EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

Similar-Scenario Retrieval and Intelligent Hyperparameter Optimization for Day-Ahead Electricity Price Forecasting

Qingzhong Wang[1], Xi Fu[2*], Weiguo Xiao[1], Chuankai He[3], Xiaogang Xie[1] and Jidong Li[4]

[1]Datang Gansu Power Generation Co., Ltd. Bikou Hydroelectric Power Plant, Longnan, Gansu, 746412, China
[2]College of Water Resources and Hydropower, Sichuan Agricultural University, Yaan, Sichuan, 625014, China
[3]Datang Hydropower Science and Technology Research Institute Co., Ltd., Chengdu, Sichuan, 610036, China
[4]College of Water Resources and Hydropower, Sichuan University, Chengdu, Sichuan, 610065, China

Abstract

INTRODUCTION: Day-ahead electricity prices exhibit pronounced volatility, nonlinearity, and temporal dependence. Training models on all available historical observations may introduce irrelevant information and reduce forecasting reliability.
OBJECTIVES: This study proposes a hybrid forecasting framework that integrates same-period similar-scenario retrieval with intelligent hyperparameter optimization to improve the forecasting performance of day-ahead market-clearing prices. By refining the composition of the training dataset and optimizing the network configuration, the proposed framework enhances the model's ability to learn the underlying dynamics of historical electricity prices.
METHODS: First, similarity measures are used to evaluate historical electricity price sequences from corresponding periods. Historical observations that are highly relevant to the target forecasting period are then selected to construct a tailored training dataset. Subsequently, a hybrid CNN–BiLSTM–AM forecasting model is developed. Within this architecture, the convolutional neural network (CNN) extracts local patterns and salient features from the input data, while the bidirectional long short-term memory network (BiLSTM) captures long-term temporal dependencies in both forward and backward directions. The attention mechanism (AM) adaptively assigns weights to the learned representations, thereby emphasizing features that contribute more substantially to the final prediction. The dung beetle optimization (DBO) algorithm is further employed to optimize the key hyperparameters of the model. Finally, the effectiveness of the proposed method is evaluated using day-ahead market-clearing price data from the Miaojiaba Hydropower Station in the Bailong River Basin, Gansu Province, China.
RESULTS: The empirical results demonstrate that the similarity-scenario-based sample reconstruction strategy effectively enhances the model's ability to capture historical electricity price patterns. Taking a representative market price of 100 CNY/MWh as the reference, the proposed method reduces the standard deviation of forecasting errors by approximately 26 CNY/MWh compared with a conventional BiLSTM model without historical similar-sequence selection, thereby helping to mitigate potential revenue losses caused by electricity price forecasting deviations.
CONCLUSION: The proposed framework effectively combines targeted sample selection, spatiotemporal feature learning, attention-based weighting, and hyperparameter optimization. It provides a reliable tool for day-ahead bidding decisions and supports efficient electricity resource allocation.

Keywords: Day-ahead market-clearing price, similar-scenario retrieval, dung beetle optimization, CNN, BiLSTM, attention mechanism

Received on 18 August 2026, accepted on 02 November 2026, published on 05 November 2026

Copyright © 2026 Qingzhong Wang *et al.*, licensed to EAI. This is an open access article distributed under the terms of the CC BY-NC-SA 4.0, which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/ew.14570

*Corresponding author. Email: author@emailaddress.com

1. Introduction

In light of China's continuous advancement in energy structure transformation and the intensified implementation of its dual-carbon targets, the promotion of electricity market reform has gained significant momentum under the guidance of the National Development and Reform Commission and the National Energy Administration [1, 2]. Against this backdrop, the national power market has gradually shifted from a centralized purchasing model to a liberalized trading mechanism. The clearing price is not only the direct basis for transaction settlement in the electricity spot market, but also a crucial foundation for studies on price formation mechanisms, market power analysis, and bidding strategy optimization for generators [3]. Accurate clearing price forecasting can help market participants (such as generation companies and electricity retailers) formulate more informed trading strategies, mitigate price-volatility risks, and optimize profits. At the same time, it provides data support for market regulatory authorities in evaluating market operation efficiency and preventing market manipulation behaviours [4].

At present, methods for forecasting clearing prices in the spot market, both domestically and internationally, can generally be classified into two categories: Methods for time-series forecasting and those driven by artificial intelligence. Time-series forecasting methods include linear regression models, exponential smoothing methods, and others. These approaches mainly rely on the historical trend of electricity prices and construct corresponding mathematical models using statistical or economic theories, thus placing high demands on the quality of historical data. Artificial intelligence–driven forecasting methods include Random Forest (RF), Support Vector Regression (SVR) [5], Convolutional Neural Networks (CNN) [6], and others. Such methods excel at modelling nonlinear and complex problems and compensating for the limitations of statistical approaches. However, they require large datasets for iterative learning and the identification of complex nonlinear relationships [7]. Each of these forecasting methods has its own advantages, but all face challenges in further improving prediction accuracy.

Currently, improvements in clearing price forecasting, both domestically and internationally, mainly follow two approaches: (i) Processing of raw data. Dong et al. [8] employed Spearman correlation analysis to select relevant electricity price drivers and constructed a similar-day dataset using weighted grey relational analysis. They subsequently developed a regularized extreme learning machine optimized by the marine predators algorithm for price forecasting. Nevertheless, owing to its shallow feedforward architecture and lack of an explicit memory mechanism, this method has limited capability to capture long-range temporal dependencies and recurring periodic patterns in historical electricity prices. Kosana V et al. [9] performed price forecasting using continuous sequences, but the heterogeneous variation characteristics made modelling and analysis more difficult. (ii) Improving forecasting models. Vlah D et al. [10] used a CNN-based surrogate model to approximate the lower-level problem in a bilevel bidding framework, avoiding repeated solution of the complex lower-level problem. This shows the potential of CNN-based representation learning and the need for careful architecture and hyperparameter selection. Liu and Mao [11] integrated CNN, BiLSTM, and attention for short-term photovoltaic power forecasting, where CNN extracts local features and BiLSTM captures bidirectional temporal dependencies. Their ablation results confirm the importance of local-feature extraction. However, existing studies rarely integrate similar-sequence retrieval, local and temporal feature extraction, and metaheuristic hyperparameter optimization within a unified day-ahead electricity-price forecasting framework.

Metaheuristic optimization has developed from early evolutionary methods, such as genetic algorithms and differential evolution, to population-based swarm-intelligence methods, including particle swarm optimization and ant colony optimization. Recent studies have further expanded this field with behaviour-inspired algorithms that use diversified search strategies to balance global exploration and local exploitation [12]. For deep learning forecasting models, hyperparameter selection is generally a derivative-free and computationally expensive task because the relationship between model configuration and forecasting accuracy is nonlinear and each candidate configuration requires repeated training and validation. Therefore, population-based metaheuristic algorithms provide a practical alternative to exhaustive grid search for hyperparameter optimization. Their effectiveness in electricity price forecasting and hybrid deep-learning frameworks has also been demonstrated in recent studies [13, 14].

In view of these limitations, and to overcome the shortcomings of traditional electricity price forecasting models in capturing complex temporal features, this paper uses historical similar sequences to construct the dataset and develops an integrated DBO-CNN-BiLSTM-AM forecasting model. Through intelligent optimization, the method collaboratively extracts spatiotemporal features, and the Dung Beetle Optimization (DBO) algorithm is applied for hyperparameter tuning to achieve optimal adaptation among the model components. Case studies demonstrate that the proposed forecasting method offers high accuracy and strong reliability. The novelty of this study lies in the task-oriented integration and empirical validation of historical similar-sequence retrieval and DBO-driven hyperparameter optimization within a CNN-BiLSTM-AM forecasting framework. Rather than proposing a completely new individual algorithm, the proposed framework combines similarity-based sample selection, local feature extraction, attention-based feature weighting, bidirectional temporal learning, and adaptive hyperparameter optimization to address the day-ahead electricity price forecasting task.

2. Historical Similar Sequences

Electricity price fluctuations vary considerably across different historical periods. Using all historical observations to train the forecasting model may introduce redundant or irrelevant samples. Similarity-day screening has been used to select historically comparable observations and reduce redundant information before electricity-price forecasting [15]. Accordingly, before constructing the training dataset, this study retrieves historical sequences that are most similar to the recent electricity price state. The procedure consists of three steps. First, the electricity prices during the three days preceding the forecasting day are used as the recent-state query sequence, while historical three-day sequences from the corresponding time periods constitute the candidate set. Second, the candidate sequences are compared with the recent-state query sequence from two complementary perspectives: Euclidean distance measures differences in electricity price levels, whereas angular distance measures differences in fluctuation trends. Finally, the two distance measures are combined into a composite similarity score. The candidate sequences are ranked in ascending order of this score, and the most similar sequences are retained to construct the training set. Before similarity evaluation, the original electricity prices are globally Min-Max normalized and linearly mapped to the interval [0, 1]. The detailed mathematical definitions are provided in Sections 2.1–2.3.

2.1. Euclidean Distance

Euclidean distance measures the overall difference in price levels between two electricity price sequences of equal length. A smaller Euclidean distance indicates that the two sequences have more similar absolute price values. Let $A = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \dots\dots, \mathbf{a}_L]$ and $B = [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots\dots, \mathbf{b}_L]$ denote two normalized electricity price sequences observed over the same L time periods.

$$E(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{t=1}^{L} (\mathbf{a_t}\text{-}\mathbf{b_t})^2} \tag{1}$$

Where $E(A, B)$ denotes the Euclidean distance between sequences A and B; L is the number of time periods in each sequence; and a_t and b_t are the normalized electricity prices of sequences A and B, respectively, at time period t..

2.2. Angular Distance

Angular distance measures the difference in fluctuation trends between two electricity price sequences. Unlike Euclidean distance, it focuses on the directions of successive price changes rather than the absolute price levels. Therefore, two sequences with different price levels but similar upward and downward movements will have a small angular distance. For sequences A and B, the direction vectors between two adjacent observations at position t are defined as:

$$C(\mathbf{A}, \mathbf{B}) = \frac{1}{\pi(L-1)} \sum_{t=2}^{L} \theta_t \tag{2}$$

$$\theta_t = arccos\left(\frac{\mathbf{c}_t^A \cdot \mathbf{c}_t^B}{\|\mathbf{c}_t^A\| \cdot \|\mathbf{c}_t^B\|}\right) \tag{3}$$

$$\mathbf{c}_t^A = (1, \mathbf{a_t}\text{-}\mathbf{a_{t-1}}), \mathbf{c}_t^B = (1, b_t\text{-}b_{t-1}) \tag{4}$$

Where $C(A, B)$ represents the angular distance between time series A and B; $\mathbf{c}_t^A$ and $\mathbf{c}_t^B$ are the direction vectors of the two sequences between time periods t-1 and t; θ_t is the angle between the two direction vectors. L is the sequence length; and t=2,……,L. The factor π normalizes the angular distance to the interval [0, 1]. A smaller value of $C(A, B)$ indicates more similar fluctuation trends.

2.3. Similarity Measure

In this paper, a similarity metric denoted as S is introduced. A smaller value of S corresponds to a higher degree of similarity between time series A and B. The metric is computed using the following formula.

$$S(\mathbf{A}, \mathbf{B}) = \alpha E(\mathbf{A}, \mathbf{B}) + \ (1 - \alpha)\ C(\mathbf{A}, \mathbf{B}) \tag{5}$$

Where $S(A, B)$ is the similarity metric; $\alpha \in [0, 1]$ is a weighting parameter that can be adjusted according to practical needs; in this paper, α=0.5.

In this paper, the historical electricity prices of the same time periods for the three days adjacent to the forecasting period are taken as the “most recent state.” Using this recent state as the query window, the similarity metric is computed between the recent state and historical electricity price sequences of equal length and corresponding time periods. Historical sequences that meet the similarity threshold are then selected as similar sequences.

3. Model Construction

Building on the similar-sequence retrieval procedure described in Section 2, the selected historical similar sequences are used to construct the training dataset for the forecasting model. Historical similar states are used as the training inputs, while the most recent state is designated as the training output to construct the training dataset. During the testing stage, the most recent state sequence is used as the test input, and the model outputs the electricity price for the forecasting day. The proposed CNN-BiLSTM-AM model processes the input in four stages. First, two 1×1 convolutional layers with 32 and 64 filters, respectively, extract local feature representations. Second, the attention module generates channel-wise weights and applies them to the convolutional features. Third, the BiLSTM layer learns bidirectional dependency representations from the attention-weighted features. Finally, a fully connected

regression layer maps the learned representation to the predicted electricity price. DBO is used only to optimize the learning-rate drop factor, the initial learning rate, and the number of BiLSTM hidden units. The following subsections describe each network component in detail.

3.1. Convolutional Neural Networks

CNNs are effective at extracting local feature representations through shared convolutional kernels. As illustrated in Figure 1, a CNN progressively transforms the input into higher-level feature maps through convolution and nonlinear activation. In the proposed forecasting model, CNN serves as the initial feature-extraction component, extracting local features from the input values before attention weighting and BiLSTM encoding.

In the proposed model, CNN is used as the feature-extraction backbone, while the pooling operation is incorporated into the attention module for channel-weight generation, as described in Section 3.3.

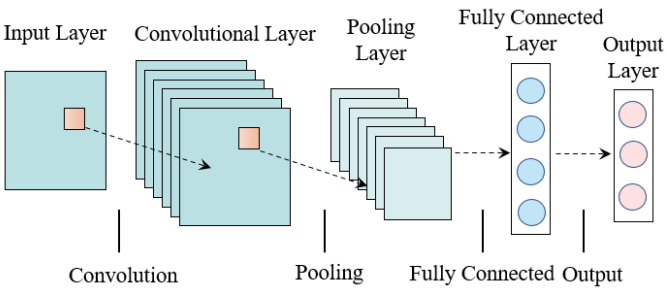


Figure 1. CNN network architecture

In the proposed model, CNN serves as the feature-extraction backbone. Specifically, each input sample is reshaped into a [3, 1, 1] tensor to match the input format required by the two-dimensional convolutional layers, where 3 denotes the number of input electricity price values, and the remaining two dimensions represent the height and width of the feature map, respectively. The CNN contains two 1×1 convolutional layers with 32 and 64 filters, respectively, and each convolutional operation is followed by ReLU activation. These layers extract and transform local feature representations from the three input price values, and the resulting 64-channel convolutional features are subsequently passed to the attention module for channel-wise weighting. In a conventional CNN, convolutional layers extract local feature maps, pooling layers aggregate feature information, and fully connected layers map the learned representations to the output space; in the proposed model, however, the convolutional layers serve as the feature extractor, while global average pooling and fully connected mapping are incorporated into the attention module to generate channel-wise weights. Equations (6)–(8) present the standard mathematical operations of convolution, global average pooling, and fully connected mapping, respectively, and the mathematical representation of the convolutional operation is given below.

$$\mathbf{e}_j^l = \sigma\left(\sum_{i\in D_j} \mathbf{e}_j^{l-1} \times \mathbf{M}_{\mathbf{i,j}}^{\mathbf{l}} + \mathbf{b}_{\mathbf{j}}^{\mathbf{l}}\right) \tag{6}$$

$$f_j^l = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} X_{i,\ j,\ c} \tag{7}$$

$$\mathbf{e}_j^{l+1} = \sum_{i=1}^{n} \mathbf{M}_{i,j}^{l}\, e_j^l + \mathbf{b}_j^l \tag{8}$$

Where $\mathbf{e}_j^l$ denotes the input to the l-th layer from the j-th convolution kernel (which corresponds to the output of the l-1-th layer); $\sigma$is the activation function; D_j is the set of input mapping layers; $\mathbf{M}_{i,j}^l$ is the weight matrix of the convolution kernel; $\mathbf{b}_{\mathbf{j}}^{\mathbf{l}}$ stands for the bias term; $X_{i,j,c}$ indicates the value of the input feature map at spatial position (i, j) and channel c; f_j^l is the scalar output value for channel c, representing the average over all spatial positions; $\mathbf{e}_j^{l+1}$ is defined as the j-th output value of the (l+1)-th layer.

3.2. Bidirectional Long Short-Term Memory Network

After attention weighting, the convolutional features are fed into the BiLSTM layer for bidirectional dependency learning. A BiLSTM consists of a forward LSTM and a backward LSTM, which process the input feature sequence in opposite directions. By combining information from both directions, BiLSTM provides a more comprehensive representation for electricity price forecasting. The resulting bidirectional feature representation is subsequently mapped to the predicted electricity price through a fully connected regression layer.

$$\begin{cases} \mathbf{f}_t = \sigma(\mathbf{W_f}[h_{t-1}, x_t] + \mathbf{b_f}) \\ \mathbf{i_t} = \sigma(\mathbf{W}_g[h_{t-1}, x_t] + \mathbf{b}_g) \\ \tilde{C}_t = tanh(\mathbf{W}_c[h_{t-1}, x_t] + \mathbf{b_c}) \\ C_t = f_c c_{t-1} + \mathbf{i_t}\tilde{C}_t \\ \mathbf{o_t} = \sigma(\mathbf{W}_o[h_{t-1}, x_t] + \mathbf{b_o}) \\ h_t = o_t\, tanh(C_t) \end{cases} \tag{9}$$

Where f_t, i_t and o_t stand for the forget, input, and output gates, in that order; C_t and $\tilde{C}_t$ are the current cell state and the candidate cell state, respectively; the sigmoid and hyperbolic tangent activation functions are represented by σ and tanh; the weight matrices W_f, W_g, $W_c$and $W_o$correspond to their respective gate units; the bias vectors b_f, b_g, b_c and $b_o$represent the bias terms for each gate.

By processing sequences bidirectionally, BiLSTM effectively captures temporal features. This bidirectional approach supports a more complete extraction of data-inherent relationships, which contributes to higher prediction accuracy [16]. The BiLSTM architecture is

depicted in Figure 2. The number of BiLSTM hidden units is treated as a hyperparameter and is optimized by DBO. The BiLSTM output is connected to a fully connected regression layer to produce the final electricity price prediction.

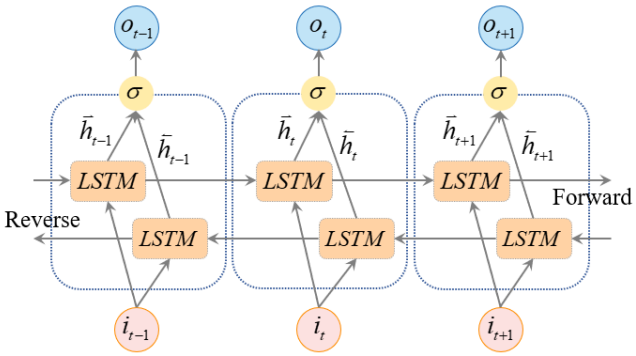


Figure 2. BiLSTM network architecture

3.3. Attention Mechanism

AM is a feature-reweighting mechanism that enables a neural network to assign different importance weights to different features. Rather than treating all convolutional channels equally, AM allows the model to emphasize informative features and suppress less relevant ones.

In the proposed model, AM is introduced between CNN feature extraction and BiLSTM encoding. Unlike the conventional CNN structure illustrated in Figure 1, global average pooling is not used as a direct feature-compression layer. Instead, it is used to generate channel-wise statistics for attention-weight calculation. Specifically, the output of the first convolutional layer is processed by global average pooling to obtain a 32-dimensional channel descriptor. The descriptor is then passed through a bottleneck multilayer perceptron with 16 hidden units and 64 output units. After sigmoid activation, a 64-dimensional channel-weight vector is generated. The weight vector is multiplied element-wise with the 64-channel output of the second convolutional layer, thereby emphasizing informative channels and suppressing less relevant channels. The resulting attention-weighted features are subsequently passed to the BiLSTM layer.

$$w = \sigma(\mathbf{W}_2\delta(\mathbf{W}_1 z + \mathbf{e}_1) + \mathbf{e}_2) \tag{10}$$

Where W and W are learnable weight matrices, W having dimensions 16×32 and W having dimensions 64×16; δ denotes the ReLU activation function; σ denotes the sigmoid activation function; z is the 32-dimensional channel descriptor; w is the resulting 64-dimensional channel-weight vector; e_1 and e_2 are the bias vectors of the multilayer perceptron. The channel-weight vector w is multiplied element-wise with the 64-channel output of the second convolutional layer before BiLSTM encoding.

3.4. Dung Beetle Optimization

Hyperparameter optimization for deep learning forecasting models is a derivative-free problem because the relationship between model configuration and forecasting performance is nonlinear and difficult to formulate explicitly. As discussed in Section 1, population-based metaheuristic algorithms provide an effective search strategy for such problems by balancing global exploration and local exploitation [12]. DBO, originally proposed by Xue and Shen [17], is a population-based metaheuristic algorithm that simulates four complementary behaviours of dung beetles: ball rolling, breeding, foraging, and stealing. These diversified update strategies enable the algorithm to search the solution space from different perspectives. In this study, DBO is employed to optimize three hyperparameters of the CNN-BiLSTM-AM model: the learning-rate drop factor, the initial learning rate, and the number of BiLSTM hidden units. The four DBO update behaviours are summarized as follows.

(i) Ball-rolling behavior. This behavior simulates the dung beetle pushing a dung ball to perform global exploration.

$$x_k^{h+1} = x_k^h + \beta \times l \times x_k^{h-1} + b \times \triangle x \tag{11}$$

$$\triangle x = |x_k^h - x_{best}^h| \tag{12}$$

Where x_k^{h+1} denotes the position of the k-th dung beetle during the h-th iteration; l is the deflection coefficient employed to simulate uneven ground conditions; b is a fixed coefficient, $\in(0,1)$; $\triangle x$ describes the difference between the current position and the optimal one; x_{best}^h denotes the global optimal position in the h-th iteration, obtained by traversing all fitness values.

(ii) Breeding behaviour. This behaviour simulates dung beetles producing dung balls and laying eggs.

$$b_k^h = x^* + d_1 \times (x_k^h - L_b) + d_2 \times (x_k^h - U_b) \tag{13}$$

Where b_k^h is the position of the breeding ball produced by the k-th dung beetle; x^* is the current global optimal position; d_1, d_2 are random weight coefficients, $\in (0,1)$; L_b, U_b specify the minimum and maximum bounds of the breeding region, respectively.

(iii) Foraging behaviour and stealing behaviour. The position update formulas are as follows:

$$x_k^{h+1} = x^t + C_1 \times (x_k^h - Lb^h) + C_2 \times (x_k^h - Ub^h) \tag{14}$$

$$x_k^{h+1} = x^* + f \times g \times (|x_k^h - x^*| + |x_k^h - x^{**}|) \tag{15}$$

Where C_1, C_2 are adaptive step-size coefficients; Lb^h, Ub^h are the dynamic search boundaries in the k-th iteration; x^{**} is the current suboptimal solution; f is the stealing-

intensity coefficient, a constant typically ranging from 0.5~1.0; g is the Gaussian disturbance term.

In this study, the DBO population size is set to pop=25, the maximum number of iterations is set to Max_iter = 20 and the optimization dimension is dim=3. The three optimized variables are the learning-rate drop factor, the initial learning rate, and the number of BiLSTM hidden units. Their search ranges are $[10^{-10}, 10^{-2}]$, [0.001, 0.02], and [1, 10], respectively. The population is divided into ball-rolling, breeding, foraging, and stealing subgroups in proportions of 20%, 30%, 30%, and 20%, respectively. The fixed coefficient b and deflection coefficient l are both set to 0.1. For each candidate hyperparameter configuration, the RMSE produced by the CNN-BiLSTM-AM model is used as the fitness value. The configuration with the minimum RMSE is selected as the optimal solution.

3.5. Predictive Model

The complete forecasting workflow of the proposed model is illustrated in Figure 3 and consists of the following five steps.

Step 1: Similar-sequence retrieval. The electricity prices from the three days preceding the forecasting day are used as the recent state. Historical three-day sequences at the corresponding intraday time period are ranked according to the composite distance $S(A, B)$, and the lowest 3% are retained as historical similar sequences.

Step 2: Dataset construction. The retained historical similar states are used as the training inputs, while the most recent state is designated as the training output, thereby forming the training dataset. The most recent state sequence is subsequently used as the test input to predict the electricity price on the forecasting day.

Step 3: DBO hyperparameter optimization. DBO searches for the learning-rate drop factor, the initial learning rate, and the number of BiLSTM hidden units. The optimal hyperparameter combination is then selected.

Step 4: Model training. The CNN-BiLSTM-AM model is trained using the training dataset and the optimal hyperparameter combination obtained by DBO.

Step 5: Day-ahead prediction. The trained model takes the most recent state sequence as input and outputs the predicted electricity price for the forecasting day.

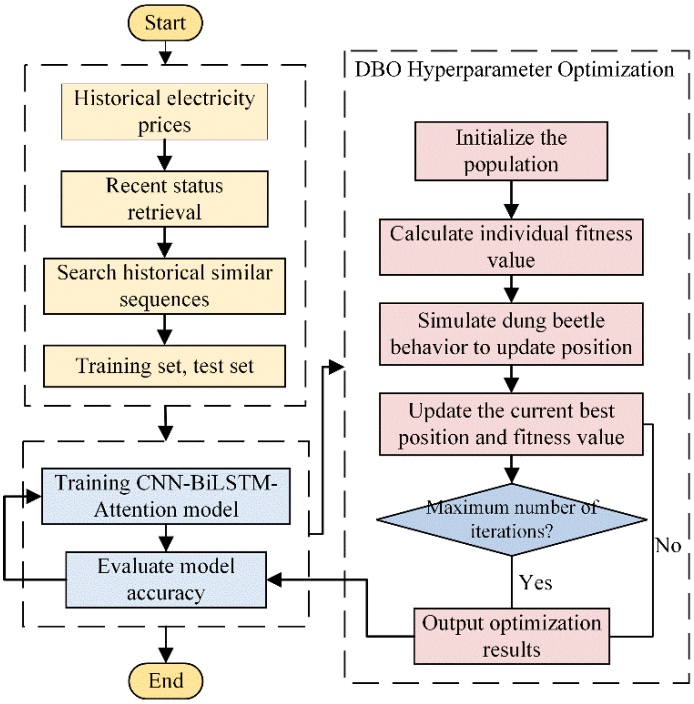


Figure 3. Ensemble model prediction flow

In the construction of the combined electricity price forecasting model, CNN-BiLSTM is responsible for high-dimensional feature extraction and temporal feature mining. AM applies weighting to the features generated by the convolutional layers, which further amplifies the influence of essential features on the final prediction outcomes. The complete structure of the ensemble model is depicted in Figure 4.

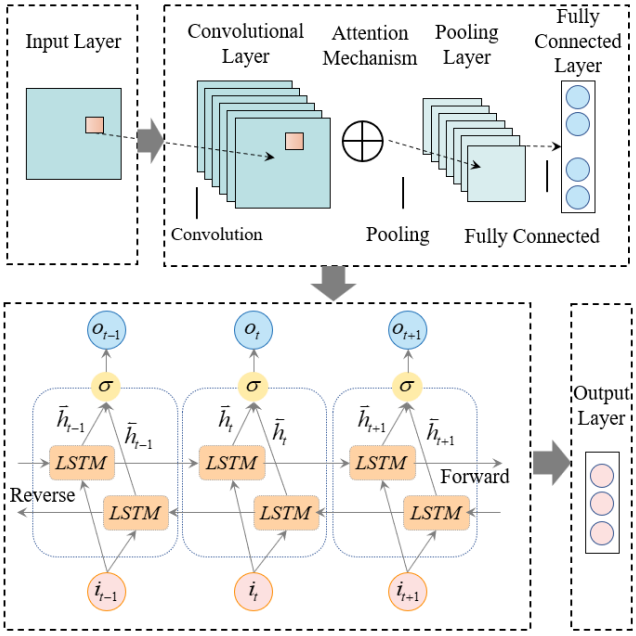


Figure 4. The combined structure of CNN-BiLSTM-AM

4. Case Analysis

4.1. Data Source

This study uses day-ahead market-clearing price data for the Miaojiaba Hydropower Station in the Bailong River Basin, Gansu Province, China. The data were exported directly from the internal information system of Datang Bikou Hydropower Plant. The dataset covers the period from January 1 to December 5, 2024, at a 15-minute temporal resolution, with electricity prices expressed in CNY/MWh.

4.2. Prediction Process

Taking the prediction of the day-ahead clearing price for the first time period on December 5, 2024, as an example, the day-ahead clearing prices of the same time period from December 2 to December 4 are selected as the most recent state. The similarity indices between the recent state and historical sequences are calculated, and the historical sequences that meet the similarity threshold are selected.

To determine the appropriate retention proportion of historical similar sequences, a sensitivity analysis was conducted for k=1%, 2%, 3%, 4%, 5%, 8% and 10%. The CNN-BiLSTM-AM model was used with fixed hyperparameters: four BiLSTM hidden units, an initial learning rate of 0.001, and a learning-rate drop factor of 0.003. The experiments were repeated over 10 random seeds, and RMSE and R^2 were evaluated as defined in Section 4.3. The sensitivity results for different retention proportions are summarized in Table 1 and Figure 5.

Table 1. Forecasting performance under different retention proportions k

k	Training samples	Mean RMSE	Mean R^2
1%	96	46.83	0.7128
2%	192	35.84	0.8431
3%	288	31.66	0.8782
4%	384	32.44	0.8722
5%	480	33.96	0.8598
8%	768	34.95	0.8510
10%	1056	34.84	0.8514

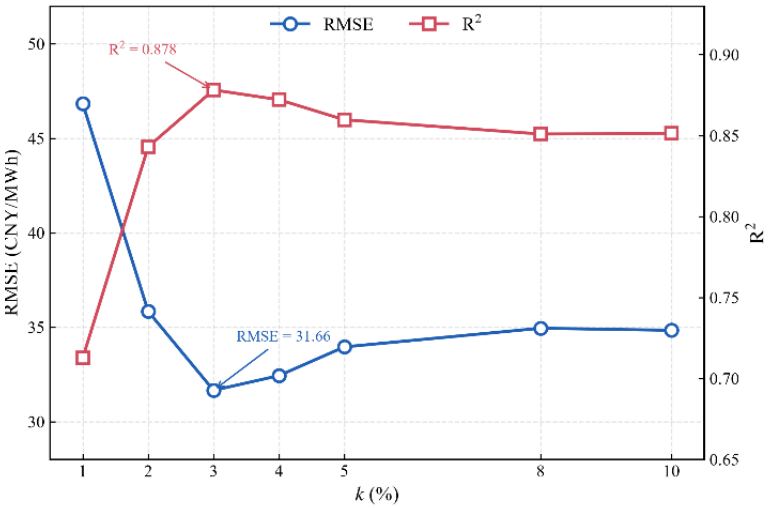


Figure 5. Comparison of forecasting performance under different retention proportions k

Table 1 and Figure 5 show that forecasting performance improves as k increases from 1% to 3%, and then declines slightly. The 3% setting yields the lowest mean RMSE (31.66 CNY/MWh) and the highest mean R^2 (0.8782). Thus, 3% was selected for the present dataset, balancing the number and similarity of retained historical sequences. The recent state and its corresponding similar states under this setting are shown in Table 2.

Table 2. Recent status and related similar status

Time series	Date	Electricity price (CNY/MWh)
	12.02	480
recent state	12.03	480
	12.04	390
	1.16	450
similar state1	1.17	480
	1.18	450
	2.03	400
similar state2	2.04	450
	2.05	400
	2.27	450
similar state3	2.28	410
	2.29	390

Upon identification of analogous historical states, these states serve as input variables, with the most recent state designated as the output target, thereby forming the training dataset. Subsequently, the DBO algorithm is utilized to fine-tune three critical hyperparameters within the integrated CNN-BiLSTM-AM model. The parameters adjusted encompass the learning rate decay factor, the initial learning rate, and the quantity of neurons in the BiLSTM layer. Upon completion of the DBO optimization process, the optimized parameters are incorporated into the CNN-BiLSTM-AM model for training. After model training is completed, the most recent state is used as the input to predict the day-ahead clearing price for the corresponding time period on December 5.

4.3. Evaluation Criteria

This study selects three core metrics to evaluate model performance: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the coefficient of determination (R^2). Among these, MAE directly reflects the average absolute deviation between predicted and actual values, while RMSE assigns greater weight to larger errors through squaring. Both are error-based indicators, and lower values indicate smaller prediction errors and higher model accuracy. R^2 measures the extent to which the model explains the variability of the target variable, with values ranging from 0 to 1. A value closer to 1 signifies better model fit and stronger explanatory power. In summary, reductions in MAE and RMSE, along with an R^2 value approaching 1, collectively indicate improved predictive accuracy of the model. The specific mathematical expressions for each metric are provided below:

$$\mathrm{MAE} = \frac{1}{N}\sum_{n=1}^{N}\left|y_n - \hat{y}_n\right| \tag{16}$$

$$\mathrm{RMSE} = \frac{1}{N}\sqrt{\sum_{n=1}^{N}\left(y_n - \hat{y}_n\right)^2} \tag{17}$$

$$R^2 = 1 - \frac{\sum_{n=1}^{N}\left(y_n - \hat{y}_n\right)^2}{\sum_{n=1}^{N}\left(y_n - \frac{1}{N}\sum_{n=1}^{N} y_n\right)^2} \tag{18}$$

Where N denotes the number of samples; y_n represents the actual electricity price; $\hat{y}_n$ is the predicted electricity price.

4.4. Model Validation and Assessment

4.4.1 Validation of similar-sequence retrieval and model performance

To evaluate the proposed framework from two complementary perspectives, comparative experiments were conducted to assess: (i) the contribution of historical similar-sequence retrieval to forecasting accuracy; and (ii) the performance of the proposed model relative to structural ablation models and contemporary benchmark models. The first experiment compares matched model structures trained with and without historical similar-sequence retrieval. The second experiment compares the proposed DBO-CNN-BiLSTM-AM model with CNN, CNN-AM, CNN-BiLSTM, CNN-BiLSTM-AM, BiLSTM-AM, BiLSTM, XGBoost, and Transformer. All models in the second experiment used the same 3% similar-sequence dataset, the same three lagged electricity-price inputs, and the same chronological test set.

The parameter settings of the proposed model and benchmark models are reported in Table 3. For the structural ablation experiment, the compared deep-learning models were trained using the same 3% similar-sequence dataset, the same three lagged electricity-price inputs, and the same chronological test set. The baseline hyperparameters of the ablation models were kept fixed, whereas DBO was used only in the proposed framework to optimize the learning-rate drop factor, the initial learning rate, and the number of BiLSTM hidden units. Transformer and XGBoost were evaluated using the same input variables and test partition. This design reduces the influence of inconsistent data partitioning and allows the observed performance differences to be attributed primarily to sample selection, model structure, and hyperparameter optimization.

Table 3. Hyperparameter settings for the proposed model and benchmark models

Network Layer	Parameter Settings
DBO	Population size: pop=25; Optimization variable dimension: dim=3; Maximum iterations: Max_iter=20; Subgroup proportions (rolling/breeding/foraging/stealing): 20%/30%/30%/20%; Fixed coefficient: b=0.1; Deflection coefficient: l=0.1;
CNN	Convolution kernel size: Kernel size=[1, 1]; Activation function: activation=ReLU; Number of filters: filters=64.
BiLSTM	Number of BiLSTM layers: 1; Bidirectional: return sequences=False; Number of neurons: 5.
AM	AM activation function: activation=sigmoid; Fully connected layer dimension=64; Weight type: single attention vector = true.
Transformer	Input time steps: 3; embedding dimension: 8; number of attention heads: 2; encoder layers: 1; feed-forward dimension: 16; LSTM readout units: 4; optimizer: Adam; maximum epochs: 18; mini-batch size: 32; initial learning rate: 0.001.
XGBoost	Number of trees: 200; maximum tree depth: 3; learning rate: 0.03; subsampling ratio: 0.80; minimum samples per leaf: 10; L2 regularization coefficient: 5.0; minimum split gain: 0.01.

To isolate the effect of historical similar-sequence retrieval from other factors, three representative forecasting architectures—BiLSTM, CNN-BiLSTM-AM, and DBO-CNN-BiLSTM-AM—were evaluated under two data settings: with and without historical similar-sequence retrieval. The comparative results are presented in Table 4.

Table 4. Effect of historical similar-sequence retrieval on forecasting accuracy

Model	Similar-sequence retrieval	MAE	RMSE	R^2
DBO-CNN-BiLSTM-AM	Yes	22.641	31.675	0.878
DBO-CNN-BiLSTM-AM	No	38.150	48.497	0.715
CNN-BiLSTM-AM	Yes	23.790	32.361	0.873
CNN-BiLSTM-AM	No	35.866	48.684	0.713
BiLSTM	Yes	27.644	41.021	0.796
BiLSTM	No	39.936	51.479	0.679

As shown in Table 4, historical similar-sequence retrieval consistently improves forecasting accuracy across all three model structures. For DBO-CNN-BiLSTM-AM, the RMSE decreases from 48.4969 to 31.6754 CNY/MWh after similar-sequence retrieval is introduced. The RMSE values of CNN-BiLSTM-AM and BiLSTM decrease from

48.6839 to 32.3609 CNY/MWh and from 51.4785 to 41.0209 CNY/MWh, respectively. These results demonstrate that retrieving historically relevant price patterns provides a more informative training set and improves the robustness of different forecasting architectures.

To further evaluate the contribution of individual model components and benchmark the proposed framework against contemporary methods, the proposed model was compared with structural ablation models and two representative benchmark models, XGBoost and Transformer. The forecasting results are summarized in Table 5.

Table 5. Forecasting accuracy comparison under 3% similar-sequence retrieval

Model	MAE	RMSE	R^2
DBO-CNN-BiLSTM-AM	22.641	31.675	0.878
CNN-BiLSTM-AM	23.790	32.361	0.873
CNN-BiLSTM	23.749	32.484	0.872
CNN-AM	24.411	33.275	0.866
CNN	26.720	35.237	0.850
BiLSTM-AM	29.061	38.698	0.819
BiLSTM	27.644	41.021	0.796
XGBoost	31.915	44.651	0.758
Transformer	48.920	58.943	0.579

As shown in Table 5, DBO-CNN-BiLSTM-AM achieves the best overall forecasting performance, with an R^2 of 0.8784, an RMSE of 31.6754 CNY/MWh, and an MAE of 22.6413 CNY/MWh.

The structural ablation results demonstrate the contributions of the individual components. Compared with BiLSTM, CNN reduces RMSE from 41.0209 to 35.2370 CNY/MWh, indicating that convolutional feature extraction is useful for capturing local electricity-price fluctuation patterns. Adding AM to CNN further reduces RMSE to 33.2750 CNY/MWh, while incorporating BiLSTM reduces RMSE to 32.4840 CNY/MWh. These results suggest that both attention-based feature weighting and bidirectional temporal dependency learning improve the representation of electricity-price dynamics.

The CNN-BiLSTM-AM model achieves a slightly lower RMSE than CNN-BiLSTM, decreasing from 32.4840 to 32.3609 CNY/MWh, with R^2 increasing from 0.8721 to 0.8731. This result indicates a marginal incremental contribution of AM after CNN and BiLSTM are jointly used. Relative to the manually configured CNN-BiLSTM-AM reference model, DBO-based hyperparameter optimization further decreases RMSE from 32.3609 to 31.6754 CNY/MWh and increases R^2 from 0.8731 to 0.8784, confirming the benefit of adaptive hyperparameter selection.

Compared with the contemporary benchmark models, the proposed model reduces RMSE by 29.1% relative to XGBoost and by 46.3% relative to Transformer. Therefore, the proposed framework performs better than both the tree-based ensemble-learning baseline and the Transformer-based deep-learning baseline for the present electricity-price forecasting task.

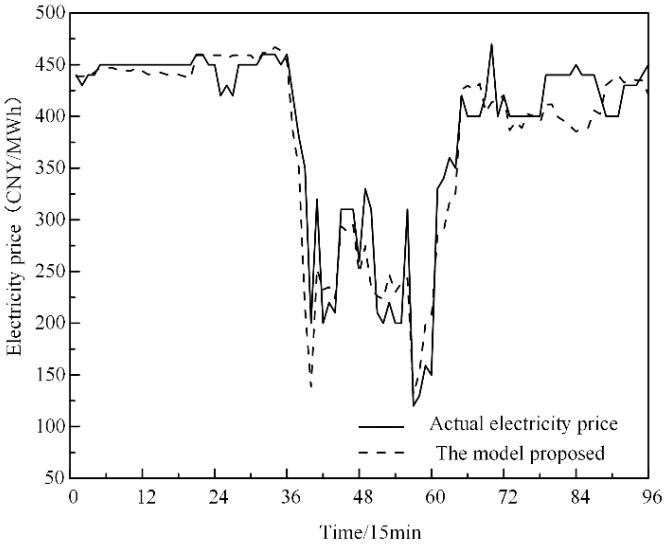


Figure 6. Comparison between the predicted and the actual electricity prices

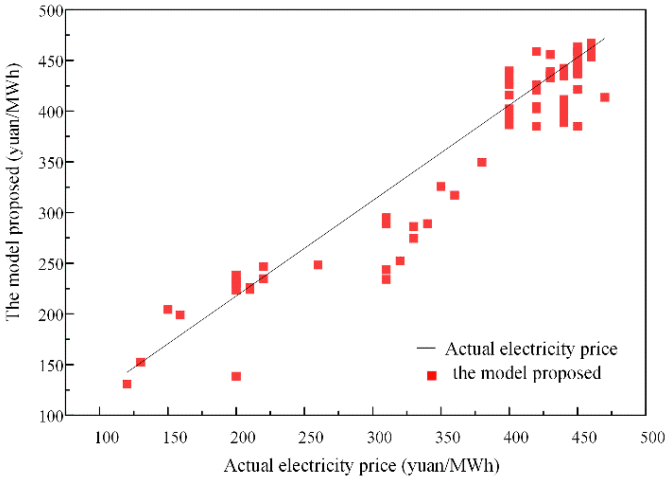


Figure 7. Scatter plot of predicted versus actual electricity prices

Figures 6 and 7 provide a visual assessment of the proposed DBO-CNN-BiLSTM-AM model. Figure 6 shows that the predicted electricity-price trajectory generally follows the observed variation pattern, including the major peak and trough periods. Figure 7 shows that most predicted values are distributed close to the diagonal reference line, indicating good consistency between predicted and actual electricity prices. The remaining deviations are mainly observed in periods with sharp price changes, which remain challenging for day-ahead electricity-price forecasting.

4.4.2 Comparative Experiments of Hyperparameter Optimization Algorithms

To evaluate the performance advantage of the DBO algorithm in tuning the hyperparameters of the CNN-BiLSTM-AM model, this study introduces two widely-used optimization algorithms—the Sparrow Search Algorithm (SSA) and Particle Swarm Optimization (PSO)—for comparative purposes. These algorithms were respectively applied to optimize the following three parameters: the learning rate decay factor, the initial learning rate, and the number of neurons in the BiLSTM layer. The parameter configurations employed by the compared algorithms are detailed in Table 6. RMSE is selected as the key evaluation criterion within the fitness function, with the corresponding prediction outcomes illustrated in Figures 8 and 9.

Table 6. Parameter Settings for Various Optimization Algorithms in Comparative Experiments

Algorithm	Parameter Settings
DBO	population size pop=25; Optimization variable dimension: dim=3; Subgroup proportions (rolling/breeding/foraging/stealing): 20%/30%/30%/20%; Fixed coefficient: b=0.1; Deflection coefficient: l=0.1.
SSA	Population size: pop=25; Discoverers proportion: PD=0.5; Warning agents' proportion: SD=0.2; Safety threshold: ST=0.6; Exploration factor: EF=1.0; Learning factor: LF=1.5.
PSO	Particle number: pop= 25; Individual learning factor: C1= 2.0; Inertia weight:$\omega = 0.9$; Social learning factor: C2=2.0.

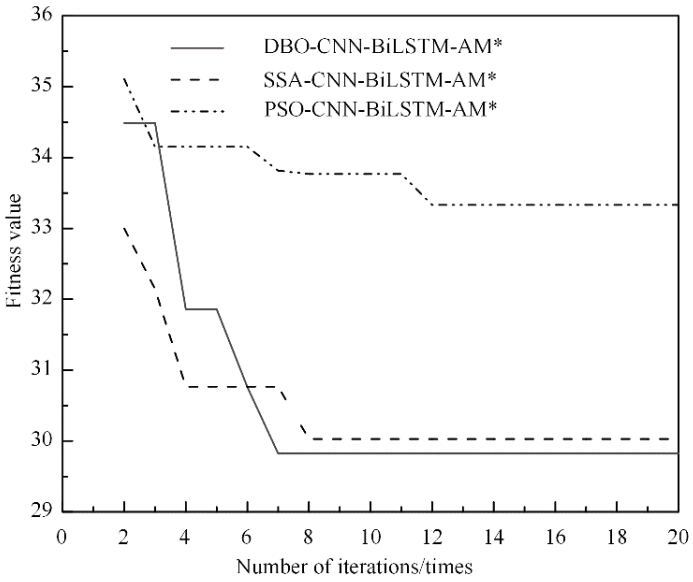


Figure 8. Comparison of Fitness Curves for Optimization Algorithms
(Note: Models marked with '' incorporate similarity consideration)*

As shown in Figure 8, the DBO algorithm achieves near-optimal performance by the 7th iteration, indicating a rapid convergence rate. After convergence, the curve remains stable at the minimum fitness value without large fluctuations, indicating strong robustness and a low tendency to fall into local oscillations. In contrast, PSO lacks sufficient convergence accuracy, while SSA is prone to falling into local optima and exhibits the slowest convergence speed.

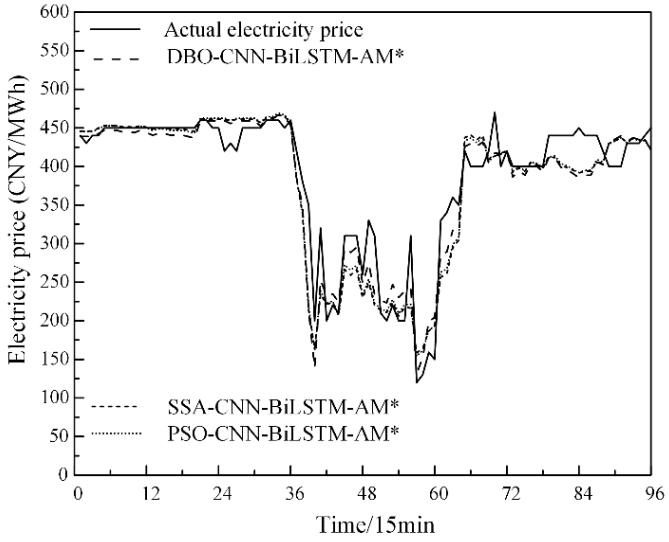


Figure 9. Comparative Experimental Results of Optimization Algorithms
(Note: Models marked with '' incorporate similarity consideration)*

Figure 9 demonstrates how the DBO algorithm skilfully emulates various natural dung beetle behaviours. These simulated actions are then systematically mapped to corresponding mathematical search strategies. After parameter optimization, the model optimized by DBO produces power-value predictions that are closer to the actual values, with smoother fitting curves and the smallest prediction error. Although the SSA and PSO algorithms also show performance improvements after optimization, their results fluctuate considerably. In comparison, both algorithms are outperformed by DBO with respect to convergence accuracy and overall robustness. The MAE, RMSE, and R^2 of the test dataset are summarized in Table 7.

Table 7. Evaluation of Prediction Accuracy for Comparative Experiments of Optimization Algorithms

Model (with similarity consideration)	MAE	RMSE	R^2
DBO-CNN-BiLSTM-AM	22.641	31.675	0.878
PSO-CNN-BiLSTM-AM	23.271	33.766	0.861
SSA-CNN-BiLSTM-AM	23.809	35.209	0.850

As shown in Table 7, among the three optimization algorithms, the DBO-CNN-BiLSTM-AM model exhibits the best predictive performance. Compared with the PSO-CNN-BiLSTM-AM model, DBO reduces MAE and RMSE by 2.7% and 6.2%, respectively, while increasing R^2 from 0.861 to 0.878. Compared with the SSA-CNN-BiLSTM-AM model, MAE and RMSE are reduced by 4.9% and 10.0%, respectively, with R^2 increasing from 0.850 to 0.878. These results collectively highlight the significant advantage of the DBO algorithm in hyperparameter optimization, enabling it to identify more optimal parameter configurations and thereby substantially improving predictive performance.

5. Conclusions

This paper proposes a DBO–CNN–BiLSTM–AM day-ahead clearing price forecasting method enhanced with a historical similar-sequence identification mechanism. Using the day-ahead clearing price data of the Miaojiaba Hydropower Station in the Bailong River Basin of Gansu Province as a case study, simulation experiments and comparative analyses against multiple benchmark models yield the following conclusions:

(i) An innovative historical similar-sequence identification mechanism is introduced, which substantially improves the forecasting performance by constructing a more relevant training dataset. This mechanism not only uncovers potential temporal dependencies embedded in historical electricity price data but also effectively reduces redundant inputs. As a result, it enhances model training efficiency while strengthening the structural representation capability of the extracted features.

(ii) Compared with the traditional BiLSTM model, the proposed model achieves significantly higher prediction accuracy and better generalization through dynamic optimization, feature enhancement, and an attention mechanism, thereby realizing a deeper integration of optimization algorithms and deep learning models. The model employs the DBO algorithm to adaptively optimize network parameters, improving convergence efficiency; utilizes CNN to extract local spatial features of price sequences; and leverages BiLSTM to capture bidirectional long- and short-term dependencies. Furthermore, the attention mechanism emphasizes critical time periods, resulting in a hybrid forecasting model equipped with dynamic adjustment capability, multi-feature fusion, and key-pattern recognition. This integrated structure leads to substantial improvements in both prediction accuracy and generalization performance.

Future work may incorporate Change Point Detection (CPD) algorithms to identify periods of abrupt price variations, introduce additional market structure and real-time state features, and refine the model's loss function to place greater emphasis on prediction accuracy in high-value regions.

Declarations

Data Availability

The processed datasets used in the sensitivity analysis, together with the test-set prediction results and evaluation metrics for all compared models, are provided as Supplementary Materials. The original operational data are available from the corresponding author upon reasonable request.

Acknowledgements

This work was financially supported by the Science and Technology Department of Sichuan Province within the framework of the Regional Innovation Cooperation Project of Sichuan Province (Grant No. 2025YFHZ0281).

Authors' Contributions

Qingzhong Wang: Conceptualization and Methodology; Xi Fu: Model Construction and Computation; Weiguo Xiao: Verification and Analysis; Chuankai He: Initial Draft Writing; Xiaogang Xie: Data Curation; Jidong Li: Article Review and Revision.

References

[1] Feng S, Zhang X. An electricity-carbon joint market optimization model for coal-fired power system under

China's "dual-carbon" target[J]. Journal of Cleaner Production, 2023; 423: 138746.
[2] Zhao X, Hu S, Wang H, et al. Energy, economic, and environmental impacts of electricity market-oriented reform and the carbon emissions trading: A recursive dynamic CGE model in China[J]. Energy, 2024; 298: 131416.
[3] Iftikhar H, Turpo-Chaparro J E, Rodrigues P C, et al. Forecasting Day-Ahead Electricity Prices for the Italian Electricity Market Using a New Decomposition-Combination Technique[J]. Energies, 2023; 16(18): 6669.
[4] Bottieau J, Wang Y, De Greve Z, et al. Interpretable Transformer Model for Capturing Regime Switching Effects of Real-Time Electricity Prices[J]. IEEE Transactions on Power Systems, 2023; 38(3): 2162-2176.
[5] Abroun M, Jahangiri A, Shamim A G, et al. Predicting long-term electricity prices using modified support vector regression method[J]. Electrical Engineering, 2024; 106: 4103-4114.
[6] Tang C, Ye Z, Zhao H, et al. DeepSCNN: a simplicial convolutional neural network for deep learning[J]. Applied Intelligence, 2025; 55: 281.
[7] Xu Y, Huang X, Gao Z, et al. A novel electricity price forecasting approach based on multi-attention feature fusion model optimized by variational mode decomposition[J]. Measurement, 2025; 253: 117596.
[8] Dong J, Dou X, Bao A, et al. Day-Ahead Spot Market Price Forecast Based on a Hybrid Extreme Learning Machine Technique: A Case Study in China[J]. Sustainability, 2022; 14(13): 7767.
[9] Kosana V, Madasthu S, Teeparthi K. A novel hybrid framework for wind speed forecasting using autoencoder-based convolutional long short-term memory network[J]. International Transactions on Electrical Energy Systems, 2021; 31(11): e13072.
[10] Vlah D, Sepetanc K, Pandzic H. Solving Bilevel Optimal Bidding Problems Using Deep Convolutional Neural Networks[J]. IEEE Systems Journal, 2023; 17(2): 2767-2778.
[11] Liu W, Mao Z. Short-term photovoltaic power forecasting with feature extraction and attention mechanisms[J]. Renewable Energy, 2024; 226: 120437.
[12] Salgotra R, Sharma P, Raju S, et al. A Contemporary Systematic Review on Meta-heuristic Optimization Algorithms with Their MATLAB and Python Code Reference[J]. Archives of Computational Methods in Engineering, 2024; 31: 1749-1822.
[13] Ghimire S, Deo R C, Casillas-Perez D, et al. Half-hourly electricity price prediction with a hybrid convolution neural network-random vector functional link deep learning approach[J]. Applied Energy, 2024; 374: 123920.
[14] Nie Y, Li P, Wang J, et al. A novel multivariate electrical price bi-forecasting system based on deep learning , a multi-input multi-output structure and an operator combination mechanism[J]. Applied Energy, 2024; 366: 123233.
[15] Wang K, Yu M, Niu D, et al. Short-term electricity price forecasting based on similarity day screening, two-layer decomposition technique and Bi-LSTM neural network[J]. Applied Soft Computing, 2023; 136: 110018.
[16] Zhang X. Short-term electrical load forecasting strategy based on EEMD-SSA-BiLSTM[J]. Signal Image and Video Processing, 2025; 19: 751.
[17] Xue J, Shen B. Dung beetle optimizer: a new meta-heuristic algorithm for global optimization[J]. The Journal of Supercomputing, 2023; 79: 7305-7336.