IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 17 May 2023, accepted 31 May 2023, date of publication 9 June 2023, date of current version 16 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3284678

RESEARCH ARTICLE

Machine Learning-Based Day-Ahead Prediction of Price-Setting Scheduled Energy in the Korean Electricity Trading Mechanism

DONGHUN LEE[1], KWANHO KIM[1], AND SANG HWA SONG[2]
[1]Department of Industrial and Management Engineering, Incheon National University, Incheon 22012, Republic of Korea
[2]Graduate School of Logistics, Incheon National University, Incheon 22012, Republic of Korea

Corresponding authors: Sang Hwa Song (songsh@inu.ac.kr) and Kwanho Kim (khokim@inu.ac.kr)

This work was supported in part by the Research Assistance Program, Incheon National University, in 2020; and in part by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry and Energy (MOTIE) of the Republic of Korea under Grant 20212020900090.

ABSTRACT Power generation companies, which participate in the electricity trading mechanism, need to determine optimal capacity bidding strategy to maximize their operational profit in Korea. Since the Price-setting Power Generation Schedule determines the profit of power generators, it is important to predict Price-setting Scheduled Energy during the trading day before the bidding phase. We propose a methodology for predicting the Pricing-setting Scheduled Energy from the power exchange without optimizing it. Instead of simulating the planning process, machine learning algorithms are applied and compared in the process of predicting the Pricing-setting Scheduled Energy. The input variables consist of seasonal and price information including calendar, fuel cost, and system marginal price. Three categories of machine learning (ML) algorithms including single, bagging and boosting approaches are implemented and tested to compare their performances. The computational experiments show that ML algorithms with price variable are shown to be better in terms of the considered measures. In addition, boosting approach is more effective than single and bagging approaches.

INDEX TERMS Electricity power generation schedule, electricity trading, machine learning, price-setting scheduled energy.

NOMENCLATURE

Symbols and acronyms:

KPX	Korea Power Exchange.
PPGS	Price-setting Power Generation Schedule.
PSE	Price-setting Scheduled Energy.
SMP	System marginal price.
DT	Decision tree.
SVM	Support vector machine.
RF	Random forest.
ET	Extra trees.
GB	GradientBoost.
CB	CatBoost.
XGB	XGBoost.
TP	True positive.
TN	True negative.
FP	False positive.
FN	False negative.

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

I. INTRODUCTION

A central power exchange, known as the Korea Power Exchange (KPX), coordinates electricity supply with the anticipated demand across the entire country in the Korean electricity trading market. According to the KPX market operation rules, the power exchange controls market operations including demand forecasting, bid management, power generation scheduling, monitoring, and dispute mediation.

It determines the next-day electricity production schedule of each plant to meet the expected electricity demand while considering each generator's operational characteristics.

Power generation companies participate in the pool to determine the transaction price and volume of electricity to be produced the next day. The price of electricity determined in the electricity market follows the same principle as the price of general products determined in a competitive market at the point of equilibrium between supply and demand. The market price in Korea's electricity market is determined hourly, one day before the day of electricity trading when the predicted electricity demand and supply curves formed by generators participating in supply bidding meet and intersect system constraints. Power generation companies determine their bidding strategies based on the demand forecast and competitors' bidding plan prediction. If more companies participate in the market, the price of electricity decreases, and the profits of the participants become lower. Therefore, it is quite important for a power generation company to accurately predict the next day's power generation schedule of the power exchange.

The power generation planning program, which is called the Price-setting Power Generation Schedule (PPGS), determines the generation output of each power plant while minimizing the total cost of the generators participating in supply bidding. The power exchange forecasts the electricity demand on the trading day and determines the power generation schedule by considering the bidding data of the participating power plants and forecasted demand load. The objective of the PPGS is to minimize the system marginal price (SMP). The unit production cost of the most expensive power generator from the plan determines the marginal price, which becomes the baseline for the cost settlement of each power generator. In a typical setting, either nuclear or coal plants determine the marginal price during off-peak hours, whereas expensive liquefied natural gas (LNG) and oil plants determine the price during peak hours as depicted in Fig. 1. The power generation level of each plant from the PPGS is called Price-setting Scheduled Energy (PSE), which is important information to power generation company. Based on the PSE information, power generation companies develop their own bidding strategy.

Once the PSE is determined, the power exchange generates an operational power generation schedule by considering detailed constraints such as the network-wide distribution restriction and finalizes the unit commitment and power dispatch solution as show in Fig. 2. Power generation companies in the pool build their own production plans based on the finalized schedule from the power exchange. The power exchange revises the operational power generation schedule in real time during the trading day, reflecting any changes in electricity demand and supply. After the trading day, the power exchange pays the cost of power generation based on the price settlement rule specified in the operation rules.

Power generation companies, such as LNG and oil generators, must determine their optimal capacity bidding strategies to maximize operational profit. Because the PPGS determines each power generator's profit, the generation company forecasts the electricity demand load and predicts the system marginal price and PSE. The PSE of a power generator is the power generation level obtained from the PSE of the power exchange. If a power generation company has any resource constraints, it is better to restrict the bidding capacity when its power generation level is expected to be low. Therefore, it is important for power generation companies to predict their PSE on the trading day.

Predicting the PSE is important; however, existing studies have mainly focused on power demand forecasting and power generation plan optimization. For demand forecasting, many studies have been conducted that applied traditional demand forecasting techniques such as regression analysis, support vector regression (SVR) and autoregressive integrated moving average (ARIMA) [2], [3], [4]. Recently, demand forecasting based on deep learning has been investigated [5], [6]. To determine the power generation plan, optimization techniques, such as priority ordering, Lagrangian relaxation, variable neighborhood search, and mixed integer programming, have been successfully applied [7], [8], [9], [10].

In this study, we propose a methodology for predicting the PSE from power exchange without optimization. Optimizing the power generation plan is expected to be more effective in predicting the PSE. However, it is difficult for each power generation company to secure all necessary data to optimize the power generation schedule. To simulate the power generation planning process of the power exchange, it is essential to collect production information of each participating company in the pool and forecast the electricity demand during the trading day. If there are any mismatches, the prediction quality of the resulting power generation plan and the associated PSE of the power generation company are severely affected. Instead of simulating the power generation planning process, machine learning algorithms were applied to predict the PSE. Since the devised algorithms predict the future PSE solely based on time-series history data, they do not require any detailed information about production characteristics. The contributions of this study are listed as follows.

(1) To find the outstanding machine learning algorithm among the well-known machine learning algorithms for the considered problem, we compare PSE prediction performances by implementing them.

(2) Through grid search with a cross-validation strategy, the optimal hyper-parameters for each algorithm that result in the best PSE prediction are determined in this study.

(3) The results of various experiments suggest that the SMP variable is essential to improve PSE prediction performances. Moreover, it is found that XGB is to be more effective in predicting the PSE in the real-world for all seasons.

The rest of this research is organized as follow. Section II reviews the previous studies related to this research. The proposed research framework and methods are presented in Section III. In Section IV, the comparison analysis with ML algorithms for PSE prediction is performed. Finally, we provide a summary and implications of this study in Section V.

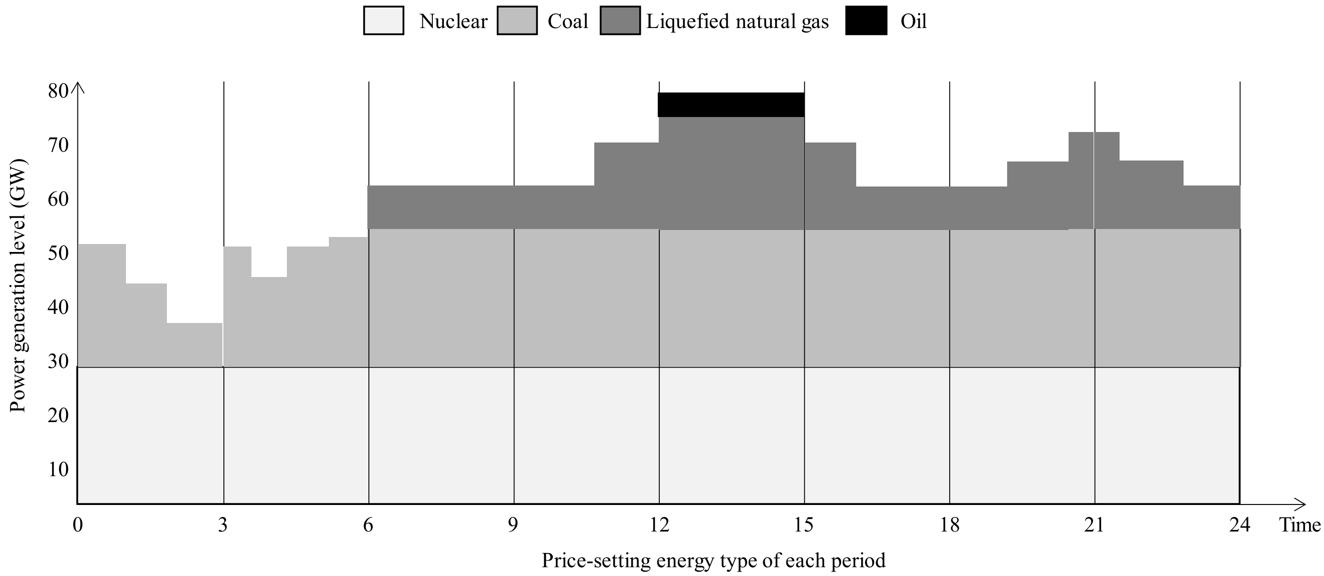


FIGURE 1. An example of price-setting power generation schedule illustrated in the KPX homepage [1].

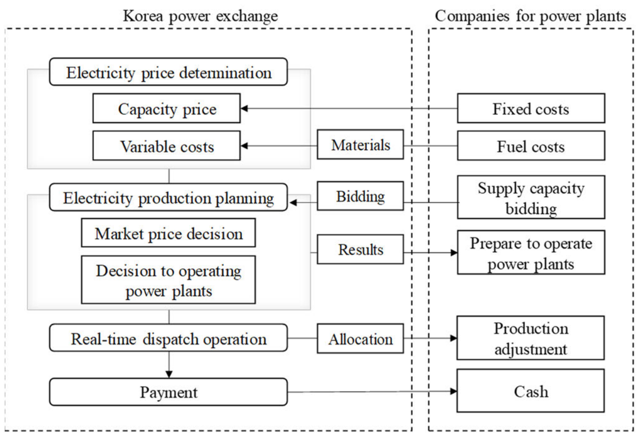


FIGURE 2. A framework for electricity bidding and settlement procedures from the KPX homepage.

II. LITERATRUE REVIEWS

In terms of energy operation, multiple mixed integer linear programming was formulated [11]. Qin et al. [12] suggested a swarm optimization approach dealing with uncertain renewable energy generation and heat demands. Another study proposed a combined method of simulation and optimization approaches to find optimal strategies in a district heating system [13]. More recently, a hybrid method to operate a residential energy system composed of a photovoltaic, fuel cell, boiler, and storage units [14]. Optimization approaches are expected to optimally operate energy system. However, their practical application in real-world scenarios may be limited by the high computational costs associated with their implementation.

Prior studies have been employed statistics approach to predict energy demands. A study suggested ARMIA-based model to predict energy demand [15]. Another study compared the energy demand prediction performances between ARIMA and seasonal ARIMA (SARIMA) by using real-time load data of Assam [16]. This research found that the SARIMA that considers the seasonal trends provides better performances than the ARIMA. Wang et al. [17] conducted comparison analysis with various SARIMA models by using electricity demand obtained from China. Although these methods are easily implemented in the real-world environment, the performances may not be guaranteed when energy demand dynamically changes. This is because these approaches rely on historical demand trends and seasonal patterns, which may not always accurately reflect current and future energy requirements.

As alternatives, machine learning (ML) and neural network (NN) approaches have been utilized to improve energy demand prediction. Guo et al. [18] used SVR, extreme learning, and NN with correlation analysis to develop ahead energy demand prediction models. This study found that the optimal number of hidden layer nodes and feature sets through experiments. A study executed a comparative analysis extreme gradient boosting (XGB) with NN [19]. Recently, an extended comparative analysis for energy demand in a district heating system conducted by employing various ML and NN approaches [20]. Through numerical experiments, the LSTM and XGB are consistently provide better performances than others. In addition, a study proposed hybrid method consisting of gate recurrent network (GRU), masked multi head attention, and light gradient boosting machine for predicting energy demand [21] and demonstrated the hybrid method is more effective to predict energy demand compared to other ML and NN. The previous results suggest that while ML and NN are useful to predict energy demands, accurate model selection is required through a comparative analysis since the performances between approaches are different according to the problem considered.

III. MACHINE LEARING ALGORITHMS FOR PSE PREDICTION

This section introduces the machine learning (ML) algorithms proposed for predicting PSE in advance. First, Sections II.A and II.B describe the proposed framework and considered variables, respectively. Next, we explain in Section II.C the proposed ML algorithms, including the single, bagging ensemble, and boosting ensemble algorithms. Finally, in Section II.D, grid search and cross-validation strategies are explained, which search for the best hyper-parameters that can improve the performance of the ML algorithms. Table 1 show notations and nomenclature used in this paper.

A. PROPOSED FRAMEWORK

The proposed framework for PSE prediction and compares their performances is presented in Fig. 3. The framework consisted of training and testing phases. In the first training stage, we prepared datasets for training that contained calendar information, SMP, fuel cost, and PSE schedule history. In Step 1.1, the preprocessing is conducted to redesign the PSE schedule history for each period. If an electricity production task is required, it is one; otherwise, it is zero. Next, the algorithms considered in this study is to set up presented in Step 1.2. Subsequently, a pool of model parameters is determined to develop PSE prediction models. Finally, ML algorithm-based PSE prediction models are implemented using the preprocessed training dataset and a grid search with a cross-validation strategy which is adopted to search for the best hyper-parameters to enhance the performance of PSE prediction [22].

In the testing phase, the identical preprocessing employed in the training phase is carried out using the testing dataset presented in Step 2.1. In Step 2.2, we perform the comparison analysis utilizing the selected algorithms that demonstrate the best prediction performances in the training phase and the preprocessed testing dataset through the various experiment settings.

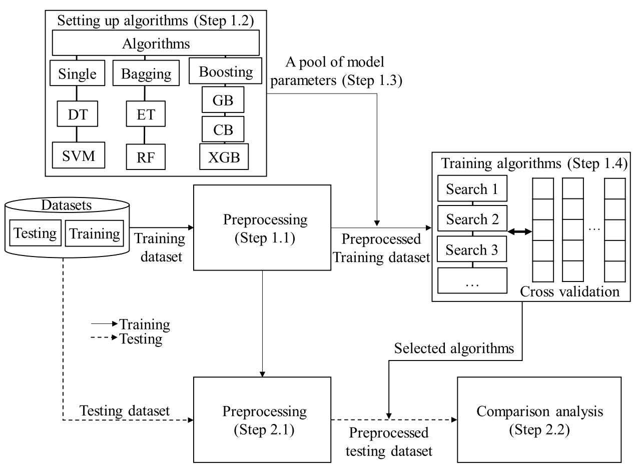


FIGURE 3. Proposed framework for PSE prediction models.

B. VARIABLES

The input and output features considered in this study are represented in Table 1. The input variables consist of seasonal and price information. The seasonal variables contain hourly, daily, and monthly information, and the price variables include the fuel cost and SMP information. These variables are reasonable to consider since they are directly associated with changes in PSE schedules. In particular, the SMP is strongly related to the expected schedule of PSE a day ahead, as the schedule of PSE is determined after the expected SMP is calculated from the KPX [23]. Thus, we investigate the differences in prediction performances when the SMP variable is used (Section III-B).

TABLE 1. Variables considered in this study.

Symbol	Category	Variable
Inputs	Calendar	Hourly
		Daily
		Monthly
	Price	Fuel cost
		SMP
Output	PSE schedule	PSE

C. PSE PREDICTION MODELS

As introduced in [24], The ML algorithms are typically categorized into single, bagging, and boosting approaches. Single machine learning approach is commonly used in many applications and effective in situations where the data is well-understood and the problem is well-defined. Bagging approach is one of ensemble method that aims to reduce variance of a model by training multiple weak models in parallel and combining their predictions. This approach trains a weak model on each sample by using bootstrap sampling which helps with random sampling from the training data. Unlike bagging approach, which trains weak models in parallel, boosting approach trains weak models sequentially by emphasizing the misclassified samples in each iteration. Each model tries to improve the prediction of the previous model by focusing on the samples that were misclassified.

The previous study related to this research applied ML algorithms to compare their performances in energy prediction problem separately for each category of single, bagging, and boosting [25]. Accordingly, these approaches are selected in this study as shown in Table 2, and we conduct comparison analysis by utilizing them.

TABLE 2. Applied machine learning algorithms.

Single	Bagging	Boosting
Decision tree	Random forest	GradientBoost
Support vector machine	Extra trees	CatBoost
		XGBoost

First, for the single approach, the decision tree (DT) algorithm searches for the best path that can observe the hourly PSE schedule a day ahead by hierarchically investigating the nodes connected to each other [26]. The advantage of this algorithm is that it can interpret why an output value is achieved using an input value from the nodes [27]. SVM is based on training to find the best hyperplane among multiple hyperplanes in a high-dimensional space that enables accurate prediction of the hourly PSE schedule a day ahead [28].

Next, bagging approaches such as RF and extra trees (ET) are considered. The purpose of bagging is to improve the prediction performance of a PSE day-ahead schedule by combining a set of algorithms [25]. RF creates multiple different trees in the training phase to search for the best tree that shows high prediction accuracy [29] compared to DT. However, the outcome interpretation of RF is more difficult than that of DT, because the investigation of all trees is difficult. The output function is denoted as Z and is calculated using Equation (1) where $p_t(y\,|\,x)$ is the probability distribution of each tree t, x is a set of samples, and T is total trees, respectively.

$$Z = argmax \frac{1}{T} \sum_{t}^{T} p_t(y\,|\,x) \tag{1}$$

Finally, ET is very similar to RF, whereas ET randomly chooses candidate trees among multiple trees to split nodes. It successfully reduces the training time and biases compared to RF [24].

Lately, for the boosting approach, gradient boosting (GB), CatBoost (CB), and XGBoost (XGB) are employed. GB attempts to select the new tree to enable accurate forecasting of a PSE schedule by using categorical cross-entropy from the previous tree. A GB uses decision trees as base predictors in each training step to reduce the loss function. As noted by [30], the output function is presented as follows.

$$Z = \sum_{j=1}^{J} \beta_j g(x; b_j) \tag{2}$$

where the function $g(x; b_j)$ is a predictor x is the input values, β_j is the expansion coefficients, and b_j is the parameters of the applied model. CB is useful not only for supporting categorical and numerical variables, but also for improving training performance by using a level-wise method. The output function is explained by [31] and calculated using Equation (3) where c_j is the leaf value, and R_j represents the set of disjoint regions that correspond to the leaves of the tree.

$$Z = \sum_{j=1}^{J} c_{j\{x \in R_j\}} \tag{3}$$

As XGB is efficient in overcoming the overfitting problem by utilizing a regularization method [32], it is likely to forecast a day-ahead PSE schedule.

D. GRID SERACH WITH CORSS-VALIDATION STRATEGY

The considered algorithms require considerable computation time to obtain optimal hyper-parameters that yield the best outcome from the number of parameters. Moreover, they may experience an overfitting problem when the best hyper-parameters are obtained from only a particular test dataset. Thus, we applied grid search with a cross-validation strategy to easily obtain the best hyper-parameters by exploring the combinations of grouped hyper-parameters [32]. Here, five cross-validation tasks are used as the most popular [25], [32], [33]. First, different hyper-parameters for each algorithm within reasonable ranges are set up. Second, the data is equally separated into four folds that are utilized for training, and the remaining fold is used for evaluation of prediction performance. Finally, each algorithm searched for the best hyper-parameters by evaluating the performance based on the accuracy of all combinations of grouped hyper-parameters. This task is repeated when five cross-validations were completed.

IV. EXPERIMENTS

A. EXPERIMENTAL SETTINGS

We collected the PSE schedules of a power generator in South Korea in 2019. Among them, 7,416 training instances were employed in the grid search phase. 1,344 test instances representing for every two weeks in February, May, August, and November were utilized to evaluate the prediction performance of the applied ML algorithms across seasons. SMP and fuel cost data were obtained from the KPX database.

An example of the input and output values, such as seasonal data, SMP data, and the PSE schedule on a particular day is shown in Table 3. Seasonal variables are simply presented as numerical values [34]. The units for the SMP and fuel cost are Korean republic won (KRW) for 1 Gcal, corresponding to the considered power generator. Each PSE schedule is expressed in binary values of 0 or 1 through preprocessing in Steps 1.1 and 2.1 presented in Fig. 3.

TABLE 3. Example of input and output values in a day.

	Time	01	02	…	24
Input	Daily	3	3	…	3
	Monthly	1	1	…	1
	Hourly	1	2	…	24
	Fuel cost	65,496	65,496	…	65,412
	SMP	107.99	107.05	…	113.89
Output	Daily	3	3	…	3

To evaluate the predictive ability of each algorithm through Step 2.2 in Fig. 3, the *accuracy*, *precision*, *recall*, and $F1$ metrics are adopted and defined as Equations (4)-(7). TP and FP are defined as the number of electricity production tasks that are correctly and incorrectly estimated from the number of true production tasks, respectively. Further, TN and FN are the number of correctly and incorrectly predicted tasks among the number of false production tasks, respectively.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (7)$$

These measures are typically utilized to examine the performance in many classification problems [35], [36], [37]. The $F1$ metric is the weighted average of precision and recall scores. Based on this metric, we can consider the prediction ability of an algorithm in terms of both precision and recall. The experiments are executed in the following hardware settings: Ryzen 3900X CPU and 32 GB RAM on 64-bit Windows. The Python SciKit-learn library was used to implement the ML algorithms [38].

The obtained best hyper-parameters of each algorithm from grid search with cross-validation strategy through Steps 1.2-1.4 are listed in Table 4. The hyper-parameter values C, G, and L are cost, gamma, and learning rate, respectively. Notably, the hyper-parameters of each algorithm are differently acquired depending on whether SMP data is used or not. Therefore, the grid search is essential to be addressed regardless of ML algorithms when the considered input variables change. The sets of the designed hyper-parameters for each algorithm for Step 1.3 are introduced in Table 8 in the Appendix.

TABLE 4. Best hyper-parameters observed by grid search for each algorithm.

Algorithm	Hyper-parameters	
	Without SMP	Without SMP
DT	Max depth: 3	Max depth: 4
SVM	C: 0.1, G: 0.001	C: 0.1, G: 0.001
ET	Tree numbers: 80 Max depth: 2	Tree numbers: 20 Max depth: 4
RF	Tree numbers: 80 Max depth: 2	Tree numbers: 80 Max depth: 4
GB	L: 0.01 Tree numbers: 40 Max depth: 2	L: 0.001 Tree numbers: 40 Max depth: 4
CB	Tree numbers: 4 Iteration: 1,000	Tree numbers: 6 Iteration: 1,000
XGB	Tree numbers: 80 Max depth: 6	Tree numbers: 100 Max depth: 6

B. EXPERIMENT RESULTS

Figs. 4 (a)-(d) visualizes the accuracy changes obtained through grid search for SVM, ET, RF, and XGB among all algorithms when SMP data is additionally used as an input variable. SVM shows less effective performances. The others are approximately the same, but XGB has the best prediction performances of the day-ahead PSE schedule. These results indicate that the prediction performances change according to the hyper-parameters; thus, it is essential to investigate the best hyper-parameters by using grid search.

Figs. 5 (a) and (b) presents the receiver operating characteristic (ROC) curve results of each algorithm according to the variables used; their area under the curve (AUC) results are displayed in Table 5. For each algorithm, a ROC curve closer to the red dotted line indicates that this algorithm precisely forecasts the PSE schedule. In contrast, a ROC curve closer to the black dotted line represents random prediction. Furthermore, the AUC indicates the accurate prediction ability of an algorithm; a value closer to 1 means that an algorithm perfectly forecasts a day-ahead PSE schedule.

As shown in Figs. 5 and Table 5, SVM has the same performances for PSE schedules regardless of the usage of SMP data. Although RF and CB underperform regarding the prediction of PSE schedules when SMP data is used, the others improved their performances. Specifically, XGB exhibits the best prediction performances (i.e., 0.960). These results indicate that SMP information is necessary to enhance the prediction performances of the day-ahead PSE schedule. Since SMP prediction models have been developed [39], [40] to show excellent performances, we believe that the SMP can be utilized as an input variable. In addition, the boosting ensemble approach is more suitable for estimating a day-ahead PSE schedule than the other approaches. Additionally, XGB is likely to accurately forecast PSE schedules, whereas SVM tends to randomly predict PSE schedules.

TABLE 5. AUC results of each model against used variables.

Algorithm	Used variables	
	Without SMP	With SMP
DT	0.912	0.915
SVM	0.733	0.733
ET	0.901	0.918
RF	0.929	0.919
GB	0.911	0.915
CB	0.950	0.945
XGB	0.952	0.960

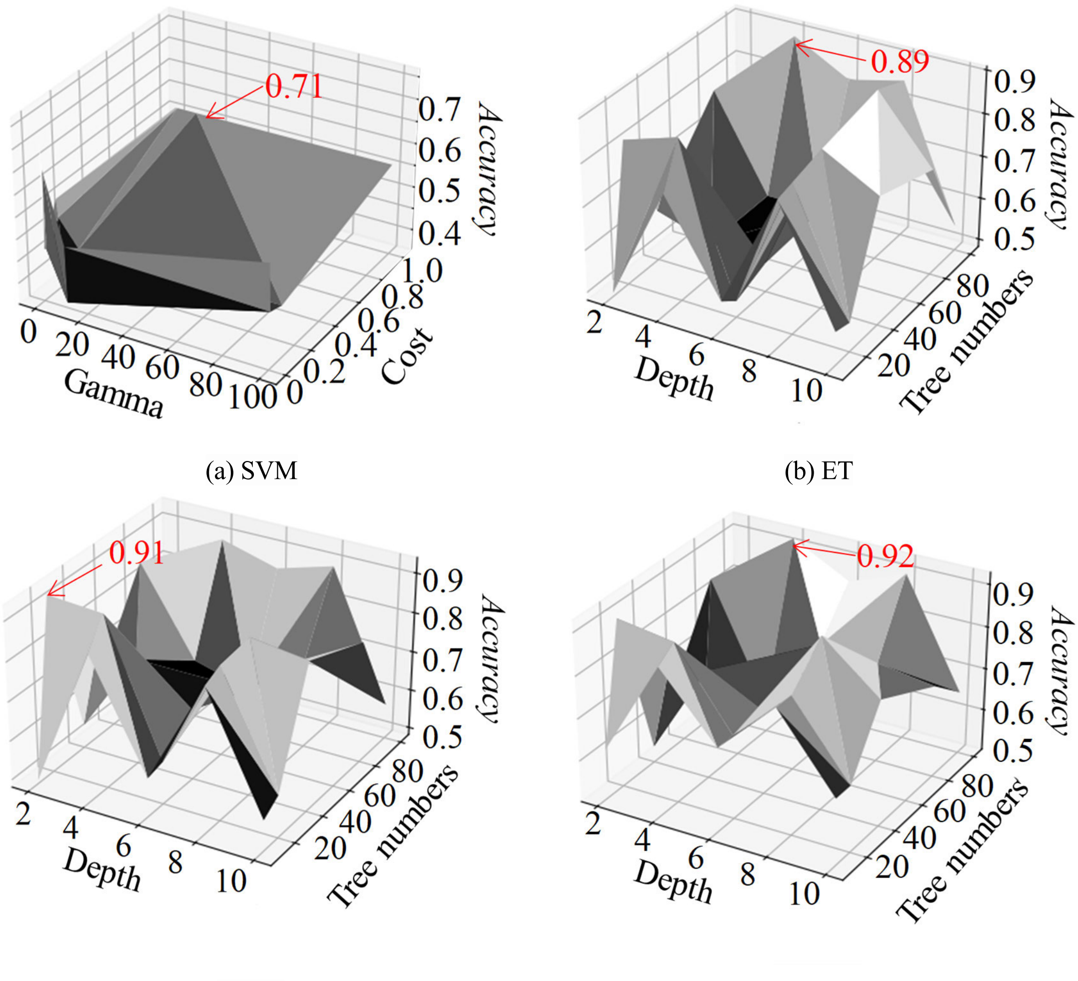
Sensitivity analysis is carried out using D1-D6 datasets which are set up as presented in Table 6.

D1, D2: Testing instances for the first-second and the third-forth weeks in December, March, June, and September, and training instances for the remaining instances

D3, D4: Testing instances for the first-second and the third-forth weeks in January, April, July, and October, and training instances for the remaining instances

D5, D6: Testing instances for the first-second and the third-forth weeks in February, May, August, and November, and training instances for the remaining instances

The results for sensitivity analysis are presented in Figs. 6 (a) and (b). The findings suggest that there are performance differences between the models according to the approach that divides into training and testing datasets. However, it is important to note that although no one model is



(a) SVM (b) ET

(c) RF (d) XGB

FIGURE 4. Detailed training results of SVM, ET, RF, and XGB according to grid search.

TABLE 6. The results for sensitivity analysis using D1-D6.

Algorithm	D1	D2	D3	D4	D5	D6	Avg.	Std.
DT	0.87	0.84	0.81	0.89	0.85	0.93	0.87	0.04
SVM	0.64	0.61	0.61	0.68	0.62	0.78	0.66	0.06
ET	0.84	0.81	0.80	0.89	0.83	0.84	0.84	0.03
RF	0.89	0.81	0.81	0.85	0.86	0.93	0.86	0.04
GB	0.87	0.78	0.80	0.9	0.85	0.93	0.85	0.05
CB	0.89	0.84	0.81	0.91	0.85	0.93	0.87	0.04
XGB	0.90	0.85	0.83	0.89	0.86	0.94	0.88	0.03

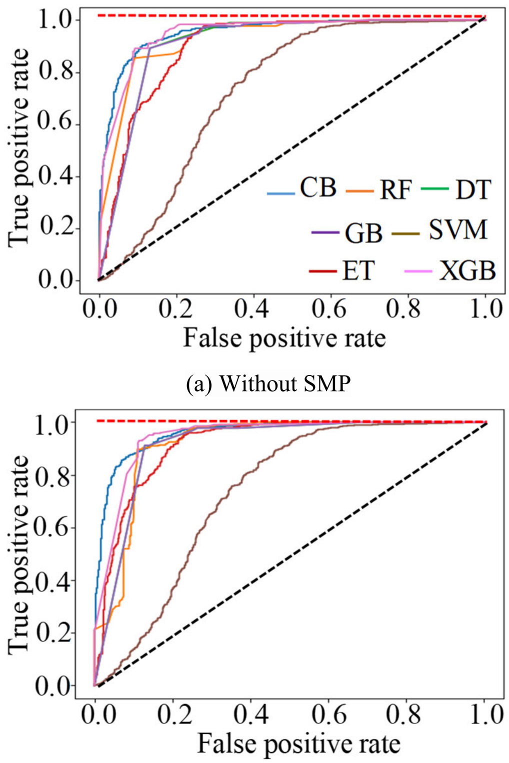
TABLE 7. F1 scores observed by each algorithm according to seasons (bold means best performance).

Algorithm	Spring	Summer	Autumn	Winter	Avg.	Std.
DT	0.942	0.906	0.674	0.934	0.864	0.111
SVM	0.624	0.671	0.567	0.708	0.642	0.053
ET	0.876	0.933	0.601	**0.936**	0.837	0.138
RF	0.876	0.976	0.611	0.914	0.844	0.139
GB	0.962	0.906	0.674	0.934	0.869	0.114
CB	0.922	0.952	0.603	0.926	0.851	0.144
XGB	**0.973**	**0.982**	**0.765**	0.928	**0.912**	0.087

dominant in performance for all datasets, XGB consistently demonstrates better performances than the other models.

In this section, the prediction performances among the algorithms considered with respect to each season are compared since the change in the day-ahead PSE schedule largely depends on the season. In general, relatively large changes are observed in spring and autumn, whereas comparatively small changes are observed in summer and winter. This means that the prediction of a day-ahead PSE schedule is more complicated in spring and autumn than in other seasons.

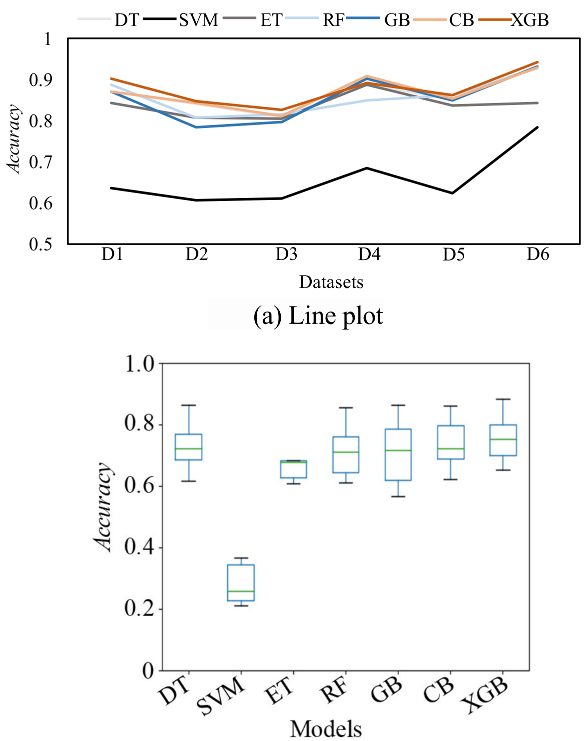
The $F1$ scores obtained by each algorithm against each season are presented in Table 7. Here, bold numbers indicate the best performances among all algorithms. In autumn, all algorithms show less effectiveness to forecast day-ahead PSE schedules. Interestingly, the prediction performances of XGB are relatively high, exhibiting $F1$ score over 70%.



(a) Without SMP

(b) With SMP

FIGURE 5. Detailed testing results of SVM, ET, RF, and XGB.



(a) Line plot

(b) Box plot

FIGURE 6. The results of sensitivity analysis.

Moreover, for the other seasons, XGB identifies a day-ahead PSE schedule with over 90 % $F1$ score. As a conclusion, XGB is successful in observing day-ahead PSE schedules for all seasons. Therefore, it is believed that this model could contribute to operating the power generator and maximize revenue in the real world.

TABLE 8. Designed hyper-parameters of each algorithm.

Algorithm	Hyper-parameters
DT	Max depth: {1, 2, 3, 4, 5, 6, 7, 8}
SVM	$C = \{0.1, 1, 10, 100\}$, $L = \{0.001, 0.01, 0.1, 1\}$
ET	Tree numbers = {10, 20, 40, 80, 100} Max depth = {2, 4, 6, 8, 10}
RF	Tree numbers = {10, 20, 40, 80, 100} Max depth = {2, 4, 6, 8, 10}
GB	L = {0.001, 0.01, 0.1, 1} Tree numbers = {10, 20, 40, 80, 100} Max depth = {2, 4, 6, 8, 10
CB	Tree numbers = {2, 4, 6, 8, 10} Iteration = {100, 500, 700, 1000}
XGB	Tree numbers = {10, 20, 40, 80, 100} Max depth = {2, 4, 6, 8, 10}

V. CONCLUSION

In this study, we applied the most popular ML algorithms to investigate the most suitable algorithm for predicting the PSE from power exchange. The proposed algorithms learn to understand the relationship between input and output values. These methods perform grid search with a cross-validation strategy to search for the best hyper-parameters that exhibit excellent prediction performance.

The research presented in this paper aims to explore a suitable ML algorithm for predicting the PSE day-ahead. To this end, the study conducts a numerical comparative analysis between different ML algorithms, including DT, SVM, ET, RF, GB, CB, and XGB. The findings in Fig. 5 and Table 5 show that the SMP variable plays a crucial role in enhancing the performance of PSE prediction when applied XGB. This finding has important practical implications, as an accurate prediction of SMP can facilitate the accurate prediction of PSE in real-world settings.

To further evaluate the performance of the different ML algorithms, the study presents the results of the analysis across four different seasons, as shown in Table 7. The findings demonstrate that the boosting approach consistently outperforms the other algorithms across all seasons. In particular, XGB outperforms the other algorithms with the highest prediction accuracy and the lowest error rate. This finding is consistent with previous studies related to energy prediction problems [25], [41] and provides further evidence for the effectiveness of the XGB algorithm.

Still, the study has certain limitations. As it is not easy to obtain PSE schedules in the real-world, this study only used PSE schedules for one year as the training and test datasets. Moreover, we compared PSE prediction performances of ML algorithms without implementation of deep learning techniques since deep learning techniques are known underperforming when the number of training instances is low [42]. Hence, in the future, we plan to compare the PSE prediction performances of deep learning techniques through more data collection.

APPENDIX

The appendix presents the designed hyper-parameters for each algorithm. The method for deriving hyper-parameters when deploying the ML algorithms was based on the recommendations provided in [24]. Based on these sets of hyper-parameters, each algorithm searches for the best hyper-parameters according to the input variables.

REFERENCES

[1] Korea Power Exchange. (2023). *Price-Setting Power Generation Schedule*. [Online]. Available: http://new.kpx.or.kr/menu.es?mid=a1040104000

[2] L. Suganthi and A. A. Samuel, ''Energy models for demand forecasting—A review,'' *Renew. Sustain. Energy Rev.*, vol. 16, no. 2, pp. 1223–1240, Feb. 2012.

[3] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, and H. Zareipour, ''Energy forecasting: A review and outlook,'' *IEEE Open Access J. Power Energy*, vol. 7, pp. 376–388, 2020.

[4] I. Ghalehkhondabi, E. Ardjmand, G. R. Weckman, and W. A. Young, ''An overview of energy demand forecasting methods published in 2005–2015,'' *Energy Syst.*, vol. 8, no. 2, pp. 411–447, May 2017.

[5] N. G. Paterakis, E. Mocanu, M. Gibescu, B. Stappers, and W. van Alst, ''Deep learning versus traditional machine learning methods for aggregated energy demand prediction,'' in *Proc. IEEE PES Innov. Smart Grid Technol. Conf. Eur.*, Sep. 2017, pp. 1–6.

[6] S. Stankoski, I. Kiprijanovska, I. Ilievski, J. Slobodan, and H. Gjoreski, ''Electrical energy consumption prediction using machine learning,'' in *Proc. Int. Conf. ICT Innov.*, in Communications in Computer and Information Science, 2019, pp. 1–10.

[7] I. Abdou and M. Tkiouat, ''Unit commitment problem in electrical power system: A literature review,'' *Int. J. Electr. Comput. Eng.*, vol. 8, no. 3, p. 1357, Jun. 2018.

[8] Q. P. Zheng, J. Wang, and A. L. Liu, ''Stochastic optimization for unit commitment—A review,'' *IEEE Trans. Power Syst.*, vol. 30, no. 4, pp. 1913–1924, Jul. 2015.

[9] N. Muralikrishnan, L. Jebaraj, and C. C. A. Rajan, ''A comprehensive review on evolutionary optimization techniques applied for unit commitment problem,'' *IEEE Access*, vol. 8, pp. 132980–133014, 2020.

[10] H. Abdi, ''Profit-based unit commitment problem: A review of models, methods, challenges, and future directions,'' *Renew. Sustain. Energy Rev.*, vol. 138, Mar. 2021, Art. no. 110504.

[11] M. Leśko, W. Bujalski, and K. Futyma, ''Operational optimization in district heating systems with the use of thermal energy storage,'' *Energy*, vol. 165, pp. 902–915, Dec. 2018.

[12] C. Qin, Q. Yan, and G. He, ''Integrated energy systems planning with electricity, heat and gas using particle swarm optimization,'' *Energy*, vol. 188, Dec. 2019, Art. no. 116044.

[13] B. Talebi, F. Haghighat, P. Tuohy, and P. A. Mirzaei, ''Optimization of a hybrid community district heating system integrated with thermal energy storage system,'' *J. Energy Storage*, vol. 23, pp. 128–137, Jun. 2019.

[14] R. Nourollahi, V. S. Tabar, S. G. Zadeh, and A. Akbari-Dibavar, ''A hybrid optimization approach to analyze the risk-constrained operation of a residential hybrid energy system incorporating responsive loads,'' *Comput. Chem. Eng.*, vol. 157, Jan. 2022, Art. no. 107603.

[15] P. Chen, T. Pedersen, B. Bak-Jensen, and Z. Chen, ''ARIMA-based time series model of stochastic wind power generation,'' *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 667–676, May 2010.

[16] K. Goswami and A. B. Kandali, ''Electricity demand prediction using data driven forecasting scheme: ARIMA and SARIMA for real-time load data of assam,'' in *Proc. Int. Conf. Comput. Perform. Eval. (ComPE)*, Jul. 2020, pp. 570–574.

[17] Y. Wang, J. Wang, G. Zhao, and Y. Dong, ''Application of residual modification approach in seasonal ARIMA for electricity demand forecasting: A case study of China,'' *Energy Policy*, vol. 48, pp. 284–294, Sep. 2012.

[18] Y. Guo, J. Wang, H. Chen, G. Li, J. Liu, C. Xu, R. Huang, and Y. Huang, ''Machine learning-based thermal response time ahead energy demand prediction for building heating systems,'' *Appl. Energy*, vol. 221, pp. 16–27, Jul. 2018.

[19] D. Chakraborty and H. Elzarka, ''Advanced machine learning techniques for building performance simulation: A comparative analysis,'' *J. Building Perform. Simul.*, vol. 12, no. 2, pp. 193–207, Mar. 2019.

[20] J. Runge and E. Saloux, ''A comparison of prediction and forecasting artificial intelligence models to estimate the future energy demand in a district heating system,'' *Energy*, vol. 269, Apr. 2023, Art. no. 126661.

[21] S. Liang, T. Deng, A. Huang, N. Liu, and X. Jiang, ''Energy consumption prediction using the GRU-MMattention-LightGBM model with features of prophet decomposition,'' *PLoS ONE*, vol. 18, no. 1, Jan. 2023, Art. no. e0277085.

[22] J. Bergstra and Y. Bengio, ''Random search for hyper-parameter optimization,'' *J. Mach. Learn. Res.*, vol. 13, no. 2, pp. 1–15, 2012.

[23] Y. Chae, M. Kim, and S.-H. Yoo, ''Does natural gas fuel price cause system marginal price, vice-versa, or neither? A causality analysis,'' *Energy*, vol. 47, no. 1, pp. 199–204, Nov. 2012.

[24] B. Mahesh, ''Machine learning algorithms-a review,'' *Int. J. Sci. Res.*, vol. 9, pp. 381–386, Jan. 2020.

[25] B. Carrera and K. Kim, ''Comparison analysis of machine learning techniques for photovoltaic prediction using weather sensor data,'' *Sensors*, vol. 20, no. 11, p. 3129, Jun. 2020.

[26] J. Liang, Z. Qin, S. Xiao, L. Ou, and X. Lin, ''Efficient and secure decision tree classification for cloud-assisted online diagnosis services,'' *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 4, pp. 1632–1644, Jul. 2021.

[27] D. Kumar, ''Decision tree classifier: A detailed survey,'' *Int. J. Inf. Decis. Sci.*, vol. 12, pp. 246–269, Jan. 2020.

[28] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, ''Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection,'' *IEEE Access*, vol. 6, pp. 33789–33795, 2018.

[29] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, ''A comparison of random forest variable selection methods for classification prediction modeling,'' *Expert Syst. Appl.*, vol. 134, pp. 93–101, Nov. 2019.

[30] J. H. Friedman, ''Stochastic gradient boosting,'' *Comput. Statist. Data Anal.*, vol. 38, no. 4, pp. 367–378, Feb. 2002.

[31] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, ''CatBoost: Unbiased boosting with categorical features,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–22.

[32] A. Ogunleye and Q. Wang, ''XGBoost model for chronic kidney disease diagnosis,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 17, no. 6, pp. 2131–2140, Nov. 2020.

[33] V. Domala, W. Lee, and T.-W. Kim, ''Wave data prediction with optimized machine learning and deep learning techniques,'' *J. Comput. Des. Eng.*, vol. 9, no. 3, pp. 1107–1122, Jun. 2022.

[34] D. Lee and K. Kim, ''Recurrent neural network-based hourly prediction of photovoltaic power output using meteorological information,'' *Energies*, vol. 12, no. 2, p. 215, Jan. 2019.

[35] S. Chang, W. Dong, and H. Jun, ''Use of electroencephalogram and long short-term memory networks to recognize design preferences of users toward architectural design alternatives,'' *J. Comput. Des. Eng.*, vol. 7, no. 5, pp. 551–562, Oct. 2020.

[36] T. Baek and Y.-G. Lee, ''Traffic control hand signal recognition using convolution and recurrent neural networks,'' *J. Comput. Design Eng.*, vol. 9, no. 2, pp. 296–309, Feb. 2022.

[37] J. Song, J. Lee, K. Ko, W.-D. Kim, T.-W. Kang, J.-Y. Kim, and J.-H. Nam, ''Unorganized point classification for robust NURBS surface reconstruction using a point-based neural network,'' *J. Comput. Des. Eng.*, vol. 8, no. 1, pp. 392–408, Jan. 2021.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, and M. Blondel, ''Scikit-learn: Machine learning in Python,'' *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Jan. 2012.

[39] F. H. Jufri, S. Oh, and J. Jung, ''Day-ahead system marginal price forecasting using artificial neural network and similar-days information,'' *J. Electr. Eng. Technol.*, vol. 14, no. 2, pp. 561–568, Mar. 2019.

[40] W. Yang, S. Sun, Y. Hao, and S. Wang, "A novel machine learning-based electricity price forecasting model based on optimal model selection strategy," *Energy*, vol. 238, Jan. 2022, Art. no. 121989.

[41] A. Torres-Barrán, Á. Alonso, and J. R. Dorronsoro, "Regression tree ensembles for wind energy and solar radiation prediction," *Neurocomputing*, vols. 326–327, pp. 151–160, Jan. 2019.

[42] A. Belhi, A. K. Al-Ali, A. Bouras, S. Foufou, X. Yu, and H. Zhang, "Investigating low-delay deep learning-based cultural image reconstruction," *J. Real-Time Image Process.*, vol. 17, no. 6, pp. 1911–1926, Dec. 2020.

DONGHUN LEE received the M.S. degree in industrial and management engineering from Incheon National University, South Korea, in 2019. He is currently pursuing the Ph.D. degree with the Industrial Intelligence Laboratory, Department of Industrial and Management Engineering.

His current research interests include deep learning and reinforcement learning-based decision-making for logistics and manufacturing systems, the efficiency operations of energy systems, and business intelligence.

KWANHO KIM received the Ph.D. degree from the Department of Industrial Engineering, Seoul National University, in 2012.

He is currently an Associate Professor with the Department of Industrial and Management Engineering, Incheon National University. His research interests include statistical methodologies to analyze and manage information, such as machine learning, text mining, information retrieval, and recommendation systems.

SANG HWA SONG received the Ph.D. degree from the Korea Advanced Institute of Science and Technology, in 2003.

He is currently a Professor with the Graduate School of Logistics, Incheon National University. His research interest includes the application of mathematical optimization to energy industry.