \ldots, e_m, e_{m+1}, \ldots, e_{n-1}\}$, (i.e., $e_i = (l_j, l_i)$)
(iii) The set of leaf nodes of CART, L $= \{l_1, \ldots, l_m\}$ (i.e., $L \subset N$)
(iv) The set of samples of nodes, S $= \{s_{l_1}, \ldots, s_{l_m}, s_{l_{m+1}}, \ldots, s_{l_n}\}$

procedure recursive-pruned-CART (N, E, L, S)
  while $l \in L$
    $N, E, L \leftarrow$ pruned-CART (N, E, L, S)
  end while
  if check-sample (L, S) = True end if
    break
end procedure

e_1 and e_2 are deleted from set E. Then, node l_7 becomes a leaf node of set L. Similarly, l_3 (resp. l_5) and l_4 (resp. l_6) are pruned because the sample of l_4 (resp. l_5) has less than 15, and they have the same parent. Therefore, l_8 (resp. l_9) become leaf nodes of set L and the edges e_3 (resp. e_5) and e_4 (resp. e_6) are deleted from set E.

After pruning, the first pruned-CART is obtained, as demonstrated in Figure 3(b), and it has five nodes and four edges. The pruned-CART with one-layer depth is recursively pruned by using recursive-pruned-CART, where the samples of all leaf nodes are checked using a check-sample. Ultimately, the pruned-CART is executed with three nodes and two edges, as shown in Figure 3(c). Two leaf nodes of pruned-CART have enough samples to train the forecasting model.

C. DEEP BELIEF NETWORK (DBN)

The DBN model was introduced by Hinton [43]. It has become popular in forecasting areas using real-world data such as sunspot data, exchange rate data, electricity load data, and generation data [43]. It is also combined with other traditional time series models to optimize the forecasting performance [44]. The DBN reconstructs the inputs with consideration to the probability and includes various learning modules with less complexity. It mainly consists of the restricted Boltzmann machine (RBM) using unsupervised learning to pre-train the network with each pair of layers. As depicted in Figure 4, the RBM model has a double-layer neural network containing a visible layer and a hidden layer with Boolean hidden units.

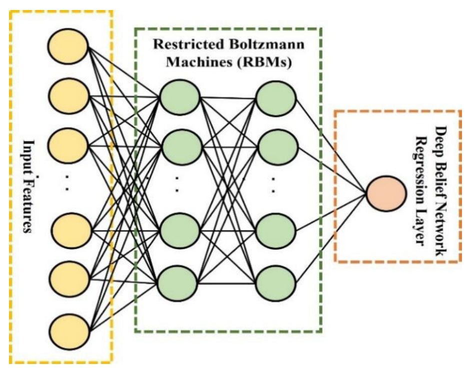

FIGURE 4. The structure of DBN Regression with RBMs.

This model is the pre-trained model with generative energy that can learn a probability distribution over a range of input sets. In addition, it has only a single hidden layer of hidden units as there are no interconnections within the same layer. Thus, it consists of symmetrical connections to a visible layer of units. However, it still requires the formation of a bipartite graph with its neurons. The learning process of layer-wise configuration and the probability distribution of the hidden layer [45] is written as,

$$EE(v,h) = -\sum\nolimits_{i=1}^{n_v} a_i v_i - \sum\nolimits_{j=1}^{n_h} b_j h_j - \sum\nolimits_{i=1}^{n_v} \sum\nolimits_{j=1}^{n_h} h_j w_{j,i} v_i \tag{3}$$

$$P(v,h) = \frac{e^{-E(v,h)}}{\sum_v \sum_h e^{-E(v,h)}} \tag{4}$$

where,

$v_i =$ the binary state of i^{th} neuron in the visible layer,
$n_v =$ the number of neurons in the visible layer,

h_j = the binary state of the j^{th} Boolean neuron within the hidden layer,

n_h = the number of neurons in the hidden layer,

$w_{j,i}$ = the weight matrix between the visible layer and the hidden layer,

a_i, b_j = bias vectors for the visible and hidden layers.

The activation functions of the visible and hidden layer are described as,

$$P(v_i = 1|h) = \sigma(a_i + \sum_{j=1}^{n_h} w_{j,i}h_j) \quad (5)$$

$$P(h_i = 1|v) = \sigma(b_j + \sum_{j=1}^{n_v} w_{j,i}v_i) \quad (6)$$

where σ refers to the sigmoid activation function [46].

The following steps describe how the DBN regression model predicts the future daily load.

- After achieving the pruned-CART process, different DBN models for each terminal node are created for training.
- Both training and testing datasets are normalized to rescale the original values of each feature between 0 and 1 by using the MinMax scaler function and then fed into the training model. The way how to normalize the data is indicated in Eq. 7, where x_i, $min(x)$, $max(x)$, and new x_i represent the original value of the input feature, the minimum value in input feature, the maximum value in input feature, and the new rescaled value of x_i.

$$new\ x_i = \frac{x_i - min(x)}{\max(x) - min(x)} \quad (7)$$

- The same training parameters are applied to train all DBN models of each terminal node of the pruned-CART in the training process.
- Testing data from Apr 2020 to Mar 2021 are predicted to be associated with each leaf node's respective trained $DBN_i(1 <= i <= m)$.
- Error metrics measure the forecasting performance on the test data.

III. DESIGN OF EXPERIMENTS, RESULTS, AND DISCUSSIONS

The section highlights the necessary steps involved in the design of experiments, and later, the discussions of experimental results are also provided.

A. EXPERIMENTAL DATA

Electrical data recording every thirty minutes per day is provided by the electricity generating authority of Thailand (EGAT). The net peak load data in megawatt (MW) from Apr 2018 to Mar 2021 is applied to train the proposed model. Firstly, due to some outliers for specific periods, the data is categorized using CART, which will improve the forecasting accuracy. Figure 5 illustrates load patterns for the first week in May 2018 to reveal the difference between weekdays and weekends. Among weekdays, there is one holiday, so-called International Workers' Day, on May 1. The load fluctuation on holiday reacts differently from other regular days.

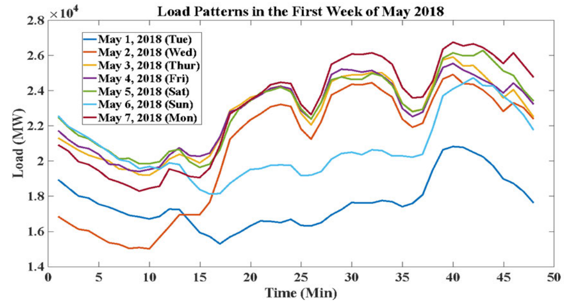

FIGURE 5. Different load patterns between weekdays and weekends in May 2018.

In general, the reaction of load patterns on Saturday is almost like weekdays. Nevertheless, the usage of load demand on Sunday is lower than other days, so the load pattern fluctuates inversely.

B. DATA SEGMENTATION

The training dataset from Apr 2018 to Mar 2020 and the testing dataset from Apr 2020 to Mar 2021 are divided for training and testing. The whole training set is applied to build the original CART for grouping similar load patterns with low variance. After that, leaf nodes of this CART that do not have enough training samples are pruned with predefined conditions. Then, one test set is started to feed into the model. For instance, the first leaf node from a tree is grouped only by Monday in the training set. If the first test set is Monday, then it is gone into the first leaf node. The following test set is provided and falls into the associated leaf node. The whole test set continues sliding, and the same procedure is conducted until the end of the classification process. It is called the walk-forward train, and the test routine is shown in Figure 6.

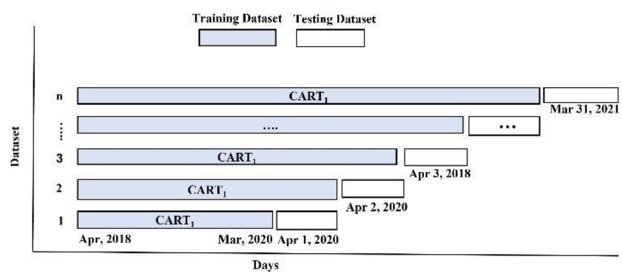

FIGURE 6. Walk-forward train and test routine of the CART with DBN.

C. DATA ARRANGEMENT

The input selection is one of the crucial factors in network modeling for forecasting purposes. Furthermore, the input variables are dominated, varying from day to day, week to week, or month to month. Consequently, the following independent and dependent variables are applied to train the CART and handle the non-linear relationship between input and output variables. Independent variables are holiday or

TABLE 1. Data arrangement of leaf node 1 for case I.

	Inputs					Target
	No. of day	YD_t(D-1)	Hol	B_Hol	DoW	$FD_t^4(d)$
Training Dataset	1	02/04/18 (Mon)	0	0	1	09/02/18 (Mon)
	.	.	.	.	.	.
	.	.	.	.	.	.
	.	.	.	.	.	.
	88	23/03/20 (Mon)	0	0	1	30/03/20 (Mon)
	Inputs					Output
	No. of day	YD_t(D-1)	Hol	B_Hol	DoW	$FD_t^4(d)$
Testing Dataset	1	30/03/20 (Mon)	0	0	1	13/04/20 (Mon)

non-holiday (*Hol*), bridging holiday or non-bridging holiday (*B-Hol*), day of the week (*DoW*), and month of the year (*MoY*). Binary variables (1 or 0) are used for *Hol* and *B_Hol*. For the *DoW* variable, one is for Monday, two is for Tuesday, and so on. Another variable *MoY* uses 1 for Jan, 2 for Feb, 3 for Mar, and so forth. The dependent variable of CART is the actual load demand, which is a continuous data type.

For the DBN forecasting model, yesterday's load demand ($YD_t(d-1)$) from the same leaf, *Hol*, *B_Hol*, *DoW*, and *MoY*, seasonal index (*SI*) is input variables to predict daily forecasted load demand ($FD_t(d)$). *SI* for each period is calculated by dividing the actual load demand of each period by the average daily load demand. This paper conducts two different cases to train both classification and forecasting models with different variables. The former case does not include the *MoY* variable, whereas the latter one includes *MoY* and *SI*. The same duration of train and test are used for both cases. The parameters for each DBN model are listed: (i) hidden layer = 10, (ii) learning rate of RBM = 0.001, (iii) number of epochs for RBM = 10, (iv) number of iterations for backpropagation = 100, (v) batch size = 50, and (vi) activation function = rectified linear unit (ReLU).

Alternatively, DBN's forecasting inputs are trained for all benchmark models, followed by the training parameters of each benchmark model. The first LSTM trains with 50 epochs, 256 batch sizes, Adam optimizer, and MSE loss function. DNN uses the ReLU function with 50 hidden layers, 100 training cycles, ten epochs, and stochastic gradient descent (SGD) for backpropagation. In ANN, a sigma activation function, two hidden layers, 0.01 learning rate, 0.9 momentum, and 100 training cycles are applied. The next SVM trained model used a dot kernel type and 0.001 convergence epsilon to optimize parameters. The last LR trained model is trained with M5 prime for feature selection.

1) DATA ARRANGEMENT FOR CASE I

For the classification module, three independent variables: *Hol*, *B_Hol*, and *DoW* are included to create the CART in case I. On the other hand, the forecasting model has four input variables such as *Hol*, *B_Hol*, *DoW,* and ($YD_t(d-1)$) to forecast $FD_t^4(d)$, which represents the forecasted load demand of four inputs. The forecasting equation for the case I is denoted in Eq. 8, where c_o is a constant, and $c_1, \ldots, c_4$ are coefficients of each input. The sample data arrangement of leaf node 1 for the case I is indicated in Table 1.

$$FD_t^4(d) = a_0 + a_1 YD_t(d-1) + a_2(Hol) + a_3(B\_Hol) + a_4 DoW \quad (8)$$

2) DATA ARRANGEMENT FOR CASE II

Case II includes four independent variables: *Hol*, *B_Hol*, *DoW*, and *MoY* for training CART. Nonetheless, the DBN model of case II consists of two more additional inputs, such as *MoY* and *SI*, so case II has a total of six inputs for forecasting. The forecasting equation of case II is indicated in Eq. 9, where c_o is constant, $c_1, \ldots, c_6$ are coefficients of each input and $FD_t^6(d)$, is the forecasted load of six inputs. The sample data arrangement of leaf node 6 for case II is represented in Table 2.

$$FD_t^6(d) = c_0 + c_1 YD_t(d-1) + c_2(Hol) + c_3(B\_Hol) + c_4 DoW + c_5 MoY + c_6 SI \quad (9)$$

D. EXPERIMENTAL RESULTS

Mean absolute percentage error (MAPE) and MSE are calculated to measure the forecasting accuracy. These error metrics reveal how many units of the forecasted demand deviate from the actual demand, contributing to Eq. 10 and Eq. 11. For both cases, the proposed model is compared to DBN with MC. Furthermore, the generated results of the proposed model are compared with other standalone benchmark models, such as LSTM, DNN, ANN, SVM, and LR. In addition, we combine CART with all standalone benchmarking models to ensure that the combination of classification gives better performance on forecasting models. Five categories are regarded as weekdays (Tuesday-Friday), weekends, Monday, holidays,

TABLE 2. Data arrangement of leaf node 6 for case II.

	INPUTS							TARGET
TRAINING DATASET	NO. OF DAY	$YD_t(D\text{-}1)$	*HOL*	*B_HOL*	*DOW*	*MOY*	*SI*	$FD_t^6(d)$
	1	01/04/18 (Sun)	0	0	7	4	1.03	08/04/18 (Sun)
	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
	105	22/03/20 (Sun)	0	0	7	3	0.98	29/03/20 (Sun)
	INPUTS							OUTPUT
TESTING DATASET	NO. OF DAY	$YD_t(D\text{-}1)$	*HOL*	*B_HOL*	*DOW*	*MOY*	*SI*	$FD_t^6(d)$
	1	29/03/20 (Sun)	0	0	7	4	0.99	05/04/20 (Sun)

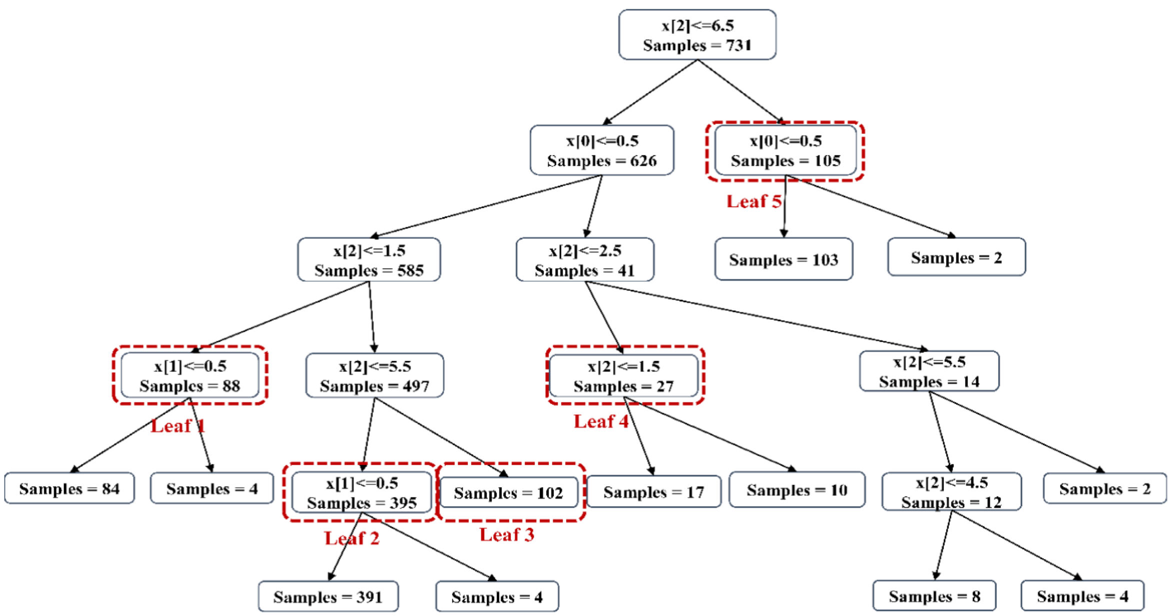

FIGURE 7. CART before and after pruning in case I.

and bridging holidays to make the monthly MAPE and MSE comparisons for each case. Before discussing MAPE and MSE comparisons, the differences between CART before pruning and CART after pruning for both cases are explained.

$$MAPE = \left(\frac{1}{48}\right)\sum_{t=1}^{48} abs\left(\frac{AD_t - FD_t}{AD_t}\right) \times 100\% \quad (10)$$

$$MSE = \left(\frac{1}{48}\right)\sum_{t=1}^{48} (AD_t - FD_t)^2 \quad (11)$$

1) CASE I: DIFFERENCE BETWEEN ORIGINAL CART AND PRUNED-CART

In case I, the original CART is primarily built based on the *DoW* independent variable, and holidays are also detected separately. This CART executes twelve terminal nodes observed at the last depth layer before pruning, as indicated in Figure 7. In Figure 7, $x[0]$, $x[1]$, and $x[2]$ refer to the independent input variables such as *Hol*, *B_Hol*, and *DoW*, respectively. Five nodes with the black boxes are regarded as the final terminal nodes after pruning CART. The first leaf node has only loaded a group of Mondays, showing 88 training days without holidays. The group of Tuesdays, Wednesdays, Thursdays, and Fridays is included in the second terminal node with 390 training days. The third and fifth leaf nodes are grouped with loads of Saturdays without holidays and Sundays with holidays. However, the load group for holidays, including Mondays to Saturdays, is generated at the fourth leaf node. The detail of leaf nodes on each

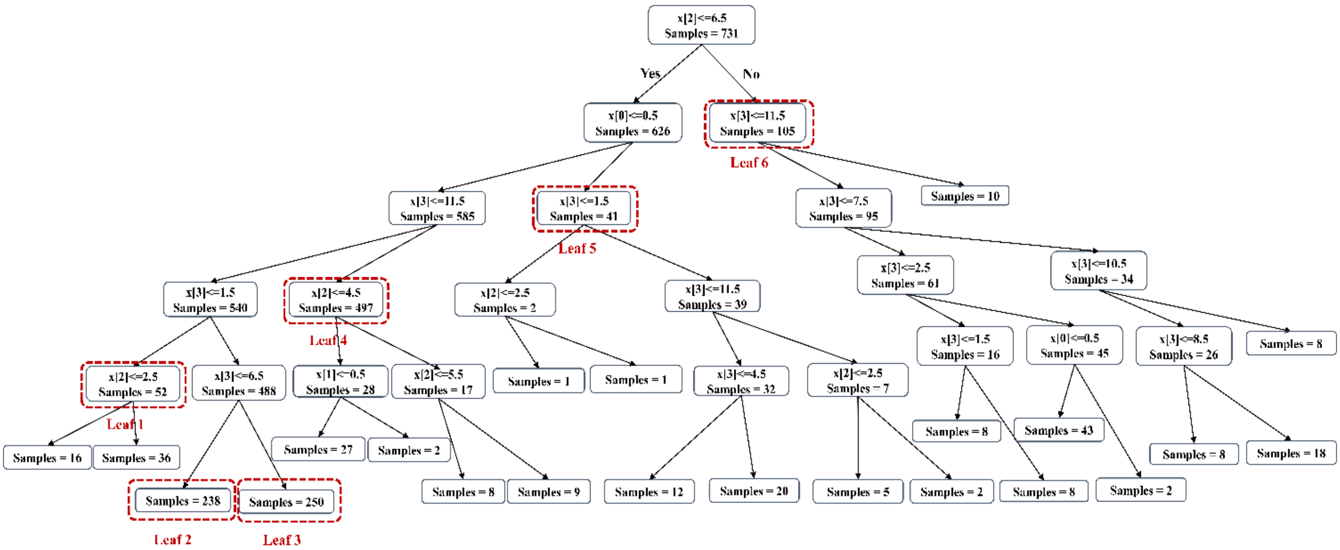

FIGURE 8. CART before and after pruning in case II.

TABLE 3. Details of five leaf nodes in case I.

Leaf Node	*Hol*	*B_Hol*	*DoW*	Group (*DoW*)	Total Days
1	0	1,0	1	Mon	88
2	0	1,0	3-5	Tue-Fri	390
3	0	0	6	Sat	102
4	1	0	1-2	Mon-Sat	41
5	1,0	0	7	Sun	105

load group, total training days, and categories of independent variables for the case I is revealed in Table 3. Certainly, CART works very well for the classification module.

2) CASE II: DIFFERENCE BETWEEN ORIGINAL CART AND PRUNED-CART

The additional *MoY* variable is given in the training process of CART so that the seasonality of data is affected on a tree in case II. There are twenty-two terminal nodes in the last depth layer of the original CART. Only six terminal nodes remain after pruning the tree, as shown in Figure 8. $x[0]$, $x[1]$, $x[2]$, and $x[3]$ in the tree represent the independent variables regarded as *Hol*, *B_Hol*, *DoW*, and *MoY*, respectively. In this case, the tree is mainly classified in terms of the *MoY* variable. Leaf 1 is grouped loads from Monday to Saturday in January, amounting to 52 days. Load demands from Mondays to Saturdays are categorized in leaf 3, leaf 4, and leaf 5 in February-June, July-November, and December, with 238, 250, and 45 training days. The fifth leaf node is split into loads on holidays in the training set, with 41 days total. The last terminal node has only Sunday's loads from January to December, including holidays in total days of 105. The detail of the load groups under *DoW* and *MoY*, full training days, and categories of independent variables for case II is indicated in Table 4.

TABLE 4. Details of six-leaf nodes in case II.

Leaf Node	*Hol*	*B_Hol*	*DoW*	*MoY*	Group (*DoW*)	Group (*MoY*)	Total Days
1	0	0	1-6	1	Mon-Sat	Jan	52
2	0	1,0	1-6	3-6	Mon-Sat	Feb-Jun	238
3	0	1,0	1-6	7-11	Mon-Sat	Jul-Nov	250
4	0	1,0	1-6	12	Mon-Sat	Dec	45
5	1	0	1-6	1-12	Mon-Sat	Jan-Dec	41
6	1,0	0	7	1-12	Sun	Jan-Dec	105

3) MONTHLY MAPE AND MSE COMPARISONS FOR THE CASE I

For the case I, the monthly MAPE and MSE comparisons between DBN with CART and DBN with MC on test predictions from Apr 2020 to Mar 2021, regarding five categories, are revealed in Table 5 and Table 6 correspondingly. In general, the proposed model provides better performance than DBN with MC in all categories, except weekdays. MAPEs and MSEs of October, December, and January are high due to long holidays and special days near holidays. The other months' errors for weekdays ranged from 2% to 5% in MAPE and approximately 350GW to 2500GW in MSE. Both CART and MC perform well for the weekend category, showing 3.11% and 3.16% of MAPE, and 790.45GW and 810.40GW in MSE. The proposed model has a slightly lower average MAPE and MSE than DBN with MC for both Mondays and holidays. Alternatively, the average error percentage of MC is around twice in MAPE and MSE more than that of CART in the category of bridging holidays.

TABLE 5. Monthly MAPE comparison between DBN with CART and DBN with MC for the case I in percent (%).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	CART	MC	CART	MC	CART	MC	CART	MC	CART	MC		
Apr, 2020	5.41	3.91	1.80	1.87	6.35	7.65	10.16	2.30	-	-	1	-
May, 2020	4.29	3.00	4.39	4.51	5.20	2.21	6.69	6.77	3.17	5.79	4	1
Jun, 2020	4.28	3.83	5.44	5.51	1.86	2.15	2.08	10.20	-	-	1	-
Jul, 2020	2.77	2.13	3.06	3.10	1.86	3.07	4.10	7.24	-	-	4	-
Aug, 2020	3.38	2.68	2.44	2.45	2.52	3.29	3.09	8.90	-	-	1	-
Sep, 2020	2.12	1.70	1.83	1.86	1.72	2.88	6.53	2.62	-	-	2	-
Oct, 2020	6.37	3.33	4.23	4.29	3.64	5.46	4.69	10.57	-	-	2	-
Nov, 2020	5.35	3.94	1.79	1.81	2.16	3.74	5.80	5.92	-	-	2	-
Dec, 2020	13.15	6.05	3.19	3.25	6.95	9.23	44.80	31.36	-	-	1	-
Jan, 2020	13.66	6.63	2.19	2.19	5.08	7.90	30.77	48.16	-	-	1	-
Feb, 2020	5.15	2.55	2.06	2.10	1.66	3.77	6.15	4.55	-	-	2	-
Mar, 2020	5.40	3.26	4.87	4.97	2.80	5.22	-	-	-	-	1	-
Total average	5.95	3.58	3.11	3.16	3.48	4.71	11.35	12.60	3.17	5.79		

TABLE 6. Monthly MSE comparison between DBN with CART and DBN with MC for the case I in gigawatt (GW).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	CART	MC	CART	MC	CART	MC	CART	MC	CART	MC		
Apr, 2020	2206.67	1147.57	269.27	283.79	2346.43	3266.00	5543.65	318.31	-	-	1	-
May, 2020	1635.99	847.94	1105.56	1151.84	2531.42	433.44	2762.90	2928.44	965.32	1932.62	4	1
Jun, 2020	1462.25	1072.55	2154.69	2186.77	265.86	311.36	276.33	5148.73	-	-	1	-
Jul, 2020	734.18	396.61	779.95	803.95	247.10	482.62	1537.69	3266.27	-	-	4	-
Aug, 2020	990.34	561.87	471.45	476.14	408.53	493.19	760.29	4228.43	-	-	1	-
Sep, 2020	363.30	246.24	272.61	281.73	214.93	305.50	3126.02	414.94	-	-	2	-
Oct, 2020	2782.57	824.77	1033.98	1054.55	801.12	1016.07	1118.37	4981.11	-	-	2	-
Nov, 2020	2181.79	1244.94	190.52	198.32	274.74	403.45	2345.38	1995.17	-	-	2	-
Dec, 2020	8676.31	2008.80	631.66	640.62	2198.79	2669.84	37661.53	20324.38	-	-	1	-
Jan, 2020	9291.08	2438.87	330.53	329.79	1391.89	1422.09	13732.38	34981.56	-	-	1	-
Feb, 2020	1975.71	539.81	270.26	279.89	204.93	415.35	1998.73	1214.59	-	-	2	-
Mar, 2020	2695.37	1053.53	1974.92	2037.48	645.00	1750.90	-	-	-	-	1	-
Total average	2916.30	1031.96	790.45	810.40	960.90	1080.82	6442.12	7254.72	965.32	1932.62		

4) MONTHLY MAPE AND MSE COMPARISONS FOR CASE II

For case II, the monthly MAPE and MSE comparisons on test predictions between CART and MC are indicated concerning five categories, as presented in Table 7 and Table 8, respectively. In this case, the performance of the proposed model for weekdays is improved. Moreover, the BH category indicates that CART is still better than MC, as in case I. However, MAPEs and MSEs of CART of holidays are much higher than MC in Apr, Jan, Feb, and Dec. It is indicated that adding *MoY* and *SI* variables in the proposed model provides worse forecasting performance, whereas it provides better performance for the forecasting model alone.

5) MONTHLY MAPE AND MSE COMPARISONS BETWEEN CASE I AND CASE II

The error comparisons using the proposed model between case I and case II are also enumerated in Table 9 and Table 10. The experimentation results show that additional input features cannot improve the forecasting performance even though it is suitable for the classification model. Total average error percentage of $F_t^4(d)$ outperforms $F_t^6(d)$ in all groups, except weekdays. In the category of weekdays, MAPE performance of $F_t^4(d)$ deteriorates in comparison to $F_t^6(d)$. In case $F_t^4(d)$, the error performance is 5.95% in MAPE and 2916.30GW in MSE, while it is 4.65% in MAPE

TABLE 7. Monthly MAPE comparison between DBN with CART and DBN with MC for case II in percent (%).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	CART	MC	CART	MC	CART	MC	CART	MC	CART	MC		
Apr, 2020	5.24	4.55	4.77	4.30	11.08	5.90	20.61	3.25	-	-	1	-
May, 2020	3.29	5.67	9.64	5.20	4.40	5.99	9.15	7.57	4.55	7.37	4	1
Jun, 2020	4.78	4.46	6.32	8.16	6.90	3.21	8.37	8.68	-	-	1	-
Jul, 2020	3.55	2.69	5.37	4.70	3.63	6.43	5.21	7.64	-	-	4	-
Aug, 2020	3.77	4.46	4.45	4.02	4.00	4.51	2.73	9.40	-	-	1	-
Sep, 2020	2.97	2.89	5.80	3.31	4.54	4.87	7.04	3.51	-	-	2	-
Oct, 2020	3.78	4.92	4.85	4.47	9.10	8.06	5.64	13.75	-	-	2	-
Nov, 2020	3.42	5.14	2.53	2.28	6.85	5.85	2.28	3.37	-	-	2	-
Dec, 2020	6.70	9.52	5.29	5.01	10.26	11.71	55.75	37.76	-	-	1	-
Jan, 2020	7.24	7.66	2.67	2.22	11.25	10.06	53.97	53.28	-	-	1	-
Feb, 2020	6.83	4.41	2.62	2.55	10.88	6.14	14.86	3.87	-	-	2	-
Mar, 2020	4.28	6.15	6.27	7.39	3.90	7.67	-	-	-	-	1	-
Total average	4.65	5.21	5.05	4.47	7.23	6.70	16.87	13.82	4.55	7.37		

TABLE 8. Monthly MSE comparison between DBN with CART and DBN with MC for case II in gigawatt (GW).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	CART	MC	CART	MC	CART	MC	CART	MC	CART	MC		
Apr, 2020	2061.70	1593.86	1497.86	1130.00	6803.93	2875.69	21803.75	592.38	-	-	1	-
May, 2020	897.95	3525.96	5580.12	1554.22	1342.38	4003.91	5469.08	3470.00	1721.85	2877.82	4	1
Jun, 2020	2015.88	1765.97	2410.16	4142.25	2776.04	784.50	6470.64	4712.74	-	-	1	-
Jul, 2020	966.51	594.76	2093.77	1455.04	1010.93	3906.45	2139.29	3736.44	-	-	4	-
Aug, 2020	1076.24	1881.32	1341.20	978.42	998.59	1413.34	377.34	5061.28	-	-	1	-
Sep, 2020	716.15	707.02	2477.45	789.28	1303.01	1243.66	3394.74	731.92	-	-	2	-
Oct, 2020	1163.90	1792.97	1549.10	1284.01	4133.12	2874.80	1820.68	7932.38	-	-	2	-
Nov, 2020	985.00	1825.85	550.90	380.08	2486.54	1104.56	463.54	892.67	-	-	2	-
Dec, 2020	2666.26	4335.00	1727.33	1545.93	5330.42	4312.75	57980.43	28201.15	-	-	1	-
Jan, 2020	3250.26	5189.45	590.85	423.59	8095.52	3368.70	41145.83	41113.33	-	-	1	-
Feb, 2020	2869.13	1501.18	541.94	481.95	5652.62	1803.28	10713.75	1078.55	-	-	2	-
Mar, 2020	1782.28	3175.38	2881.11	3815.71	1311.70	4154.25	-	-	-	-	1	-
Total average	1704.27	2324.06	1936.82	1498.37	3437.07	2653.82	13798.10	8865.71	1721.85	2877.82		

and 1704.27GW in MSE for $F_t^6(d)$. Nevertheless, case I provides the minimum of 1.66% MAPE and 204.93GW MSE in Feb 2021, in the category of Monday, while case II executes the minimum 2.28% MAPE and 463.54GW MSE in Nov 2020 for the H category.

6) MAPE COMPARISON BETWEEN THE PROPOSED MODEL AND STANDALONE FORECASTING MODELS

Comparative analysis between the proposed and benchmark standalone forecasting models is also performed, where MAPE is evaluated using case I. The opted benchmark models are DBN, LSTM, DNN, ANN, SVM, and LR. Four input features and MC are used for training all benchmark models on test predictions from Apr 2020 to Mar 2021. From the analysis of results (exhibited in Figure 9) obtained from detailed experiments, it is noteworthy that our proposed model outperformed all benchmark models and achieved the lowest average MAPE. The MAPE value is lower than the 4% for weekends, Mondays, and BH. On the other side, weekdays and H provide high MAPE, showing just under 6%

TABLE 9. Monthly MAPE comparison between $F_t^4(d)$ and $F_t^6(d)$ of the proposed model in percent (%).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$		
Apr, 2020	5.41	5.24	1.80	4.77	6.35	11.08	10.16	20.61	-	-	1	-
May, 2020	4.29	3.29	4.39	9.64	5.20	4.40	6.69	9.15	3.17	4.55	4	1
Jun, 2020	4.28	4.78	5.44	6.32	1.86	6.90	2.08	8.37	-	-	1	-
Jul, 2020	2.77	3.55	3.06	5.37	1.86	3.63	4.10	5.21	-	-	4	-
Aug, 2020	3.38	3.77	2.44	4.45	2.52	4.00	3.09	2.73	-	-	1	-
Sep, 2020	2.12	2.97	1.83	5.80	1.72	4.54	6.53	7.04	-	-	2	-
Oct, 2020	6.37	3.78	4.23	4.85	3.64	9.10	4.69	5.64	-	-	2	-
Nov, 2020	5.35	3.42	1.79	2.53	2.16	6.85	5.80	2.28	-	-	2	-
Dec, 2020	13.15	6.70	3.19	5.29	6.95	10.26	44.80	55.75	-	-	1	-
Jan, 2020	13.66	7.24	2.19	2.67	5.08	11.25	30.77	53.97	-	-	1	-
Feb, 2020	5.15	6.83	2.06	2.62	1.66	10.88	6.15	14.86	-	-	2	-
Mar, 2020	5.40	4.28	4.87	6.27	2.80	3.90	-	-	-	-	1	-
Total average	5.95	4.65	3.11	5.05	3.48	7.23	11.35	16.87	3.17	4.55		

TABLE 10. Monthly MSE comparison between $F_t^4(d)$ and $F_t^6(d)$ using DBN with CART in gigawatt (GW).

Month	Weekdays		Weekends		Monday		Holidays (H)		Bridging Holidays (BH)		Count H	Count BH
	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$	$F_t^4(d)$	$F_t^6(d)$		
Apr, 2020	2206.67	2061.70	269.27	1497.86	2346.43	6803.93	5543.65	21803.75	-	-	1	-
May, 2020	1635.99	897.95	1105.56	5580.12	2531.42	1342.38	2762.90	5469.08	965.32	1721.85	4	1
Jun, 2020	1462.25	2015.88	2154.69	2410.16	265.86	2776.04	276.33	6470.64	-	-	1	-
Jul, 2020	734.18	966.51	779.95	2093.77	247.10	1010.93	1537.69	2139.29	-	-	4	-
Aug, 2020	990.34	1076.24	471.45	1341.20	408.53	998.59	760.29	377.34	-	-	1	-
Sep, 2020	363.30	716.15	272.61	2477.45	214.93	1303.01	3126.02	3394.74	-	-	2	-
Oct, 2020	2782.57	1163.90	1033.98	1549.10	801.12	4133.12	1118.37	1820.68	-	-	2	-
Nov, 2020	2181.79	985.00	190.52	550.90	274.74	2486.54	2345.38	463.54	-	-	2	-
Dec, 2020	8676.31	2666.26	631.66	1727.33	2198.79	5330.42	37661.53	57980.43	-	-	1	-
Jan, 2020	9291.08	3250.26	330.53	590.85	1391.89	8095.52	13732.38	41145.83	-	-	1	-
Feb, 2020	1975.71	2869.13	270.26	541.94	204.93	5652.62	1998.73	10713.75	-	-	2	-
Mar, 2020	2695.37	1782.28	1974.92	2881.11	645.00	1311.70	-	-	-	-	1	-
Total average	2916.30	1704.27	790.45	1936.82	960.90	3437.07	6442.12	13798.10	965.32	1721.85		

and about 7%, respectively. The DBN model is ranked as the second model, whose MAPE performance is approximately 3% to 8% for all categories. Afterward, LSTM, DNN and LR come with MAPE performance varying between 6% and 9%. The worst MAPE performance is depicted by ANN and SVM models. For all categories, their MAPE performance varies between 7% and 11%. The errors of the H category in all benchmark models are higher than the proposed model.

Similarly, all forecasting models for case II on test predictions from Apr 2020 to Mar 2021 are compared in five categories, as represented in Figure 10. The proposed model outperforms benchmark models for weekdays, at around 4% in total average. About 6% of MAPE in CART with DBN and DBN models is given, while other models are delivered around 7% on weekends. In the case of Monday groups, the proposed model has less performance than all baseline models, except SVM. The CART with DBN is ranked as

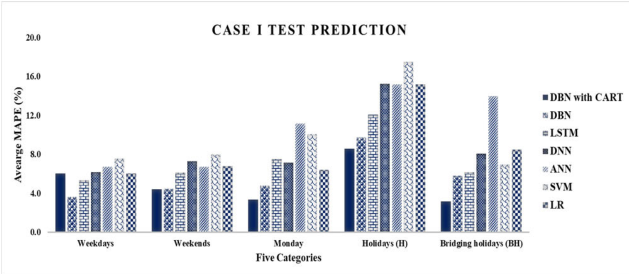

FIGURE 9. MAPE comparison for case I.

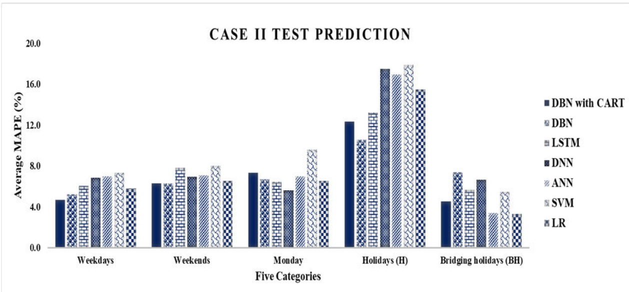

FIGURE 10. MAPE comparison for case II.

TABLE 11. Error performance on test prediction of case I.

Models	Approach	MSE (GW)	MAPE (%)
DBN	Standalone	1595.47	4.36
	With CART	2505.41	5.36
LSTM	Standalone	11325.57	14.41
	With CART	2210.84	5.43
DNN	Standalone	10891.33	12.82
	With CART	2398.58	5.30
ANN	Standalone	12749.61	13.65
	With CART	2376.15	5.38
SVM	Standalone	8988.55	11.90
	With CART	2904.93	6.12
LR	Standalone	12187.60	13.59
	With CART	2340.97	5.27

a second good model in the holiday group, representing near 12%. ANN and LR give better accuracy than others for the BH category, although the proposed model provides a reasonable error, at approximately 4%. It is revealed that the effect of external input factors has the drawback for improving accuracies in the groups of holidays and Mondays.

7) MONTHLY MAPE AND MSE COMPARISONS BETWEEN CASE I AND CASE II

All standalone benchmark models are combined with the CART model to measure the improvement of forecasting

TABLE 12. Error performance on test prediction of case II.

Models	Approach	MSE (GW)	MAPE (%)
DBN	Standalone	2803.20	6.04
	With CART	2742.80	5.39
LSTM	Standalone	7855.55	10.21
	With CART	2550.12	5.58
DNN	Standalone	13235.55	14.41
	With CART	2010.84	4.80
ANN	Standalone	14874.05	15.32
	With CART	2027.18	4.94
SVM	Standalone	9737.85	12.47
	With CART	3087.35	6.59
LR	Standalone	13189.94	14.37
	With CART	2570.67	5.91

performance. Therefore, two error metrics on the whole testing set are computed for all alternative hybrid and standalone models and compared to each other. Overall, all forecasting models achieve better accuracy using our approach, and they have approximately 5% MAPE and 2500GW MSE, as represented in Table 11 and Table 12. Regardless of errors in the DBN standalone model in case I, the combination of the CART and six forecasting models improve the performance compared with all standalone forecasting models. Consequently, the usage of the classification algorithm could improve the performance of short-term forecasting.

IV. CONCLUSION

In conclusion, this article focuses on the daily load forecasting for the electric industry. The CART model is proposed to classify the load data and handle non-linear problems between input and output variables. Moreover, grouping similar load patterns can help the improvement in the forecasting model's training process on unseen data. The DBN forecasting model is applied for predicting the daily load demand. The historical load data provided by EGAT is used. Our proposed model has been compared with six standalone forecasting models by using two error metrics. It outperforms all benchmark models, giving a minimum MAPE of 0.462%. Additionally, the CART is combined with all forecasting models and measured accuracies. Consequently, the combination of classification and forecasting models ensures the improvement in the accuracy performance on STLF. Besides, the CART provides the insight classification of seasonality effect better than MC.

APPENDIX

See Table 13.

TABLE 13. Literature review summary of classification on electricity load data Units for Magnetic Properties.

Ref #	Classification	Variables	Pruning	Results
[4]	CART	Load, temperature	-	Applied CART classifier for forecasting and recognition. Then compared with MLP
[5]	Frequency domain	Non-residential load	-	Used harmonics-based features for electricity customer classification
[6]	Frequency domain	Hourly load	-	Created classification tree using different complexity parameter values to finetune terminal nodes
[7]	Selection, typification, and clustering of load curves (STCL) algorithm	Load	-	Provided good performance in recognizing patterns in samples with heterogeneous data in the electric sector
[8]	k-means algorithm, logistic regression, support vector machines	Load in power-zone area	-	Clustered load patterns to serve temporary power installation and executed more efficient forecasting accuracy
[9]	Deep forest regression with Gini index parameter	Power load, weather condition data	-	Used the model for both classification and regression by providing minimum percentage error for STLF
[11]	k-means algorithm	Load	-	Classified daily load curves by functional clustering procedure
[12]	Cluster using threshold	Half-an-hour load, average temperature, forecasted temperature, day of week	-	Proved that results from clustering-based approach gave better results than without clustering
[13]	CART algorithm with Gini index parameter	Load, weather	-	Applied the CART for cluster analysis along with association analysis
[14]	Random forest	Power consumption data, weather, date, day of the week, holiday, academic year	-	Categorized electrical load data by grouping pattern similarity
[18]	CART	Day type, load, temperature, humidity	Pruned by computing the cross-validation error and the standard error	Evaluated the globally optimal regression tree through systematic rules
[19]	CART with simplified fuzzy inference	Day type, load, temperature, humidity	Pruned by computing the cross-validation error and the standard error	Used simplified fuzzy inference for the split values to enhance the accuracy of the regression tree
[21]	k-means algorithm for classification and RF and CART for regression	Total load, peak load, flat load, valley load, reactive power, max power, rate, time, category	-	Used directly as the classification-regression model based on the size of the data file
[28]	CART with "If…Then" structure to create rules	Load, weather	Pruned by measuring the cost-complexity	Clustered by choosing the same season of the year
[42]	CART	-	-	Introduced the CART by these authors
[23]	Alopex perceptron decision tree (APDT) algorithm	IRIS and wine classification data	Applied learning automata (LA) and genetic algorithm (GA) for pruning	Classified for 2-class problems
[24]	CART	Temperature	Pruned by complexity and loss	Used for model selection
[25]	Fuzzy clustering and CART with Gini index parameter	Power consumption data	Pruned by the cost complexity function	Executed forecasted results based on seven clusters
[26]	Tree classification methods	Electricity customers' loads	-	Classified on customer load
[27]	CART	Demographic information for households	Pruned by objective functions and side-constraints	Approached optimal prune CART along with TRANSIMS

FUTURE WORK

In the future, we will extend our combination approach using other advanced DL models, such as MLP, LSTM, CNN, or hybrid methods combining two or more DL models.

ACKNOWLEDGMENT

The authors want to acknowledge the Electricity Generating Authority of Thailand (EGAT) for providing data used in this research.

REFERENCES

[1] K. Chapagain, S. Kittipiyakul, and P. Kulthanavit, "Short-term electricity demand forecasting: Impact analysis of temperature for Thailand," *Energies*, vol. 13, no. 10, pp. 1–29, 2020, doi: 10.3390/en13102498.

[2] E. G. Henrichon and K.-S. Fu, "A nonparametric partitioning procedure for pattern classification," *IEEE Trans. Comput.*, vol. C-18, no. 7, pp. 614–624, Jul. 1969.

[3] W. S. Meisel and D. A. Michalopoulos, "A partitioning algorithm with application in pattern classification and the optimization of decision trees," *IEEE Trans. Comput.*, vol. C-22, no. 1, pp. 93–103, Jan. 1973, doi: 10.1109/T-C.1973.223603.

[4] L. Atlas, J. Connor, D. Park, M. El-Sharkawi, R. Marks, A. Lippman, R. Cole, and Y. Muthusamy, "A performance comparison of trained multilayer perceptrons and trained classification trees," *Proc. IEEE*, vol. 78, no. 10, pp. 1614–1619, Oct. 1990, doi: 10.1109/5.58347.

[5] E. Carpaneto, G. Chicco, R. Napoli, and M. Scutariu, "Electricity customer classification using frequency–domain load pattern data," *Int. J. Electr. Power Energy Syst.*, vol. 28, no. 1, pp. 13–20, Jan. 2006, doi: 10.1016/j.ijepes.2005.08.017.

[6] S. Zhong and K.-S. Tam, "Hierarchical classification of load profiles based on their characteristic attributes in frequency domain," *IEEE Trans. Power Syst.*, vol. 30, no. 5, pp. 2434–2441, Sep. 2015, doi: 10.1109/TPWRS.2014.2362492.

[7] A. M. S. Ferreira, C. A. M. T. Cavalcante, C. H. O. Fontes, and J. E. S. Marambio, "A new method for pattern recognition in load profiles to support decision-making in the management of the electric sector," *Int. J. Electr. Power Energy Syst.*, vol. 53, pp. 824–831, Dec. 2013, doi: 10.1016/j.ijepes.2013.06.001.

[8] A. M. Kotriwala, P. Hernandez-Leal, and M. Kaisers, "Load classification and forecasting for temporary power installations," in *Proc. IEEE PES Innov. Smart Grid Technol. Conf. Eur. (ISGT-Eur.)*, Oct. 2018, pp. 1–6, doi: 10.1109/ISGTEurope.2018.8571849.

[9] L. Yin, Z. Sun, F. Gao, and H. Liu, "Deep forest regression for short-term load forecasting of power systems," *IEEE Access*, vol. 8, pp. 49090–49099, 2020, doi: 10.1109/ACCESS.2020.2979686.

[10] S. M. Gorade and P. A. Deo, "A study some data mining classification techniques," *Int. J. Mod. Trends Eng. Res.*, vol. 4, no. 1, pp. 210–215, 2017, doi: 10.21884/ijmter.2017.4031.zt9tv.

[11] A. Goia, C. May, and G. Fusai, "Functional clustering and linear regression for peak load forecasting," *Int. J. Forecasting*, vol. 26, no. 4, pp. 700–711, 2010, doi: 10.1016/j.ijforecast.2009.05.015.

[12] A. Jain and B. Satish, "Clustering based short term load forecasting using support vector machines," in *Proc. IEEE Bucharest PowerTech*, Jun. 2009, pp. 1–8, doi: 10.1109/PTC.2009.5282144.

[13] X. Zhang, M. Cheng, Y. Liu, D. H. Li, and R. M. Wu, "Short-term load forecasting based on big data technologies," *Appl. Mech. Mater.*, vols. 687–691, pp. 1186–1192, Nov. 2014, doi: 10.4028/www.scientific.net/AMM.687-691.1186.

[14] J. Moon, Y. Kim, M. Son, and E. Hwang, "Hybrid short-term load forecasting scheme using random forest and multilayer perceptron," *Energies*, vol. 11, no. 12, pp. 1–20, 2018, doi: 10.3390/en11123283.

[15] G. Kesavaraj and S. Sukumaran, "A study on classification techniques in data mining," in *Proc. 4th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2013, pp. 1–7, doi: 10.1109/ICCCNT.2013.6726842.

[16] A. K. Srivastava, "Short term load forecasting using regression trees: Random forest, bagging and M5P," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 2, pp. 1898–1902, Apr. 2020, doi: 10.30534/ijatcse/2020/152922020.

[17] W.-Y. Loh, "Classification and regression trees," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 1, no. 1, pp. 14–23, 2011, doi: 10.1002/widm.8.

[18] H. Mori, N. Kosemura, K. Ishiguro, and T. Kondo, "Short-term load forecasting with fuzzy regression tree in power systems," in *Proc. IEEE Int. Conf. Syst., Man Cybern. e-Syst. e-Man Cybern. Cyberspace*, Oct. 2001, pp. 1948–1953, doi: 10.1109/ICSMC.2001.973654.

[19] H. Mori and N. Kosemura, "Optimal regression tree based rule discovery for short-term load forecasting," in *Proc. IEEE Power Eng. Soc. Winter Meeting. Conf.*, Jan. 2001, pp. 421–426, doi: 10.1109/PESW.2001.916878.

[20] M. Hambali, A. Akinyemi, J. Oladunjoye, and N. Yusuf, "Electric power load forecast using decision tree algorithms," *Comput. Inf. Syst. Dev. Informat. Allied Res. J.*, vol. 7, no. 4, pp. 29–42, 2016.

[21] J. Lei, T. Jin, J. Hao, and F. Li, "Short-term load forecasting with clustering–regression model in distributed cluster," *Cluster Comput.*, vol. 22, no. S4, pp. 10163–10173, Jul. 2019, doi: 10.1007/s10586-017-1198-4.

[22] Z. Guo, K. Zhou, X. Zhang, and S. Yang, "A deep learning model for short-term power load and probability density forecasting," *Energy*, vol. 160, pp. 1186–1200, Oct. 2018, doi: 10.1016/j.energy.2018.07.090.

[23] S. Shah and P. S. Sastry, "New algorithms for learning and pruning oblique decision trees," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 29, no. 4, pp. 494–505, Nov. 1999, doi: 10.1109/5326.798764.

[24] S. Gey and E. Nedelec, "Model selection for CART regression trees," *IEEE Trans. Inf. Theory*, vol. 51, no. 2, pp. 658–670, Feb. 2005.

[25] Y. Qi, B. Luo, X. Wang, and L. Wu, "Load pattern recognition method based on fuzzy clustering and decision tree," in *Proc. IEEE Conf. Energy Internet Energy Syst. Integr. (EI)*, Nov. 2017, pp. 1–5, doi: 10.1109/EI2.2017.8245714.

[26] F. Biscarri, I. Monedero, A. García, J. I. Guerrero, and C. León, "Electricity clustering framework for automatic classification of customer loads," *Expert Syst. Appl.*, vol. 86, pp. 54–63, Nov. 2017, doi: 10.1016/j.eswa.2017.05.049.

[27] H. D. Sherali, A. G. Hobeika, and C. Jeenanunta, "An optimal constrained pruning strategy for decision trees," *INFORMS J. Comput.*, vol. 21, no. 1, pp. 49–61, Feb. 2009, doi: 10.1287/ijoc.1080.0278.

[28] Y.-C. Guo, "Knowledge-enabled short-term load forecasting based on pattern-base using classification & regression tree and support vector regression," in *Proc. 5th Int. Conf. Natural Comput.*, Aug. 2009, pp. 425–429, doi: 10.1109/ICNC.2009.248.

[29] A. S. Khwaja, X. Zhang, A. Anpalagan, and B. Venkatesh, "Boosted neural networks for improved short-term electric load forecasting," *Electr. Power Syst. Res.*, vol. 143, pp. 431–437, Feb. 2017, doi: 10.1016/j.epsr.2016.10.067.

[30] K. Gajowniczek and T. Zabkowski, "Short term electricity forecasting using individual smart meter data," *Proc. Comput. Sci.*, vol. 35, pp. 589–597, Jan. 2014, doi: 10.1016/j.procs.2014.08.140.

[31] J. Che and J. Wang, "Short-term load forecasting using a kernel-based support vector regression combination model," *Appl. Energy*, vol. 132, pp. 602–609, Nov. 2014, doi: 10.1016/j.apenergy.2014.07.064.

[32] P. Lusis, K. R. Khalilpour, L. Andrew, and A. Liebman, "Short-term residential load forecasting: Impact of calendar effects and forecast granularity," *Appl. Energy*, vol. 205, pp. 654–669, Nov. 2017, doi: 10.1016/j.apenergy.2017.07.114.

[33] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015, doi: 10.1016/j.neunet.2014.09.003.

[34] M. Cai, M. Pipattanasomporn, and S. Rahman, "Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques," *Appl. Energy*, vol. 236, pp. 1078–1088, Feb. 2019, doi: 10.1016/j.apenergy.2018.12.042.

[35] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[36] K. Amarasinghe, D. L. Marino, and M. Manic, "Deep neural networks for energy load forecasting," in *Proc. IEEE 26th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2017, pp. 1483–1488, doi: 10.1109/ISIE.2017.8001465.

[37] W. He, "Load forecasting via deep neural networks," *Proc. Comput. Sci.*, vol. 122, pp. 308–314, Dec. 2017, doi: 10.1016/j.procs.2017.11.374.

[38] P. P. Phyo, C. Jeenanunta, and K. Hashimoto, "Electricity load forecasting in Thailand using deep learning models," *Int. J. Electr. Electron. Eng. Telecommun.*, vol. 8, no. 4, pp. 221–225, 2019, doi: 10.18178/ijeetc.8.4.221-225.

[39] S. Kouhi and F. Keynia, "A new cascade NN based method to short-term load forecast in deregulated electricity market," *Energy Convers. Manage.*, vol. 71, pp. 76–83, Jul. 2013, doi: 10.1016/j.enconman.2013.03.014.

[40] C. Jeenanunta and K. D. Abeyrathn, "Combine particle swarm optimization with artificial neural networks for short-term load forecasting," *Int. Sci. J. Eng. Technol.*, vol. 1, no. 1, pp. 25–30, 2017. [Online]. Available: https://ph02.tci-thaijo.org/index.php/isjet/article/view/175773.

[41] S. W. Hnin and C. Jeenanunta, "Bayesian optimization in a support vector regression model for short-term electricity load forecasting," *Eng. Appl. Sci. Res.*, vol. 46, no. 3, pp. 267–275, 2019, doi: 10.14456/easr.2019.30.

[42] A. L. Breiman, H. J. Friedman, A. R. Olshen, and C. J. Stone, *Classification and Regression Trees*, vol. 67. Boston, MA, USA: Wadsworth Inc, 1984.

[43] W. Xu, H. Peng, X. Zeng, F. Zhou, X. Tian, and X. Peng, "Deep belief network-based AR model for nonlinear time series forecasting," *Appl. Soft Comput.*, vol. 77, pp. 605–621, Apr. 2019, doi: 10.1016/j.asoc.2019.02.006.

[44] X. Kong, C. Li, F. Zheng, and C. Wang, "Improved deep belief network for short-term load forecasting considering demand-side management," *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1531–1538, Mar. 2020, doi: 10.1109/TPWRS.2019.2943972.

[45] T. Ouyang, Y. He, H. Li, Z. Sun, and S. Baek, "Modeling and forecasting short-term power load with copula model and deep belief network," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 2, pp. 127–136, Apr. 2019, doi: 10.1109/TETCI.2018.2880511.

[46] H. Z. Wang, G. B. Wang, G. Q. Li, J. C. Peng, and Y. T. Liu, "Deep belief network based deterministic and probabilistic wind speed forecasting approach," *Appl. Energy*, vol. 182, pp. 80–93, Nov. 2016, doi: 10.1016/j.apenergy.2016.08.108.

PYAE PYAE PHYO received the B.Sc. degree in electrical power engineering from Mandalay Technological University, Myanmar, and the M.Sc. and Ph.D. degrees in engineering and technology from the Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. During her Ph.D. program, she joined a machine learning (ML) camp in South Korea. She is currently working as an ML Researcher at the ML Laboratory, Jeju National University, South Korea. Her research interests include artificial intelligence, machine learning, deep learning, prediction analysis, time series analysis, electricity forecasting, and energy consumption. She received the Excellent Foreign Student (EFS) Scholarship Award throughout her graduate studies. She was awarded a Canada-ASEAN Scholarships and Educational Exchanges for Development (SEED) to research at the University of Calgary, Canada.

CHAWALIT JEENANUNTA received the B.Sc. degree in mathematics and computer science and the M.Sc. degree in management science from the University of Maryland, and the Ph.D. degree in industrial and systems engineering from the Virginia Polytechnic Institute and State University. He joined the Sirindhorn International Institute of Technology, Thammasat University, Thailand, as a Lecturer, where he is an Associate Professor. He was the Chair of the management technology curriculum, the Head of the School of Management Technology, and the Deputy Director for building, ground, and properties. He is also the Head of the Center of Excellence in Logistics and Supply Chain Systems Engineering (LogEn) and the Center for Technology Transfer of Industry 4.0. His research interests include applications of operations research and innovation. He received research funding from the Thailand Research Fund, the Institute of Developing Economies and Japan External Trade Organization (IDE-JETRO), the Economic Research Institute for ASEAN and East Asia (ERIA), and the PTT Logistics and Electricity Generating Authority of Thailand (EGAT).