energies

MDPI

Article

An Ensemble Learner-Based Bagging Model Using Past Output Data for Photovoltaic Forecasting

Sunghyeon Choi [1] and Jin Hur [2,*]

[1] Enel X, Seoul 04511, Korea; sunghyeon.choi@enel.com
[2] Department of Energy Grid, Sangmyung University, Seoul 03016, Korea
* Correspondence: jinhur@smu.ac.kr; Tel.: +82-02-781-7576

Received: 19 February 2020; Accepted: 17 March 2020; Published: 19 March 2020

Abstract: As the world is aware, the trend of generating energy sources has been changing from conventional fossil fuels to sustainable energy. In order to reduce greenhouse gas emissions, the ratio of renewable energy sources should be increased, and solar and wind power, typically, are driving this energy change. However, renewable energy sources highly depend on weather conditions and have intermittent generation characteristics, thus embedding uncertainty and variability. As a result, it can cause variability and uncertainty in the power system, and accurate prediction of renewable energy output is essential to address this. To solve this issue, much research has studied prediction models, and machine learning is one of the typical methods. In this paper, we used a bagging model to predict solar energy output. Bagging generally uses a decision tree as a base learner. However, to improve forecasting accuracy, we proposed a bagging model using an ensemble model as a base learner and adding past output data as new features. We set base learners as ensemble models, such as random forest, XGBoost, and LightGBMs. Also, we used past output data as new features. Results showed that the ensemble learner-based bagging model using past data features performed more accurately than the bagging model using a single model learner with default features.

Keywords: photovoltaic power forecasting; machine learning; lagged data; ensemble; decision tree; bagging; random forest; XGBoost; Light GBM

1. Introduction

The 196 countries that signed the Paris Agreement in 2015 agreed to make efforts to reduce their artificial greenhouse gas emissions to zero in the second half of the 21st century. This agreement highlighted the need to generate energy through renewable resources and was motivated by research on how to manage and integrate variable power generation systems, such as solar and wind power, into the grid [1]. Focusing on solar energy, the proportion of solar energy in the power system has been growing with the large drop in photovoltaic (PV) prices [2]. Photovoltaic power is now one of the fastest growing renewable energy technologies, and is ready to play an important role in the future global electricity generation mix. According to International Energy Agency(IEA)'s Renewable 2018, solar power plants accounted for more than two-thirds of the world's net electricity capacity growth in 2017. The world's total renewable-based power capacity is expected to grow 50 percent between 2019 and 2024, with solar power accounting for 60 percent of the rise [3,4].

The high penetration of PV in the power system provides many economic benefits, but solar energy with the characteristics of variability and intermittency can bring challenge to the safe operation and reliability of the power system. Power system operators must ensure an accurate balance between electricity production and consumption at any time, and effective forecasting techniques have become important to prepare for the grid integration of renewable energy sources with this instability [5]. The solar power forecasting has following advantages: (1) the effective operation of the power grid [6],

(2) the optimal management of the energy fluxes occurring into the solar system, (3) estimating the reserves, (4) scheduling the power system, (5) congestion management, (6) the optimal management of the storage with the stochastic production, (7) trading the produced power in the electricity market, and (8) reduction of the costs of solar power generation. Accurate predictions can not only contribute to the reduction of uncertainty in power generation forecasts, but also add to the stable operation of the system. In addition, it allows photovoltaic plant operators to avoid penalties that may arise from differences between forecasted and produced energy, and benefits from cost saving for energy consumers [7–10].

The forecast of solar power can be performed by several methods and machine learning is the typical method that we focused on in this study. Many algorithms on photovoltaic power forecasting have been proposed and remarkable results have been achieved by experts and scholars. The following is the state-of-the-art of photovoltaic prediction based on machine learning methods. Artificial Neural Networks (ANNs) are useful in data analysis and prediction, and increasingly used for nonlinear regression and classification problems [11]. Markov chains are stochastic processes having the Markov property. In a Markov process, the present state fully captures all the information that could affect the future evolution of the process [12]. K-Nearest Neighbor (k-NN) is one of the simplest machine learning algorithms based on pattern recognition. It compares the current state with training sets in a feature space [13]. Support Vector Machine (SVM) stands out for its ability to deal with nonlinear problems. SVM has three main parameters that highly affect the performance of the technique and these regulate the kernel function, used to transform the predictors. SVM has shown great potential in many studies [14]. Recently, Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machines (LightGBMs), and deep neural networks have been used in many studies and these have proved useful in predicting time series [15–17]. Light GBM and XGBoost have recently gained attention in the Kaggle platform as having good performance. However, both XGBoost and LightGBM tend to overfit at times, as they are both based on the decision trees. To improve forecasting accuracy and reduce overfitting at the same time, we proposed an ensemble learner-based bagging model.

2. Machine Learning

Machine learning is a subfield of computer science and is classified as an artificial intelligence. It has an advantage that a model can solve problems that are impossible to be represented by explicit algorithms. [18] Machine learning models the relationships between inputs and outputs, even though the representation is impossible. There are three main ways of learning methods: supervised learning, unsupervised learning, and ensemble learning. In supervised learning, the computer is given inputs and outputs, and the goal is to learn a general rule that maps inputs and outputs [19]. In contrary with supervised learning, an unsupervised learning model does not need outputs. It is able to find hidden structure in its inputs [20]. The basic concept of ensemble learning is to train multiple base learners as ensemble members, and to combine their predictions into a single output. In general, the base learner of the ensemble model is a decision tree and the ensemble model is known to have better results than the method of using a single model. In this paper, we propose an ensemble model bagging, and as a base learner we used an ensemble model, not a decision tree. Thus, we call it an ensemble of ensemble.

2.1. Ensemble Methods

2.1.1. Decision Tree

A decision tree is a representative base learner used in most ensemble models of machine learning. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. At each node in the tree, the tree splits into branches based on its condition. Depending on the outcome of the condition, either the left or the right sub-branch of the tree is selected. Eventually, a leaf node is reached where the branch that does not split any further is the decision. [21–23]. A decision tree has some advantages, it is easy to understand, requires less data

cleaning, the data type is unconstrained, and it is a nonparametric method. But it causes overfitting. Overfitting is one of the most practical difficulties for decision tree models.

2.1.2. Bagging

Bagging used in statistical classification and regression is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms. The algorithm reduces variance, which affects the performance of the forecasting model and helps to prevent overfitting. [24]. The bagging model is shown in Figure 1 and algorithm as follows:

(1) Create B bootstrap samples $\mathcal{L}^{*(b)}$ $(b = 1, \cdots, \mathrm{B})$, where $\mathcal{L} = (x_i, y_i)_{i=1}^{n}$ are learning sets;
(2) For each bootstrap sample, $\mathcal{L}^{*(b)}$, build a forecasting model $f^{(b)}(x)$;
(3) Aggregate B forecasting model to final model $\hat{f}$;
(4) When it comes to the regression model, final models are the average value of the sum of each forecasting model, $\hat{f}(x) = argmax_k(\sum_{b=1}^{B} I(f^{(b)}(x) = k))/\mathrm{B}$.

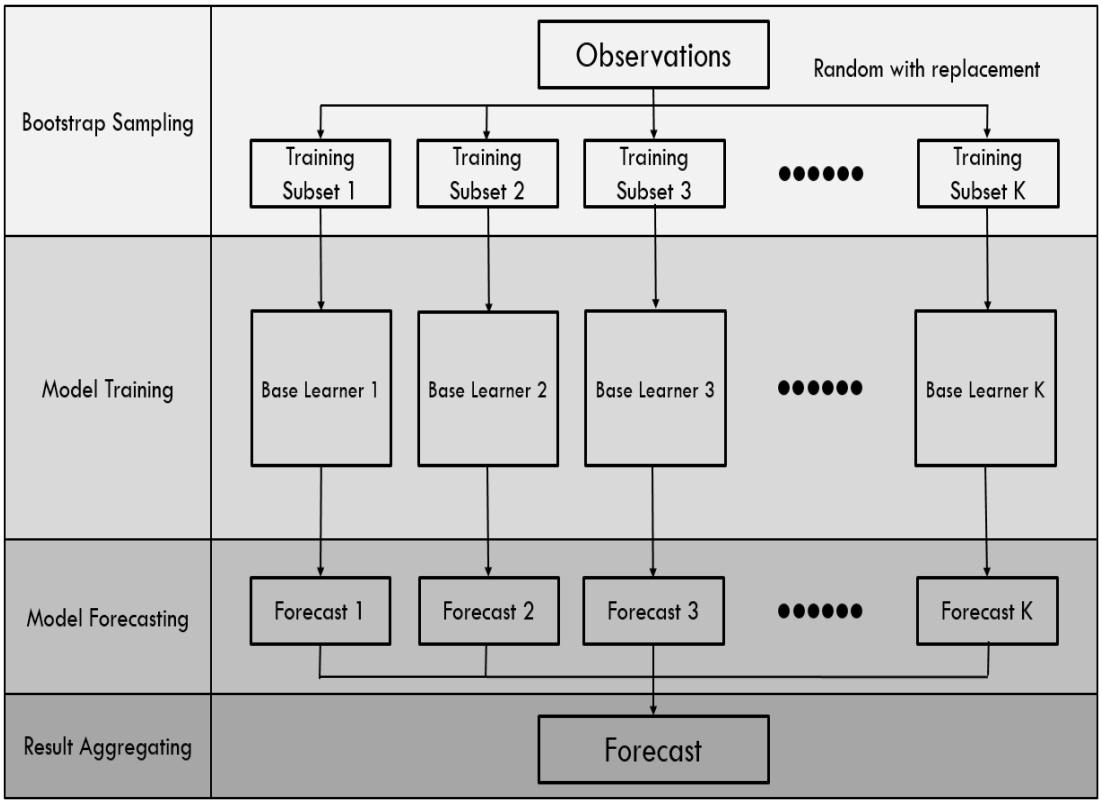

Figure 1. Bagging model algorithm.

As to why bagging can improve predictability, it can be explained based on the fact that the expected loss of the average forecasting model is less than the expected loss of a single forecasting model. Model $\hat{f}(x)$, built using given learning sets $\mathcal{L}$, highly depends on the $\mathcal{L}$. In order to highlight this, it is written as $\hat{f}(x) = f(x, \mathcal{L})$ and for a given forecasting model, the mean forecasting model is defined as $f_A(x) = E_{\mathcal{L}} f(x, \mathcal{L})$. Here, the expected value uses the distribution of the population from which the training data were obtained, and the theorem below shows that the expected loss of the average forecasting model is less than the expected loss of a single forecasting model.

Let us say that (X, Y) is a future observation that is independent of $\mathcal{L}$. For the square loss function, $L(y, a) = (y - a)^2$, the expected losses of $f(x, \mathcal{L})$ and $f_A(x)$, R, and R_A are defined as follows:

$$R = E_{(X,Y)} E_{\mathcal{L}} L(Y, f(X, \mathcal{L})), R_A = E_{(X,Y)} L(Y, f_A(X)) \quad (1)$$

As the square function is a convex function, Equation (2) is established by the Jensen inequality:

$$E_{(X,Y)}E_{\mathcal{L}}f^2(X,\mathcal{L}) \geq E_{(X,Y)}f_A(X)^2 \quad (2)$$

Then, R is always greater than or equal to R_A [25]. The bagging model can be applied not only in PV forecasting but also in various fields [26–29].

2.1.3. Random Forest

Random forests randomize not only input data but also input variables. By averaging results from multiple trees, it is able to reduce the variance and the overall performance of the model improves [30]. In particular, when a random forest has a large number of input variables, it often shows better performance than bagging and boosting. The random forest algorithm follows:

(1) For sets $\mathcal{L} = (x_i, y_i)_{i=1}^n$, $x_i \in \mathbb{R}$, create a bootstrap sample $\mathcal{L}^* = \{y_i{}^*, x_i{}^*\}_{i=1}^n$ using n of data.
(2) In the bootstrap sample $\mathcal{L}^*$, only k of the input variables are randomly selected to create decision trees. At this time, the decision trees are carried out to the specified level of s.
(3) These generated decision trees are linearly combined to create the final learner.

While bagging is a method of reconstructing data to make the model diverse, random forest reconstructs variables as well as data, and reduces the variance of the model, resulting in better performance than general bagging. The variance of the bagging model consists of the variance of the trees and their covariance, respectively (Equation (3)).

$$\mathrm{Var}(\mathrm{X}+\mathrm{Y}) = Var(X) + Var(Y) + 2Cov(X,Y) \quad (3)$$

Although samples were selected randomly with replacement from the all data sets, as each tree has a large number of overlapping data, it is hard to say that they are independent. In short, $\mathrm{Cov}(\mathrm{X},\mathrm{Y})$ is not zero and it means that as the tree increases, the variance across the model may increase. A way to reduce the covariance between each tree is needed, which is accomplished by using random forests. The studies of using PV forecasting with a random forest can be found in references [31–34].

2.1.4. Boosting

Boosting is an ensemble model using decision trees as weak learners and building the model in a stagewise manner by optimizing a loss function [35]. It is a method that converts weak learners into strong learners.

In this paper, we used XGBoost and LightGBM, which are types of gradient boosting. Gradient boosting is a generalization of boosting to arbitrary differentiable loss functions. Gradient boosting is an accurate and effective off-the-shelf procedure that can be used for both regression and classification problems.

Let us say that there is a model h_0 that takes an input x and predicts the variable y (Equation (4)):

$$\mathrm{y} = h_0(x) + error \quad (4)$$

If the error is not unpredictable random noise, the most intuitive way to increase the forecasting performance is to eliminate the error. Removing this error is the basic concept of gradient boosting. In short, rather than predicting y, as the gradient boost moves to the next step, it predicts error and lowers the error. The process is shown in Equation (5):

$$\begin{aligned} error &= h_1(x) + error_2 \\ error_2 &= h_2(x) + error_3 \\ error_3 &= h_3(x) + error_4 \\ &\vdots \\ \mathrm{y} &= h_1(x) + h_2(x) + h_3(x) + h_4(x) + \ldots + small\ error \end{aligned} \tag{5}$$

Unlike gradient boosting, however, XGBoost adds a regularization term to the object function, which prevents the model from overfitting. Through the regularization term, XGBoost imposes penalties on complex models. The mathematical formula is given in Equation (6):

$$\mathrm{obj}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{t} \Omega(f_k) \tag{6}$$

where t is the number of trees, f_k is the output value of k^{th} tree, Ω is a regularization that measures the complexity of the model and avoids overfitting, and l is the distance between y_i and $\hat{y}_i$, which is used to measure the training error [36].

LightGBM is a gradient boosting framework that uses tree-based learning algorithms. It is designed to be distributed and efficient with the following advantages: faster training speed and higher efficiency, lower memory usage, better accuracy, support of parallel and GPU learning, and capable of handling large-scale data. While XGBoost uses levelwise loss, LightGBM uses leafwise loss to further reduce loss [Figure 2]. It is capable of being more than twice as fast as XGBoost based on the same parameters and requires a large amount of training data because it is sensitive to overfitting. XGboost and LightGBM are used in forecasting field and are found in references [37,38].

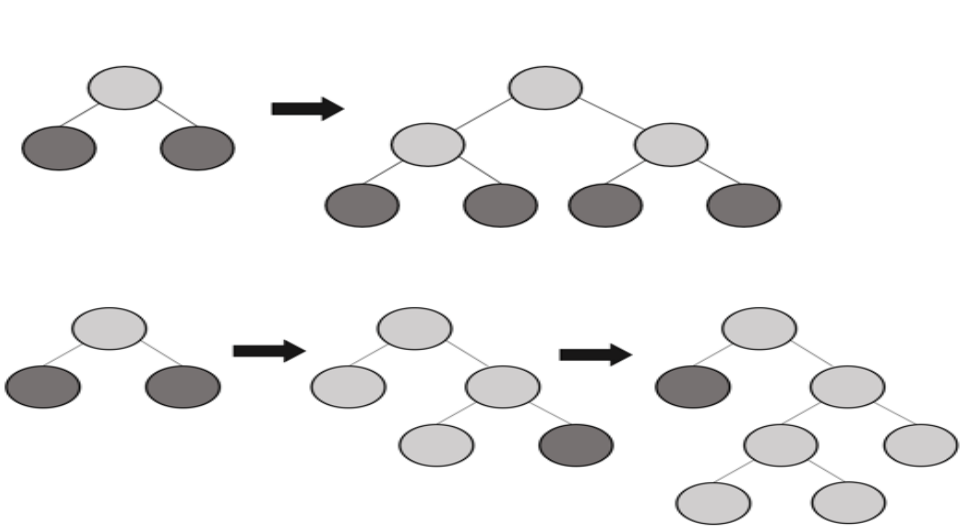

Figure 2. Levelwise (XGBoost) and Leafwise (LightGBM).

2.2. Ensemble Model

In this paper, we used an ensemble model as a base learner of the bagging model instead of a decision tree, which is widely used as a base learner of the bagging algorithm. In other words, the ensemble model itself is viewed as a base learner [Figure 3].

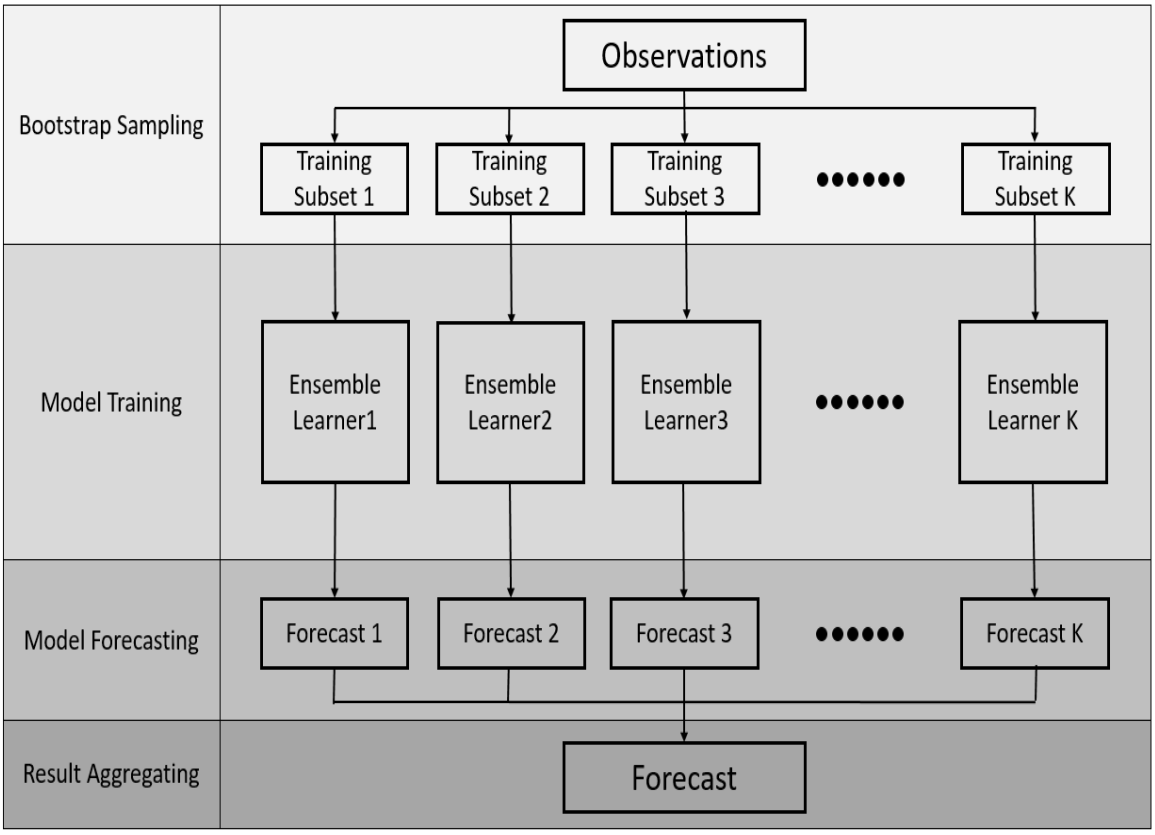

Figure 3. Ensemble learner-based bagging model algorithm.

Basically, using an ensemble model provides better performance than using a single model. The algorithms of boosting models, such as XGBoost and LightGBM used in this study, tend to be sensitive to hyperparameters. Forecasting performance changes, depending on how we set up the hyperparameter. If we use many models, accordingly there would be more hyperparameters created and it makes the result less susceptible to hyperparameters. It means that regardless of tuning the hyperparameter, we can expect that good performance can be ensured. It also reduces inefficient time consumption in hyperparameter tuning. In short, by using an ensemble model as a base learner of the bagging algorithm, we can expect two outcomes:

(1) better performance than single model by using an ensemble model as a base learner;
(2) good performance regardless of hyperparameter tuning.

3. Framework of Ensemble of Ensemble

3.1. Datasets and Preprocessing

In this study, the prediction model was modeled using one year of data (2016) of a PV plant in South Korea, with data consisting of hourly temperatures, humidity, irradiation, and actual output. Solar power is an energy source greatly affected by weather conditions. In particular, irradiation is the most influential factor in output, and many studies have recognized that it is very important to be able to predict the solar irradiation effectively. As for this, some overviews can be found in references [39–41]. As the power system in the Republic of Korea is operated one-hour based, the data used for forecasting solar power output were one-hour timeslot data. The ratio of training data to test data was set to 80:20, and the test data were set at the last three to five days of each month to take into account the monthly and seasonal characteristics rather than random extraction. The following lists the general characteristics of a PV plant in South Korea [42–46]:

- PV technology, crystalline silicon;
- rated power, 340 W;
- tracking, dual-axis tracking system;
- solar panel tilt angle, 25°;
- location, South Korea, 35.9° N latitude and 127.7° E longitude;
- average daily solar radiation, 2.56–5.48 kWh/m^2.

3.2. Feature Engineering and Selection

The data features used were date, time, humidity, temperature, irradiation, and actual output. The value at time t was greatly affected by the value at time t–1. The past values are known as lags and in this study two lagged outputs were added as new features.

First, the date, time, humidity, temperature, irradiation, one-day-ahead (D-1) and two-days-ahead (D-2) output were set as new features, and the output set as a label, i.e., the target. Second, the Pearson correlation coefficient was calculated to correlate the total variables. It has a value between −1 and 1. The Pearson correlation coefficient is mainly used to approximate the characteristics of variables by identifying one-to-one correspondence between continuous variables [47]. As one variable increases and the other increases, it becomes positive and closer to 1; in the other case, it becomes negative and closer to −1. If there is no relationship, it is close to zero. The formula for calculating the Pearson correlation coefficient of two n-dimensional vectors X and Y is given by Equation (7):

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \quad (7)$$

where $\overline{x}$ is the mean of X and $\overline{y}$ is the mean of Y.

The Pearson coefficient method was used to evaluate the correlation between features and label. The results are shown in Table 1.

Table 1. The Pearson coefficient between variables and photovoltaic output.

Variables	Pearson Coefficient	Variables	Pearson Coefficient
Hour	0.149857	Irradiation	0.956225
Temperature	0.291227	Lagged output (D-1)	0.925120
Humidity	−0.628640	Lagged output (D-2)	0.762727

(D-1: one-day-ahead D-2: two-days-ahead).

From Table 1, the following results were drawn:

- The order of correlation is irradiation, lagged output, humidity, temperature, and time (absolute valued based).
- There is a positive relationship with all variables except humidity.
- Lagged output has a relationship greater than the weather conditions except for irradiation.

3.3. Modeling and Results

The hyperparameters adjusted in this prediction model were of three types: a learning rate tuning parameter in an optimization algorithm that determines the step size at each iteration, the number of samples, and the maximum depth of the tree. The hyperparameters were tuned to 0.1, 100, and 5, in that order. for all models. To verify the effectiveness of the model and assess its performance, mean absolute error (MAE) and root mean square error (RMSE) were used as an indicator. MAE shows the average distance between the measured values and the model predictions. The root mean square error (RMSE) is more sensitive to big forecast errors, and hence is suitable for applications where

small errors are more tolerable and larger errors cause disproportionately high costs. It is probably the reliability factor that is most appreciated and used [23]. The MAE and RMSE are shown in Equations (8) and (9):

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{8}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{9}$$

where n is the number of samples in the test set, $\hat{y}_i$ is the forecasted photovoltaic power output, and y is the real photovoltaic power output. In this study, the major contributions are listed as follows:

- We used lagged output data as a new feature to improve forecast accuracy.
- We used an ensemble model as a base learner in bagging model, which gave better performance than the base learner using a single model. In this paper, we used a decision tree.

We named two features used in this study as follows:

- Default features—the features that used the given variables, including hour, temperature, humidity, and irradiation.
- New features—the features of past output data that were added to the default features, including hour, temperature, humidity, irradiation, D-1 output, and D-2 output.

Figures 4 and 5 show the predicted output by the base learner of the bagging model. First of all, use of the new features showed better accuracy than the output using the default features. As for the base learner, bagging with an ensemble base learner generally provided a better result than the one with a single model.

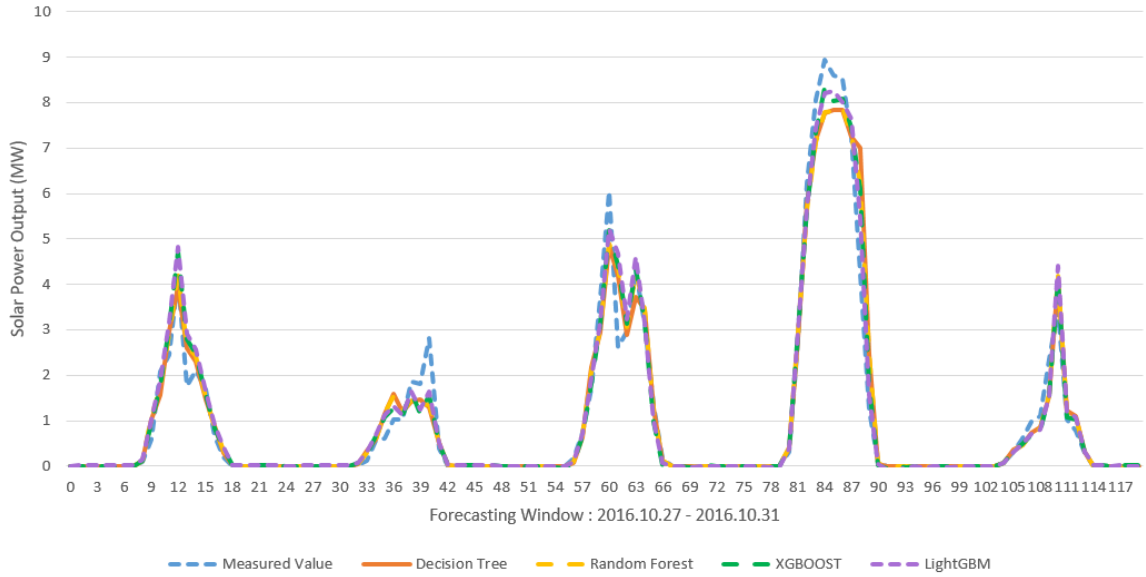

Figure 4. Comparison between the measured value and forecasted values (default features).

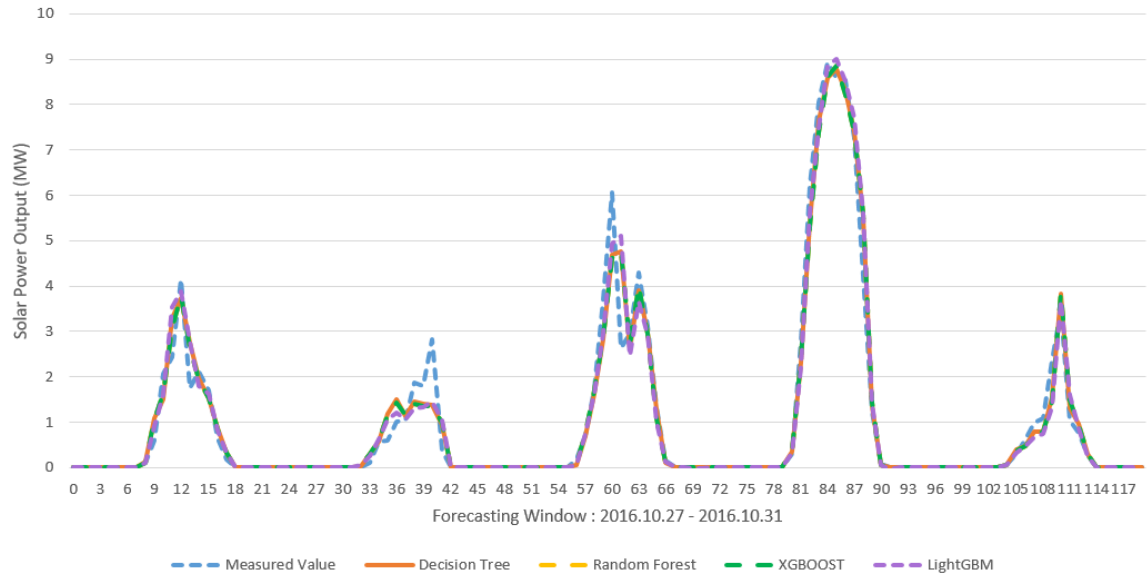

Figure 5. Comparison between the measured value and forecasted values (new features).

From Tables 2 and 3 and Figures 6–9, except for June and October, the error of the other 10 months decreased for MAE. Similarly, in regard to RMSE, the error of 9 months decreased except for June, July, and October. In May and August, especially, the error decreased remarkably, as shown. Tables 4 and 5 show how much monthly error decreased when using the new features.

Table 2. Mean absolute error (MAE) and root mean square error (RMSE) comparison in the default features.

	Default Features							
Base Learner	**Decision Tree**		**Random Forest**		**XGBoost**		**LightGBM**	
	MAE	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
Jan	0.19755	0.24290	0.18960	0.23644	0.19321	0.25418	0.19892	0.26533
Feb	0.31988	0.50108	0.29083	0.38489	0.29043	0.43317	0.28835	0.43567
Mar	0.31099	0.40583	0.29232	0.34503	0.28656	0.31936	0.28802	0.30982
Apr	0.22439	0.20006	0.22188	0.18288	0.20093	0.13914	0.20435	0.14659
May	0.36698	0.98405	0.36528	0.98833	0.37329	1.01233	0.37396	1.05930
Jun	0.21572	0.18469	0.21287	0.17216	0.20059	0.16217	0.20760	0.15663
July	0.22677	0.18154	0.22460	0.16996	0.20998	0.14262	0.21544	0.15201
Aug	0.50317	1.17744	0.49694	1.14226	0.50409	1.12412	0.48971	1.10444
Sep	0.16023	0.15951	0.14592	0.13070	0.14519	0.11332	0.14862	0.11784
Oct	0.18628	0.17626	0.18131	0.16307	0.17477	0.13875	0.17512	0.13226
Nov	0.23167	0.29501	0.22490	0.26160	0.22249	0.24148	0.22239	0.24293
Dec	0.37042	0.64288	0.35329	0.57893	0.34518	0.52126	0.33349	0.50318

Table 3. MAE and RMSE comparison in the new features.

Base Learner	New Features							
	Decision Tree		Random Forest		XGBoost		LightGBM	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Jan	0.18306	0.20791	0.17465	0.20080	0.16943	0.19153	0.16887	0.19261
Feb	0.26577	0.32186	0.23252	0.27144	0.22896	0.27347	0.22881	0.28659
Mar	0.19251	0.19801	0.18846	0.18871	0.19471	0.19648	0.19047	0.19747
Apr	0.13213	0.07050	0.12418	0.06371	0.11922	0.05734	0.12170	0.05746
May	0.25107	0.33340	0.24862	0.34152	0.25912	0.35030	0.26826	0.39825
Jun	0.23083	0.17935	0.21850	0.16174	0.22152	0.16184	0.22509	0.16368
July	0.21527	0.18678	0.20443	0.16226	0.18758	0.14779	0.19421	0.15449
Aug	0.29739	0.36561	0.31298	0.41447	0.30205	0.36531	0.29154	0.35976
Sep	0.14408	0.09420	0.13370	0.08136	0.13084	0.07199	0.13066	0.07343
Oct	0.17426	0.15925	0.17450	0.14947	0.18007	0.16552	0.18061	0.16895
Nov	0.16680	0.12603	0.14749	0.12773	0.18266	0.12687	0.18393	0.13263
Dec	0.27169	0.41678	0.26387	0.36839	0.25871	0.31906	0.24655	0.30048

Table 4. Error reduction rate (%) from the default features to the new features (MAE).

Base Learner	Decision Tree	Random Forest	XGBoost	LightGBM
Jan	7.334851936	7.885021097	12.30785156	15.10657551
Feb	16.91571839	20.04951346	21.16516889	20.64851743
Mar	38.09768803	35.52955665	32.05262423	33.86917575
Apr	41.11591426	44.03281053	40.66590355	40.44531441
May	31.58482751	31.93714411	30.58480002	28.26505509
Jun	−7.004450213	−2.64480669	−10.43421905	−8.424855491
July	5.071217533	8.980409617	10.66768264	9.854251764
Aug	40.89671483	37.01855355	40.08014442	40.46680689
Sep	10.07926106	8.374451754	9.883600799	12.08451083
Oct	6.452651922	3.755998014	−3.032557075	−3.134993148
Nov	28.00103596	34.41974211	17.90192818	17.29394307
Dec	26.65352843	25.31065131	25.05069819	26.06974722

Table 5. Error reduction rate (%) from the default features to the new features (RMSE).

Base Learner	Decision Tree	Random Forest	XGBoost	LightGBM
Jan	14.40510498	15.07359161	24.64788732	27.40847385
Feb	35.76674383	29.47595417	36.86774246	34.21825794
Mar	51.20863416	45.30620526	38.47695391	36.26299141
Apr	64.76057183	65.16294838	58.78970821	60.80223753
May	66.11960774	65.44474012	65.39665919	62.40441801
Jun	2.89133142	6.052509294	0.203490165	−4.501053438
July	−2.886416217	4.530477759	−3.625017529	−1.631471614
Aug	68.94873624	63.71491604	67.5025798	67.42602586
Sep	40.94414143	37.75057383	36.47193788	37.68669382
Oct	9.650516283	8.339976697	−19.29369369	−27.74081355
Nov	57.27941426	51.1735474	47.46148749	45.40402585
Dec	35.16986063	36.36709101	38.79062272	40.28379506

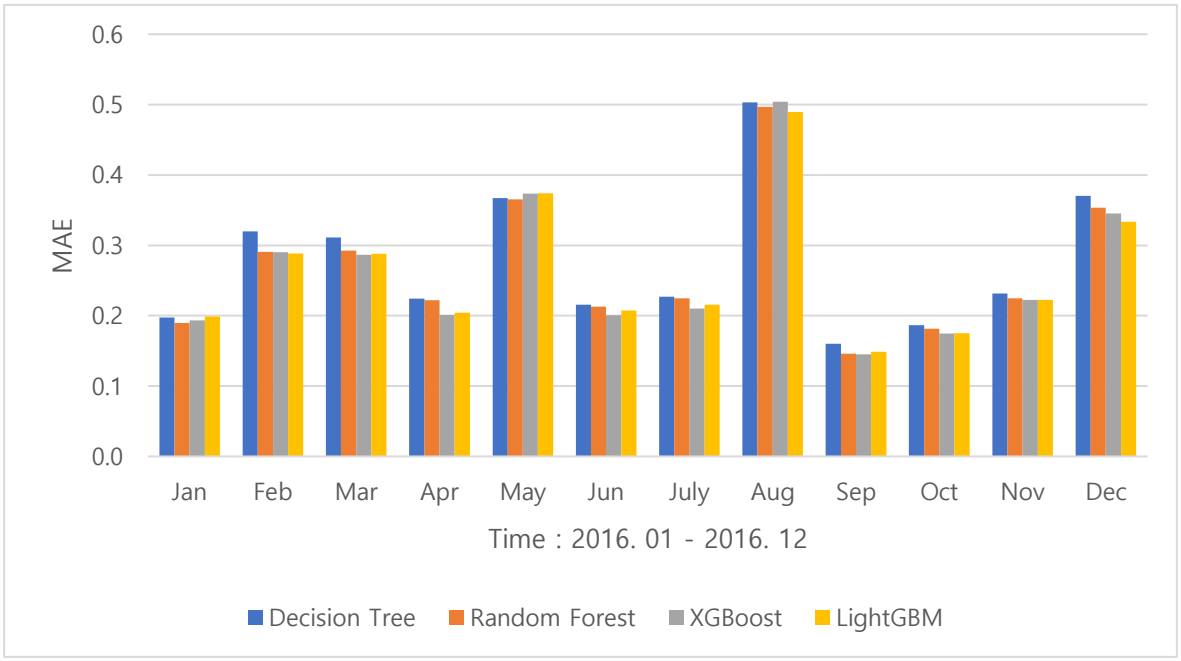

Figure 6. MAE value comparison by base learner of the bagging model (default features).

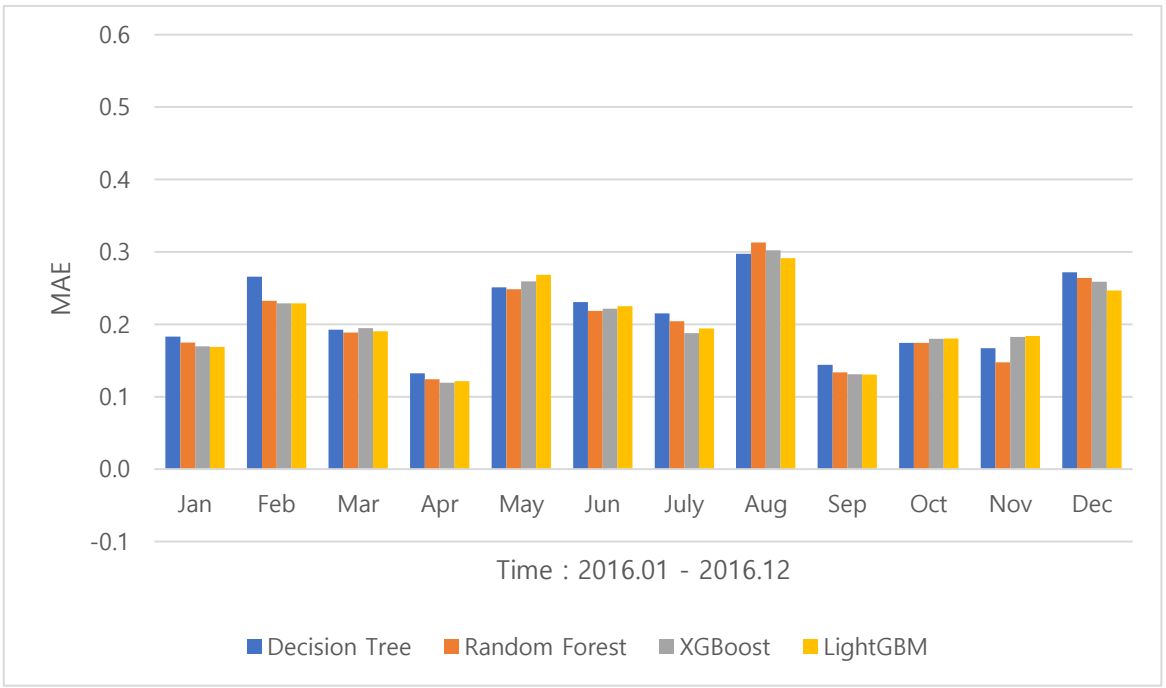

Figure 7. MAE value comparison by base learner of the bagging model (new features).

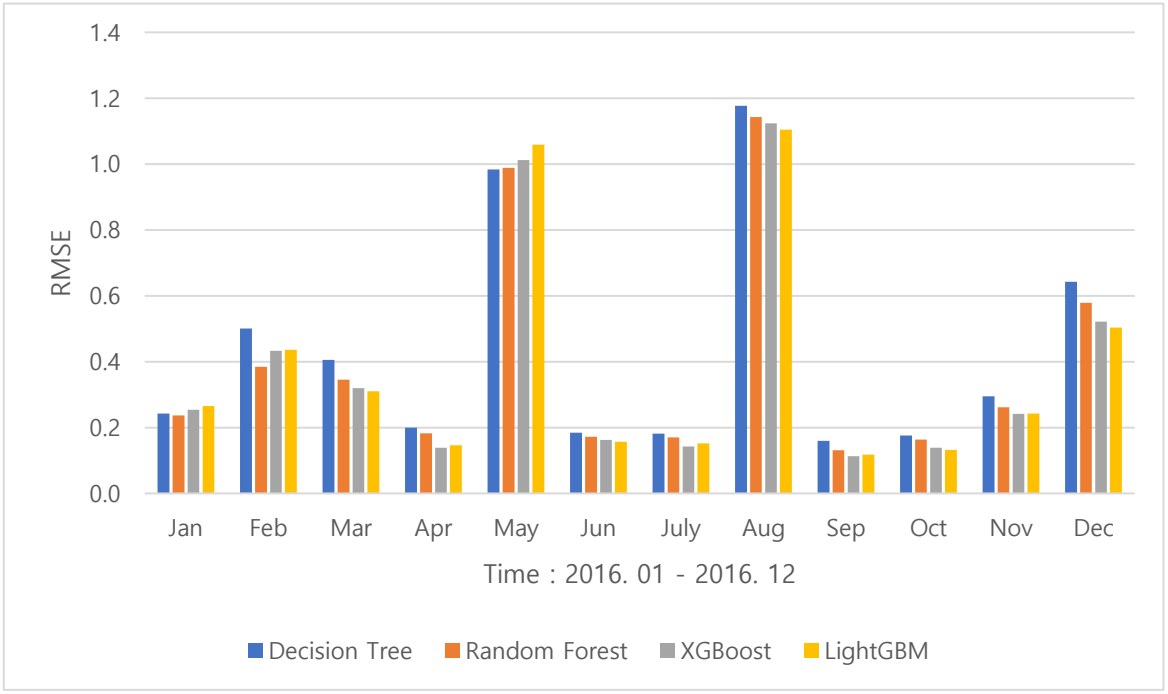

Figure 8. RMSE value comparison by base learner of the bagging model (default features).

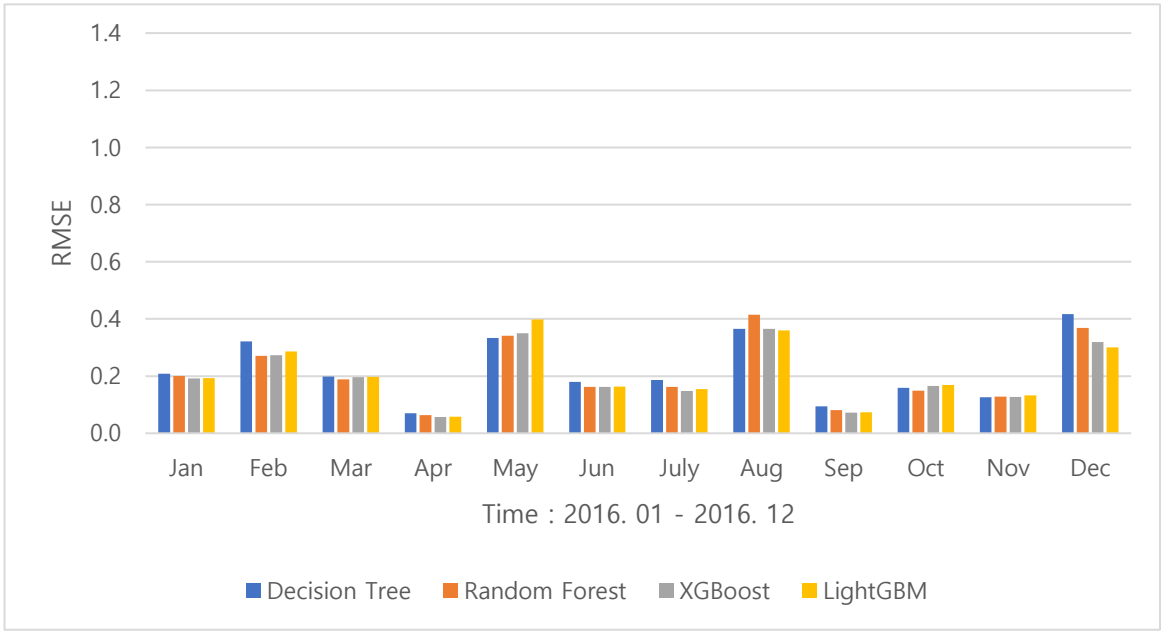

Figure 9. RMSE value comparison by base learner of the bagging model (new features).

Tables 6 and 7 show the average MAE and RMSE rank of the default features and new features, and all bagging models using ensemble base learner show lower error than the when using the single model as a base learner.

Table 6. Rank by average performance metrics from January to December (default features).

	MAE	RMSE
1	LightGBMs (0.26216)	XGBoost (0.38349)
2	XGBoost (0.26623)	LightGBMs(0.38550)
3	Random Forest (0.26665)	Random Forest (0.39635)
4	Decision Tree (0.27617)	Decision Tree (0.42927)

Table 7. Rank by average performance metrics from January to December (new features).

	MAE	RMSE
1	Random Forest (0.20199)	XGBoost (0.20229)
2	LightGBMs (0.20256)	LightGBMs(0.20715)
3	XGBoost (0.20291)	Random Forest (0.21097)
4	Decision Tree (0.21041)	Decision Tree (0.22164)

4. Conclusions

The change in energy mix from conventional to sustainable energy will increase the penetration of solar energy to the power system. In order to cope with solar penetration, which is highly affected by weather conditions, and to form a benefit value chain ranging from grid operator to plant operator and energy consumer, improved forecasting technology is necessary.

In this study, we formed a bagging model for photovoltaic forecasting and in order to improve forecasting performance, the following two models were proposed. First, we proposed an ensemble model as a base learner of the bagging algorithm rather than using a decision tree, which is generally used as a base learner of the bagging model. Second, we proposed using past output data as a new feature. The proposed model showed better performance than the existing method and details of simulation conclusions follow:

(1) The overall performance of using an ensemble model as a base learner in the bagging predictor was better than using a decision tree-based bagging predictor.
(2) The results showed that adding past output as new features instead of just using weather conditions provided better performance to make predictions and it reduced the error rate by up to 50% or more.
(3) The ensemble models used as a base learner of the bagging model were random forest, XGBoost, and LightGBM, and they did not show much difference in performance.

The improvement achieved through the model was demonstrated in the data reported. However, the problem is that the MAE metric is still quite high. Clearly, the model presented showed good performance, but needs to be improved to be an accurate model. As mentioned previously, in this study, in order to compare performance difference between the single model learner-based bagging model and the ensemble model learner-based bagging model, we did not tune each base learner's hyperparameter for optimization. In other words, all the same hyperparameters were used for all models. The impact of the hyperparameter was reduced by using the ensemble model as the base learner, but it could certainly be lower than it is now if the optimal hyperparameter were set and the prediction was made. To improve the model's accuracy, we plan to do additional studies, such as optimization of the hyperparameter, data cleaning, and others.

Author Contributions: J.H. conceived and designed the overall research; S.C. implemented each forecasting model and conducted the experimental simulation; J.H. and S.C. wrote the paper; and J.H. guided the research direction and supervised the entire research process. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the Korea Electric Power Corporation (No. R18XA06-55).

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

References

1. Antonanzas, J.; Osorio, N.; Escobar, R.; Urraca, R.; Martinez-de-Pison, F.J.; Antonanzas-Torres, F. Review of Photovolt Power Forecast. *Sol. Energy* **2016**, *136*, 78–111. [CrossRef]
2. GTM Research/SEIA. U.S. Solar Market Insight, Report Q2. In *Executive Summary*; National Renewable Energy Lab. (NREL): Golden, CO, USA, 2015.
3. IEA. Renewables 2018—Market Analysis and Forecast from 2018 to 2023. Available online: https://www.iea.org/renewables2018 (accessed on 7 February 2020).
4. IEA. Renewables 2019—Market Analysis and Forecast from 2019 to 2024. Available online: https://www.iea.org/renewables2019 (accessed on 7 February 2020).
5. Lorenz, E.; Remund, J.; Müller, S.C.; Traunmüller, W.; Steinmaurer, G.; Pozo, D.; Ruiz-Arias, J.A.; Fanego, V.L.; Ramirez, L.; Romeo, M.G. Benchmarking of Different Approaches to Forecast Solar Irradiance, others. In Proceedings of the 24th European Photovoltaic Solar Energy Conference, Hamburg, Germany, 21–25 September 2009.
6. Paulescu, M.; Paulescu, E.; Gravila, P.; Badescu, V. *Weather Modeling and Forecasting of PV Systems Operation*; Springer: London, UK, 2013.
7. Espinar, B.; Aznarte, J.-L.; Girard, R.; Moussa, A.M.; Kariniotakis, G. Photovoltaic Forecasting: A State of the Art, OTTI—Ostbayerisches Technologie-Transfer-Institut. Available online: https://hal-minesparistech.archives-ouvertes.fr/hal-00771465/document (accessed on 4 March 2015).
8. Moreno-Munoz, J.J.G.; De la Rosa, R.; Posadillo, F. Very short term forecasting of solar radiation. In Proceedings of the 33rd IEEE Photovoltaic Specialists Conference 2008 PVSC 08, San Diego, CA, USA, 11–16 May 2008.
9. Diagne, H.M.; Lauret, P.; David, M. Solar Irradiation Forecasting: State-of-Theart and Proposition for Future Developments for Small-Scale Insular Grids. Available online: https://hal.archives-ouvertes.fr/hal-00918150/document (accessed on 4 March 2015).
10. Heinemann, D.; Lorenz, E.; Lückehe, B. Short-term forecasting of solar radiation: A statistical approach using satellite data. *Sol. Energy* **1999**, *67*, 139–150. [CrossRef]
11. Kalogirou, S. Artificial neural networks in renewable energy systems applications: A review. *Renew. Sustain. Energy Rev.* **2001**, *5*, 373–401. [CrossRef]
12. Torre, M.C.; Poggi, P.; Louche, A. Markovian model for studying wind speed time series in corsica. *Int. J. Renew. Energy Eng.* **2001**, *3*, 311–319.
13. Hugo, T.C.; Carlos, P.; Coimbra, F.M. Assessment of Forecasting Techniques for Solar Power Production with no Exogenous Inputs. Available online: https://www.sciencedirect.com/science/article/abs/pii/S0038092$\\times$12001429?via%3Dihub (accessed on 7 February 2020).
14. Joao, G.; da Silva, F., Jr.; Takashi, O.; Takumi, T.; Gentarou, K.; Yoshihisa, U.; Kazuhiko, O. Use of Support Vector Regression and Numerically Predicted Cloudiness to Forecast Power Output of a Photovoltaic Power Plant in Kitakyushu, Japan. Available online: https://onlinelibrary.wiley.com/doi/abs/10.1002/pip.1152 (accessed on 7 February 2020).
15. Li, P.; Zhang, J.-S. A new hybrid method for China's energy supply security forecasting based on arima and xgboost. *Energies* **2018**, *11*, 1687. [CrossRef]
16. Zhang, W.; Quan, H.; Srinivasan, D. Parallel and reliable probabilistic load forecasting via quantile regression forest and quantile determination. *Energy* **2018**, *160*, 810–819. [CrossRef]
17. Abdel-Nasser, M.; Mahmoud, K. Accurate photovoltaic power forecasting models using deep LSTM-RNN. *Neural Comput. Appl.* **2019**, *31*, 2727–2740. [CrossRef]
18. Inman, R.H.; Pedro, H.T.C.; Coimbra, C.F.M. Solar forecasting methods for renewable energy integration. *Prog. Energy Combust. Sci.* **2013**, *39*, 535–576. [CrossRef]
19. Badescu, V. *Modeling Solar Radiation at the Earth's Surface: Recent Advances*; Springer Science & Business Media: Berlin, Germany, 2008.

20. Gala, Y.; Andez, A.F.; Díaz, J.; Dorronsoro, J.R. Hybrid machine learning forecasting of solar radiation values. *Neurocomputing* **2016**, *176*, 48–59. [CrossRef]
21. Mori, H.; Takahashi, A. A data mining method for selecting input variables for forecasting model of global solar radiation. In Proceedings of the Transmission and Distribution Conference Exposition 2012 IEEE PES, Orlando, FL, USA, 7–10 May 2012; pp. 1–6.
22. Mori, N.K.H. Optimal Regression Tree Based Rule Discovery for Short-term Load Forecasting. In Proceedings of the 2001 IEEE Power Engineering Society Winter Meeting, Conference Proceedings (Cat. No.01CH37194), Columbus, OH, USA, 28 January–1 February 2001; Volume 2, pp. 421–426.
23. Troncoso, S.; Salcedo-Sanz, C.; Casanova-Mateo, J.C.; Riquelme, L. Prieto, Local models-based regression trees for very short-term wind speed prediction. *Renew. Energy* **2015**, *81*, 589–598. [CrossRef]
24. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
25. Choi, H.; Kim, Y.; Kim, J.; Song, J.; Park, C. *Data Mining with R*; CRC Press: Boca Raton, FL, USA, 1973.
26. Massaoudi, M.; Chihi, I.; Sidhom, L.; Trabelsi, M.; Refaat, S.S.; Oueslati, F.S. PV Power Forecasting Using Weighted Features for Enhanced Ensemble Method. *arXiv* **2019**, arXiv:1910.09404.
27. Tu, M.C.; Shin, D.; Shin, D.K. Effective diagnosis of heart disease through bagging approach. In Proceedings of the 2nd International Conference Biomedical Engineering and Informatics, Tianjin, China, 17–19 October 2009; pp. 1–4.
28. De Oliveira, E.M.; Oliveira, F.L.C. Forecasting mid-long term electric energy consumption through bagging ARIMA and exponential smoothing methods. *Energy* **2018**, *144*, 776–788. [CrossRef]
29. Patrick, B.; Nekipelov, D.; Ryan, S.P.; Yang, M. Machine Learning Methods for Demand Estimation. *Am. Econ. Rev.* **2015**, *105*, 481–485.
30. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
31. Abuella, M.; Chowdhury, B. Random forest ensemble of support vector regression models for solar power forecasting. In Proceedings of the2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 23–26 April 2017; pp. 1–5.
32. Zamo, M.; Mestre, O.; Arbogast, P.; Pannekoucke, O. A benchmark of statistical regression methods for short-term forecasting of photovoltaic electricity production, part I: Deterministic forecast of hourly production. *Sol. Energy* **2014**, *105*, 792–803. [CrossRef]
33. Huang, J.; Troccoli, A.; Coppin, P. An analytical comparison of four approaches to modelling the daily variability of solar irradiance using meteorological records. *Renew. Energy* **2014**, *72*, 195–202. [CrossRef]
34. Mohammed, A.; Yaqub, W.; Aung, Z. *Probabilistic Forecasting of Solar Power: An Ensemble Learning Approach in Intelligent Decision Technologies*; Springer: Berlin, Germany, 2015; pp. 449–458.
35. Cheng, H.-Y. Hybrid solar irradiance now-casting by fusing Kalman filter and regressor. *Renew. Energy* **2016**, *91*, 434–441. [CrossRef]
36. Zheng, H.; Yuan, J.; Chen, L. Short-term load forecasting using emd-lstm neural networks with a xgboost algorithm for feature importance evaluation. *Energies* **2017**, *10*, 1168. [CrossRef]
37. Xiao, C.; Yi, W.; Jialun, Z.; Jing, S.; Bingjie, L.; Chongqing, K. Data-Driven Load Data Cleaning and Its Impacts on Forecasting Performance. Available online: https://www.researchgate.net/profile/Yi_Wang137/publication/337707536_Data-Driven_Load_Data_Cleaning_and_Its_Impacts_on_Forecasting_Performance/links/5de64bde4585159aa45d1828/Data-Driven-Load-Data-Cleaning-and-Its-Impacts-on-Forecasting-Performance.pdf (accessed on 7 February 2020).
38. Ruijin, Z.; Weilin, G.; Xuejiao, G. Short-Term Photovoltaic Power Output Prediction Based on k-Fold Cross-Validation and an Ensemble Model. Available online: https://www.mdpi.com/1996-1073/12/7/1220/pdf (accessed on 7 February 2020).
39. Elliston, B.; MacGill, I. The potential role of forecasting for integrating solar generation into the Australian national electricity market. In Proceedings of the Solar 2010, the 48th AuSES Annual Conference, Canberra, Australia, 1–3 December 2010.
40. Heinemann, D.; Lorenz, E.; Girodo, M. Forecasting of Solar Radiation. In *Solar Energy Resource Management for Electricity Generation from Local Level to Global Scale*; Nova Science Publishers: Hauppauge, NY, USA, 2006.
41. Lauret, P.; Voyant, C.; Soubdhan, T.; David, M.; Poggi, P. A benchmarking of machine learning techniques for solar radiation forecasting in an insular context. *Sol. Energy* **2015**, *112*, 446–457. [CrossRef]

42. Mohammed, H.; Alsharif, I.D.; Jeong, K.; Jin, H.K. Opportunities and Challenges of Solar and Wind Energy in South Korea: A Review. Available online: https://www.mdpi.com/2071-1050/10/6/1822 (accessed on 7 February 2020).
43. Alsharif, M.H.; Kim, J. Hybrid Off-Grid SPV/WTG Power System for Remote Cellular Base Stations Towards Green and Sustainable Cellular Networks in South Korea. *Energies* **2016**, *10*, 9. [CrossRef]
44. Korea Meteorological Administration (KMA). *Annual Climatological Report 2013*; Korea Meteorological Administration: Seoul, South Korea, 2013. Available online: http://web.kma.go.kr/eng/index.jsp (accessed on 22 May 2018).
45. NASA Surface Meteorology and Solar Energy Web Site. Available online: https://eosweb.larc.nasa.gov/cgibin/sse/homer.cgi?email=skip%40larc.nasa.gov&step=1&lat=37.499&lon=126.54958&submit=Submit&ms=1&ds=1&ys=1998&me=12&de=31&ye=1998&daily=swv_dwn (accessed on 22 May 2018).
46. National Institute of Meteorological Sciences (NIMS). Cumulative Solar Irradiance Map. Available online: http://www.greenmap.go.kr/02_data/data02_1_1.do#2#2#1 (accessed on 22 May 2018).
47. Zhou, H.; Deng, Z.; Xia, Y.; Fu, M. A new sampling method in particle filter based on pearson correlation coefficient. *Neurocomputing* **2016**, *216*, 208–215. [CrossRef]