

Article

# Short-Term Photovoltaic Output Prediction Based on Decomposition and Reconstruction and XGBoost under Two Base Learners

Weihui Xu <sup>1</sup>, Zhaoke Wang <sup>1</sup>, Weishu Wang <sup>1,\*</sup>, Jian Zhao <sup>2,\*</sup>, Miaoja Wang <sup>1</sup> and Qinbao Wang <sup>1</sup>

<sup>1</sup> School of Energy and Power Engineering, North China University of Water Resources and Electric Power, Zhengzhou 450045, China

<sup>2</sup> State Grid Henan Electric Power Company Electric Power Science Research Institute, Zhengzhou 450052, China

\* Correspondence: wangweishu@ncwu.edu.cn (W.W.); zhaojian07@126.com (J.Z.)

**Abstract:** Photovoltaic power generation prediction constitutes a significant research area within the realm of power system artificial intelligence. Accurate prediction of future photovoltaic output is imperative for the optimal dispatchment and secure operation of the power grid. This study introduces a photovoltaic prediction model, termed ICEEMDAN-Bagging-XGBoost, aimed at enhancing the accuracy of photovoltaic power generation predictions. In this paper, the original photovoltaic power data initially undergo decomposition utilizing the Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (ICEEMDAN) algorithm, with each intrinsic mode function (IMF) derived from this decomposition subsequently reconstructed into high-frequency, medium-frequency, and low-frequency components. Targeting the high-frequency and medium-frequency components of photovoltaic power, a limiting gradient boosting tree (XGBoost) is employed as the foundational learner in the Bagging parallel ensemble learning method, with the incorporation of a sparrow search algorithm (SSA) to refine the hyperparameters of XGBoost, thereby facilitating more nuanced tracking of the changes in the photovoltaic power's high-frequency and medium-frequency components. Regarding the low-frequency components, XGBoost-Linear is utilized to enable rapid and precise prediction. In contrast with the conventional superposition reconstruction approach, this study employs XGBoost for the reconstruction of the prediction output's high-frequency, intermediate-frequency, and low-frequency components. Ultimately, the efficacy of the proposed methodology is substantiated by the empirical operation data from a photovoltaic power station in Hebei Province, China. Relative to integrated and traditional single models, this paper's model exhibits a markedly enhanced prediction accuracy, thereby offering greater applicational value in scenarios involving short-term photovoltaic power prediction.

**Keywords:** bagging; continuous multi-day; nonlinear fusion reconstruction; SSA; XGBoost; ICEEMDAN



**Citation:** Xu, W.; Wang, Z.; Wang, W.; Zhao, J.; Wang, M.; Wang, Q. Short-Term Photovoltaic Output Prediction Based on Decomposition and Reconstruction and XGBoost under Two Base Learners. *Energies* **2024**, *17*, 906. <https://doi.org/10.3390/en17040906>

Academic Editors: J. Carlos García-Díaz and Óscar Trull

Received: 15 December 2023

Revised: 4 February 2024

Accepted: 8 February 2024

Published: 15 February 2024



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

## 1. Introduction

In addressing climate change, air pollution, and the energy security concerns stemming from conventional energy sources, numerous countries are vigorously advancing renewable energy development. With the ongoing expansion of the total installed capacity of photovoltaic systems, there is a significant mitigation of environmental issues. Nevertheless, the intrinsic high variability of photovoltaic power output presents considerable challenges to the safe and stable functioning of power systems. Forecasting of photovoltaic power generation is pivotal in proactively managing potential power supply gaps, optimizing the resource distribution, judiciously allocating reserve capacity, and reducing the risks associated with power undersupply and oversupply. Photovoltaic power forecasting is categorized into long-term, medium-term, and short-term forecasts, based on the range of time they cover in their predictions [1]. Long-term forecasts, which extend beyond a

month, are utilized for strategic energy planning and system expansion [2]. Medium-term forecasts, projecting the power output for the forthcoming days, assist in load balancing and power dispatch [3]. Short-term forecasts, focusing on power predictions ranging from a few minutes to several hours, are essential for real-time system scheduling, a swift demand response, and maintaining grid stability [4–6]. Given the high dependence of photovoltaic power generation on weather conditions and its inherent variability, the accuracy of these forecasts directly impacts the efficiency and stability of grid operations. Therefore, there is a need for greater precision and robustness in the forecasting models.

In addressing the traditional photovoltaic power forecasting issue, numerous scholars both domestically and internationally have conducted extensive research on theoretical methods, including physical models [7], statistical models [8], machine learning models [9,10], and hybrid models that integrate these approaches [11]. Among these methods, machine-learning-based photovoltaic power forecasting has emerged as the dominant approach. Reference [12] introduced a photovoltaic power prediction method combining the genetic algorithm (GA) and particle swarm optimization (PSO) to refine the adaptive neuro-fuzzy inference system (ANFIS), with its effectiveness validated using data from the Beijing Goldwind microgrid system. Reference [13] proposed an advanced Improved Grey Wolf Algorithm (DIGWO) in conjunction with Bidirectional Long Short-Term Memory (BiLSTM) for developing a fault diagnosis model for photovoltaic arrays. Reference [14] employed two hybrid models, namely Convolutional Neural Network-LSTM (CNN-LSTM) and Convolutional LSTM (ConvLSTM), for short-term photovoltaic power forecasting, corroborating their effectiveness with data from a Moroccan solar power plant. Reference [15] proposed a photovoltaic power prediction method utilizing an improved ant colony optimization algorithm (ACO) and support vector machine (SVM), which demonstrates a higher forecasting accuracy compared to a traditional SVM. Traditional machine learning methods exhibit drawbacks in terms of their generalization ability and prediction accuracy. As a forefront research area within machine learning, deep learning can delve deeper into understanding the inherent laws and representation levels of the sample data. Deep learning has been extensively researched in the context of photovoltaic power forecasting. Reference [16] utilizes the random forest algorithm for photovoltaic power prediction. Reference [17] employed an eight-layer fully convolutional network (FCN-8) and an enhanced bidirectional gated recurrent unit (EBiGRU) to develop a deep hybrid network. However, the complexity of the deep learning network architecture, particularly the choice of the number of hidden layers and nodes, significantly impacts the training outcomes. Improper network design often leads to issues such as falling into local minima and overfitting. These characteristics pose limitations to the further advancement of deep learning in photovoltaic power forecasting.

In recent years, tree ensemble algorithms have demonstrated significant advancements. In the Kaggle data science competition, tree ensemble algorithms, notably XGBoost, outperformed many deep learning algorithms. Reference [18] employed XGBoost to develop a fault diagnosis model targeting various fault types in photovoltaic arrays. Reference [19] suggests developing a photovoltaic power prediction model for various weather types by integrating the Fuzzy C-Means (FCM) clustering algorithm with XGBoost. Reference [20] introduces an adaptive transfer learning framework for XGBoost-based photovoltaic power prediction, validating the model with distributed photovoltaic power generation data from the Alice Springs region. Reference [21] integrates LSTM with XGBoost to diminish the error rate of individual models and enhance their accuracy in photovoltaic power prediction. Reference [22] integrates the physical aspects of distributed photovoltaics with XGBoost for more accurate distributed photovoltaic forecasting.

Signal decomposition technology provides a powerful tool within the machine learning framework for photovoltaic systems, enhancing aspects such as data processing [23], feature extraction, performance optimization, and predictive maintenance [24]. In the application of photovoltaic power forecasting, these technologies are primarily used to accurately extract key features from the time-series data on photovoltaic power, such as pe-

riodicity and trends, and are also employed for data denoising, which involves separating and removing environmental and equipment noise to enhance the accuracy of predictions. Furthermore, signal decomposition technology enables models to adaptively adjust according to changes in the environmental and operational conditions, thereby maintaining the flexibility and accuracy of the forecasting strategy. Reference [25] employed Fast Iterative Filtering Decomposition (FIFD) to extract the complex features of photovoltaic power time series. Reference [26] introduces Variational Mode Decomposition (VMD) to address the volatility of raw photovoltaic data. However, VMD requires presetting parameters such as the number of decomposition modal components and the quadratic penalty term, where inappropriate values may affect the prediction outcomes. Reference [27] utilized Empirical Mode Decomposition (EMD) to decompose photovoltaic power data. They employed the Sine Cosine Algorithm (SCA) and Extreme Learning Machine (ELM) to develop a model, validating its effectiveness with distributed photovoltaic data from the electrical department of SOA University in Bhubaneswar, Odisha, India. EMD can effectively remove noise interference and improve the prediction accuracy, but it faces the issues of frequency aliasing and poor noise robustness. Reference [28] employs Ensemble Empirical Mode Decomposition (EEMD) to decompose photovoltaic power data into multiple stable components, and by calculating the Sample Entropy (SE) of each component and reconstructing those with similar SE values, it reduces the superposition errors, lowers the computational costs, and enhances the prediction accuracy. Reference [29] proposes a secondary decomposition technique combining VMD and Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) to process historical PV output data, enhancing the model's predictive performance. Thus, by employing decomposition techniques to analyze photovoltaic data, it is possible to effectively unearth the inherent nonlinear characteristics of the data, thereby significantly enhancing the precision and reliability of predictive models.

In the context of rapid advancements in artificial intelligence technology in photovoltaic power forecasting, this paper proposes a short-term photovoltaic power forecasting model based on ICEEMDAN-Bagging-XGBoost. The paper initially outlines the algorithm's mechanism and then segregates it into high-frequency, medium-frequency, and low-frequency components based on the zero-crossing rate of each IMF component decomposed using ICEEMDAN. Bagging-XGBoost is utilized for predicting the high-frequency and medium-frequency components, while the SSA is incorporated to reduce the time required for XGBoost hyperparameter optimization. XGBoost-Linear is capable of accurately predicting the low-frequency component, demonstrating superior smoothing and curve-fitting capabilities, along with a rapid calculation speed. The output of the first-layer prediction model is fed into the second-layer model, comprising SSA-XGBoost, for nonlinear fusion and reconstruction, yielding the final prediction result. Compared with the traditional superposition reconstruction method, this study employs XGBoost to perform nonlinear fusion-based reconstruction of the high-frequency, medium-frequency, and low-frequency components of the predicted output. In the case study, this method's efficacy was validated using real power generation data from a photovoltaic power station in Hebei Province, China. The results demonstrate that the proposed model exhibits a superior stability and minimal prediction errors compared to other models, thereby underscoring its significant practical value.

## 2. Basic Principles

### 2.1. ICEEMDAN Algorithm Mechanism

ICEEMDAN represents an advancement over EEMD and CEEMDAN, aiming to further enhance the precision in processing non-stationary and nonlinear signals. EEMD reduces mode mixing by adding white noise to the signal and performing multiple decompositions, yet it may leave residual noise. CEEMDAN builds upon this by adaptively adjusting the noise level to further decrease the reconstruction error. In contrast, ICEEMDAN introduces specific adaptive white noise  $E_k(w^{(i)})$  during the extraction of each

intrinsic mode function, effectively mitigating the impact of residual noise and false modal components, thus improving the accuracy of the signal decomposition and reconstruction. The detailed procedure of analysis encompasses the following steps:

Step 1: Add special noise  $E_1(w^{(i)})$  to the original signal  $x$ , as shown in Formula (1):

$$x^{(x)} = x + \beta_0 E_1(w^{(i)}) \quad (1)$$

In the formula,  $w^{(i)}$  is the  $i$ -th Gaussian white noise added, ( $i = 1, 2, \dots, M$ ),  $\beta_0$  is the noise standard deviation of the first decomposed signal, and  $\sigma(\cdot)$  is the standard. The difference operator,  $E(\cdot)$ , is the operator for computing the EMD decomposition.

Step 2: Calculate the local average value of each  $x_1^{(i)}$  signal, and obtain the residual  $Res_1$  of the first decomposition, as shown in Formula (2):

$$r_1 = \langle M(x_1^{(i)}) \rangle \quad (2)$$

In the formula,  $\langle \cdot \rangle$  is the operator for calculating the average value of  $M$  signals, and  $M(\cdot)$  is the operator for calculating the local average value of the signal.

Step 3: Subtract the first residual  $r_1$  from the original signal  $x$ , thereby obtaining the first  $IMF_1$  component of the original signal. As shown in Formula (3):

$$IMF_1 = x - r_1 \quad (3)$$

Step 4: Determine the second  $IMF$  component,  $IMF_2$ , as follows:

$$IMF_2 = Res_1 - \langle M(r_1 + \beta_1 E_2(w^{(i)})) \rangle \quad (4)$$

Step 5: For  $k = 3, 4, \dots, N$ , calculate the  $k$ -th  $IMF$  component,  $IMF_k$ , as follows:

$$IMF_k = r_{k-1} - \langle M(r_{k-1} + \beta_{k-1} E_k(w^{(i)})) \rangle \quad (5)$$

Step 6: Increment  $k$  by 1 and repeat Step 5 until the extraction of the final  $IMF$  component is achieved.

## 2.2. XGBoost Algorithm Mechanism

XGBoost is a prominent example of the Boosting ensemble algorithm, falling within the category of a Gradient Boosting Decision Tree (GBDT). The underlying principle of the traditional GBDT algorithm involves utilizing CART as the base learner. In each iteration, the new base learner persistently adapts to the residual generated by the previous base learner, thereby reducing the loss function of this iteration expeditiously. Ultimately, the weak learners produced in each iteration are combined to form a strong learner. Building upon a GBDT, XGBoost incorporates a regularization term into the loss function to manage the model's complexity, and subsequently executes a second-order Taylor expansion on the loss function. In contrast to a GBDT, XGBoost enhances the model's generalization ability.

The algorithm mechanism of XGBoost is as follows. For the dataset  $D = \{(x_i, y_i)\}$ , ( $|D| = n$ ), the ensemble model of the tree is shown using Equation (6):

$$\hat{y}_i^{(t)} = \sum_{k=1}^t f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (6)$$

In the formula,  $\hat{y}_i^{(t)}$  is the prediction result after the  $t$ -th iteration,  $\hat{y}_i^{(t-1)}$  is the sum of the prediction results of the previous  $t - 1$  trees, and  $f_t(x_i)$  is the newly added  $t$ -th tree. The loss function (objective function) of XGBoost consists of two parts:

$$Obj^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \sum_{i=1}^t \Omega(f_i) \quad (7)$$

$$\Omega(f_t) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (8)$$

In the formula, the first part of the objective function of the  $t$ -th iteration is the error between the predicted value  $\hat{y}_i^{(t)}$  and the true value  $y_i$ , and the second part is the regularization term, that is, the sum of the complexity of each tree.  $\gamma$  and  $\lambda$  are the penalty coefficients,  $T$  is the number of leaves in the tree, and  $w$  is the leaf weight.

Given that the complexity of the first  $t - 1$  trees is a known constant, Equation (7) can be reformulated as:

$$Obj^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + constant \quad (9)$$

XGBoost performs second-order Taylor expansion on the loss function, defining  $g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$  and  $h_i = \partial_{\hat{y}_i^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)})$  as the first and second derivatives of the objective function  $Obj^t$ , respectively. Bring  $g_i$  and  $h_i$  into the objective function to get the approximate objective function:

$$Obj^{(t)} \cong \sum_{i=1}^n [l(y_i, \hat{y}_i^{(t-1)} + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i))] + \Omega(f_t) + constant \quad (10)$$

Due to the error  $\hat{y}_i^{(t-1)}$  between the predicted value  $l(y_i, \hat{y}_i^{(t-1)})$  and the actual value  $y_i$  in the  $t$ -th iteration and the complexity constant being a definite constant value, which has no effect on the function optimization, the objective function is approximated as:

$$Obj^{(t)} \cong \sum_{i=1}^n \left[ g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad (11)$$

The XGBoost algorithm adopts an incremental training method where, in each iteration, new trees are continually added to fit the preceding error values, thereby minimizing the objective function, as depicted in Equation (11).

$$\begin{cases} \hat{y}_i^{(0)} = 0 \\ \hat{y}_i^{(1)} = f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\ \hat{y}_i^{(2)} = f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\ \vdots \\ \hat{y}_i^t = \sum_{k=1}^t f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \end{cases} \quad (12)$$

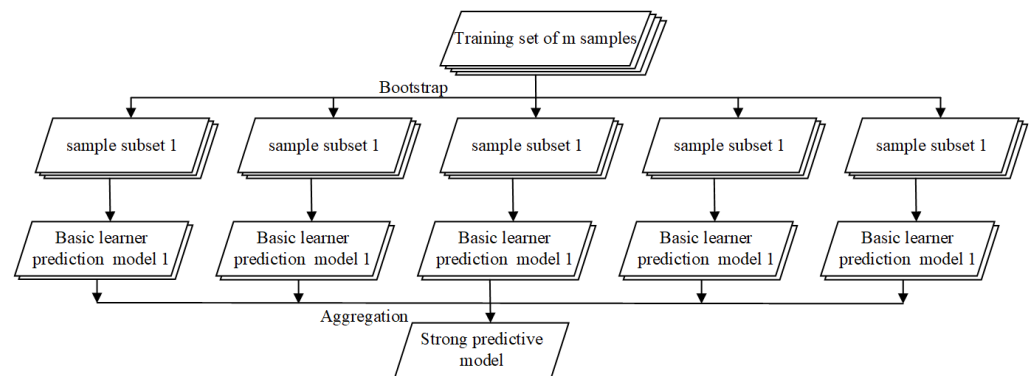
Within the XGBoost framework, users have the option to select from two types of base learners: the Tree Booster and the Linear Booster, with the latter also referred to as XGBoost-Linear. These learners offer flexible solutions tailored to accommodate diverse data characteristics and predictive task requirements. The Tree Booster elucidates the complex structure of the data through the construction of a series of decision trees. It enhances traditional Gradient Boosted Decision Trees with several optimizations, including efficient tree-splitting algorithms, regularization to prevent overfitting, and parallel processing techniques to expedite the training process. These improvements endow the XGBoost Tree Model with remarkable capabilities in handling nonlinear relationships and feature interactions, making it applicable to a wide range of datasets and machine learning tasks.

Conversely, XGBoost-Linear provides a succinct and efficient solution for scenarios where there exists a linear relationship between the features and the target or when dealing with high-dimensional data. By incorporating L1 and L2 regularization, XGBoost-Linear effectively mitigates overfitting, showcasing its advantages in terms of computational efficiency and model simplicity. Although the tree-based ensemble of XGBoost is more

commonly utilized in practical applications, XGBoost-Linear demonstrates unique strengths under specific conditions, especially in datasets characterized by smoothness and strong regularity, offering superior fitting capabilities and an exceptional generalization performance.

### 2.3. Parallel Ensemble Learning Method of the Bagging Mechanism

Bagging, also known as the Bagging method, is a parallel ensemble learning method that can effectively reduce variance. Its algorithmic architecture is depicted in Figure 1. The core concept involves sampling the original dataset using the self-sampling method (Bootstrap) to generate multiple random datasets. For example, for a dataset  $D$  with  $m$  samples, random sampling with replacement is performed on it, and a new dataset  $D_i$  with the same number of samples as the original dataset is obtained by looping  $m$  times. Due to random sampling with replacement, there will be duplicate samples in the resulting dataset. After  $n$  rounds of self-sampling,  $n$  new datasets  $\{D_1, D_2, \dots, D_{n-1}, D_n\}$  containing  $m$  samples are obtained, and  $n$  new datasets are used to train  $n$  independent base learners  $\{V_1, V_2, \dots, V_{n-1}, V_n\}$ . In regression problems, the outputs from the trained  $n$  base learners are aggregated and averaged to yield the final result.



**Figure 1.** Bagging parallel ensemble learning architecture.

### 2.4. Sparrow Search Optimization Algorithm Mechanism

The SSA represents a novel heuristic swarm intelligence optimization algorithm, mimicking the predation and anti-predation behaviors of sparrows. In the SSA, the key roles include the discoverer, joiner, and predator. Assuming an  $N$ -dimensional search space, the position (or potential solution) of the  $i$ -th sparrow is represented as follows:

$$X_i^t = \{x_{i,1}^t, x_{i,2}^t, \dots, x_{i,N}^t\}, i = 1, 2, \dots, m \quad (13)$$

In the formula,  $t$  is the current iteration number, and  $m$  is the population size.

The formula for updating the position of the finder within the sparrow population is as follows:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t \exp(-\frac{i}{\varphi M_{\max}}), R_2 < S_T \\ x_{i,j}^t + \sigma l, R_2 \geq S_T \end{cases} \quad (14)$$

In the formula,  $x_{i,j}^t$  is the position of the  $i$ -th sparrow in the  $j$ -th dimension in the  $N$ -dimensional search space;  $M_{\max}$  is the maximum number of iterations;  $\varphi$  is a random number between 0 and 1;  $\sigma$  is a random number that follows a normal distribution;  $L$  is a  $1 \times N$  matrix;  $R_2 \in [0, 1]$  indicates the warning value;  $S_T \in [0.5, 1]$  indicates the safety value; when  $R_2 < S_T$ , that means that the population is in a safe area, and the discoverer can randomly search for potential solutions in the current area; and when  $R_2 \geq S_T$ , that means that there are predators around the population, and we need to move to a safe area for searching.



To ascertain a superior potential solution, the joiner might opt to follow the discoverer in pursuit of the optimal solution or independently explore other regions, with the position update occurring as follows:

$$x_{i,j}^{t+1} = \begin{cases} \sigma \exp\left(-\frac{x_{worse}^t - x_{i,j}^t}{i^2}\right), i > n/2 \\ x_{best}^{t+1} + \frac{1}{N} \sum_j^N (\text{rand}\{-1, 1\} \cdot |x_{i,j}^t - x_{best}^{t+1}|), i \leq n/2 \end{cases} \quad (15)$$

In the formula,  $x_{worse}^t$  is the worst solution in the  $j$ -th dimension of the search space in the current population;  $x_{i,j}^{t+1}$  is the optimal solution in the  $j$ -th dimension of the search space in the current population; when  $i > n/2$ , that means that the  $i$ -th joiner does not get a better solution and searches elsewhere; when  $i \leq n/2$ , the joiner will monitor the finder and compete with it for the optimal solution, thereby replacing the finder to seek out a larger space to search.

Additionally, the sparrow algorithm incorporates predators. The sparrows tasked with monitoring predators constitute 10% to 20% of the entire population. The positions of these sparrows are randomly determined and are expressed as follows:

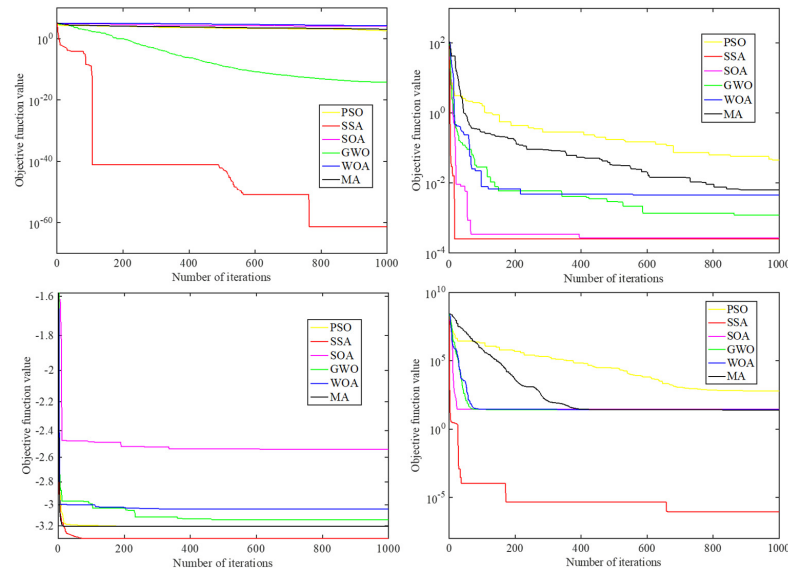
$$x_{i,j}^{t+1} = \begin{cases} x_{best}^t + \beta |x_{i,j}^t - x_{best}^t|, f_i \neq f_{best} \\ x_{i,j}^t + K \left( \frac{|x_{i,j}^t - x_{worse}^t|}{(f_i - f_{worse}) + \mu} \right), f_i = f_{best} \end{cases} \quad (16)$$

In the formula,  $\beta$  is a random number obeying the standard normal distribution;  $K \in [-1, 1]$ ;  $\mu$  is a small constant, in order to prevent the denominator from being 0;  $f_{worse}$  and  $f_{best}$  are the worst and optimal fitness of the current population;  $f_i$  is the fitness value of the  $i$ -th sparrow; when  $f_i \neq f_{best}$ , that means that the sparrow is at the edge of the population and is more threatened; when  $f_i = f_{best}$ , that means that the sparrow is aware of the danger and needs to approach other sparrows to avoid being preyed on.

In this paper, the SSA is utilized to optimize the hyperparameters of XGBoost. The SSA exhibits a fast convergence speed and a strong optimization capability. Figure 2 illustrates the performance of six heuristic optimization algorithms on various test functions—PSO, the SSA, SOA, GWO, the Whale Optimization Algorithm (WOA), and the Memetic Algorithm (MA). These convergence curves facilitate a comparative analysis of the optimization capabilities of these algorithms during the iterative process. Each subplot displays the results for different test functions, with changes in the objective function values reflecting the quality of the solutions provided by the algorithms. In the graph, lower values on the  $y$ -axis indicate better optimization results, while the  $x$ -axis represents the number of iterations, showing the progression over time.

The marked decline in the SSA during the initial iterations on each test function exemplifies its rapid initial convergence rate, suggesting its efficacy in reducing errors or costs in practical applications. The performance characteristics of the SSA are its convergence rate and the ability to avoid or escape local optima. The convergence rate is observed through the slope of the curve, where a steeper slope indicates a quicker improvement in the quality of the solution; the ability to escape local optima is inferred from the trend in the curve after its initial decline. If the curve flattens at a higher value, this implies that the algorithm is trapped in a local optimum. The absence of such flattening in the SSA's curve indicates its effective mechanism for avoiding local optima and more comprehensively exploring the search space. Especially after 1000 iterations, the SSA attains lower objective function values relative to other algorithms, indicating its efficiency in finding globally optimal or near-optimal solutions. During the later stages of algorithm optimization, the subtle enhancements it exhibits are vital for practical applications, as these minor improvements can sometimes yield substantial real-world impacts. Through comparison of it with other algorithms, Figure 2 underscores the superiority of the SSA in its optimization efficiency

and convergence speed, bolstering its dependability in optimizing the hyperparameters of intricate models like XGBoost.

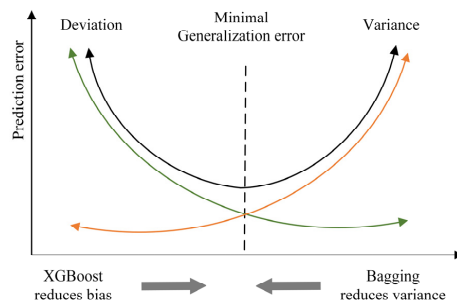


**Figure 2.** Test results of different group optimization algorithms.

### 3. A Photovoltaic Prediction Method Based on ICEEMDAN and Multi-Model Fusion

#### 3.1. The Overall Framework of the Photovoltaic Prediction Model

XGBoost continuously adds new trees during training and employs a greedy algorithm to progressively reduce the loss function value. The regularization term in the loss function moderates the model complexity, enhancing the fitting capability and consequently reducing prediction bias. However, the serial ensemble learning model, exemplified by XGBoost, often encounters excessive variance due to its complexity. Minor changes in the sample data can alter the learned model’s performance and stability, leading to a decreased prediction accuracy on the test set, indicative of overfitting. The Bagging parallel ensemble learning method conducts multiple random samplings from the original data, inputting each resulting dataset into respective basic learners for training. Subsequently, the prediction results are aggregated to mitigate the impact of data disturbances and thereby decrease the risk of overfitting. Consequently, this paper adopts XGBoost as the basic learner in the Bagging parallel ensemble learning method, harnessing the synergistic strengths of both algorithms to minimize the prediction model’s generalization error. The variation in the error during model training is illustrated in Figure 3.

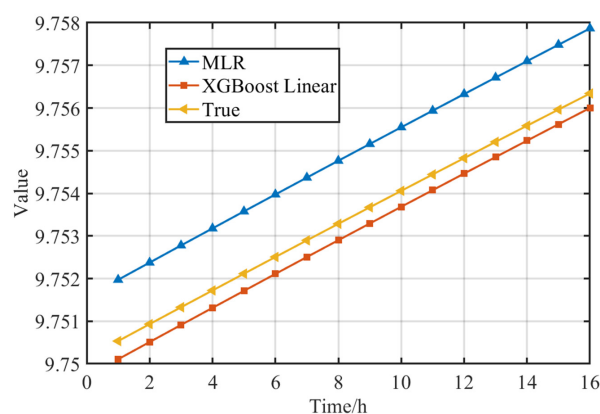


**Figure 3.** Error variation in Bagging-XGBoost training.

Figure 3 depicts the overall framework of the photovoltaic prediction model based on ICEEMDAN-Bagging-XGBoost, as proposed in this paper. The first-layer prediction model comprises SSA-Bagging-XGBoost, while the second-layer prediction model consists of SSA-XGBoost. Detailed descriptions of the model analysis and algorithm flow are provided below.



This study utilizes the ICEEMDAN algorithm to decompose the original photovoltaic power sequence. Based on the zero-crossing rate of each IMF component, these are categorized into high-frequency, intermediate-frequency, and low-frequency components. The SSA-Bagging-XGBoost model is employed for predicting the high-frequency and intermediate-frequency components, which capture the intricacies and randomness of photovoltaic power fluctuations, thereby reducing the variance and bias during training and enabling more detailed tracking of the photovoltaic power change curve. For the low-frequency components, which roughly indicate the photovoltaic power variation trends, the XGBoost-Linear model is utilized for prediction. This approach is adopted because the XGBoost-Linear model's simplicity enhances the fitting of smooth curves and minimizes the risk of overfitting. Figure 4 presents the comparison results between the XGBoost-Linear and Multiple Linear Regression (MLR) models in terms of smooth curve prediction. Through analysis, it can be observed that, compared to the MLR algorithm, which also achieves a high degree of fit for smooth curves, XGBoost-Linear demonstrates a superior prediction performance for smooth curves. It exhibits significant advantages in computation speed and parameter adjustment flexibility. Therefore, choosing XGBoost-Linear for predicting the low-frequency components is a highly suitable choice.



**Figure 4.** Curve fitting diagram.

After conducting the preceding analysis, the first-layer prediction model comprises SSA-Bagging-XGBoost. The XGBoost algorithms are utilized to calculate the feature importance scores of the input features based on tree gain. This process ascertains the contributions of characteristics including irradiance, temperature, pressure, humidity, wind direction, and wind speed to the predicted target. These feature importance scores aid in selecting the final input feature set for the prediction model. The divided high-frequency component, intermediate-frequency component, and low-frequency component are considered the target output, and the dataset required by the model is created by integrating the determined input features. Then, the dataset is segmented into an initial training set and a test set, and the initial training set is further divided into a training set and a verification.

The validation set performs two key functions. One is facilitating the optimal hyperparameter selection for XGBoost in the first-layer prediction model. Given that most machine learning algorithms encompass both hyperparameters and model parameters, these parameters significantly impact the model's performance. Model parameters can be automatically adjusted during training, while hyperparameters require manual iterative debugging to ascertain their optimal combination. XGBoost includes numerous hyperparameters, like the number of trees, tree depth, learning rate, and minimum loss value for node splitting. Therefore, this study introduces SSA optimization to assist in identifying the optimal hyperparameters for XGBoost.

Utilizing the SSA for hyperparameter optimization, both the tree-based and linear ensemble XGBoost models are fine-tuned. Subsequently, initial training samples are put into the optimally configured Bagging-XGBoost model to generate predictions for the high-frequency and mid-frequency components. Concurrently, the XGBoost-Linear model

is employed to predict the low-frequency portion of the data, leveraging its proficiency in handling smoothly varying data.

The XGBoost models, under the optimal hyperparameter configuration, are employed to make predictions on the verification and test sets, yielding prediction results. The prediction results from the validation set are merged with the corresponding meteorological factors and actual values to form a new training set. Similarly, the prediction results from the validation set are amalgamated with the corresponding meteorological factors to develop a new input feature set. Building on this, the new training set is utilized to further train XGBoost, enabling it to comprehensively learn the correlations between each frequency component, the meteorological factors, and the actual values, thus achieving the fusion and reconstruction of the preliminary prediction results.

The second function of the second-layer validation set is the optimization of the hyperparameters of the second-layer prediction model. The second-layer prediction model, composed of SSA-XGBoost, is employed to nonlinearly fuse and reconstruct the high-frequency, intermediate-frequency, and low-frequency components from the first-layer prediction to obtain the final output. The high-frequency, medium-frequency, and low-frequency components of the sub-training set from the first-layer prediction model serve as the inputs, and the corresponding actual photovoltaic power values are the target output, forming the training set for the second-layer prediction model. The high-frequency, medium-frequency, and low-frequency components predicted by the first-layer model on the verification set act as inputs, and the corresponding actual photovoltaic power values serve as the target output, forming the verification set for the second-layer prediction model. Then, by utilizing the SSA, the optimal hyperparameters for the second-layer prediction model are identified. The high-frequency, intermediate-frequency, and low-frequency components predicted by the first-layer model on the test set are put into the SSA-XGBoost model with the optimal hyperparameters for nonlinear fusion reconstruction, resulting in the output of the final prediction results. Figure 5 presents the ICEEMDAN-Bagging-XGBoost photovoltaic prediction model framework.

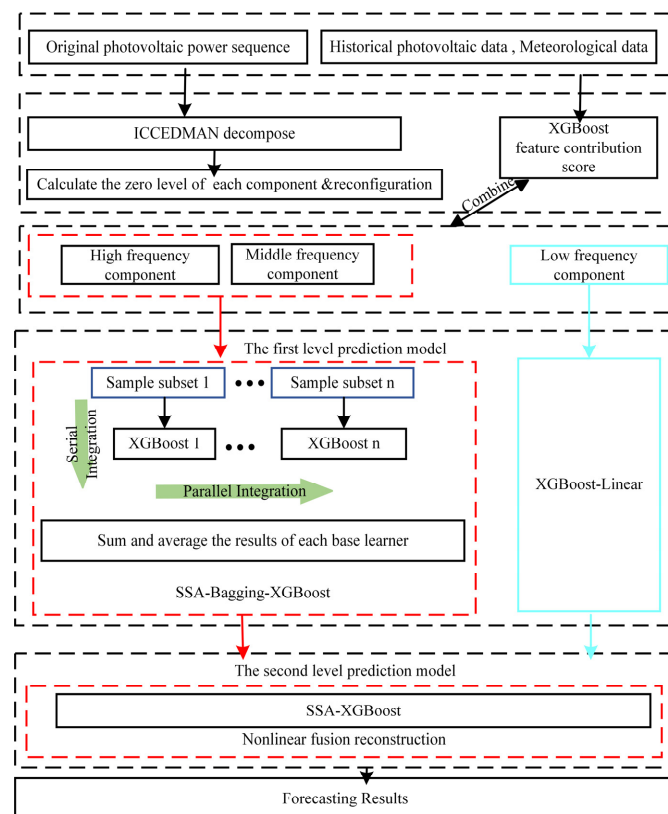


Figure 5. ICEEMDAN-Bagging-XGBoost photovoltaic prediction model framework.

### 3.2. Model Evaluation Metrics

To rigorously evaluate the performance of the predictive model outlined in this manuscript, three principal metrics are utilized: the relative root mean square error  $e_{RMSE}$ , the mean absolute error  $e_{MAE}$ , and the mean absolute percentage error  $e_{MAPE}$ .

$$e_{MAE} = \frac{1}{m} \sum_{t=1}^m |x(t) - y(t)| \quad (17)$$

$$e_{RMSE} = \sqrt{\frac{1}{m} \sum_{t=1}^m [x(t) - y(t)]^2} \quad (18)$$

$$e_{MAPE} = \frac{1}{m} \sum_{t=1}^m \left| \frac{x(t) - y(t)}{x(t)} \right| \times 100\% \quad (19)$$

In the formula,  $x(t)$  represents the actual value at time  $t$ ,  $y(t)$  denotes the predicted value, and  $m$  is the number of samples.

## 4. Case Analysis

The experimental dataset utilized in this paper originates from a centralized photovoltaic power plant in Hebei Province, China. This original dataset comprises daily photovoltaic power data from June to August 2018, with each day featuring 96 sampling points. Additionally, it encompasses meteorological information including irradiance, temperature, humidity, air pressure, wind direction, and wind speed. The model programming related to this paper was conducted in the MATLAB 2023a environment. The prediction evaluation indices employed are the relative mean square error  $e_{RMSE}$ , mean absolute error  $e_{MAE}$ , and mean absolute percentage error  $e_{MAPE}$ .

### 4.1. Photovoltaic Power Time-Series Decomposition

Photovoltaic power prediction is influenced by numerous factors, exhibiting complex volatility. To enhance the prediction accuracy, signal decomposition technology has increasingly gained prominence in photovoltaic power prediction. This technology facilitates more accurate prediction of photovoltaic power by analyzing the fluctuation patterns in time series. Currently, EMD is extensively employed, yet it encounters a modal aliasing problem during decomposition. To address this issue, the ICEEMDAN decomposition algorithm is utilized in this study. The algorithm employs a variational model to ascertain the relevant frequency band and to extract the corresponding modal components. The ICEEMDAN algorithm exhibits significant anti-noise advantages in signal processing. This advantage stems from its unique noise-assisted mechanism, which effectively reduces the modal aliasing in the signal, thus ensuring more accurate frequency separation and enhanced signal analysis reliability.

At present, most studies introducing decomposition algorithms for photovoltaic forecasting typically put the decomposed IMF components sequentially into the forecasting model for training and then aggregate the forecast results of each component to obtain the final output. Given that training each component separately can result in prolonged training durations and increased resource consumption, this approach is not optimal. Additionally, the cumulative prediction errors of each component during the final reconstruction can diminish the overall prediction accuracy.

As shown in Figure 6, the ICEEMDAN decomposition reveals no significant mode mixing in each intrinsic mode function, with each component exhibiting relatively stable frequencies. Based on this, the division into high-frequency, medium-frequency, and low-frequency parts is achieved by calculating the zero-crossing rate of each intrinsic mode function. In this paper, the zero-crossing rate is categorized as high-frequency within the range of 0.1 to 1, medium-frequency from 0.01 to 0.1, and low frequency below 0.01. The

calculation of each component’s zero-crossing rate is facilitated by Formula (1), as depicted in Figure 7.

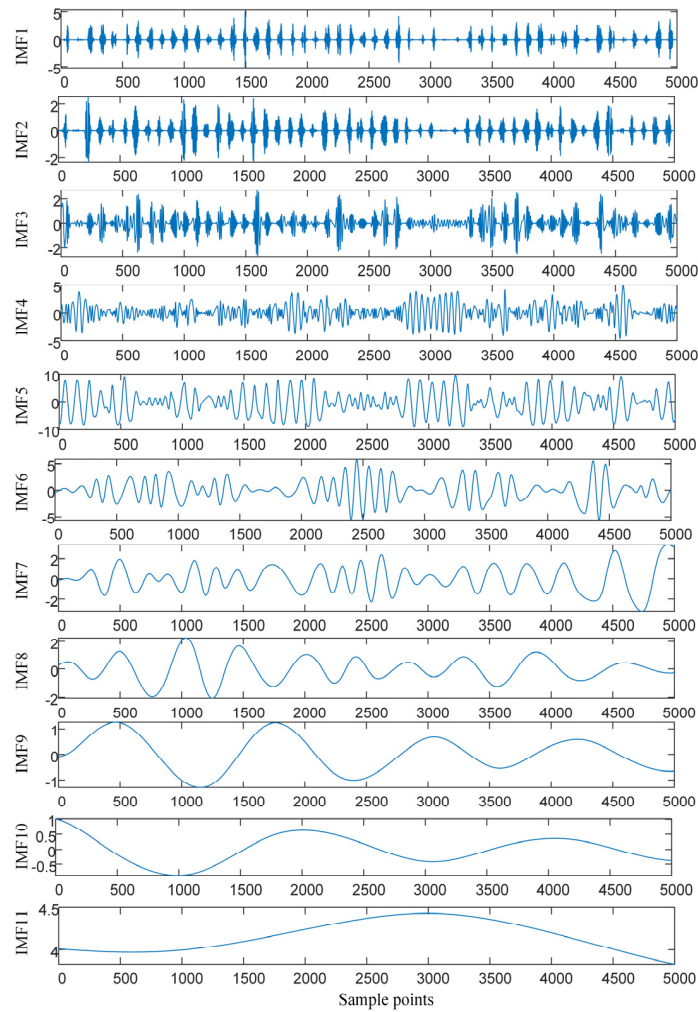


Figure 6. ICCEMDAN decomposition results.

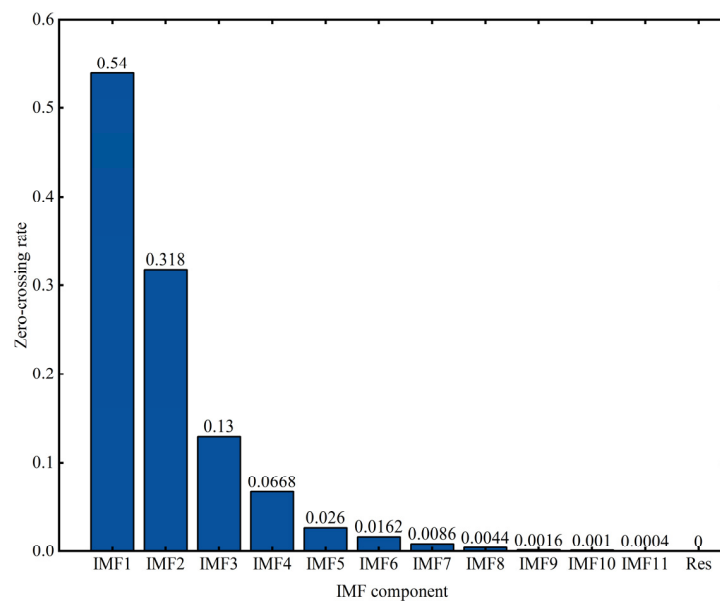
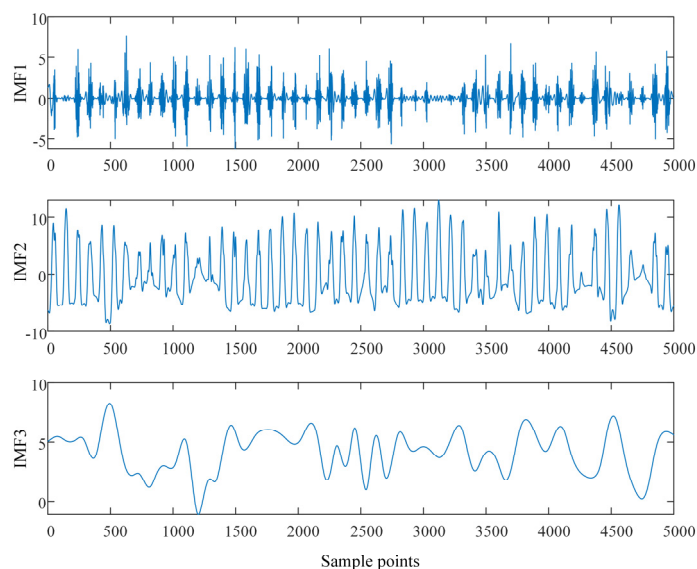


Figure 7. Zero-crossing rate of each IMF component.

Examination of the previous chart reveals that the zero-crossing rate of IMF1 to IMF3 exceeds 0.1, leading to their classification as high-frequency components. The zero-crossing rate of IMF4 to IMF8 surpasses 0.01, categorizing them as intermediate-frequency components, whereas the zero-crossing rate of IMF7 to Res falls below 0.01, thus classifying them as low-frequency components. The intrinsic modal functions corresponding to these three frequency components are amalgamated and reconstituted into new modal functions, as depicted in Figure 8.



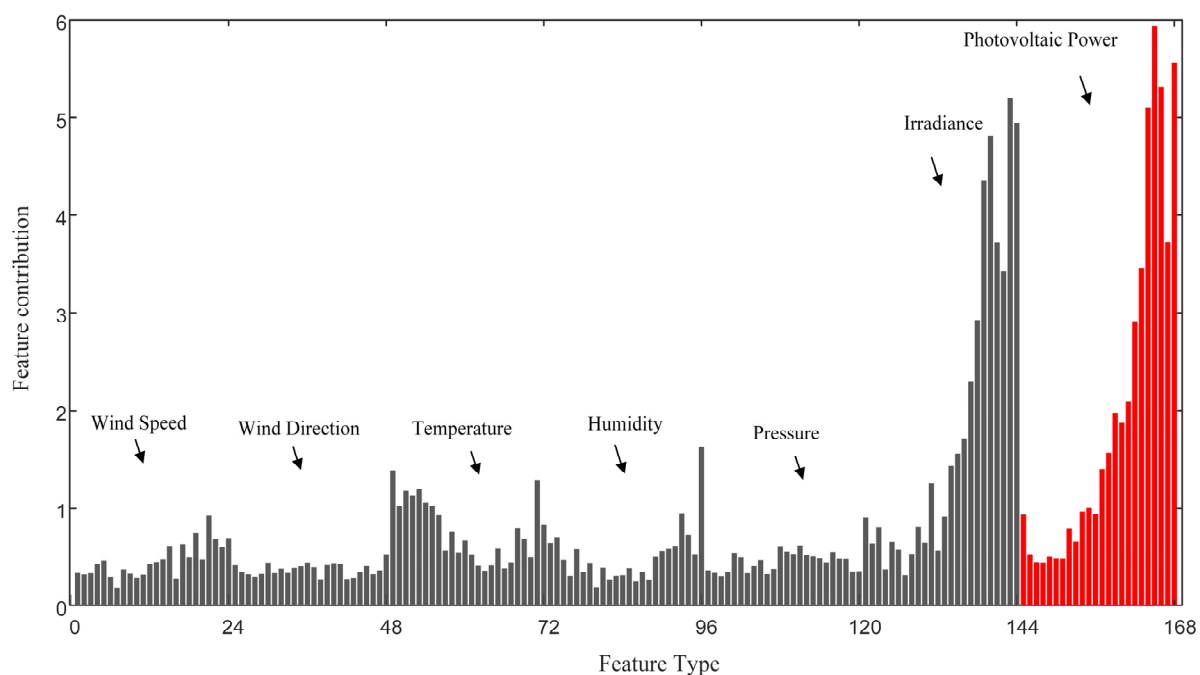
**Figure 8.** Data reconstruction results.

Following the ICEEMDAN decomposition and reconstruction of the photovoltaic processed data, the low-frequency components demonstrate a smoother data trend and enhanced periodicity. Consequently, employing multivariate linear regression for their prediction not only circumvents complex parameter tuning but also rapidly and accurately yields satisfactory results. The SSA-Bagging-XGBoost model is utilized to predict both the high-frequency and intermediate-frequency components. In this model, XGBoost serves as the basic learner within the Bagging parallel ensemble learning method, effectively compensating for the latter's limitation in reducing deviation. Concurrently, this method significantly mitigates the impact of minor data perturbations and reduces the variance effectively. Furthermore, optimization with the SSA substantially shortens the hyperparameter optimization time. Consequently, employing SSA-Bagging-XGBoost for predicting both the high-frequency and intermediate-frequency components facilitates a more effective prediction of local details in the photovoltaic power fluctuations.

In this study, the decomposed low-frequency, intermediate-frequency, and high-frequency components are considered the target outputs, and the dataset required for the model is formed by integrating the established input features. Subsequently, the dataset is segmented into an initial training set and a test set, with the initial training set further divided into a training set and a validation set.

#### 4.2. Feature Engineering Analysis

Feature importance analysis plays a vital role in training and the prediction performance. The XGBoost model can calculate the importance of each feature during the training process. The principle is based on the gain in the number of structural branches to select a feature as the split point. The importance of a feature is determined by the total number of times it appears in the tree. The more frequently it occurs, the more important it is. This paper employs XGBoost to construct the feature engineering, as shown in Figure 9.



**Figure 9.** XGBoost contribution analysis.

Figure 9 illustrates the cumulative contribution of multiple features over time (measured in hours). Each feature has 24 bars representing its cumulative importance for each hour in the past 24 h. For features such as wind speed, wind direction, temperature, humidity, air pressure, irradiance, and photovoltaic power, each bar signifies the contribution of that feature to the model's predictive results within a specific hour. The figure indicates that, in historical data, there is a high correlation between photovoltaic power and electric energy production in the target forecast period. Irradiance also shows a higher feature contribution, aligning with the expectations for a solar power generation prediction model, as irradiance directly affects the solar output.

Meteorological conditions such as wind speed, wind direction, temperature, humidity, and air pressure have a relatively lower contribution. These factors do impact photovoltaic power generation, but their effects are not as significant as irradiance and photovoltaic power. In specific periods, they may significantly influence the predictive model's output, but overall, their impact is smaller. Although temperature and humidity contribute less than irradiance and photovoltaic power, they show an increased correlation in certain specific periods, indicating their potential importance to the prediction model. Rising temperatures may reduce the efficiency of photovoltaic panels during the day, thereby affecting power generation, while changes in humidity might impact the panel performance in the morning due to dew or fog or interact with temperature during hot periods to affect the cooling efficiency of the panels. The fluctuations in these environmental variables provide important contextual information for the model, especially under conditions of drastic weather changes. Their contribution to the prediction accuracy should not be overlooked. Including temperature and humidity as input features in the construction of a photovoltaic power prediction model helps capture the variations in the power generation caused by meteorological condition fluctuations, enhancing the model's predictive capability.

Therefore, when constructing a prediction model, selecting photovoltaic power, irradiance, temperature, and humidity as inputs not only improves the model's fit to the historical data but also enhances its adaptability to future condition changes and predictive accuracy.

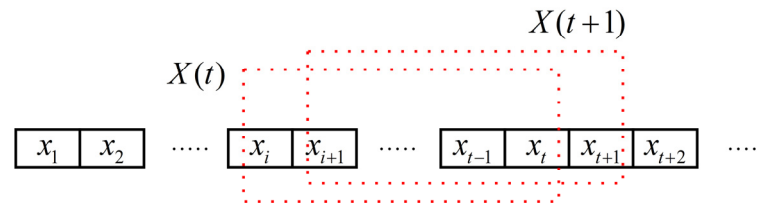
#### 4.3. Prediction Method Analysis

In this paper, the rolling prediction method is utilized for the three frequency components, while the sliding window method is employed to process the decomposed and



reconstructed component sequences. Subsequently, the original sequence is segmented into multiple sub-sequences.

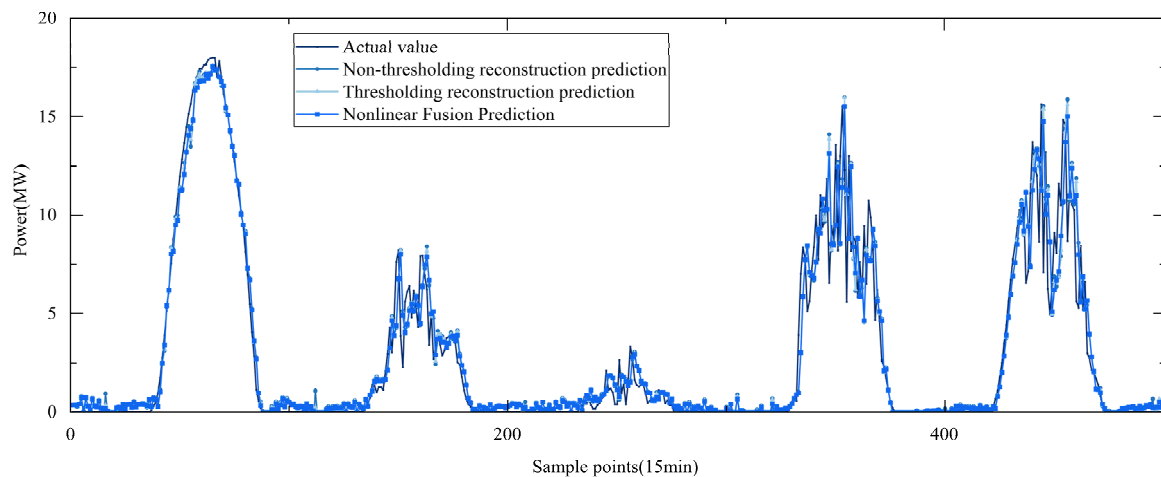
Figure 10 illustrates the assumption that  $X(t) = [x_i, x_{i+1}, \dots, x_t]$  delineates a sub-time series with a timestep of  $K$ , under the premise  $i \geq 1$  and  $t > i$ . This formulation leads to the subsequent sub-time series  $X(t+1) = [x_{i+1}, x_{i+2}, \dots, x_{t+1}]$ , and consequently, a time series of length  $N$  is segmented into  $N - K + 1$  subsequences.



**Figure 10.** Sliding window handling.

#### 4.4. Model Effect Comparison Analysis

To validate the superiority of the proposed Nonlinear Fusion Prediction method, it was compared with both the non-thresholding reconstruction and thresholding reconstruction prediction methods after decomposition, where both employed direct summation for fusion. The results of this comparison are depicted in Figure 11 and Table 1. Clearly, the Nonlinear Fusion Prediction method utilizing XGBoost achieved the highest prediction accuracy. Moreover, the evaluation metrics indicated that the prediction performance of the non-thresholding reconstruction method was slightly inferior to that of the thresholding reconstruction method. This can be attributed to the reconstruction phase, wherein the decomposed components undergo reconstruction or integration, potentially enhancing the useful information within the signal and reducing noise. Such enhanced information can be more effectively utilized in subsequent predictions, thereby improving the prediction accuracy.



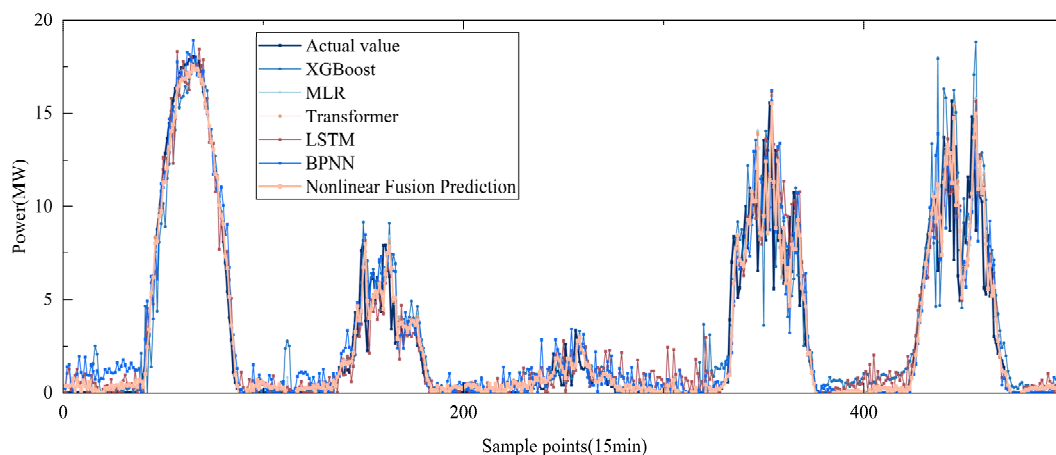
**Figure 11.** Comparison of reconstruction methods.

For high-frequency and medium-frequency components, Bagging-XGBoost effectively minimized the bias and variance, thereby enhancing its ability to track local variation trends in the photovoltaic data and improving the prediction accuracy. Regarding low-frequency components, XGBoost-Linear utilized its exceptional fitting capability to smooth the curve, thereby achieving a remarkably high prediction accuracy. XGBoost conducts Nonlinear Fusion Prediction on the high-, medium-, and low-frequency components predicted by the first-layer model, further enhancing the prediction accuracy compared to the thresholding reconstruction method.

**Table 1.** Evaluation results of different models.

Algorithm	$e_{RMSE}$	$e_{MAE}$	$e_{MAPE}$
XGBoost	1.3514	0.7364	0.1030
MLR	1.7258	1.0036	0.2925
Transformer	1.4023	0.7821	0.1302
LSTM	1.4816	0.8035	0.1581
BPNN	1.6743	0.9579	0.2731
Non-thresholding reconstruction prediction	1.3560	0.7431	0.1149
Thresholding reconstruction prediction	1.3382	0.7253	0.1004
Nonlinear Fusion Prediction	1.1932	0.5901	0.0853

To further investigate the predictive performance of this model, comparisons were made between XGBoost, MLR, Transformer, LSTM, and BPNN models. The prediction errors of each model are, respectively, presented in Table 1. As indicated by the results in Table 1 and Figure 12, the proposed model significantly outperforms the single models in its prediction accuracy, more precisely tracking changes in the photovoltaic power. In the case of a single model, there is frequently a risk of converging to a local minimum during the training and learning process, which may result in the diminished generalization ability of the model and instability in the prediction outcomes. While the XGBoost algorithm can incrementally reduce the deviation by adding new trees during training, it struggles to mitigate the data disturbance effects at later stages, leading to increased deviation and a reduced test set prediction accuracy. The proposed ICEEMDAN-Bagging-XGBoost model in this paper demonstrates a high predictive accuracy and overcomes the effects of the data variability.

**Figure 12.** Comparing the results with the single-body models.

## 5. Conclusions

This paper draws upon the latest cutting-edge research theories in the field of artificial intelligence to propose a short-term photovoltaic power prediction model. The model employs ICEEMDAN-Bagging-XGBoost and integrates SSA optimization to efficiently identify the optimal hyperparameters for XGBoost, thereby minimizing the time required for hyperparameter optimization. The analysis illustrates that the proposed prediction model in this study exhibits an exceptional performance in short-term forecasting for photovoltaic power generation. This model accurately tracks the curve of variation in the photovoltaic power during periods of significant fluctuations. Compared to single models, the model presented in this study achieves a superior prediction accuracy and demonstrates a more stable performance. Additionally, it boasts advantages such as a reduced overfitting risk and an enhanced generalization ability. Consequently, it possesses significant practical value for short-term forecasting in photovoltaic power plants.

In future work, we will continue to carry out research on the application of advanced artificial intelligence technology in power systems:

At the forecasting scenario level, the data used in this example encompass complete meteorological factors. However, at this level, the variability in photovoltaic power is a critical factor, largely due to unknown meteorological factors. Consequently, at this level, future work will involve integrating real-time weather forecasting with rolling correction of the short-term meteorological factors to enhance the accuracy of the photovoltaic power forecasting for each cycle.

At the algorithmic level, given the volatility and randomness of photovoltaic power generation systems, along with the anticipated large volume and high dimensionality of future photovoltaic system data, there are heightened requirements for photovoltaic power forecasting algorithms. It is essential to effectively utilize algorithms like XGBoost to extract critical information from the high-dimensional features and to apply batch learning techniques to photovoltaic power forecasting, thereby continuously optimizing the model's accuracy. Additionally, the implementation of functionalities for offline training and online prediction significantly enhances the practical applicability of the model.

At the application level, it is imperative to continue exploring the application of artificial intelligence technology in the prediction of renewable energy output. Furthermore, it is crucial to fully leverage the role of artificial intelligence technology, embedded into power knowledge and experience, in power grid optimization and scheduling, fault diagnosis and analysis, and other fields to ensure the power system embodies the characteristics of intelligent interaction, safety, and controllability.

**Author Contributions:** Conceptualization, W.X.; methodology, W.X. and J.Z.; software, Z.W.; validation, M.W. and Q.W.; formal analysis, W.X.; investigation, M.W. and Q.W.; resources, J.Z.; data curation, J.Z.; writing—original draft preparation, Z.W. and W.W.; writing—review and editing, W.X. and J.Z.; visualization, M.W. and Q.W.; supervision, J.Z.; project administration, W.X. and M.W.; funding acquisition, W.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Project of Central Plains Science and Technology Innovation Leading Talents of Henan Province (224200510022).

**Data Availability Statement:** The data used in this study came from a photovoltaic plant in Hebei Province, China. These data include operating data such as the power generation and environmental parameters of the photovoltaic power plant. We used these data to validate our models and analyses when conducting the simulation studies.

**Conflicts of Interest:** Author Jian Zhao comes from State Grid Henan Electric Power Company Electric Power Science Research Institute. The authors declare no conflict of interest.

## Nomenclature

FCN	Fully Convolutional Network	GA	Genetic Algorithm
ACO	Ant Colony Optimization	ICEEMDAN	Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise
ANFIS	Adaptive Neuro-Fuzzy Inference System	MA	Memetic Algorithm
BILSTM	Bidirectional Long Short-Term Memory	PSO	Particle Swarm Optimization
CEEMDAN	Complete Ensemble Empirical Mode Decomposition With Adaptive Noise	SCA	Sine Cosine Algorithm
CNN	Convolutional Neural Network	SE	Sample Entropy
ConvLSTM	Convolutional LSTM	SOA	Salp Swarm Algorithm
DIGWO	Improved Grey Wolf Algorithm	SSA	Singular Spectrum Analysis
EBiGRU	Enhanced Bidirectional Gated Recurrent Unit	SVM	Support Vector Machine
ELM	Extreme Learning Machine	VMD	Variational Mode Decomposition
EMD	Empirical Mode Decomposition	SSA	Singular Spectrum Analysis
FCM	Fuzzy C-Means	WOA	Whale Optimization Algorithm
FIFID	Fast Iterative Filtering Decomposition	XGBoost	Extreme Gradient Boosting
MLR	Multiple Linear Regression	LSTM	Long Short-Term Memory

## References

1. Louzazni, M.; Mosalam, H.; Cotfas, D.T. Forecasting of photovoltaic power by means of non-linear auto-regressive exogenous artificial neural network and time series analysis. *Electronics* **2021**, *10*, 1953. [\[CrossRef\]](#)
2. Ding, S.; Li, R.; Tao, Z. A novel adaptive discrete grey model with time-varying parameters for long-term photovoltaic power generation forecasting. *Energy Convers. Manag.* **2021**, *227*, 113644. [\[CrossRef\]](#)
3. Ahmad, T.; Zhang, D.; Huang, C. Methodological framework for short-and medium-term energy, solar and wind power forecasting with stochastic-based machine learning approach to monetary and energy policy applications. *Energy* **2021**, *231*, 120911. [\[CrossRef\]](#)
4. Mellit, A.; Pavan, A.M.; Lughi, V. Deep learning neural networks for short-term photovoltaic power forecasting. *Renew. Energy* **2021**, *172*, 276–288. [\[CrossRef\]](#)
5. Yin, W.; Han, Y.; Zhou, H.; Ma, M.; Li, L.; Zhu, H. A novel non-iterative correction method for short-term photovoltaic power forecasting. *Renew. Energy* **2020**, *159*, 23–32. [\[CrossRef\]](#)
6. Huang, Z.; Huang, J.; Min, J. SSA-LSTM: Short-Term photovoltaic power prediction based on feature matching. *Energies* **2022**, *15*, 7806. [\[CrossRef\]](#)
7. Mayer, M.J.; Gróf, G. Extensive comparison of physical models for photovoltaic power forecasting. *Appl. Energy* **2021**, *283*, 116239. [\[CrossRef\]](#)
8. Theocharides, S.; Makrides, G.; Livera, A.; Theristis, M.; Kaimakis, P.; Georghiou, G.E. Day-ahead photovoltaic power production forecasting methodology based on machine learning and statistical post-processing. *Appl. Energy* **2020**, *268*, 115023. [\[CrossRef\]](#)
9. Chahboun, S.; Maaroufi, M. Principal component analysis and machine learning approaches for photovoltaic power prediction: A comparative study. *Appl. Sci.* **2021**, *11*, 7943. [\[CrossRef\]](#)
10. Mohana, M.; Saidi, A.S.; Alelyani, S.; Alshayeb, M.J.; Basha, S.; Anqi, A.E. Small-scale solar photovoltaic power prediction for residential load in Saudi Arabia using machine learning. *Energies* **2021**, *14*, 6759. [\[CrossRef\]](#)
11. Zhang, J.; Tan, Z.; Wei, Y. An adaptive hybrid model for day-ahead photovoltaic output power prediction. *J. Clean. Prod.* **2020**, *244*, 118858. [\[CrossRef\]](#)
12. Semero, Y.K.; Zhang, J.; Zheng, D. PV power forecasting using an integrated GA-PSO-ANFIS approach and Gaussian process regression based feature selection strategy. *CSEE J. Power Energy Syst.* **2018**, *4*, 210–218. [\[CrossRef\]](#)
13. Hong, P.; Xin, Z. Improved Grey Wolf Algorithm Based on Multistrategy Dual Weighting Factors with BiLSTM for PV Fault Diagnosis. In Proceedings of the 2022 2nd International Conference on Electrical Engineering and Mechatronics Technology (ICEEMT), Hangzhou, China, 1–3 July 2022; pp. 284–290.
14. Agga, A.; Abbou, A.; Labbadi, M.; El Houm, Y. Short-term self consumption PV plant power production forecasts based on hybrid CNN-LSTM, ConvLSTM models. *Renew. Energy* **2021**, *177*, 101–112. [\[CrossRef\]](#)
15. Pan, M.; Li, C.; Gao, R.; Huang, Y.; You, H.; Gu, T.; Qin, F. Photovoltaic power forecasting based on a support vector machine with improved ant colony optimization. *J. Clean. Prod.* **2020**, *277*, 123948. [\[CrossRef\]](#)
16. Dai, Y.; Wang, Y.; Leng, M.; Yang, X.; Zhou, Q. LOWESS smoothing and Random Forest based GRU model: A short-term photovoltaic power generation forecasting method. *Energy* **2022**, *256*, 124661. [\[CrossRef\]](#)
17. Naz, A.; Javaid, N.; Asif, M.; Javed, M.U.; Ahmed, A.; Gulfam, S.M.; Shafiq, M.; Choi, J.-G. Electricity consumption forecasting using gated-FCN with ensemble strategy. *IEEE Access* **2021**, *9*, 131365–131381. [\[CrossRef\]](#)
18. Wang, H.; Sun, F. Optimal sensor placement and fault diagnosis model of PV array of photovoltaic power stations based on XGBoost. *IOP Conf. Ser. Earth Environ. Sci.* **2021**, *661*, 012025. [\[CrossRef\]](#)
19. Zhong, M.; Zhou, Y. Short-term PV Power Prediction Based on SAGA-FCM and XGBoost. *J. Phys. Conf. Ser.* **2023**, *2562*, 012024. [\[CrossRef\]](#)
20. Ding, S.; Yang, Q.; Li, X.; Yan, W.; Ruan, W. Transfer learning based photovoltaic module defect diagnosis using aerial images. In Proceedings of the 2018 International Conference on Power System Technology (POWERCON), Guangzhou, China, 6–8 November 2018; pp. 4245–4250.
21. Haiwang, T.; Qiliang, Y.; Jiangchun, X.; Kefeng, H.; Shuo, Z.; Haoyu, H. Photovoltaic power prediction based on combined XGboost-LSTM model. *Acta Energetica Solaris Sin.* **2022**, *43*, 75.
22. Choi, S.H.; Hur, J. Optimized-XG boost learner based bagging model for photovoltaic power forecasting. *Trans. Korean Inst. Electr. Eng.* **2020**, *69*, 978–984. [\[CrossRef\]](#)
23. Ghimire, S.; Deo, R.C.; Casillas-Perez, D.; Salcedo-Sanz, S. Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise Deep Residual model for short-term multi-step solar radiation prediction. *Renew. Energy* **2022**, *190*, 408–424. [\[CrossRef\]](#)
24. Zhang, T.; Zhang, X.; Chau, T.K.; Chow, Y.; Fernando, T.; Iu, H.H.-C. Highly accurate peak and valley prediction short-term net load forecasting approach based on decomposition for power systems with high PV penetration. *Appl. Energy* **2023**, *333*, 120641.
25. Li, N.; Li, L.; Zhang, F.; Jiao, T.; Wang, S.; Liu, X.; Wu, X. Research on short-term photovoltaic power prediction based on multi-scale similar days and ESN-KELM dual core prediction model. *Energy* **2023**, *277*, 127557. [\[CrossRef\]](#)
26. Netsanet, S.; Zheng, D.; Zhang, W.; Teshager, G. Short-term PV power forecasting using variational mode decomposition integrated with Ant colony optimization and neural network. *Energy Rep.* **2022**, *8*, 2022–2035. [\[CrossRef\]](#)
27. Behera, M.K.; Nayak, N. A comparative study on short-term PV power forecasting using decomposition based optimized extreme learning machine algorithm. *Eng. Sci. Technol. Int. J.* **2020**, *23*, 156–167. [\[CrossRef\]](#)

28. Li, Z.; Xu, R.; Luo, X.; Cao, X.; Du, S.; Sun, H. Short-term photovoltaic power prediction based on modal reconstruction and hybrid deep learning model. *Energy Rep.* **2022**, *8*, 9919–9932. [[CrossRef](#)]
29. Wang, S.; Liu, S.; Guan, X. Ultra-Short-Term Power Prediction of a Photovoltaic Power Station Based on the VMD-CEEMDAN-LSTM Model. *Front. Energy Res.* **2022**, *10*, 945327. [[CrossRef](#)]

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