electronics

MDPI

Article

A Short-Term Power Load Forecasting Method Based on SBOA–SVMD-TCN–BiLSTM

Mao Yang [1], Yiming Chen [1,*], Guozhong Fang [2], Chenglian Ma [1], Yunjing Liu [3] and Jinxin Wang [1]

[1] Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, Ministry of Education (Northeast Electric Power University), Jilin 132012, China; yangmao820@163.com (M.Y.)
[2] Northeast Branch of State Grid Corporation of China, Shenyang 110179, China
[3] Grid Jilin Electric Power Co., Ltd., Jilin Power Supply Company of State, Jilin 132011, China
* Correspondence: chenyiming0205@163.com; Tel.: +86-15844159969

Abstract: Short-term electricity load forecasting provides a basis for day-ahead energy scheduling. To improve the accuracy of short-term electricity load forecasts and deeply explore the temporal characteristics of load sequences, a method is proposed to extract predictable components of load sequences based on the secretary bird optimization algorithm (SBOA)-optimized successive variational mode decomposition (SVMD). This method decomposes the electricity load sequence into multiple subsequences under different time series. The combined forecasting architecture of the temporal convolutional network (TCN) and the bidirectional long short-term memory network (BiLSTM) is introduced to mine the temporal characteristics of each load component, resulting in short-term load forecasting outcomes. A case study is conducted using the annual electricity load data for the year 2018 from a specific region in Belgium. The experimental results show that the mean absolute error (MAE) of the TCN–BiLSTM model is reduced by 47.8%, 32.8%, and 11.5%, respectively, compared to other models. The root mean square error (RMSE) is reduced by 42.9%, 39.2%, and 11.3%, respectively, and the average goodness of fit R^2 is reduced by 9.81%.

Keywords: secretary bird optimization algorithm; temporal convolutional network; temporal feature extraction; short-term load forecasting

check for updates

Citation: Yang, M.; Chen, Y.; Fang, G.; Ma, C.; Liu, Y.; Wang, J. A Short-Term Power Load Forecasting Method Based on SBOA–SVMD-TCN–BiLSTM. *Electronics* **2024**, *13*, 3441. https://doi.org/10.3390/electronics13173441

Academic Editor: Ahmed Abu-Siada

Received: 31 July 2024
Revised: 17 August 2024
Accepted: 27 August 2024
Published: 30 August 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

Short-term electricity load forecasting refers to predicting the power demand for the coming days [1,2]. With the rapid economic development in our country, electricity demand has surged dramatically, and the distribution network, directly interfacing with users, faces integration challenges from new elements such as distributed renewable energy and electric vehicles. These changes have significantly altered the load characteristics of the distribution system, presenting substantial challenges to the safe and efficient operation of the power system. Accurate short-term load forecasting enables power departments to effectively formulate supply plans, optimize scheduling, and arrange electricity trading, thus ensuring a system balance and stability [3,4]. Therefore, there is an urgent need for more effective methods to analyze and predict the behavioral characteristics of the new distribution system.

Due to the influences of seasonal changes, day–night alternation, and social activity patterns, the characteristics of electricity load at different times vary significantly, showing multiple periodicities and random fluctuations over time [5]. Current short-term load forecasting methods mainly include traditional mathematical statistical methods and modern machine learning methods. Traditional methods include time series methods [6], multiple linear regression [7], and exponential smoothing [8]. These methods have simple modeling, fast computation, and a solid theoretical foundation, making them suitable for processing relatively stable data with obvious periodicity. However, they often fall short in terms

of accuracy when handling real-time, dynamically changing data due to complex internal nonlinear relationships. Modern machine learning methods, such as support vector machines [9], random forests [10], extreme learning machine [11], traditional artificial neural networks [12], and deep learning neural networks [13], can handle complex nonlinear relationships and generally offer high prediction accuracy and strong feature extraction capabilities, especially suitable for big data and dynamically changing data. However, support vector machines, random forests, extreme learning machines, and traditional artificial neural networks struggle to capture long-term dependencies and intrinsic sequences in time series data, weakening the continuity of load sequences over time.

Traditional forecasting methods have significant shortcomings in handling complex nonlinear relationships and long-term dependencies, failing to meet the needs of modern power systems. In recent years, deep learning neural networks have become a research hotspot in load forecasting due to their powerful data processing and feature extraction capabilities. Convolutional neural networks (CNNs) [14] can capture local patterns well through convolution and pooling operations but have limitations in dealing with long-term dependencies. Recurrent neural networks (RNNs) [15], like typical deep learning models, effectively capture time dependencies in time series data. However, RNNs tend to suffer from gradient vanishing and exploding problems when handling long time sequences. To address this issue, the long short-term memory network (LSTM) [16] was introduced. By incorporating input, forget, and output gates, LSTMs effectively mitigate gradient problems and significantly improve the ability to capture long-term dependencies. LSTMs can remember and forget information, thus better handling complex time series data. Although the LSTM performs well in many applications, its complex structure leads to longer training times. The gated recurrent unit (GRU) [17] neural network merges some gating mechanisms, reducing the number of parameters and improving computational efficiency. However, its simpler structure weakens the ability to capture dependencies among loads. The literature [18,19] suggests that bidirectional long short-term memory network (BiLSTMs) can more comprehensively capture time dependencies and complex features through the bidirectional processing of sequence data, improving prediction accuracy and stability. However, BiLSTMs can incur significant computational and storage overhead when processing very long sequences, potentially leading to redundant information processing and unnecessary complexity. The temporal convolutional network (TCN), as an emerging deep learning model, has gradually started to gain attention. The literature [20] suggests that TCNs, through their causal convolution and dilated convolution structures, can effectively capture long-term dependencies in time series data. By employing convolutional operations instead of recursive ones, TCNs overcome the gradient vanishing problem associated with handling long sequences. However, since convolutional operations primarily focus on local patterns, TCNs may exhibit limitations in capturing global contextual information.

To better explore and capture complex temporal features in load data, sequence decomposition techniques like empirical mode decomposition (EMD) [21] and variational mode decomposition (VMD) [22] have been introduced in recent years. EMD, a data-driven decomposition method, can decompose complex time series into several intrinsic mode functions (IMFs). VMD, an improvement upon EMD, decomposes signals into a set of sub-signals with finite bandwidth through a variational framework, enhancing decomposition robustness and stability. The literature [23] has found that VMD excels in noise suppression and mode mixing reduction, but its parameter selection is complex and requires repeated tuning for optimal decomposition results. Some research has utilized the advantage of CNNs in capturing local features to establish CNN–RNN combined forecasting models. The literature [24,25] uses CNNs to extract local features and RNNs to capture temporal dependencies, thus improving prediction accuracy. However, excessive convolution operations can reduce model efficiency in handling long-term dependencies, significantly increasing computational complexity and training time.

Despite the advancements made by modern load forecasting methods in handling nonlinear relationships, challenges remain in capturing temporal continuity. The literature [26]

suggests that while deep learning methods perform well in many applications, their complex model architectures still limit their comprehensive utilization of temporal information, particularly in effectively capturing the inherent multiple periodicities and stochastic fluctuations of the data. The literature [27] further points out that the long-term dependencies and evolving trends in load sequences are crucial for the stable operation of power systems. Therefore, how to more effectively capture and utilize the temporal continuity in load data remains a key focus and challenge in current research.

Addressing the issue of weakened temporal continuity in electricity load, this paper proposes a short-term electricity load forecasting method based on SBOA-optimized SVMD and TCN–BiLSTM, integrating deeply mined load temporal features into the forecasting model.

The contributions of this paper are as follows:

(1) A load sequence decomposition method based on SBOA-optimized SVMD is proposed to obtain more predictable load components while avoiding mode mixing;
(2) A load multi-component sequence forecasting method based on TCN temporal feature extraction and BiLSTM is introduced to extract temporal features of load sequences.

2. Electricity Load Decomposition

2.1. Successive Variational Mode Decomposition

VMD is a signal processing technique driven by data, which decomposes the original load data into K mode decomposition functions, achieving a multi-scale analysis of the original load data. When decomposing the original data, setting a large number of decompositions can lead to over-decomposition, while setting too few can result in high residual error Res. To address this issue, successive variational mode decomposition (SVMD) has been proposed. SVMD applies VMD multiple times, gradually removing the extracted mode components in each iteration, without the need to preset the value of K [28]. Compared to VMD, SVMD significantly reduces computational complexity and better captures the complex fluctuation characteristics in the electricity load. The method is as follows:

Assuming the original input signal $f(t)$ is decomposed into:

$$f(t) = \mu_L(t) + f_r(t) \tag{1}$$

In the equation, $\mu_L(t)$ is the L-th order mode; $f_r(t)$ is the residual signal, where $f_r(t)$ includes the total processed modes $\sum_{i=1}^{L-1} u_i(t)$ and the unprocessed part $f_u(t)$.

In order to fulfill the aforementioned hypothesis, it is necessary to satisfy the following conditions:

(1) To ensure that each mode is compact around its center frequency, thereby avoiding over-decomposition and noise interference, the L-th order mode minimization constraint is constructed as follows:

$$J_1 = \left\| \partial_t \left[\delta(t) + \frac{j}{\pi t} * \mu_k(t) \right] e^{-j\omega_L t} \right\|_2^2 \tag{2}$$

In the equation, ω_L represents the center frequency of the L-th order mode, $*$ represents the convolution operation.

(2) The spectral overlap between the residual signal and the L-th order mode should be minimized. To satisfy this constraint, an appropriate filter $\hat{\beta}_L(\omega)$ with the following frequency response should be selected:

$$\hat{\beta}_L(\omega) = \frac{1}{\alpha(\omega - \omega_L)^2} \tag{3}$$

In the equation, α is the bandwidth adjustment parameter of the modal function.

Based on the selected filter, the constraint can be constructed as follows:

$$J_2 = \|\beta_L(\omega) \times f_r(t)\|_2^2 \tag{4}$$

(3) To effectively distinguish between the L-th order mode and the $L-1$-th order mode, based on the J_2 constraint, an appropriate filter $\hat{\beta}_i(\omega)$ with the following frequency response should be selected:

$$\hat{\beta}_i(\omega) = \frac{1}{\alpha(\omega - \omega_i)^2}, i = 1, 2, \cdots, L-1 \tag{5}$$

Based on the selected filter, the constraint is constructed as follows:

$$J_3 = \sum_{i=1}^{L-1} \|\beta_i(\omega) \times \mu_L(t)\|_2^2 \tag{6}$$

(4) To ensure that the signal can be fully reconstructed, the following constraint is constructed:

$$f(t) = \mu_L(t) + f_u(t) + \sum_{i=1}^{L-1} u_i(t) \tag{7}$$

Therefore, the extraction of mode components can be simplified to the minimization problem of the above constraints:

$$\begin{aligned} &\min\{\eta J_1 + J_2 + J_3\} \\ &\text{s.t. } u_L(t) + f_r(t) = f(t) \end{aligned} \tag{8}$$

In the equation, η is the parameter that balances J_1, J_2, J_3.

The SVMD decomposition process is illustrated in Figure 1:

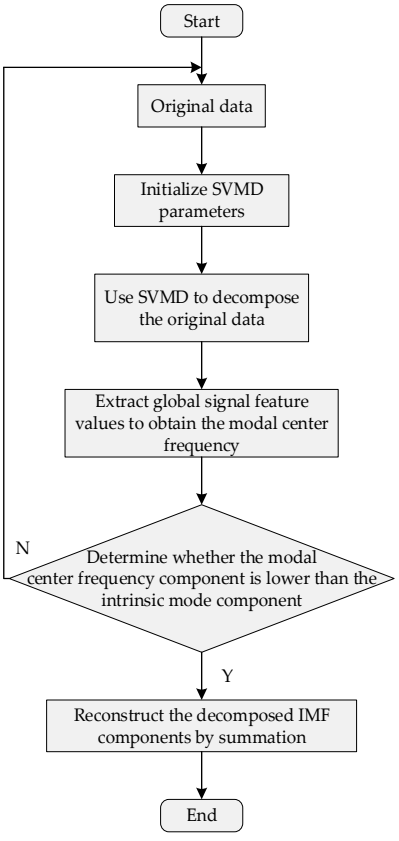


Figure 1. SVMD Flowchart.

The SVMD process is illustrated in Figure 1. Initially, the parameters of SVMD are set and initialized. Using these initialized parameters, the original data is decomposed to extract the global characteristics of the signal and to compute the center frequencies of each mode. Subsequently, these modal center frequencies are evaluated to determine if they are below a predefined intrinsic mode component. If the evaluation result is "yes", the extracted IMFs are summed to reconstruct the signal. If the result is "no", the process returns to the first step, where parameters are reinitialized, and data decomposition is performed again until the specified conditions are met.

2.2. Secretary Bird Optimization Algorithm

When performing SVMD decomposition, it is necessary to determine the parameters. Parameters with a minor impact on the decomposition are set to empirical values, while the mode compactness coefficient (maxAlpha) significantly influences the compactness and fidelity of the modes, thus having a major impact on the decomposition. To ensure that the SVMD algorithm can decompose clear characteristic information, this paper employs SBOA to optimize maxAlpha, ensuring that the modes are neither too dispersed nor too concentrated, thereby avoiding mode mixing.

2.2.1. Initial Preparation Phase

SBOA operates as a population-centric metaheuristic approach. Each secretary bird is regarded as a member of the algorithm's solution set, and its position represents a potential solution to the current problem [29]. In the initial phase, the positions of the secretary birds in the search space are randomly initialized:

$$X_{i,j} = ll_j + r \times (ul_j - ll_j), i = 1, 2, \cdots, N, j = 1, 2, \cdots, Dim \tag{9}$$

In the equation, X_i denotes the location of the i-th secretary bird, while ll_j signifies the lower limit and ul_j signifies the upper limit. r is a random value ranging from 0 to 1, and Dim represents the dimension of the problem variables.

As indicated in Equation (10), the optimization process of SBOA starts with a population of candidate solutions, which are initialized within the ul and ll limits defined by the problem.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{i,j} & \cdots & x_{1,Dim} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1} & & x_{i,j} & & x_{i,Dim} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,j} & \cdots & x_{N,Dim} \end{bmatrix}_{N \times Dim} \tag{10}$$

In the equation, X represents a set of all secretary birds, X_i represents the i-th secretary bird, $X_{i,j}$ denotes the j-th problem variable value of the i-th secretary bird, and N denotes the total count of members in the population of secretary birds.

The objective function is evaluated based on the problem variable values of the candidate solutions represented by each secretary bird, and the results are compiled into a vector F according to Equation (11).

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N \times 1} \tag{11}$$

In the equation, F_i indicates the objective function value obtained by the i-th secretary bird. The optimal candidate solution for the given problem is identified by comparing these objective function values.

Two different natural behaviors of secretary birds are used to update SBOA members. These behaviors include:

- Hunting strategy.
- Escape strategy.

Therefore, in each iteration, every candidate solution is updated in two distinct phases.

2.2.2. Hunting Strategy of Secretary Birds

The hunting process is refined into three stages: searching, consuming, and attacking prey. The entire predation process is divided into three equal time intervals.

(1) Searching for Prey Stage: In the first stage, a differential evolution strategy is adopted to enhance the algorithm's diversity and global search capabilities. By incorporating differential mutation mechanisms, it improves the likelihood of finding the global optimum while avoiding local optima. Therefore, the position update of secretary birds in the search for prey stage can be modeled by Equations (12) and (13).

$$x_{i,j}^{newP1} = x_{i,j} + (x_{random_1} - x_{random_2}) \times R_1 \, , t < \frac{1}{3}T \tag{12}$$

$$X_i = \begin{cases} X_i^{new,P1}, F_i^{new,P1} < F_i \\ X_i, \text{ else} \end{cases} \tag{13}$$

In the equations, t represents the current number of iterations, T represents the maximum number of iterations. $X_i^{new,P1}$ denotes the updated state of the i-th secretary bird during the initial phase, x_{random_1} and x_{random_2} are random candidate solutions. R_1 represents an array of dimension $1 \times Dim$ randomly generated within the interval [0, 1]. $x_{i,j}^{newP1}$ represents the value of the j-th dimension, and $F_i^{new,P1}$ represents its fitness value.

(2) Consuming Prey Stage: In the second stage, Brownian motion (RB) and "x_{best}" (individual historical best position) are introduced to simulate the random movement of secretary birds. RB can be mathematically modeled by Equation (14). By using "x_{best}", individuals can search near optimal positions, enhancing their local search capabilities. The introduction of RB randomness effectively prevents local optima and helps find better solutions. Consequently, the position updates of secretary birds during the prey consumption stage can be modeled by Equations (15) and (16).

$$RB = randn(1, Dim) \tag{14}$$

$$x_{i,j}^{newP1} = x_{best} + \exp((t/T) \wedge 4) \times (RB - 0.5) \times (x_{best} - x_{i,j}) \, , \, \frac{1}{3}T < t < \frac{2}{3}T \tag{15}$$

$$X_i = \begin{cases} X_i^{new,P1}, F_i^{new,P1} < F_i \\ X_i \, , \text{ else} \end{cases} \tag{16}$$

In the equations, $randn(1, Dim)$ is an array of dimension $1 \times Dim$ generated randomly from a standard normal distribution (mean of 0, standard deviation of 1), while x_{best} denotes the current optimal value.

(3) Attacking Prey Stage: Introducing the *Levy* flight strategy enhances the global search capability of the optimizer, and incorporating a nonlinear perturbation factor, denoted as $(1 - \frac{t}{T})(2 \times \frac{t}{T})$, improves the algorithm's performance. Therefore, the position of secretary birds in the attacking prey stage is updated by Equations (17) and (18).

$$x_{i,j}^{newP1} = x_{best} + ((1 - \frac{t}{T}) \wedge (2 \times \frac{t}{T})) \times x_{i,j} \times RL \, , t > \frac{2}{3}T \tag{17}$$

$$X_i = \begin{cases} X_i^{new,P1}, F_i^{new,P1} < F_i \\ X_i, else \end{cases} \tag{18}$$

The calculation method for RL is as follows:

$$RL = 0.5 \times Levy(Dim) \tag{19}$$

In the equations, s denotes the distribution function. The calculation method is as follows:

$$Levy(D) = s \times \frac{u \times \sigma}{|v|^{\frac{1}{\eta}}} \tag{20}$$

In the equations, s and η are fixed values, with s set to 0.01 and η set to 1.5. u and v are random numbers between 0 and 1. The calculation method for σ is as follows:

$$\sigma = \left(\frac{\Gamma(1+\eta) \times \sin(\frac{\pi\eta}{2})}{\Gamma(\frac{1+\eta}{2}) \times \eta \times 2(\frac{\eta-1}{2})} \right)^{\frac{1}{\eta}} \tag{21}$$

In the equations, Γ represents the gamma function, and the value of η is 1.5.

2.2.3. Escape Strategy of Secretary Birds

When secretary birds encounter threats from other predators, they typically employ various camouflage or escape strategies to protect themselves or their food. These strategies can be broadly divided into two categories:

- Strategy C_1: Camouflage;
- Strategy C_2: Escape.

Assuming the probabilities of the two categories are equal:

In the escape strategy of secretary birds, the birds will first choose to camouflage. If camouflage is not possible, they will choose to escape. Based on this premise, the introduction of the perturbation factor $(1 - \frac{t}{T})^2$ helps the algorithm select the optimal solution from both new and existing solutions. In summary, the two escape strategies of secretary birds can be represented by Equation (22), and the updated condition can be represented by Equation (23).

$$x_{i,j}^{new,P2} = \begin{cases} C_1 : x_{best} + (2 \times RB - 1) \times (1 - \frac{t}{T})^2 \times x_{i,j}, rand < r_i \\ C_2 : x_{i,j} + R_2 \times (x_{random} - K \times x_{i,j}), \text{else} \end{cases} \tag{22}$$

$$X_i = \begin{cases} X_i^{new,P2}, F_i^{new,P2} < F_i \\ X_i, \text{else} \end{cases} \tag{23}$$

In the equations, r_i is set to 0.5, R_2 represents an array of dimension $1 \times Dim$ randomly generated from a normal distribution, x_{random} represents the random candidate solution of the current iteration, and K represents a random choice between the integers 1 or 2, which can be calculated using Equation (24).

$$K = round(1 + rand(1, 1)) \tag{24}$$

In the equations, $rand(1, 1)$ represents a random number generated between 0 and 1.

The optimization process of SBOA is shown in Figure 2:

The SBOA process is illustrated in Figure 2. It begins with the initialization of parameters and the calculation of the fitness values of individuals to find the optimal position. In the hunting phase of the secretary birds, the entire predation process (maximum number of iterations T) is divided into three equal intervals. The current iteration count t is used to determine which interval it belongs to, and the position X_i of the i-th secretary bird is updated using Equations (12), (15) and (17). Subsequently, in the escape phase of the secretary birds, the value *rand* is assessed to see if it is greater than 0.5. If the evaluation result is

"yes", then parameter X_i is updated using the formula C_1 in Equation (22). If the evaluation result is "no", then parameter X_i is updated using the formula C_2 in Equation (22). The optimal value x_{best} for the problem is then updated from the results of each secretary bird. Finally, the algorithm checks whether the maximum number of iterations has been reached. If this condition is satisfied, the optimal value is output; otherwise, the algorithm returns to the hunting phase and continues iterating.

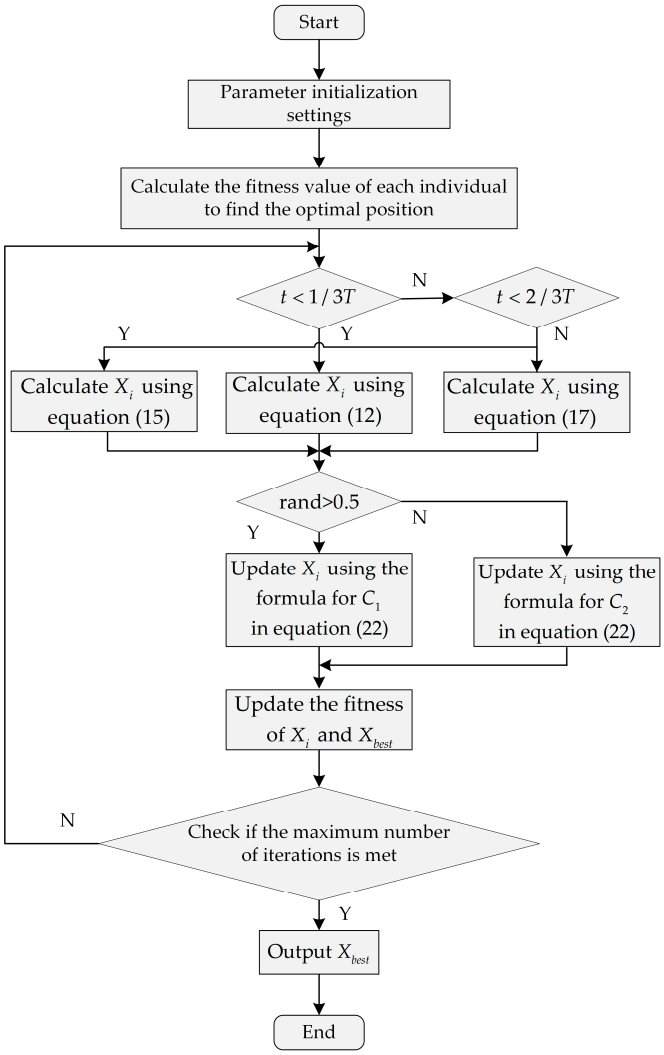


Figure 2. SBOA optimization flowchart.

3. Short-Term Power Load Forecasting Model

3.1. TCN Neural Network Structure

TCN is a novel neural network based on the CNN architecture. TCN employs structures such as dilated causal convolutions and residual blocks. The structure of TCN's dilated causal convolutions is illustrated in Figure 3. Compared to traditional CNN, causal convolutions focus only on past and current information, without considering future data. This means that the output value y_t at time t is determined solely by the input values at time t and earlier, up to input $\{x_0, x_1, \cdots x_{t-1}, x_t\}$. Traditional CNN increases the receptive field by adding pooling layers, which can lead to information loss. In contrast, TCN introduces dilated convolutions to expand the receptive field, allowing it to capture dependencies over a larger range without losing information. The operation of dilated convolution is expressed as follows:

$$F(t) = \sum_{i=0}^{k-1} f(i) * x_{t-p*i} \tag{25}$$

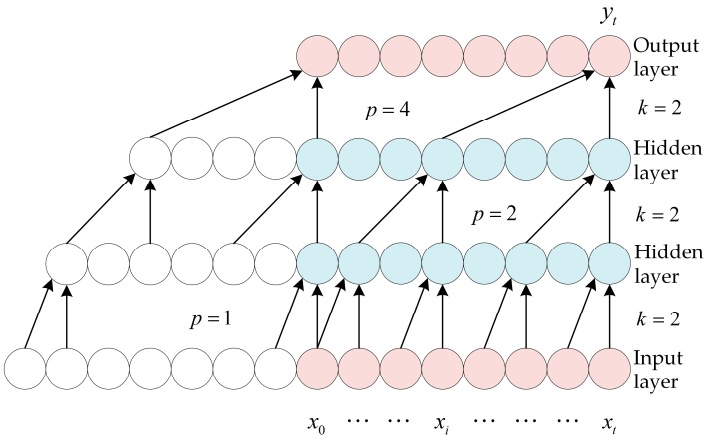


Figure 3. TCN dilated causal convolution structure.

In the equation, t represents the index of the sequence element, p denotes the dilation factor, $x_t \in R^n$ refers to the sequence element, $F(t)$ is the dilated convolution of the sequence element x_t, and k indicates the size of the convolution kernel.

The TCN residual unit is shown in Figure 4. The residual connection adds the input x to the output $f(x)$. $f(x)$ can be expressed as:

$$f(x) = h(x) - x \tag{26}$$

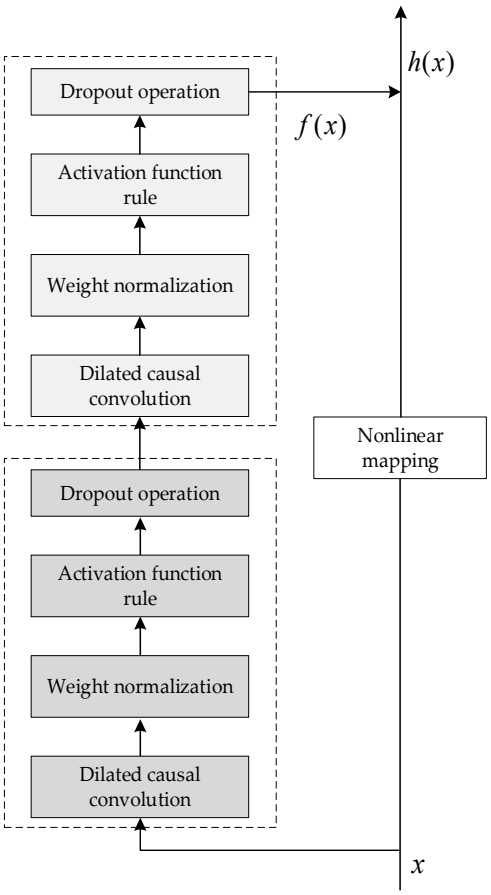


Figure 4. TCN residual unit.

In Figure 4, the addition of causal convolutions in TCN ensures that information flows strictly from the past to the future, thereby preventing information leakage from

future to past. Even with fewer layers, the application of dilated causal convolution (DCC) enables TCN to have a larger receptive field, allowing it to process longer time series data. Furthermore, DCC incorporates techniques such as the ReLU activation function, weight normalization, and dropout regularization. These methods not only enhance the model's nonlinear representation capabilities but also improve its stability and generalization performance, ultimately boosting the model's effectiveness in time series forecasting tasks.

3.2. BiLSTM Neural Network

The structure of the LSTM neural network is shown in Figure 5:

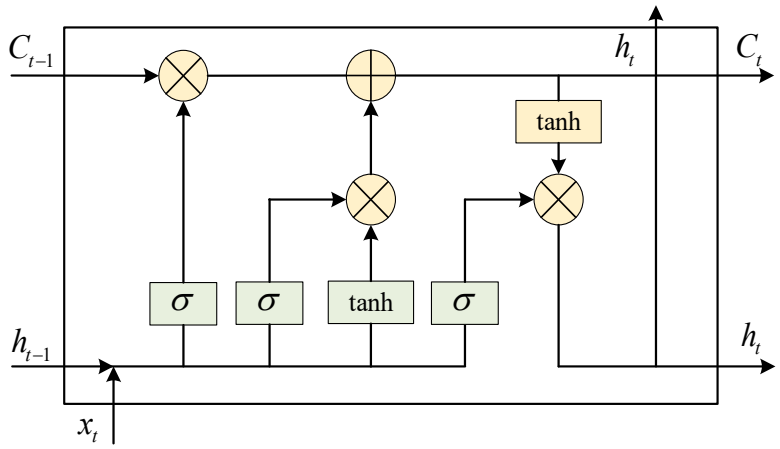


Figure 5. LSTM model structure diagram.

In Figure 5, the input gate (i_t) of the LSTM network determines whether new information should be stored in the cell state, the forget gate (f_t) selectively discards unnecessary information to prevent information overload, and the output gate (o_t) controls the state output of the memory cell. The collaboration of these three gates allows the LSTM network to effectively capture and maintain both long-term and short-term dependencies in time series data.

The computation formulas for the LSTM network are as follows:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{27}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{28}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{29}$$

$$\widetilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{30}$$

$$c_t = i_t \cdot \widetilde{c}_t + f_t \cdot c_{t-1} \tag{31}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{32}$$

In the equation, i_t is determined by the input x_t, the hidden layer output from the previous time step h_{t-1}, and the activation function σ. The weights and biases for the gates are represented by W_i, U_i, b_i, W_f, U_f, b_f, W_o, U_o, and b_o.

BiLSTM adopts a bidirectional structure, which fully leverages both past and future contextual information, enhancing the model's ability to understand and predict sequential data. The structure of BiLSTM is shown in Figure 6.

In Figure 6, h_t represents the forward hidden state, h_i represents the backward hidden state, and each level of the hidden layer h_t is composed of three parts: the existing input x_t, the forward hidden state output from the previous time step h_{t-1}, and the backward hidden state output from the previous time step h_{i-1}.

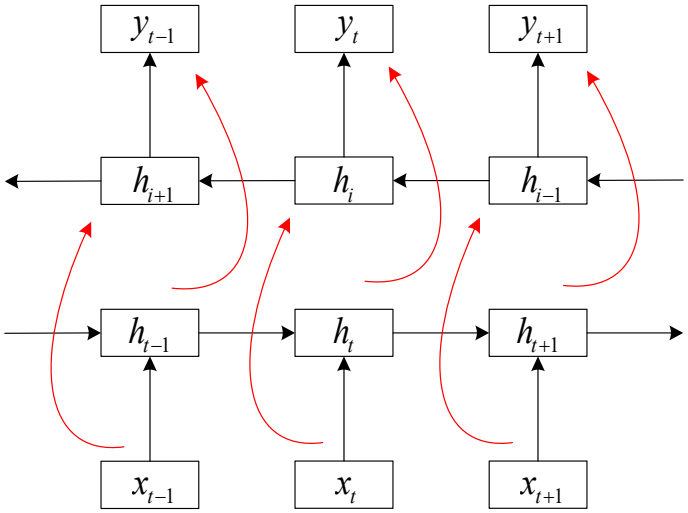


Figure 6. BiLSTM model structure diagram.

3.3. Combined Prediction Model Based on TCN–BiLSTM

The TCN–BiLSTM combined prediction neural network model is shown in Figure 7.

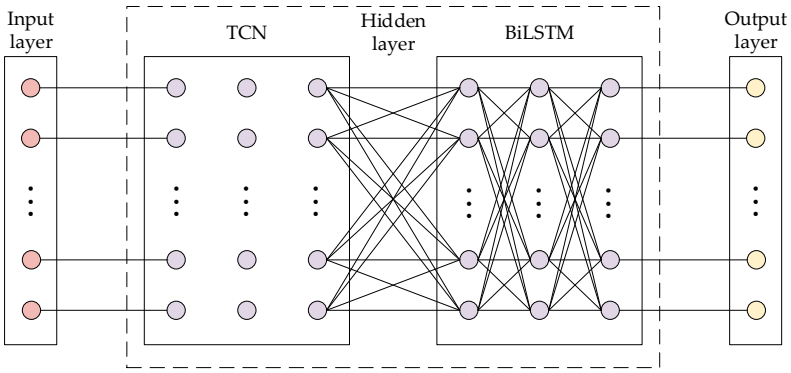


Figure 7. TCN-BiLSTM model structure diagram.

As shown in Figure 7, the model structure includes the following steps:

(1) The input layer consists of IMF components obtained from the SVMD decomposition.
(2) In the TCN layer, a 1-layer residual unit is used. The residual unit is composed of 2 convolution units and multiple nonlinear mappings. The ReLU function is used as the activation function to normalize the weights.
(3) The output vector of the TCN model serves as the input for the BiLSTM.
(4) The output layer includes 1 fully connected layer, which contains 1 neuron, and its output is the load prediction value for one time step.

The model architecture fully leverages the IMF components decomposed by SVMD, capturing short-term features in the time series through TCN, and further capturing long- and short-term dependencies in the time series using BiLSTM. The residual units in the TCN enhance the model's ability to extract complex features by introducing multiple nonlinear mapping layers. The use of the ReLU activation function helps mitigate the vanishing gradient problem, making the training process more stable and efficient. The fully connected layer maps the extracted complex features to load prediction values, enabling the model to handle high-dimensional inputs and provide accurate predictions. This design allows the model to better capture the intrinsic patterns in time series data, thereby improving prediction accuracy and reliability.

4. Combined Prediction Model Based on SVMD-SBOA-TCN-BiLSTM

In this paper, the electrical load is decomposed using SVMD optimized by SBOA, and the TCN–BiLSTM combined model is used for prediction. The model flowchart is shown in Figure 8. The specific steps are as follows:

(1) A novel swarm intelligence optimization algorithm, SBOA, is used to optimize the key parameter of SVMD.
(2) The electrical load sequence is decomposed into multiple IMFs over different time scales using SVMD.
(3) The decomposed IMFs are used as basic units and input into the TCN for feature extraction, outputting feature vectors containing rich information.
(4) These feature vectors are then used as inputs to the BiLSTM for prediction, resulting in the forecasted values for each IMFs.
(5) The final prediction result is obtained by integrating the predicted values of different IMFs.

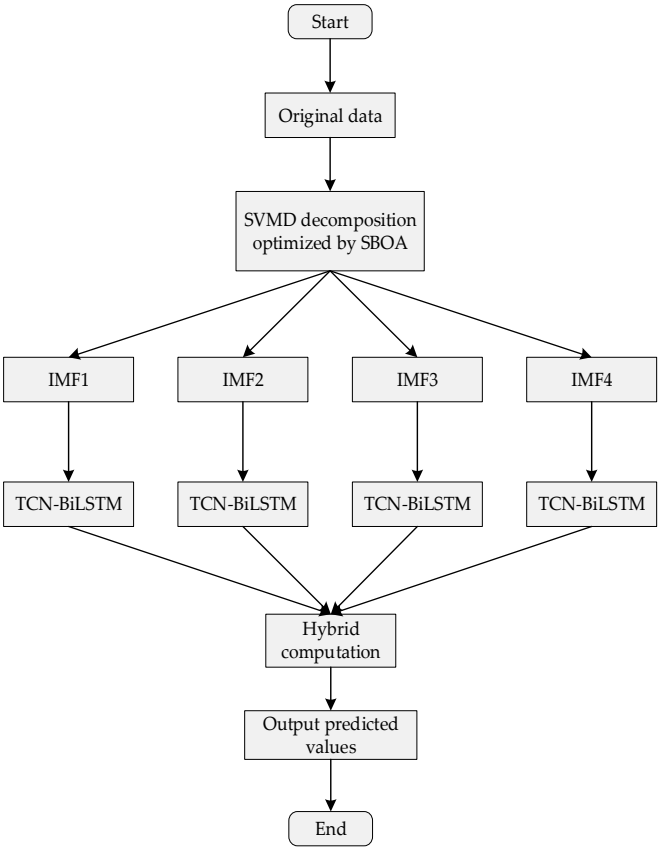


Figure 8. Prediction model flowchart.

The proposed model combines the precise decomposition capability of SVMD, the optimization advantages of SBOA, the feature extraction efficiency of TCN, and the bidirectional sequential modeling capability of BiLSTM to significantly improve the extraction and prediction accuracy of the power load sequence's temporal features. SVMD accurately decomposes power load data, allowing each IMF component to independently reflect specific fluctuation characteristics within the load data. These components better capture the data's non-linearity and non-stationarity. The SBOA-optimized SVMD parameters further enhance the decomposition accuracy, making each IMF component more representative. When processing these IMF components, TCN efficiently extracts complex patterns and long short-term dependencies from the time series through its convolutional structure, enhancing the feature vector's expressive capability. BiLSTM, with its bidirectional network structure, comprehensively understands the time series data, capturing the temporal relationships both forward and backward, thereby further improving prediction perfor-

mance. This multi-layered composite model not only enhances the accuracy of power load forecasting but also demonstrates advantages in handling complex temporal data.

5. Experimental Analysis

5.1. Data Preparation

The raw power load dataset selected for this study is sourced from a region in Belgium. The data collection spans from 1 January 2018 to 31 December 2018, with a sampling frequency of every 15 min, resulting in a total of 35,040 data points. The data are split into a training set and a test set in a ratio of 5:1. A day-ahead forecasting method is adopted, with the original dataset segmented into daily sequences of 96 data points each. Data from the first 10 months are used as the training set, while the remaining data are used for testing. During the model training, an Adam optimizer is used, and training is executed on a GPU to accelerate the process. The trained model is applied for day-ahead forecasting, with each day's forecast based on the previous day's data. The prediction results are denormalized and aggregated daily to produce a forecast curve for the next two months, which is then compared with actual measured data to evaluate the model performance. The laboratory hardware configuration includes a computer with an Intel Core i9-13900HX CPU, 16 GB of memory, and an NVIDIA GeForce RTX 4070 GPU for acceleration. The experiments are programmed using Matlab2024a.

To address issues such as missing data and abnormal fluctuations in the original power load dataset, this study employs the quartile method to detect outliers and convert them to missing data, which are then filled using an interpolation function for upsampling. Additionally, since the load IMFs obtained from SVMD decomposition have significant numerical differences, directly using these decomposed data for prediction could adversely affect the overall mapping effect of the network. Therefore, it is necessary to normalize the input sequence to prevent the problem of dimensional explosion during training. The normalization method scales the data to the range of [0, 1] to improve the efficiency and stability of model training. The normalization equation used in this study is as follows:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{33}$$

In the equation, $X_{\min}$ and $X_{\max}$ represent the minimum and maximum values of the input data matrix X. The normalized data X_{norm} ensure that the input features are on the same scale, which helps to accelerate the convergence speed of the model. To restore the normalized data to the original scale after prediction, the inverse normalization equation is used as follows:

$$X_{\text{restore}} = X_{\text{norm}} \cdot (X_{\max} - X_{\min}) + X_{\min} \tag{34}$$

5.2. Parameter Settings and Evaluation Metrics

To obtain the optimal structure of the deep learning model, this study configures the TCN with 10 convolutional filters and a filter size of 2. Additionally, one residual block is incorporated to mitigate the vanishing gradient problem and ensure the training stability of the deep network. To prevent overfitting, a dropout layer with a dropout rate of 0.02 is introduced in the TCN. The BiLSTM layer is configured with 60 hidden units, a maximum of 100 epochs, and the ReLU activation function. To further enhance the model's performance, the initial learning rate is set to 0.005 and is halved every 2 epochs through an automatic adjustment callback function. The training process utilizes the Adam optimizer. These parameter settings, determined through iterative experimentation and optimization, aim to improve the model's prediction accuracy and generalization ability. The choices of these settings have undergone multiple adjustments to ensure the model can effectively extract local features from the time series data and perform excellently in practical applications. Table 1 shows the parameter settings for SVMD and SBOA.

Table 1. Parameter settings for SVMD and SBOA.

SVMD Parameters	**Value**	Type of Stopping Criteria	4
compactness of mode	19,990.25	**SBOA Parameters**	**Value**
time step of the dual ascent	0	pop	30
tolerance of convergence criterion	1×10^{-6}	max_iteration	60

To evaluate the prediction accuracy of the SBOA–SVMD–TCN–BiLSTM prediction model, this paper utilizes MAE, RMSE, and goodness of fit R^2 as evaluation metrics. The calculation formulas are as follows:

$$\sigma_{\mathrm{MAE}} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{35}$$

$$\sigma_{\mathrm{RMSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{36}$$

$$\sigma_{\mathrm{R^2}} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2}. \tag{37}$$

In the equation, $\hat{y}_i$ represents the predicted power load at time i, y_i represents the actual power load at time i, and $\overline{y}_i$ represents the average predicted power load. The smaller the MAE and RMSE, the better the model's prediction performance. The closer R^2 is to 1, the better the fit of the prediction model.

5.3. Power Load Sequence Decomposition

In this study, normalized load data were decomposed using the SVMD method optimized by SBOA. The data were selected from a region in Belgium, covering the first two months of spring and autumn, as well as the first three months of summer and winter in 2018. For each of these months, data from the first seven days were selected at 15-min intervals, resulting in a total of 6720 data points. The decomposition results are shown in Figure 9.

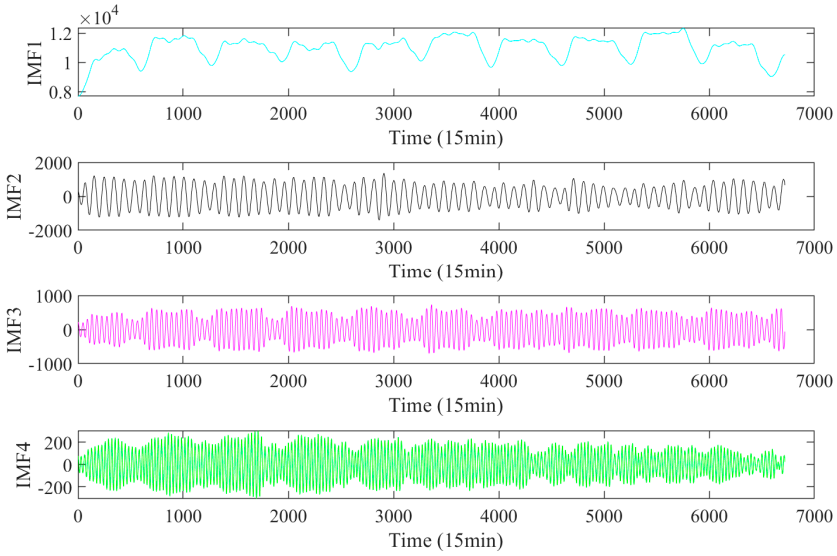


Figure 9. SVMD decomposition diagram.

The original power load sequence is decomposed into four single-frequency IMF components: IMF1 to IMF4. Each IMF component has clear features and corresponds to different central frequencies:

- IMF1: Exhibits weak volatility and does not cross the zero point, approximately representing the trend of the original power load.
- IMF2: Has a relatively low frequency with evident periodicity, reflecting the regularity of the power load to some extent.
- IMF3 to IMF4: These components have relatively higher frequencies and weaker periodicity, representing both the regularity and the anomalies or sudden events in the power load.

Each IMF component can represent the characteristics of the power load to a certain extent. This decomposition effectively solves the modal aliasing issue seen in other algorithms, ensures the integrity of the decomposed signal, and enhances the accuracy of the model's prediction.

In this study, the mode compactness parameter in SVMD is adjusted, using permutation entropy as a metric to reflect the effectiveness of SVMD parameter optimization in controlling the complexity of the sequence. Permutation entropy is a key indicator of sequence complexity, with values ranging from 0 to 1. A lower permutation entropy indicates lower complexity and higher predictability of the sequence, while values closer to 1 suggest higher complexity and increased difficulty in prediction [30,31].

Table 2 presents the different mode compactness parameters in SVMD and their corresponding permutation entropy values. It is evident that the optimized SVMD parameters achieved the lowest permutation entropy, indicating that the SVMD parameters optimized by SBOA significantly reduced sequence complexity, thereby enhancing prediction performance.

Table 2. Permutation entropy corresponding to different mode compactness parameters in SVMD.

SVMD Parameter (Compactness of Mode)	Permutation Entropy
15,000	0.3462
17,500	0.2857
19,990.25 (Optimized)	0.1245

5.4. Comparison of Different Optimization Algorithms

To verify the effectiveness and superiority of the proposed SBOA, we compared it with the sparrow search algorithm (SSA) and the grey wolf optimizer (GWO). In each iteration, the minimum permutation entropy of the decomposed sequences was used as the objective function. The performance of different optimization algorithms was evaluated by comparing the average permutation entropy after optimization. To ensure the fairness of the experiments, the population size for each algorithm was set to 30, with a maximum of 60 iterations. The optimization fitness curves of the different algorithms are shown in Figure 10.

In Figure 10, by comparing the fitness of SBOA, SSA, and GWO, it is evident that SBOA achieves a significantly lower fitness value upon convergence compared to GWO and SSA. Moreover, the fitness value stabilizes thereafter, indicating that SBOA effectively avoids over-searching and does not fall into local optima, demonstrating stability in the later stages of optimization. In contrast, SSA shows a steep decline in the first few iterations but has a slower convergence rate and exhibits some fluctuations, making it less stable than SBOA. Meanwhile, GWO's fitness curve remains relatively stable throughout the iterations, showing some global search capability, but its fitness value does not reach the level achieved by SBOA and SSA in the later stages.

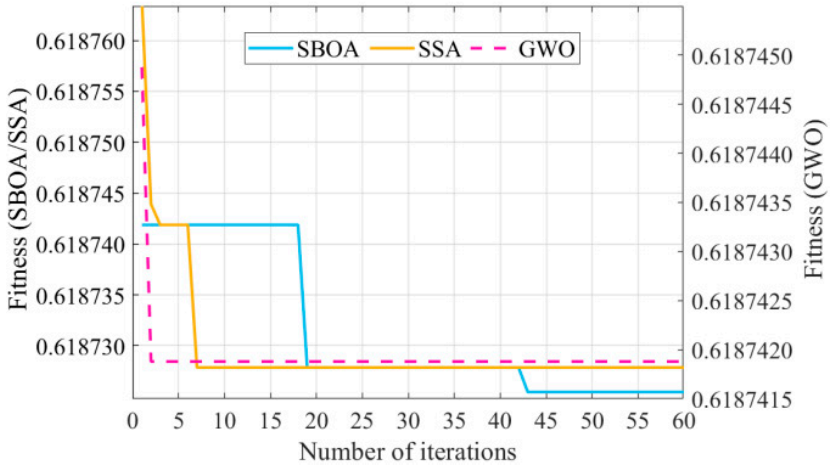


Figure 10. Optimization fitness curves.

5.5. Comparison of Different Sequence Decomposition Algorithms

To validate the superiority of the SVMD sequence decomposition algorithm used in this study, comparative models were established with CEEMDAN and ICEEMDAN decomposition methods. The parameter settings for these two methods are consistent, as shown in Table 3. The last week of the test set was selected for analysis, and the comparison of prediction results for a specific day of that week is shown in Figure 11.

Table 3. Parameter settings for CEEMDAN and ICEEMDAN.

Parameters	Value	Parameters	Value
Nstd	0.2	MaxIter criterion	1000
NR	100	SNRFlag	2

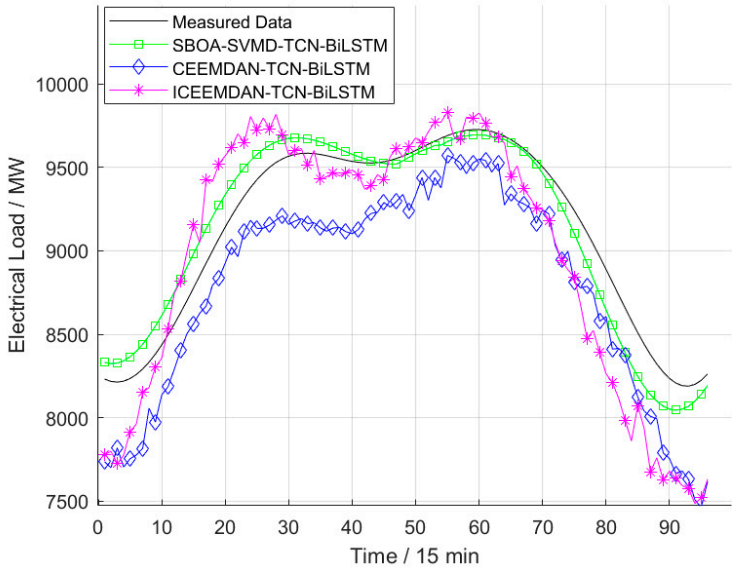


Figure 11. Comparison of prediction results with different sequence decomposition algorithms.

As shown in Figure 11, the load prediction result curve of the proposed model is the closest to the actual load curve, indicating the best load prediction performance. A detailed comparison of load prediction errors is presented in Table 4.

Table 4. Comparison of prediction errors between different decomposition methods.

Model	MAE/MW	RMSE/MW	R^2/%
CEEMDAN-TCN-BiLSTM	309.5300	402.4907	0.8854
ICEEMDAN-TCN-BiLSTM	267.1721	369.5974	0.9063
SBOA-SVMD-TCN-BiLSTM	219.7098	309.2698	0.9273

As shown in Table 4, the prediction errors of the SBOA–SVMD–TCN–BiLSTM model, in terms of MAE and RMSE, are reduced by 29.01% and 17.76%, 23.16% and 16.32%, respectively, compared to the other two models, with an improvement in metric R^2 as well. This indicates that the SVMD algorithm used in this study outperforms other decomposition algorithms in terms of prediction accuracy and model fitting. In contrast, the CEEMDAN and ICEEMDAN decomposition algorithms suffer from severe mode mixing, resulting in frequency components that lack clear periodicity and distinct features, thereby increasing prediction errors. Additionally, the increase in the number of decomposition components further amplifies errors. The SVMD algorithm effectively reduces mode mixing and accurately extracts different characteristic components of the load series, leading to more precise prediction results.

5.6. Comparison of Predictive Model Performance

To validate the necessity of SBOA–SVMD, the forecasting results of the proposed model are compared with those of the TCN–BiLSTM model. In the proposed model, the IMFs obtained from SBOA-SVMD decomposition are used as inputs for the TCN–BiLSTM model, while in the comparison experiment, the processed raw load data is directly used as input for the TCN–BiLSTM model. The analysis focuses on the forecasting results from the last two weeks of the test set, with the comparison of results for a specific day shown in Figure 12.

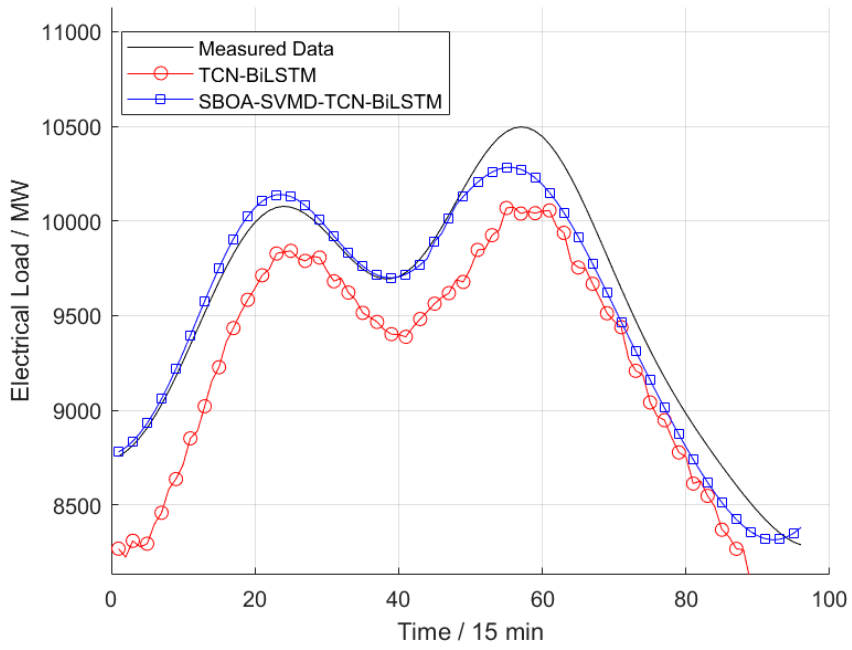


Figure 12. Comparison of prediction models with and without decomposition algorithms.

As shown in Figure 12, the proposed prediction model is closer to the actual values, demonstrating the effectiveness and necessity of the SBOA–SVMD approach. The prediction errors are listed in Table 5.

Table 5. Comparison of prediction errors between models with and without decomposition algorithms.

Model	MAE/MW	RMSE/MW	R^2/%
TCN-BiLSTM	337.4591	464.9645	0.8346
SBOA-SVMD-TCN-BiLSTM	219.7098	309.2698	0.9273

As shown in Table 5, compared to the TCN–BiLSTM model, the prediction errors MAE and RMSE of the proposed model are reduced by 34.89% and 33.49%, respectively, while R^2 is improved by 11.1%, further confirming the necessity of introducing the SBOA–SVMD optimization decomposition method.

To further verify the predictive accuracy of the SBOA–SVMD–TCN–BiLSTM model, this paper uses the IMFs obtained from the SBOA–SVMD decomposition as inputs and introduces traditional prediction models such as ELM, LSTM, and BiLSTM for comparison. The experimental results are shown in Figure 13.

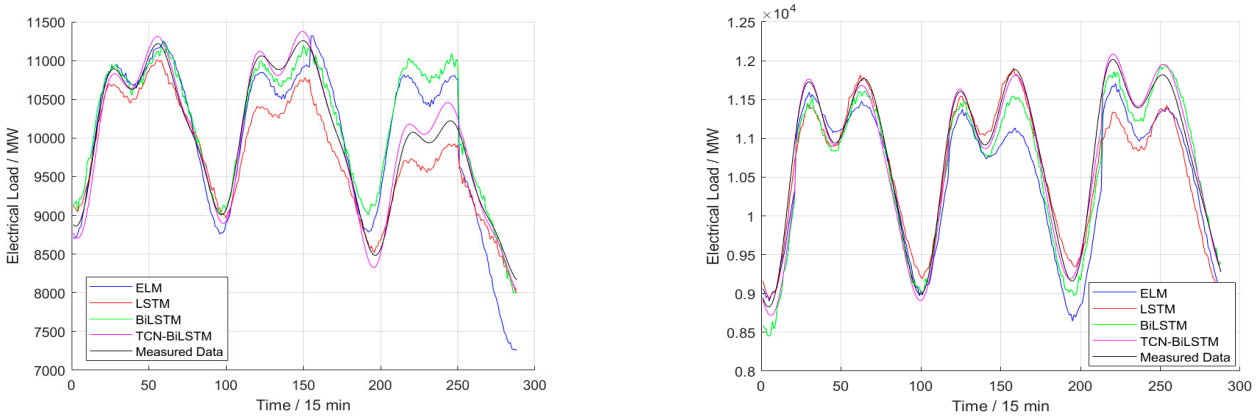


Figure 13. Prediction results of different methods. (**a**) Prediction comparison for three specific days in the first month of the test set; (**b**) Prediction comparison for three specific days in the second month of the test set.

Figure 13 compares the prediction results of different models for three randomly selected days from the test set. By comparing the model predictions with actual measured data, it can be observed that the TCN–BiLSTM model's predictions closely match the measured data, especially in peak and trough regions, showing high prediction accuracy. This model not only accurately predicts the load curve's changing pattern but also captures abrupt changes in the load curve effectively. The ELM model, while reflecting the overall trend of the power load, shows significant deviations in regions with large load changes due to its limited ability to handle long-term dependencies, resulting in predictions that are generally lower than the actual data. The LSTM model's overall prediction performance is poor, struggling with long-term dependencies and sensitivity to noise, leading to low accuracy in areas with rapid fluctuations, particularly at load peaks and troughs. The BiLSTM model performs relatively well; its bidirectional structure accounts for both forward and backward time dependencies, resulting in predictions that are mostly consistent with actual data. However, it still shows some deviations in regions with drastic load changes, indicating room for improvement in predicting high-frequency fluctuations. The TCN–BiLSTM model exhibits the best prediction performance. The model leverages multi-scale feature extraction in the TCN layer and bidirectional dependency capture in the BiLSTM layer, significantly enhancing prediction accuracy. The predictions are highly consistent with actual measurements, effectively capturing subtle load variations and validating the TCN–BiLSTM model's high precision in both high-frequency fluctuations and complex load patterns.

Table 6 shows the comparison of evaluation metrics for different models. From Table 6, it can be observed that the TCN–BiLSTM model significantly outperforms the LSTM, ELM, and BiLSTM models. Specifically, the TCN–BiLSTM model reduces the MAE by 47.8%, 32.8%, and 11.5%, respectively, compared to the LSTM, ELM, and BiLSTM models. Similarly, the RMSE is reduced by 42.9%, 39.2%, and 11.3%, and the R^2 value is significantly higher. It can be seen that the proposed model outperforms other models in terms of MAE, RMSE, and R^2 metrics. Compared to BiLSTM, the proposed model introduces TCN, a deep learning model with spatial feature extraction capabilities. TCN effectively captures useful information from signals through multi-layer convolutions that extract features at different time scales. Simultaneously, BiLSTM captures more information by considering bidirectional dependencies in time series data. This combination enhances the model's ability to handle complex time series data, thereby improving the model's stability and accuracy. Additionally, the validation shows that single prediction models have poor prediction capabilities and larger errors, further verifying the high prediction accuracy of the proposed model.

Table 6. Comparison of prediction accuracy of different models.

Model	MAE/MW	RMSE/MW	R^2/%
LSTM	420.6167	542.0793	0.7766
ELM	326.9141	508.3097	0.8035
BiLSTM	248.1770	348.7726	0.9075
TCN-BiLSTM	219.7098	309.2698	0.9273

Table 7 shows the training and prediction times of the proposed model for different IMFs.

Table 7. The training and prediction times for different IMFs.

IMF	Training Time/s	Testing Time/s
IMF1	98.5469	0.5882
IMF2	86.8445	0.1254
IMF3	85.2174	0.1010
IMF4	84.5613	0.0728

This study utilizes an offline training and online prediction approach. The total training time is recorded as 355.17 s, and the total testing time is recorded as 0.89 s. Although the training time is relatively long, the model's testing time is very short. In practical applications, the model can generate predictions quickly, so it will not be significantly impacted by data latency or computational resource constraints during actual deployment.

5.7. Comparison of Prediction Accuracy across Different Seasons

Considering the impact of seasonal factors, the data for each season in the dataset was processed and divided into training and testing sets at a 5:1 ratio. Different methods were used for load forecasting, with the prediction errors shown in Table 8.

Table 8 shows the load forecasting accuracy results of various methods for a specific day in each of the four seasons, with the error distribution shown in Figure 14. It can be observed from Table 8 that the proposed method achieves the lowest average prediction error compared to the other three methods. Specifically, the average MAE is improved by 63.2%, 47.97%, and 41%, respectively, compared to the other three methods, while the RMSE is reduced by 62.15%, 46.38%, and 38.19%, respectively. The significant improvements in both MAE and RMSE metrics indicate that the proposed short-term power load forecasting model greatly enhances overall prediction accuracy and model performance.

Table 8. Comparison of prediction accuracy across different seasons.

Model	Spring		Summer		Autumn		Winter	
	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW
LSTM	648.59	705.48	542.21	697.25	263.04	339.69	530.43	630.01
ELM	395.40	542.41	434.63	494.68	253.51	287.28	320.14	350.18
BiLSTM	337.50	379.72	442.66	490.69	206.13	258.42	251.24	323.73
TCN-BiLSTM	116.23	132.33	238.88	275.29	157.05	191.63	218.11	298.60

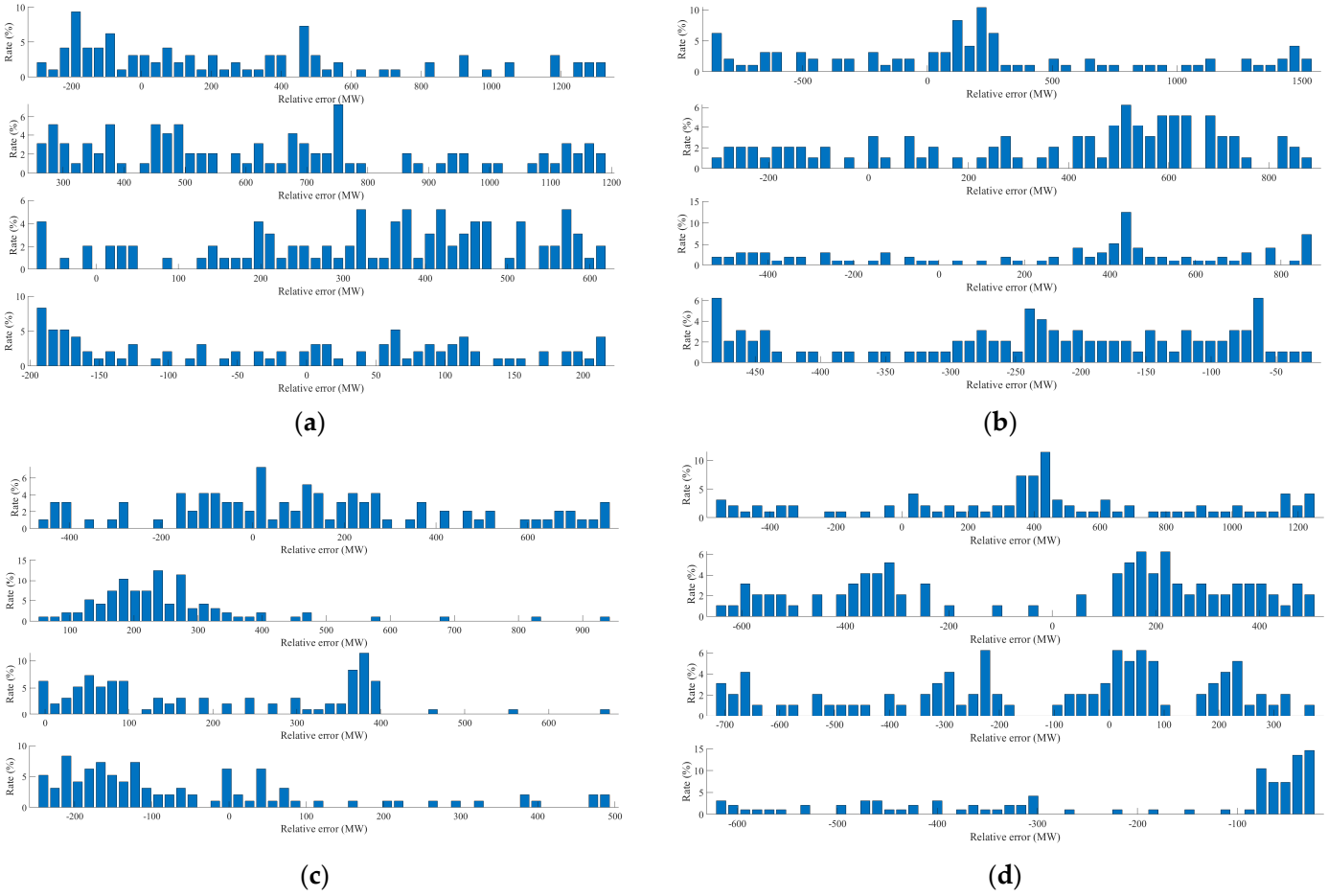


Figure 14. Error distribution of each model. (**a**) Spring; (**b**) Summer; (**c**) Autumn; (**d**) Winter.

By observing the distribution of prediction errors for different models, it can be seen that the proposed model primarily has relative errors within 200 MW during the spring and autumn seasons, whereas the errors for LSTM, ELM, and BiLSTM models are more concentrated in the range above 200 MW. In the summer and winter seasons, the prediction performance of all the models declines. Although the proposed model also exhibits high error distribution, the proportion of high errors is significantly lower compared to other models.

To evaluate the model's performance under extreme scenarios, this study considers the error during peak load periods. The time intervals from 7:00 to 9:00 and 13:00 to 15:00 on the selected four days were analyzed. The error results are presented in Table 9.

Table 9. Comparison of errors during peak periods across different seasons.

Time	Day 1		Day 2		Day 3		Day 4	
	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW	MAE/MW	RMSE/MW
7:00	180.73	180.74	150.78	151.63	213.33	214.14	192.30	194.06
8:00	189.85	189.87	97.47	98.32	141.43	142.24	240.52	241.13
9:00	194.61	194.61	65.57	65.73	94.04	95.18	279.53	286.29
13:00	107.58	111.64	210.90	211.34	227.03	231.48	131.15	155.61
14:00	191.06	191.71	250.29	250.42	224.36	235.00	276.54	280.13
15:00	213.35	213.38	271.97	272.00	173.65	175.66	405.35	412.23

Based on Table 9, it can be observed that during the peak period from 7:00 to 9:00 a.m., the MAE and RMSE values for all seasons are generally low, indicating high prediction accuracy of the model during this time. Among the seasons, the errors are smallest in summer and autumn, while spring and winter show relatively higher errors, with winter having significantly higher errors than the other seasons. In contrast, during the peak period from 1:00 to 3:00 p.m., the MAE and RMSE values for all seasons increase, with winter showing a significantly higher increase compared to the other seasons.

6. Discussion

This paper presents a novel short-term power load forecasting method that combines SBOA-optimized SVMD with TCN–BiLSTM, providing new insights into addressing the issue of weakened temporal continuity in current power load research. By deeply integrating temporal features into the prediction model, this method demonstrates significant performance improvements in short-term power load forecasting. From the perspective of power systems, accurate short-term load forecasting is crucial for the stable operation and optimal scheduling of the grid, enabling dispatch centers to allocate and manage power resources more effectively. Future research will explore the following directions:

- Incorporating other intelligent optimization algorithms to further refine the SVMD decomposition process.
- Integrating additional advanced deep learning models with TCN–BiLSTM to enhance the model's ability to capture complex temporal features.

Through ongoing research and innovation, we anticipate that this study will provide new ideas and methods for the field of power load forecasting, promoting its widespread application and development in practical settings.

7. Conclusions

This paper addresses the non-stationary and highly random characteristics of power load, aiming to improve the accuracy of short-term power load forecasting. Based on the decomposition–prediction–reconstruction framework, it deeply explores the temporal characteristics of power load sequences. By combining optimization algorithms, feature decomposition techniques, and deep learning models, a short-term power load forecasting method based on SBOA–SVMD–TCN–BiLSTM is proposed. The conclusions are as follows:

- The SBOA optimizes the key parameters of SVMD. The optimization fitness comparison experiment shows that SBOA can achieve the global optimum without falling into local optima. The decomposed IMF components can more accurately reflect the complex fluctuation characteristics of power load, effectively improving the accuracy and efficiency of decomposition.
- The decomposed IMF components are input into the TCN neural network according to the time window method, fully leveraging the TCN model's advantage in extracting potential data features, which enhances the accuracy and stability of power load forecasting.

- The feature vectors processed by TCN are used as inputs for the BiLSTM network model. By capturing the bidirectional dependencies through BiLSTM, the temporal and complex nonlinear relationships of power load data can be well analyzed.

The model comparison experiment results show that the MAE of the TCN–BiLSTM model is reduced by 47.8%, 32.8%, and 11.5% compared to the LSTM, ELM, and BiLSTM models, respectively. The RMSE is reduced by 42.9%, 39.2%, and 11.3%, and the R^2 value is significantly higher. Therefore, compared to traditional single prediction models, the proposed model effectively enhances the accuracy and fitting performance of load forecasting.

Author Contributions: Research, C.M.; conceptualization, G.F.; algorithm design, J.W.; simulation analysis, Y.C. and Y.L.; verification, J.W. and M.Y.; writing—original draft preparation, Y.C.; writing—review and editing, M.Y. and Y.C.; supervision, M.Y. All authors have read and agreed to the published version of the manuscript.

Funding: State Grid Jilin Electric Power Co., Ltd. Jilin Power Supply Company Project (Research on key technologies for power load forecasting based on a new generation of artificial intelligence technology, SGJLJL00XTJS2301427).

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: Author Guozhong Fang was employed by the company Northeast Branch of State Grid Corporation of China. Author Yunjing Liu was employed by the company Grid Jilin Electric Power Co., Ltd., Jilin Power Supply Company of State. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Kong, X.; Ma, Y.; Ai, Q. A review on modeling of electricity consumption characteristics and load forecasting for multi-user in new power system. *Autom. Electr. Power Syst.* **2023**, *47*, 2–17.
2. Zhu, J.; Dong, H.; Li, S. A review of data-driven load forecasting in integrated energy systems. *Proc. Chin. Soc. Electr. Eng.* **2021**, *41*, 7905–7924.
3. Yang, S.; Lv, J.; Guo, X. Modeling and analysis of multi-flexible resource optimal scheduling to adapt to the development of the new power system. *Power Syst. Technol.* **2024**, *48*, 1–14.
4. Hong, H.; Wu, C.; Ni, S. Optimal scheduling of the new power system considering frequency-inertia security constraints. *Electr. Power Autom. Equip.* **2024**, *44*, 1–12.
5. Kim, N.; Park, H.; Lee, J. Short-term electrical load forecasting with multidimensional feature extraction. *IEEE Trans. Smart Grid.* **2022**, *13*, 2999–3013. [CrossRef]
6. Li, D.; Qin, Z.; Lin, S. Short-term load forecasting for microgrid based on chaotic time series method. *J. Electr. Power Syst. Autom.* **2015**, *27*, 14–18.
7. Selvi, M.; Mishra, S. Investigation of performance of electric load power forecasting in multiple time horizons with new architecture realized in multivariate linear regression and feed-forward neural network techniques. *IEEE Trans. Ind. Appl.* **2020**, *56*, 5603–5612. [CrossRef]
8. Yang, G.; Zheng, H.; Zhang, H. Short-term load forecasting based on Holt-Winters exponential smoothing and temporal convolution network. *Autom. Electr. Power Syst.* **2022**, *46*, 73–82.
9. Jiang, H.; Zhang, Y.; Muljadi, E. A short-term and high-resolution distribution system load forecasting approach using support vector regression with hybrid parameters optimization. *IEEE Trans. Smart Grid* **2018**, *9*, 3341–3350. [CrossRef]
10. Chen, Y.; Xue, J.; Qiu, J. Research on short-term load forecasting based on improved random forest algorithm. *Mod. Ind. Econ. Informatiz.* **2024**, *14*, 218–225.
11. Liu, L.; Guo, K.; Chen, J. A photovoltaic power prediction approach based on data decomposition and stacked deep learning model. *Electronics* **2023**, *12*, 2764. [CrossRef]
12. Li, L.; Wei, J.; Li, C. Load model prediction based on artificial neural network. *Trans. China Electrotech. Soc.* **2015**, *30*, 225–230.
13. Luo, F.; Zhang, X.; Yang, X. Load analysis and prediction of integrated energy distribution system based on deep learning. *High Volt. Eng.* **2021**, *47*, 23–32.
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
15. Zhang, M.; Yu, Z.; Xu, Z. Short-term load forecasting using recurrent neural networks with input attention mechanism and hidden connection mechanism. *IEEE Access* **2020**, *8*, 186514–186529. [CrossRef]
16. Lu, B.; Huo, Z.; Yu, M. Short-term prediction of multi-load integrated energy systems based on LSTNet-Skip. *Proc. CSEE* **2023**, *43*, 2273–2282.

17. Wang, S.; Zhang, Z. Short-term prediction of multi-load integrated energy systems based on quantum-weighted multi-level GRU neural network. *Power Syst. Prot. Control* **2022**, *50*, 85–93.
18. Yang, L.; Wu, H.; Ding, M. Short-term load forecasting of new energy grid considering feature selection based on Bi-LSTM network. *Autom. Electr. Power Syst.* **2021**, *45*, 166–173.
19. Zhu, Q.; Zeng, S.; Chen, M. Short-term load forecasting method based on bidirectional long short-term memory model with stochastic weight averaging algorithm. *Electronics* **2024**, *13*, 3098. [CrossRef]
20. Zhang, T.; Huang, Y.; Liao, H. A hybrid electric vehicle load classification and forecasting approach based on GBDT algorithm and temporal convolutional network. *Appl. Energy* **2023**, *351*, 121768.
21. Kong, X.; Li, C.; Zheng, F. Short-term load forecasting method based on empirical mode decomposition and feature correlation analysis. *Autom. Electr. Power Syst.* **2019**, *43*, 46–52.
22. Wang, Y.; Jiang, B.; Han, D. Multichannel short-term power load forecasting model based on variational mode decomposition (VMD) data decomposition. *Fluid Meas. Control* **2024**, *5*, 25–31.
23. Wang, L.; Zhou, X.; Tian, T. Short-term power load combination forecasting model based on spatiotemporal fusion of multidimensional meteorological information and MPA-VMD. *Electr. Power Autom. Equip.* **2024**, *44*, 190–197.
24. Lu, J.; Zhang, Q.; Yang, Z. Short-term load forecasting method based on CNN-LSTM hybrid neural network model. *Autom. Electr. Power Syst.* **2019**, *43*, 131–137.
25. Zhu, L.; Xun, Z.; Wang, Y. Short-term power load forecasting based on CNN-BiLSTM. *Power Syst. Technol.* **2021**, *45*, 4532–4539.
26. Estebsari, A.; Rajabi, R. Single residential load forecasting using deep learning and image encoding techniques. *Electronics* **2020**, *9*, 68. [CrossRef]
27. Laitsos, V.; Vontzos, G.; Tsiovoulos, A. Enhanced sequence-to-sequence deep transfer learning for day-ahead electricity load forecasting. *Electronics* **2024**, *13*, 1996. [CrossRef]
28. Duan, Q.; Xue, G.; Tan, Q. Improved BWO-TimeNet short-term heat load forecasting model based on SVMD. *J. Guangxi Norm. Univ. (Nat. Sci. Ed.)* **2024**, 1–17. [CrossRef]
29. Fu, Y.; Liu, D.; Chen, J. Secretary bird optimization algorithm: A new metaheuristic for solving global optimization problems. *Artif. Intell. Rev.* **2024**, *57*, 123. [CrossRef]
30. Li, Y.; Chen, X.; Yu, J. A fusion frequency feature extraction method for underwater acoustic signal based on variational mode decomposition, duffing chaotic oscillator and a kind of permutation entropy. *Electronics* **2019**, *8*, 61. [CrossRef]
31. Liu, X.; Pu, X.; Li, J. Short-term wind power forecasting based on Bayesian optimization of VMD-GRU. *Power Syst. Prot. Control* **2023**, *51*, 158–165.

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.