

Enhancing photovoltaic power prediction using a CNN-LSTM-attention hybrid model with Bayesian hyperparameter optimization

Ning Zhou¹, Bowen Shang¹, Mingming Xu¹, Lei Peng², Yafei Zhang¹

1. State Grid Henan Electric Power Research Institute, Zhengzhou, 450052, P. R. China

2. State Grid Henan Electric Power Corporation, Zhengzhou, 450052, P. R. China



Scan for more details

Abstract: Improving the accuracy of solar power forecasting is crucial to ensure grid stability, optimize solar power plant operations, and enhance grid dispatch efficiency. Although hybrid neural network models can effectively address the complexities of environmental data and power prediction uncertainties, challenges such as labor-intensive parameter adjustments and complex optimization processes persist. Thus, this study proposed a novel approach for solar power prediction using a hybrid model (CNN-LSTM-attention) that combines a convolutional neural network (CNN), long short-term memory (LSTM), and attention mechanisms. The model incorporates Bayesian optimization to refine the parameters and enhance the prediction accuracy. To prepare high-quality training data, the solar power data were first preprocessed, including feature selection, data cleaning, imputation, and smoothing. The processed data were then used to train a hybrid model based on the CNN-LSTM-attention architecture, followed by hyperparameter optimization employing Bayesian methods. The experimental results indicated that within acceptable model training times, the CNN-LSTM-attention model outperformed the LSTM, GRU, CNN-LSTM, CNN-LSTM with autoencoders, and parallel CNN-LSTM attention models. Furthermore, following Bayesian optimization, the optimized model demonstrated significantly reduced prediction errors during periods of data volatility compared to the original model, as evidenced by MRE evaluations. This highlights the clear advantage of the optimized model in forecasting fluctuating data.

Keywords: Photovoltaic power prediction; CNN-LSTM-Attention; Bayesian optimization

0 Introduction

Solar energy is considered a renewable resource because of its abundant supply, accessibility, and economical operation [1, 2]. The escalating energy demand in China has propelled a rapid annual growth in photovoltaic (PV) power generation. Nevertheless, the variability of environmental factors such as weather and sunlight poses challenges to PV power systems, resulting in equipment damage and grid instability [3]. Reliable prediction of future PV power output would provide information for power system

Received: 23 May 2024/ Revised: 11 June 2024/ Accepted: 17 July 2024/

Published: 25 October 2024

✉ Bowen Shang
shangbw69@163.com

Ning Zhou
zhouningznzz@sina.cn

Mingming Xu
240409976@qq.com

Lei Peng
1748467752@qq.com

Yafei Zhang
1548071640@qq.com

planning, operation, optimization, and dispatch, reduce the uncertainty of input power to the grid, and ensure safe and stable operation [4]. Thus, the realization of precise PV power prediction is of paramount importance.

Various algorithms and models have been proposed to address the PV power prediction challenge. These methods are categorized into statistical, artificial intelligence (AI), and hybrid-approach methods. Statistical methods encompass diverse algorithms such as the autoregressive model [5], Bayesian method [6], Kalman filter [7], grey model [8], and Markov chain model [9]. These methods can handle linear data; however, they often have a limited ability to capture complex nonlinear relationships. AI techniques demonstrate great promise for extracting features and exploring data, effectively capturing non-linear connections between input and output variables. Common AI methodologies include neural networks [10], support vector regression [11], and adaptive fuzzy methods [12]. AI-based deep learning methods include recurrent neural networks (RNNs) [13], long short-term memory networks (LSTMs) [14], convolutional neural networks (CNNs) [15], and gated recurrent units (GRUs) [16]. Although neural networks offer a unique advantage in handling nonlinear relationships, a single network structure often outperforms others when processing only specific types of feature information. For example, CNNs are primarily used for spatial feature extraction [17] but struggle with temporal dependencies. Whereas, RNNs are designed for temporal feature learning and cannot extract spatial information. To address the limitation of single network structures being unable to simultaneously capture both the temporal and spatial features of photovoltaic power, researchers have proposed hybrid deep learning models to enhance photovoltaic power prediction performance [18]. These hybrid models include CNN-RNN [19], CNN-GRU [20], CNN-LSTM [21], and CNN-LSTM with autoencoders [22]. CNN-RNN was the earliest hybrid model; however, RNNs encounter issues with gradient vanishing and exploding. LSTM alleviates the vanishing gradient problem through its gating mechanism and uses a cell state to control long-term memory, effectively preventing it from overshadowing [23]. The GRU simplifies the LSTM; however, it underperforms in terms of handling complex time dependencies in photovoltaic forecasting compared with the LSTM. An LSTM with autoencoders introduces more intermediate layers and feature representations, thereby increasing the difficulty of model interpretability and requiring more computational resources [22]. Based on this comparison, we selected a CNN-LSTM network structure and incorporated an attention mechanism to dynamically adjust the focus of

the model on different parts of the input data, aiming for higher precision in photovoltaic power prediction.

Hybrid models comprising multiple intricate neural network components entail numerous hyperparameters that influence the model training efficiency and prediction accuracy; thus, their optimization in each sub-model is pivotal for effective PV power prediction. Owing to the challenge of balancing efficiency and precision during hyperparameter selection, automatic parameter optimization is a pragmatic solution for hyperparameter tuning. Currently, popular hyperparameter optimization algorithms used in neural networks include grid and random searches. The fundamental idea involves exhaustively validating all points in a large parameter space and returning the optimal value, which is time-consuming and labor-intensive. This approach is plagued by low training efficiency, poor handling of complex parameter spaces, and poor adaptability. In contrast, the Bayesian optimization algorithm, based on Bayesian principles, adaptively selects better sampling points and adjusts the direction of the next search based on the observed optimal results in each iteration. Hence, it gradually approaches the global optimal solution. Consequently, the Bayesian optimization algorithm exhibits higher accuracy, efficiency, and adaptability and exhibits superior performance on non-convex optimization problems [24]. It is well suited for addressing hyperparameter optimization problems in mixed neural network models. Thus, this study selected the Bayesian optimization method to accomplish hyperparameter optimization tasks.

This study proposed a PV power prediction method, termed CNN-LSTM-attention, which leverages Bayesian hyperparameter optimization. Initially, the Desert Knowledge Australia Solar Centre (DKASC) PV dataset was utilized, and pertinent algorithms were employed to identify crucial features and preprocess the data [24]. Subsequently, a CNN-LSTM-attention neural network model was constructed to facilitate model training and power data prediction. This model employed a hybrid approach that integrated three algorithms. The CNN and LSTM components extracted spatial and temporal information from the PV power and feature data, the attention mechanism emphasized key information, and the Bayesian optimization algorithm employed the tree-structured Parzen estimator (TPE) and fine-tuned the hyperparameters of the initial model. This iterative process involved updating the model through the probabilistic agent model and the acquisition function to identify the optimal hyperparameter combination. This novel hybrid network model amplified the feature extraction capabilities of PV power data and effectively addressed the intricate issue of hybrid model

hyperparameter tuning. Compared with the parallel CNN and LSTM architecture followed by an attention mechanism [25, 26], this study adopted a pipeline sequential structure, which enhanced the information interaction between the CNN and LSTM. This approach facilitated the comprehensive utilization of the strengths of both networks for deep feature learning. While this increased the training time and requires more hyperparameter tuning, the stronger integration of information rendered it more suitable for addressing complex spatiotemporal modeling problems such as PV power forecasting. The empirical findings indicate that the proposed PV power prediction approach exhibited improved accuracy with a discernible augmentation effect resulting from the hyperparameter optimization algorithm.

1 Model description and problem analysis

Power output prediction is pivotal for PV power generation to ensure efficient energy utilization. This process involves the analysis and modeling of diverse factors within a system to anticipate the future output power. Representing the PV power output as y and a set of closely related factors as $x = [x_1, x_2, \dots, x_n]$, the mathematical relationship between the two can be expressed as:

$$y = f(x) \quad (1)$$

By leveraging the obtained data of x and its corresponding power output y , the mathematical relationship f can be modeled. This enables the calculation of the anticipated PV power output using the existing data. The crux of the PV prediction technology lies in the use of algorithms to estimate the mathematical relationship f between x and y . In practice, it is often challenging to obtain entirely accurate x and y values owing to various real-world factors such as environmental conditions and equipment limitations. Moreover, f represents a complex nonlinear functional relationship. This renders the extraction of information and performing precise calculations using limited monitoring data challenging. These challenges collectively contribute to the low predictive accuracy of PV power forecasts.

To address these challenges, this study proposed an innovative CNN-LSTM-attention method for PV power prediction that incorporated Bayesian superparameter optimization. The proposed approach encompassed several key stages, as illustrated in Fig. 1.

The acquired PV datasets were subjected to essential feature screening and data pretreatment to obtain a dataset with crucial information and minimal interference factors. A CNN-LSTM-attention model was then established to estimate f , thereby facilitating power data prediction.

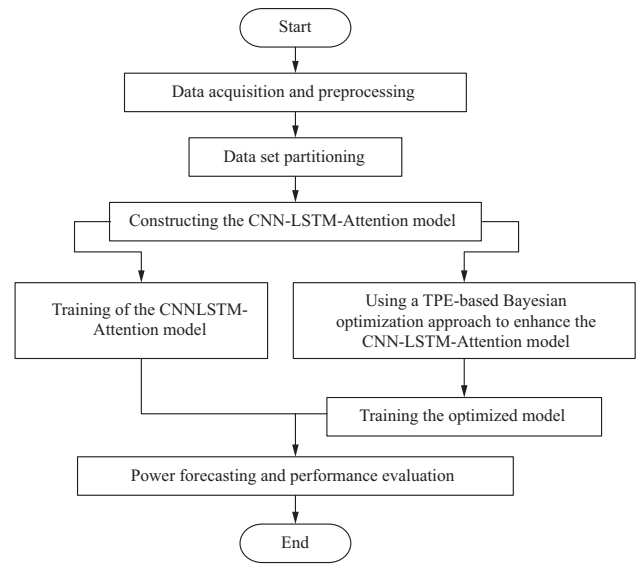


Fig. 1 General process of the PV power prediction method

The TPE-based Bayesian optimization algorithm enabled hyperparameter optimization, ultimately yielding an optimized model with the most effective hyperparameter combination. Finally, the initial, optimized. Consequently traditional models were comprehensively compared to verify the superiority of the proposed prediction method.

2 CNN-LSTM-attention PV power prediction model

Different PV power stations have diverse geographical and meteorological conditions, resulting in distinct spatial characteristics of power output data. Because PV power data represent a typical time series characterized by nonlinearity and temporal correlations, their variations are related to current factors and past power and characteristic data. Hence, the accurate capture of the spatial, temporal, and nonlinear features of PV power and modeling its mathematical relationships with other attributes are crucial. In this study, an integrated CNN-LSTM-attention hybrid model was established to enable the high-precision prediction of PV power.

2.1 Generation of input–output vectors

When constructing a predictive model, it is crucial to design input and output vector shapes that align with the CNN-LSTM-Attention model structure.

1) Input vector:

The composition of the input vector for the CNN-LSTM-attention PV power prediction model is dependent on the feature data that influence the PV power output. The effects of different features at the current and historical time

points must be considered. Let d be the number of sampling time points for the daily data. The input vector for the model comprises historical feature data from d time points of the previous day and partial feature data from the prediction time step to numerically predict the PV power output at the prediction time step.

For each time step t :

$$\mathbf{I}_t = [I_{t,1}, I_{t,2}, \dots, I_{t,l}] \quad (2)$$

where $\mathbf{I}_t$ is a $1 \times l$ vector representing the historical feature data at each time point and l is the number of historical features. Thus, the dimension of the historical feature vector from the d time points of the previous day is $d \times l$:

$$\mathbf{I}_{t-d:t-1} = \begin{bmatrix} I_{t-d,1} & I_{t-d,2} & \dots & I_{t-d,l} \\ I_{t-d+1,1} & I_{t-d+1,2} & \dots & I_{t-d+1,l} \\ \vdots & \vdots & \ddots & \vdots \\ I_{t-1,1} & I_{t-1,2} & \dots & I_{t-1,l} \end{bmatrix} \quad (3)$$

It is assumed that the feature data at the current prediction time can be represented as an $1 \times k$ vector, denoted as $\mathbf{I}'_t = [I'_{t,1}, \dots, I'_{t,k}]$, where k is the number of features at the current prediction time. Merging this with the historical features yields a complete feature vector of dimensions $d \times (l + k)$ for the time step.

$$\mathbf{I}_{t-d:t-1}, \mathbf{I}'_t = \begin{bmatrix} I_{t-d,1} & I_{t-d,2} & \dots & I_{t-d,l} & I'_{t,1} & \dots & I'_{t,k} \\ I_{t-d+1,1} & I_{t-d+1,2} & \dots & I_{t-d+1,l} & I'_{t,1} & \dots & I'_{t,k} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ I_{t-1,1} & I_{t-1,2} & \dots & I_{t-1,l} & I'_{t,1} & \dots & I'_{t,k} \end{bmatrix} \quad (4)$$

Equation (4) characterizes the feature input for a single time step in the model. For the N time steps, the final form of the input vector of the model is represented as

$$\mathbf{I} = N \times d \times (l + k) \quad (5)$$

2) Output vector:

The ultimate predictive goal of the proposed PV power forecasting model is the PV power output. Thus, the output vector comprises only the PV power data. At each time step, the model input corresponds to the PV power output value, resulting in the final form of the output vector

$$\mathbf{O} = N \times 1 \quad (6)$$

Once the model's data input-output vectors ($\mathbf{I}$, $\mathbf{O}$) are established, the mathematical process of the PV power forecasting model can be expressed as

$$\mathbf{O} = \Phi(\mathbf{I}) \quad (7)$$

By constructing and iteratively training the CNN-LSTM-attention model, the mathematical relationship Φ between $\mathbf{I}$ and $\mathbf{O}$ can be modeled, thereby accurately representing $\mathbf{f}$ to the best extent possible.

2.2 Construction of the model network structure

The CNN-LSTM-attention PV power prediction model

network architecture comprised the input, CNN, LSTM, dropout, attention, and fully connected layers (Fig. 2). The CNN layer captured local features at different positions in the input data. The LSTM layer modeled the long-term dependencies from the feature sequence output of the convolutional layer. The attention mechanism calculated the attention weights for each time step or position to enhance the focus on different parts of the sequence and extract crucial information. The fully connected layer produced the final output. Finally, the two dropout layers randomly removed neurons and their connections from the neural network to prevent overfitting.

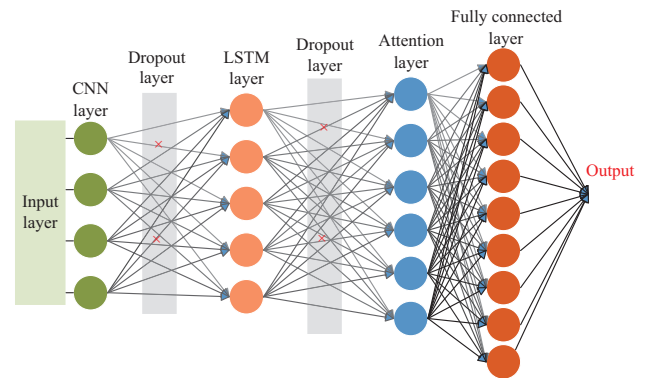


Fig. 2 Network architecture of CNN-LSTM-attention

The power output computation process using the CNN-LSTM attention model is as follows:

1) The input vector satisfying these requirements was fed into the input layer and denoted as $\mathbf{I}$.

2) In the CNN layer, convolution operations (also known as point convolutions) were conducted using convolutional kernels of size one and various filters to produce the point convolution output. Assuming that the weights of the point convolution kernel are W_{conv} , the output of the CNN layer is expressed as

$$Conv(\mathbf{I}) = dot(\mathbf{I}, W_{conv}) \quad (8)$$

where “dot” indicates the product of the corresponding elements at each position of the two matrices, followed by summation.

3) Within the LSTM layer, a single LSTM unit comprised forget, input, and output gates. The forget gate determined the information that should be forgotten, the input gate determined the information that must be updated and stored in the state, and the output gate, combined with the updated state, determined the state and output of the current time step. Assuming that the LSTM layer had H hidden units (the size of the hidden layer), the memory cell, input gate, forget gate, output gate, and candidate cell state were represented by C_p , R_p , F_p , O_p , G_p . The LSTM update process is expressed as follows:

$$\begin{aligned} H_t &= LSTM(Conv(\mathbf{I}), H_{t-1}) \\ C_t, R_t, F_t, O_t, G_t &= LSTM_cell(H_t) \end{aligned} \quad (9)$$

where $LSTM$ is the LSTM layer update operation, and $LSTM_cell$ is the LSTM unit computation process [27].

4) In the attention layer, assuming that A is the attention weight and H_{att} is the output of the attention layer, the calculation process is expressed as:

$$H_{att} = Attention(H_t, H_t, H_t) = \text{soft max}\left(\frac{H_t H_t^T}{\sqrt{d_k}}\right) H_t \quad (10)$$

where soft max represents the SoftMax function [28] and d_k denotes the weighted sum parameters (keys).

5) The final output was generated by a fully connected layer. Assuming that W_{fc} and b_{fc} are the weights and biases of the fully connected layer, respectively, the output is calculated as

$$\mathbf{O} = \text{Softplus}(H_{att} W_{fc} + b_{fc}) \quad (11)$$

where Softplus is the Softplus activation function.

Dropout layers were included following the CNN and LSTM layers. If the output of a neuron is denoted as x' during the dropout operation, the probability that the output of that neuron will be zeroed out is p , whereas the probability of retaining it is $1 - p$. The neuronal output is thus expressed as

$$y' = \begin{cases} 0, & \text{with probability } p \\ \frac{x'}{1-p} & \text{otherwise} \end{cases} \quad (12)$$

This process delineates the data flow and transformation from the model input to output. The specific steps involved in model training are as follows: 1) Forward propagation: Input the model data vectors and compute the initial power output predictions via steps (1)–(5); 2) Loss computation: Contrast the predicted value with the actual power data and calculate the error via a loss function; 3) Backward propagation: Derive the weight parameter gradients using the loss function to determine the effect of each parameter on the loss function. 4) Parameter optimization: Update the weight parameters using optimization algorithms based on gradient information to progressively minimize the loss function. Through multiple iterative training sessions, the weight parameters were continuously adjusted and optimized based on the disparity between the actual power data and the predicted values. This yielded a high-performance predictive model. This modeling process derived the relationship Φ between the model inputs and outputs, characterizing the mathematical relationship $\mathbf{f}$ between the power output y and feature data x as follows:

$$\mathbf{f} \approx \Phi \quad (13)$$

Thus, to forecast the power for a specific period, the existing feature data must only be input into a well-trained

model that enables the computation of the corresponding PV power prediction data.

The entire model construction and training process involved numerous parameters that were broadly categorized into weights (i.e., learned in a model through training) and hyperparameters (i.e., set before model training to control the training process). The key hyperparameters included the number of convolutional kernels (*filters*), hidden LSTM layer size (*units*), neuron output retention probabilities of the dropout layers (*drop_C* and *drop_L*), and batch size during model training (*batch_size*). For a single model with a few hyperparameters, trial and error can yield suitable hyperparameter combinations; however, this approach is ineffective for complex models with numerous hyperparameters and longer training times. Thus, the workload of parameter tuning must be reduced and appropriate hyperparameter adjustment methods must be determined to improve the PV power prediction accuracy.

3 Bayesian theory-based hyperparameter optimization for the PV power prediction model

Hybrid neural network models require robust hyperparameter optimization algorithms because of their complex network structures and numerous hyperparameters. Bayesian optimization stands out for its adaptive nature, efficiency, and superior performance in nonconvex optimization problems. This renders it well suited for the high computational and time-intensive demands of hyperparameter tuning. Therefore, we selected Bayesian optimization as the strategy for optimizing the hyperparameters of our model.

3.1 Bayesian optimization theory based on TPE

Bayesian optimization, first introduced in 1998 [29], applies a Bayesian formula to optimize complex black box functions. The core of the algorithm is the idea of combining Bayesian statistics and probabilistic modeling to determine the optimal solution of a function within a limited number of iterations, considering a finite set of sample points, and constructing the posterior probability of the output of the black box function.

The Bayesian optimization algorithm primarily comprises two essential components: a probabilistic surrogate model and an acquisition function (Fig. 3). Initially, a few sampling points are randomly selected from the objective function, based on which the probabilistic surrogate model estimates the objective function [30, 31]. The acquisition function evaluates the impact weights of

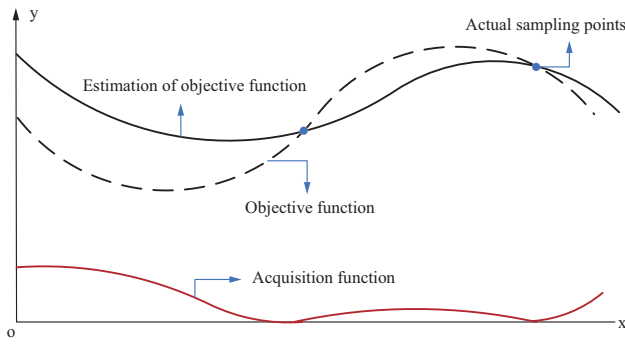


Fig. 3 Schematic of Bayesian optimization

the sampling positions to estimate the objective function, and selects the position with the highest impact as the next sampling point for subsequent observations, typically choosing the point with the highest acquisition function value. The surrogate model is updated accordingly, and the acquisition function was recalculated to determine the next sampling point. This process is iteratively repeated until the termination condition is satisfied, ultimately yielding an approximate distribution of the objective function and determining its optimum value.

In the optimization process, the selection of appropriate probabilistic surrogate models and acquisition functions is crucial. Gaussian processes are commonly selected as probabilistic surrogate models; however, they incur high computational costs and perform poorly when optimizing large datasets and high-dimensional spaces. By contrast, the TPE algorithm [32], with its unique tree-like structure, demonstrates greater efficiency, robustness, and adaptability. The commonly used acquisition functions include the probability of improvement [33], expected improvement (EI) [34], and upper confidence bounds [35]. In the proposed optimization algorithm, the TPE algorithm was employed as the probabilistic surrogate model, the EI method was used as the acquisition function, and the loss value obtained after the predictive model completed the prediction served as the objective function. Through iterative computation, the algorithm obtained an optimal hyperparameter combination that minimized the objective function value.

During the initial computations, the known finite initial sample set is denoted as:

$$D = \{[(v_1, \dots, z_1), e_1], \dots, [(v_i, \dots, z_i), e_i]\} \quad (14)$$

where $(v_i, \dots, z_i)$ represents different combinations of five hyperparameters: *filters*, *units*, *drop_C*, *drop_L*, and *batch_size*, e_i is the objective function for each hyperparameter combination, and the Set D comprises the known initial hyperparameter combinations and their corresponding objective function values.

The TPE algorithm redefines the likelihood distribution

$p(m, n)$ as follows:

$$p(m, n) = \begin{cases} l(m) & n < n^* \\ g(m) & n \geq n^* \end{cases} \quad (15)$$

where m is the observed value, indicating a specific hyperparameter combination selected at that time; n is the output value of the objective function corresponding to the observed value m ; n^* is a threshold to assess the quality of the observed value m , where $n < n^*$, indicating that m is superior, otherwise inferior; $l(m)$ is the density estimate of observations where $n < n^*$; and $g(m)$ is the density estimate of observations where $n \geq n^*$.

Using EI as the acquisition function and substituting (15) into the EI formula,

$$\begin{aligned} EI_{n^*}(m) &= \int_{-\infty}^{n^*} (n^* - n) p(n | m) dn \\ &= \int_{-\infty}^{n^*} (n^* - n) \frac{p(m | n) p(n)}{p(m)} dn \end{aligned} \quad (16)$$

Let $\gamma = p(n < n^*)$, then:

$$\begin{aligned} p(m) &= \int_{-\infty}^{n^*} p(m | n) p(n) dn \\ &= \gamma l(m) - (1 - \gamma) g(m) \end{aligned} \quad (17)$$

$$\begin{aligned} \int_{-\infty}^{n^*} (n^* - n) p(m | n) p(n) dn \\ = \gamma n^* l(m) - l(m) \int_{-\infty}^{n^*} p(n) dn \end{aligned} \quad (18)$$

Substituting (18) and (19) into (17) yields:

$$\begin{aligned} EI_{y^*}(m) &= \frac{\gamma n^* l(m) - l(m) \int_{-\infty}^{n^*} p(n) dn}{\gamma l(m) - (1 - \gamma) g(m)} \\ &\propto \left(\gamma + \frac{g(m)}{l(m)} (1 - \gamma) \right)^{-1} \end{aligned} \quad (19)$$

In (19), the value of EI is directly proportional to $l(m)/g(m)$. Thus, when m has a maximum probability of $l(m)$ and a minimum probability of $g(m)$, the maximum EI value is obtained.

$$m^* = \arg \max EI_{y^*}(m) \quad (20)$$

where m^* represents the hyperparameter combination that minimizes the objective function value. Applying the predictive model yields the corresponding value e^* for the objective function, resulting in a new sample (m^*, e^*) . By adding (m^*, e^*) to the known sample set D , updating the TPE probability model [(15)–(20)], and iteratively acquiring new samples via EI, the process continues until the termination condition is satisfied. This iterative approach determines the optimal hyperparameter combination that satisfies the specified requirements.

3.2 Adjustment of power prediction model hyperparameters based on TPE Bayesian optimization

Figure 4 shows the application of the TPE-based Bayesian

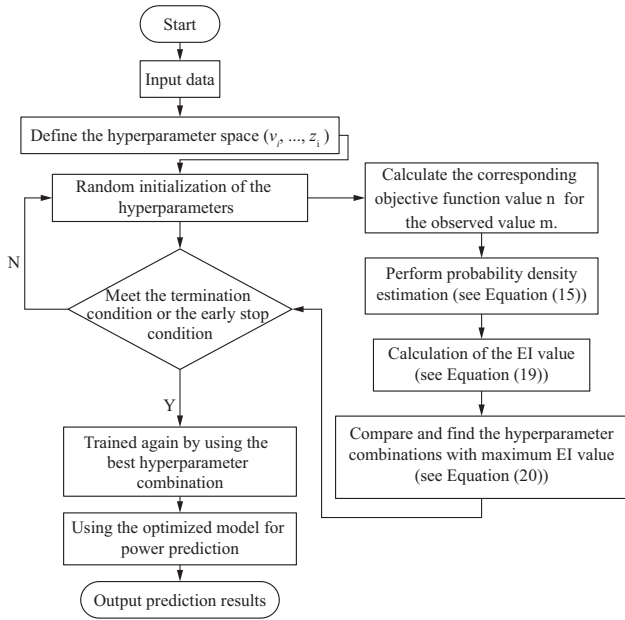


Fig. 4 Procedure for applying Bayesian optimization to optimize the hyperparameters of the prediction model

optimization method for tuning the hyperparameters of the PV power prediction model.

Step 1: Defining the hyperparameter space. Five hyperparameters were optimized: *filters*, *units*, *drop_C*, *drop_L*, and *batch_size*. Table 1 lists the ranges and step sizes for each hyperparameter.

Step 2: Defining the optimization objective function. The initial hyperparameter combination was applied to the CNN-LSTM-attention model. The model was trained on the training set, and the power data of the test set were predicted using the trained model. The mean absolute error (MAE) between the actual and predicted values was calculated and used as the loss value as an objective function for the optimization process.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (21)$$

where y_i represents actual power data, $\hat{y}_i$ denotes predicted power data, and MAE is the mean absolute error.

Table 1 Settings for the parameter space

Parameter name	Value range	Variable step size
<i>filter</i>	(30, 260)	4
<i>units</i>	(30, 260)	4
<i>drop_C</i>	(0.000001, 0.01)	random
<i>drop_L</i>	(0.000001, 0.01)	random
<i>batch_size</i>	(20, 70)	10

Step 3: Defining parameters, for example, the maximum

number of iterations for the optimization objective function and early stopping criteria. The total iteration count for the Bayesian optimization process was set to 100. An early stopping condition was established at 20 iterations; if the objective function (loss value) did not exhibit a significant improvement for 20 consecutive iterations, the optimization algorithm stopped prematurely.

Step 4: Executing the Bayesian optimization process. The posterior probability distributions of the objective function and hyperparameter combinations were continuously updated through iterations until the termination condition was satisfied, thereby identifying the optimal hyperparameter combination.

Such hyperparameter optimization for the original model yielded an optimized PV power prediction model.

4 Results and analysis

4.1 Preparation prior to experimentation

4.1.1 Data preprocessing

A subset of PV data from the DKASC dataset [36], encompassing 2020–2021 with a 5 min sampling frequency, was selected as the final original dataset Q for analysis. The DKASC dataset comprises monitoring data from multiple PV and battery energy storage systems, meteorological data, and building and system parameters and provides a comprehensive consideration of the features that may affect PV power generation. The dataset is characterized by ample and complete data collection, which renders it highly suitable for PV power prediction.

The dataset includes twelve feature variables: received active energy (AE_Power), current phase average (Current), active power (Power), wind speed (Wind_Speed), temperature (Temp), weather relative humidity (Humidity), global horizontal irradiance (GHI), diffuse horizontal irradiance (DHI), wind direction (Wind_dir), rainfall (Rainfall), global tilted radiation (RGT), and diffuse tilted radiation (RDT). Descriptive statistics, including mean, median, standard deviation, and maximum and minimum values, were calculated for each of the 12 variables. The results of this analysis are summarized in Table 2.

Table 2 Descriptive statistics of dataset Q features

Feature data	Mean	Median	Standard Deviation	Min	Max
AE_Power / kWh	254,721.78	254,307.5	11,340.59	23,5804	27,4999
Current/A	4.54	0.30	5.67	0	19.98
Power/kW	2.24	0	2.85	−0.04	9.64

continue

Feature data	Mean	Median	Standard Deviation	Min	Max
Wind_Speed/(m/s)	2.29	2.03	3.39	-919.53	54.39
Temp/ $^{\circ}$ C	20.74	21.70	11.36	-39.99	61.92
Humidity/%	34.64	28.81	21.98	0	131.16
GHI/(W/m ^{2})	269.04	9.32	369.28	0	1,524.54
DHI/(W/m ^{2})	54.03	6.63	91.19	0	769.86
Wind_dir/Degrees	28.62	32.69	15.44	-6.98	1,701.37
Rainfall/mm	0.29	0	2.05	0	52.00
RGT/(W/m ^{2})	290.80	9.44	387.29	-0.01	1,408.90
RDT/(W/m ^{2})	57.52	8.37	92.02	0.07	766.95

The numerical analysis in Table 2 reveals that the data points for Current, Power, and Temp were relatively clustered with a few outliers. Whereas, variables such as GHI, DHI, and RGT exhibited greater dispersion. Certain features even contain erroneous values, such as the negative values observed in the Power and RGT. Notably, data quality directly impacts the accuracy and applicability of models. Raw data collection is often influenced by external factors, such as environmental conditions and equipment issues, leading to numerous missing, erroneous, and unknown noise values. Therefore, appropriate preprocessing of raw data is essential before undertaking model training and prediction.

First, crucial feature data were selected. The original dataset comprised 11 sets of feature data, including wind speed, temperature, humidity, wind direction, rainfall, total irradiance, scattering irradiance, received functional quantity, and average current phase (denoted as T_a , $a = 1, \dots, 11$), along with the corresponding PV power data (denoted as P). Failure to filter the feature data results in an excessive number of input dimensions for the predictive model, thereby increasing the model complexity and significantly affecting the ultimate training efficacy. To address this concern, the correlation between the feature data and the associated PV power was quantified using Pearson's correlation coefficients. Consequently, features with a high degree of correlation were selected for use in the final predictive model.

The Pearson correlation coefficient (r) is defined as:

$$r = \frac{\sum_{i=1}^b (T_{ai} - \bar{T}_a)(P_i - \bar{P})}{\sqrt{\sum_{i=1}^b (T_{ai} - \bar{T}_a)^2} \sqrt{\sum_{i=1}^b (P_i - \bar{P})^2}} \quad (22)$$

where r is the Pearson correlation coefficient, T_a and P are the individual feature data and PV power, respectively, and

b is the number of data points. The computed results of r ranged between 0–1, with values closer to 1 indicating a higher inter-dataset correlation. Using (22), the correlation coefficients between the 11 sets of feature data and PV power can be computed separately, thereby enabling the selection of features with correlation values closest to 1 as the ultimate feature dataset.

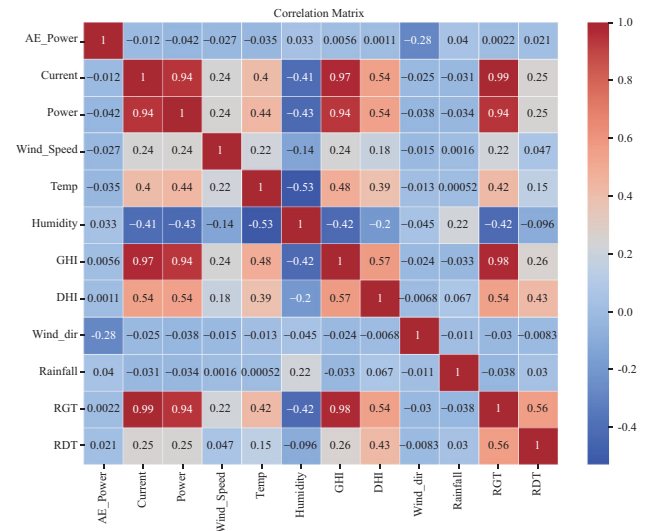


Fig. 5 Heatmap of the correlation analysis

As shown in Fig. 5, the GHI, Current, and RGT features exhibited a very strong relationship with Power, followed by DHI. Humidity and Temp exhibited an approximate correlation of 44% correlation with Power, whereas the remaining feature variables exhibited lower correlations with Power.

Subsequently, redundant data were eliminated, missing values were filled in, abnormal data were processed, and data smoothing techniques were applied. The function of a PV power system requires ample sunshine and sufficient daylight. Therefore, data pertaining to night-time power generation were excluded. Missing daytime values were addressed using the k-nearest neighbors mean method [37]. Anomalies were identified and removed using the isolation forest approach [38] for specific aberrant data points, and periods of excessive fluctuation in the original PV power data, likely influenced by noise, were rectified using the simple moving average method for data smoothing.

Finally, all the data were normalized. The different feature data, T_a and power data P possessed distinct dimensional scales and units that directly influenced the analytical outcomes. To mitigate the dimensional impact and ensure data uniformity, the min-max normalization method was applied to map all data within the range [0,1]:

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (23)$$

where X_{new} and X are the normalized and original data, respectively, and X_{max} and X_{min} are the maximum and minimum original data values, respectively. By individually transforming the numerical values of various feature data Ta and power data P into the range of 0–1 using (23), the dimensional influence could be mitigated. This approach yielded a more stable and robust model that could better accommodate numerical disparities across features.

Through these operations, highly correlated feature data and effective PV power data were obtained, thereby preparing the groundwork for generating model data vectors.

4.1.2 Data partitioning

The PV power prediction algorithm based on the CNN-LSTM-attention model comprised two main parts: model training and model prediction. Therefore, the original dataset Q must be divided into three non-overlapping parts: training, validation, and test sets. The training set was used to train the model parameters, and the validation set was used for model performance evaluation and parameter tuning. The ratio of the training to validation sets was 9:1. The test set was used for the model prediction. The predictive capability and accuracy of the model were assessed and validated by measuring its performance using previously unknown data. In this study, data from July 1, 2020, to December 31, 2020, were used as the training set for model training, whereas data from July 1, 2021, to July 20, 2021, served as the validation set for model verification. To assess the predictive performance of the model, two typical data periods were selected: smooth and fluctuating. The smooth data period was characterized by stable environmental and meteorological conditions, resulting in relatively steady photovoltaic power generation. Conversely, the fluctuating data period was marked by significant changes in the environmental and meteorological conditions, leading to substantial variability in photovoltaic power generation. These periods were used to validate the prediction accuracy, stability, and superiority of the models.

4.1.3 Results of input vector generation

Because PV power generation is significantly influenced by factors such as GHI and $Temp$, seven sets of historical feature data, namely, $Power$, $Current$, GHI , $Humidity$, DHI , RGT and $Temp$, were introduced as neural network input vectors. In addition, because $Power$ is affected by current factors, such as weather, two sets of real-time data, including current $Temp$ and $Humidity$, were included as part of the input vector. Thus, the neural network input vector comprised eight sets of data. The shape of the final model input vector $\mathbf{I}$ was $N \times 288 \times 9$.

4.1.4 Initial model network structure and parameter configuration

Table 3 presents the network architecture parameters of the unoptimized CNN-LSTM-Attention model. The values of the hyperparameters such as $filters$, $units$, $drop_C$, and $drop_L$ were obtained based on empirical knowledge.

Table 3 Network structure (layers) and parameter settings

Input	CNN	LSTM	Attention	Fully Connected
Number of layers: 1				
Number of nodes: 8	$filters$ 64			Number of nodes: 1
	Activation Function: ReLu	$units$ 64	Activate Function: softmax	
	$drop_C = 1$			Activation Function: linear

The model training was conducted using the Python Keras framework. The training process involved 30 epochs, which enabled the model to undergo 30 parameter updates and effectively learn the dataset features. A batch size of 50 was established by dividing the dataset into 50 equal batches. To reduce memory consumption and accelerate training, each batch was sequentially inputted into the neural network for training. The Adam optimizer was chosen, which adaptively adjusted the learning rate based on the first and second moment estimates of the gradients, resulting in faster convergence than traditional optimizers. The MAE was used as a loss function (21).

4.1.5 Model performance evaluation metrics

To quantitatively evaluate the predictive performance, the coefficient of determination (goodness of fit), mean absolute percentage error (MAPE), MAE, root mean square error (RMSE), and training time were adopted as evaluation metrics for the final predictions. The coefficient of determination assesses the ability of the model to explain the variance in the observed data, and ranges as 0–1, with values closer to 1 indicating a better model fit. The MAPE quantifies the accuracy of predictions by computing the average ratio of the absolute error of each forecasted value to its corresponding actual value. A smaller MAPE indicates a higher accuracy of the predicted values and better performance of the model. The MAE measures the average magnitude of errors in a set of predictions without considering their direction. Equation (21) is used to compute the MAE. RMSE measures the deviation between the observed and true values, with the square root applied to ensure that the error is on the same scale as the data, thus

better depicting the predictive accuracy. The formulae are as follows:

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (24)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \quad (25)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (26)$$

where R^2 is the coefficient of determination, $\hat{y}_i$ is the predicted values, y_i is the true values, and $\bar{y}$ is the mean of the true values.

Owing to variations in device or platform performance, the training times for the models cannot be generalized. To mitigate this effect, we measured the training time of each model relative to that of the LSTM model. The training times of the other models were normalized by the ratio of their training times to that of the LSTM model, allowing for an indirect comparison of the training durations across models. (All model trainings in this experiment were conducted on the same platform.)

4.2 Validation of CNN-LSTM-attention model effectiveness

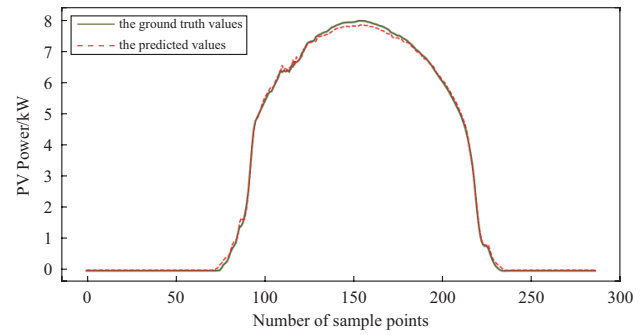
To comprehensively evaluate the final performance of the CNN-LSTM-attention model in predicting the PV power, its predictive effects during smooth and fluctuating periods in the test dataset were evaluated, validating its effectiveness across different timeframes. In addition, under the same dataset conditions, the predictive performances of the LSTM, GRU, CNN-LSTM, CNN-LSTM with autoencoders, and parallel CNN-LSTM-attention models were obtained for comparative analysis to demonstrate the advantages of the proposed methodology.

4.2.1 Prediction results of the proposed model for different time periods

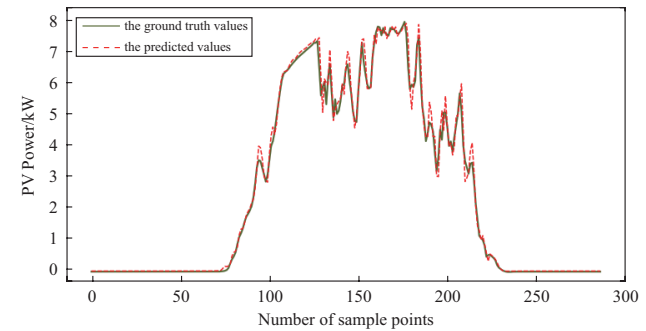
Figures 6(a) and 6(b) show the comparison curves between the predicted and ground truth values of the model during smooth and fluctuating data periods, respectively. Table 4 presents the evaluation metrics for these scenarios.

Table 4 Evaluation of prediction results of the proposed model for different data periods

	Data smoothing	Data fluctuation
Determination coefficient	0.99948	0.99425
MAPE	69.356%	57.723%
MAE	0.05819	0.12049
RMSE	0.07564	0.22127



(a) Prediction results of the proposed model during the data smoothing period



(b) Prediction results of the proposed model during the data fluctuation period

Fig. 6 Comparison curves between the predicted and ground-truth values of the model

When the power data exhibited smoother trends, the predicted curve of the model closely tracked the changes in the ground truth values, capturing minor fluctuations. However, in periods of substantial power fluctuations, the model's predictions followed the general trend of the true values but struggled to accurately capture abrupt changes, exhibiting relatively weaker tracking performance. In terms of the goodness of fit, MAPE, MAE, and RMSE, the model performed better during smooth data periods than during fluctuating periods (Table 4). A comparison between the MAE and MAPE metrics for forecasting results across the two data segmentation periods yielded unique outcomes. The MAE evaluation showed superior performance for smoothed period predictions, whereas the MAPE evaluation indicated poorer performance. This phenomenon can be attributed to the nature of the error metrics; MAE measures the average magnitude of prediction errors, which are highly sensitive to outliers, where larger actual values result in larger errors. During smoothed periods, stable power values led to smaller error magnitudes, and hence, a lower MAE. Conversely, during fluctuating data periods characterized by larger variations in power values, the model struggled to track rapid changes, resulting in more high-magnitude errors and consequently higher MAE values. In contrast,

MAPE, measures the average absolute percentage error relative to the actual values, emphasizing the sensitivity to error proportions relative to actual values. Despite the larger absolute errors during data fluctuations, the proportional error relative to the high actual values tended to be smaller, potentially yielding better MAPE performance during these periods. This underscores the significant impact of power data fluctuations on the model forecasting outcomes.

4.2.2 Comparative analysis of predictive results for different time periods for the various models

Figures 7 and 8 illustrate the comparative prediction curves for the LSTM, GRU, CNN-LSTM, CNN-LSTM with autoencoders, and parallel CNN-LSTM-attention models during the data smoothing and fluctuation periods, respectively. Table 4 lists the evaluation metrics for the prediction results of these scenarios. Observing the evaluation of the five metrics on the predictions of these models, it is evident that our model exhibited superior performance in terms of goodness of fit, MAE, and RMSE during periods of data smoothness and fluctuations (Table 5). The evaluation results for MAE and RMSE are particularly notable. Our model required the least training time, except for the GRU and LSTM models. Despite the slightly higher goodness of fit and MRE evaluations for the CNN-LSTM with the autoencoder model during smooth data periods, it significantly increased the training time compared with ours. This discrepancy in training efficiency underscores the fact that the CNN-LSTM with the autoencoder model offers less cost-effectiveness compared to ours, achieving only marginal improvements in prediction accuracy at nearly four times the training duration. Furthermore, compared to

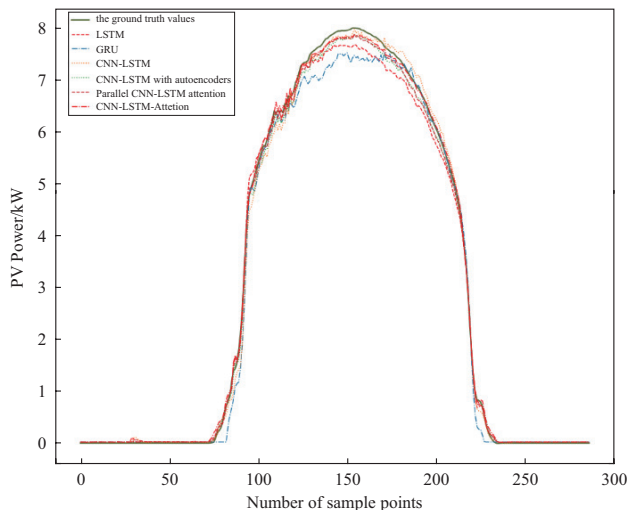


Fig. 7 Comparison of prediction results of different models during the data smoothing period

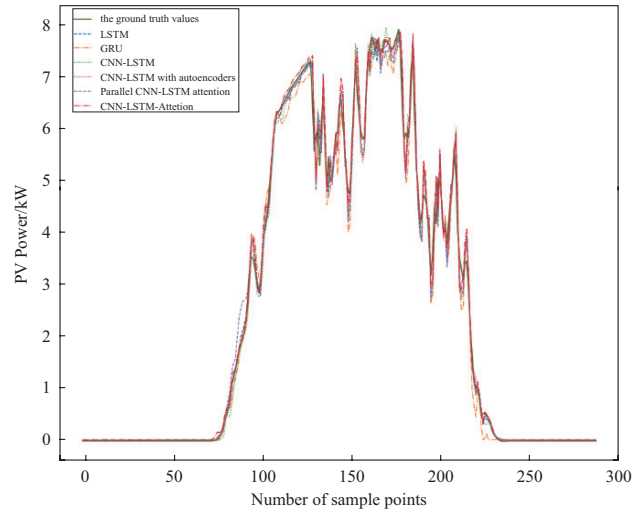


Fig. 8 Comparison of prediction results of different models during the data fluctuation period

the Parallel CNN-LSTM-attention model, despite slightly longer training times for our model, evaluations of goodness of fit, MAE, and RMSE consistently favored our model during both smooth and fluctuating data periods. Overall, our model demonstrated significant superiority in predicting photovoltaic power under acceptable training time costs, regardless of data smoothness or fluctuations.

Table 5 Evaluation of the prediction results of different models during the different data periods

	Determination coefficient	MRE	MAE	RMSE	Training time
Data smoothing					
LSTM	0.99716	40.419%	0.12116	0.17748	1
GRU	0.99376	371.93%	0.16740	0.26581	0.89552
CNN-LSTM	0.99801	32.735%	0.09393	0.14845	1.06536
CNN-LSTM with autoencoders	0.99876	39.335%	0.08706	0.11737	3.99570
Parallel CNN-LSTM attention	0.99910	44.713%	0.07580	0.09982	1.32298
CNN-LSTM-Attention	0.99948	69.355%	0.05819	0.07564	1.50024
Data fluctuation					
LSTM	0.99423	62.546%	0.12600	0.22282	1
GRU	0.99165	54.548%	0.16546	0.26803	0.89552
CNN-LSTM	0.99483	69.166%	0.11500	0.21083	1.06536
CNN-LSTM with autoencoders	0.99425	66.002%	0.12696	0.22239	3.99570
Parallel CNN-LSTM attention	0.99290	50.912%	0.13690	0.24709	1.32298
CNN-LSTM-Attention	0.99431	57.723%	0.12049	0.22127	1.50024

Special attention should be paid to the hyperparameter settings of the models used for the predictive performance comparison. The LSTM model comprised one layer with a hidden size of 64. The GRU model also had one layer of GRU units with a hidden dimension of 64. The CNN-LSTM model incorporated one CNN layer with 64 filters of size 1, followed by an LSTM layer with one layer and a hidden size of 64. For the CNN-LSTM with the autoencoder model, there were three CNN layers with filter sizes sequentially set to 32, 64, and 128, each with a kernel size of 3. The LSTM layer in this model had one layer with a hidden size of 64. The parallel CNN-LSTM-attention model included two CNN layers with filter sizes sequentially set to 32 and 64 and kernel sizes of 1 and 3. In addition, this model incorporated two pooling layers with window sizes of 2 each. The LSTM layer in the parallel CNN-LSTM-attention model comprised one layer with a hidden size of 32. Notably, all aforementioned parameter settings were empirically determined.

4.3 Validation of Bayesian optimization method effectiveness

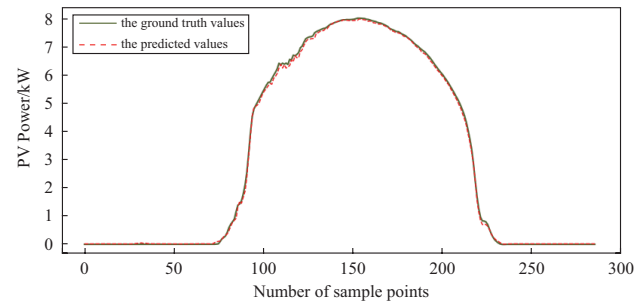
The proposed optimized model was used to predict power data during smooth and fluctuating data periods. The predictions obtained from the optimized and non-optimized models were compared, thereby validating the superiority of the Bayesian optimization-based CNN-LSTM-attention model for PV power prediction.

4.3.1 Results of hyperparameter optimization

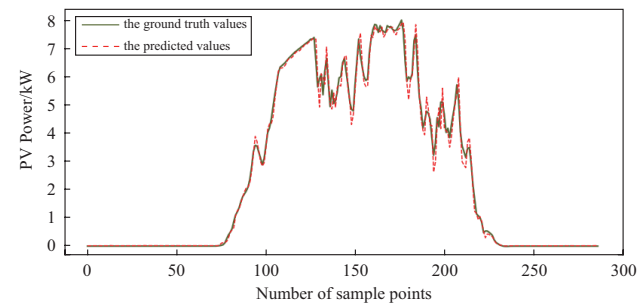
The model hyperparameters were *filters*, *units*, *drop_C*, *drop_L*, and *batch_size*. The final optimization results were 80, 64, 0.5969, 0.1392, and 60, respectively.

4.3.2 Predictive results of the optimized model at different time periods

Figures 9(a) and 9(b) present the comparison curves of the predictions of the optimized model with the actual values during the data smoothing and fluctuation periods. Table 5 lists the evaluation metrics of the optimized model. In both data periods, the prediction curves of the optimized model closely tracked the variations in the actual values with minimal errors. The evaluation results of the R^2 coefficient of determination, MAE, and RMSE indicate that the predictive accuracy during periods of data fluctuation lagged slightly behind that of the data smoothing periods, yet still demonstrated commendable performance (Table 6). Overall, the prediction accuracy of the PV power improved in both data periods, with a notable reduction in discrepancies. These findings suggest that the optimized model enhanced predictive capabilities during data smoothing periods and accurately captured power data fluctuations.



(a) Prediction results of the optimized model during the data smoothing period



(b) Prediction results of the optimized model during the data fluctuation period

Fig. 9 Comparison curves between the predicted and ground-truth values of the optimized model

Table 6 Evaluation of prediction results of the optimized model for different data periods

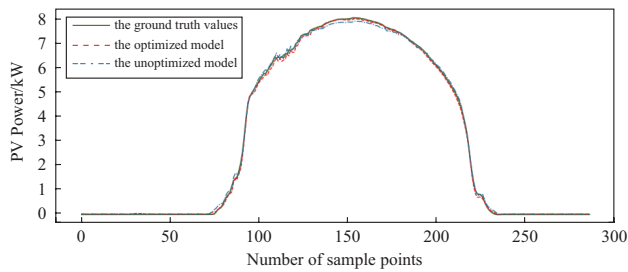
	Data smoothing	Data fluctuation
determination coefficient	0.99961	0.99431
MAPE	42.028%	26.666%
MAE	0.04673	0.11927
RMSE	0.06555	0.22126

4.3.3 Comparative analysis of predictive results before and after model optimization at different time periods

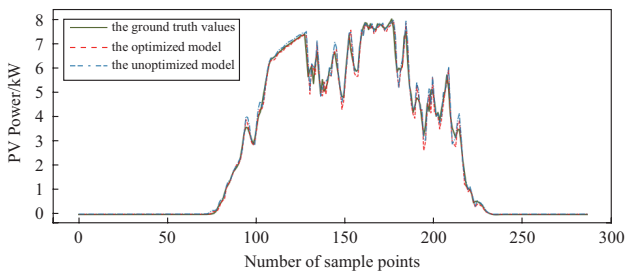
Figures 10(a) and 10(b) illustrate a comparison between the predicted values before and after model optimization during the data smoothing and fluctuation periods, respectively. Table 7 lists the evaluation metrics for the prediction results. Notably, after Bayesian hyperparameter optimization, the predicted values exhibited a markedly improved fit to the actual PV power data compared to the predictions before model optimization. During the data smoothing period, compared with the unoptimized model, the optimized model exhibited reductions in MAPE, MAE, and RMSE by 39.40%, 19.69%, and 13.34%, respectively (Table 7). Similarly, during the data fluctuation period, the optimized model exhibited a decrease in MAPE of 53.80% and MAE of 1.01%, with a relatively smaller reduction in

RMSE. These results indicate that compared to the non-optimized model, the optimized model showed significant improvements in predictive performance, as measured by the MAPE, MAE, and RMSE. Particularly, during periods of data volatility, the optimized model exhibited a 53.8% decrease in MAPE, reflecting a substantial enhancement. This suggests a marked improvement in the ability of the optimized model to track fluctuating data.

In conclusion, under acceptable time-cost trade-offs, the CNN-LSTM-attention PV power prediction model proposed in this study demonstrated superior predictive accuracy compared to the LSTM, GRU, CNN-LSTM, CNN-LSTM with autoencoders, and parallel CNN-LSTM-attention models. Leveraging Bayesian optimization for hyperparameter tuning further enhanced the predictive capability of the model for the PV power data. In addition, it maintained robust prediction stability even under significant fluctuations in the data.



(a) Comparison of prediction results during the data smoothing period before and after model optimization



(b) Comparison of prediction results during the data fluctuation period before and after model optimization

Fig. 10 Comparison of prediction results before and after model optimization

Table 7 Evaluation of prediction results during the different data periods before and after model optimization

	Optimized model	Original model
Data smoothing		
R^2 determination coefficient	0.99961	0.99948
MAPE	42.028%	69.356%
MAE	0.04673	0.05819
RMSE	0.06555	0.07564

continue

	Optimized model	Original model
Data fluctuation		
R^2 determination coefficient	0.99431	0.99425
MAPE	26.666%	57.723%
MAE	0.11927	0.12049
RMSE	0.22126	0.22127

5 Conclusions

In response to the limitations observed in existing research on PV power prediction, which include low model information capture and challenges in tuning model hyperparameters, this study proposed a Bayesian optimization-based CNN-LSTM-attention method for PV power prediction.

1) In terms of predicting power data during smoothing and fluctuation periods within acceptable time costs, the proposed CNN-LSTM-attention model demonstrated superior predictive accuracy compared with the LSTM, GRU, CNN-LSTM with autoencoders, and parallel CNN-LSTM-attention models. Following the Bayesian algorithm optimization of the hyperparameters, the model significantly improved the prediction accuracy and demonstrated stability during data fluctuation. Beyond PV power prediction, it exhibited advantages in medical image analysis, personalized recommendations, and time-series analyses, indicating its potential for wider application and enhancement.

2) The study improved the PV power prediction accuracy, but encountered challenges such as lengthy hyperparameter optimization runtimes and uncertainty regarding the global optimum. Future research should further explore these aspects to gain a comprehensive understanding.

Acknowledgments

This work was supported by the State Grid Science & Technology Project (5400-202224153A-1-1-ZN).

Declaration of Competing Interest

We declare that we have no conflict of interest.

References

- [1] iNCi M (2019) Design and analysis of dual level boost converter based transformerless grid connected PV system for residential

- applications. Proceedings of the 4th IEEE International Conference on Power Electronics and their Applications. 2019 in Elazig, Turkey, pp: 1-6
- [2] Celik O, Tan A, Inci M, et al. (2020) Improvement in energy harvesting capability of grid-connected photovoltaic microinverters. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, 1-25
 - [3] Dolara A, Leva S, Manzolini G (2015) Comparison of different physical models for PV power output prediction. *Solar Energy*, 119: 83-99
 - [4] Wang F, Zhang Z Y, Liu C, et al. (2019) Generative adversarial network and convolutional neural network-based weather classification models for day-ahead short-term photovoltaic power forecasting. *Energy Conversion and Management*, 181: 443-462
 - [5] Habib S, Alyahya S, Islam M, et al. (2022) Design and implementation: An IoT-based automated wastewater irrigation system. *Electronics*, 12(1): 28
 - [6] Zuhair M, Shaikh F A, Tanweer W, et al. (2022) Faults feature extraction using discrete wavelet transform and artificial neural network for induction motor availability monitoring—internet of things enabled environment. *Energies*, 15(21): 7888
 - [7] Yang D Z (2019) On post-processing day-ahead NWP forecasts using Kalman filtering. *Solar Energy*, 182: 179-181
 - [8] Muhammad T, Khan A U, Chughtai M T, et al. (2022) An adaptive hybrid control of grid tied inverter for the reduction of total harmonic distortion and improvement of robustness against grid impedance variation. *Energies*, 15(13): 4724
 - [9] Wang Y, Wang J Z, Wei X (2015) A hybrid wind speed forecasting model based on phase space reconstruction theory and the Markov model: A case study of wind farms in northwest China. *Energy*, 91: 556-572
 - [10] Wang J, Zhang N, Lu H (2019) A novel system based on neural networks with linear combination framework for wind speed forecasting. *Energy Conversion and Management*, 181: 425-442
 - [11] Deo R C, Wen X H, Qi F (2016) A wavelet-coupled support vector machine model for forecasting global incident solar radiation using limited meteorological dataset. *Applied Energy*, 168: 568-593
 - [12] Sharifian A, Ghadi M J, Ghavidel S, et al. (2018) A new method based on a Type-2 fuzzy neural network for accurate wind power forecasting under uncertain data. *Renewable Energy*, 120: 220-230
 - [13] Abdel-Nasser M, Mahmoud K (2019) Accurate photovoltaic power forecasting models using deep LSTM-RNN. *Neural Computing and Applications*, 31(7): 2727-2740
 - [14] Zhang J X, Chi Y Y, Xiao L P (2018) Solar power generation forecast based on LSTM. Proceedings of the 9th IEEE International Conference on Software Engineering and Service Science. 2018 in Beijing, China, 869-872
 - [15] Zang H X, Cheng L L, Ding T, et al. (2020) Day-ahead photovoltaic power power-forecasting approach based on deep convolutional neural networks and meta-learning. *International Journal of Electrical Power & Energy Systems*, 118: 105790
 - [16] Han T, Muhammad K, Hussain T, et al. (2020) An efficient deep-learning framework for intelligent energy management in IoT networks. *IEEE Internet of Things Journal*, 8(5): 3170-3179
 - [17] Khan Z A, Hussain T, Ullah F, et al. (2022) Randomly initialized CNN with a densely connected stacked autoencoder for efficient fire detection. *Engineering Applications of Artificial Intelligence*, 116: 105403
 - [18] Khan Z A, Hussain T, Baik S W (2022) Boosting energy harvesting via deep learning-based renewable power generation prediction. *Journal of King Saud University*, 34(3): 101815
 - [19] Kim J, Moon J, Hwang E, et al. (2019) Recurrent inception convolution neural network for multi-short-term load forecasting. *Energy and Buildings*, 194: 328-341
 - [20] Sajjad M, Khan Z A, Ullah A, et al. (2020) A novel CNN-GRU-based hybrid approach for short-term residential load forecasting. *IEEE Access*, 8: 143759-143768
 - [21] Qu J Q, Qian Z, Pei Y (2021) Day-ahead hourly photovoltaic power forecasting using attention-based CNN-LSTM neural network embedded with multiple relevant and target variables prediction pattern. *Energy*, 232: 120996
 - [22] Khan Z A, Hussain T, Ullah A, et al. (2020) Towards efficient electricity forecasting in residential and commercial buildings: A novel hybrid CNN with an LSTM-AE based framework. *Sensors*, 20(5): 1399
 - [23] Li G L, Yang J, Zhou M G (2022) Power prediction of photovoltaic generation based on improved temporal convolutional network. *Laser & Optoelectronics Progress*, 59(8): 480-489
 - [24] Victoria A H, Maragatham G (2021) Automatic tuning of hyperparameters using Bayesian optimization. *Evolving Systems*, 12(1), 217-223
 - [25] Rai A, Shrivastava A, Jana K C, (2023) Differential attention net: Multi-directed differential attention based hybrid deep learning model for solar power forecasting. *Energy*, 263(C): 125746
 - [26] Chung W H, Gu Y H, Yoo S J (2022) District heater load forecasting based on machine learning and parallel CNN-LSTM attention. *Energy*, 246: 123350
 - [27] Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Computation*, 9(8): 1735-1780
 - [28] Bridle J S (1990) Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In: Soulié FF, Hérault J (eds) *Neurocomputing: NATO ASI Series*, vol 68. Springer, Berlin, pp 227-236
 - [29] Pelikan M, Goldberg D E, Cantu-Paz E (1999) BOA: Bayesian optimization algorithm. Proceedings of the Genetic and Evolutionary Computation Conference, Orlando, FL, USA, pp: 525-532
 - [30] Frazier P I (2018) A tutorial on bayesian optimization. Preprint arXiv:1807.02811
 - [31] Shahriari B, Swersky K, Wang Z Y, et al. (2016) Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1): 148-175
 - [32] Bergstra J, Bardenet R, Bengio Y, et al. (2011) Algorithms for

hyperparameter optimization. Proceedings of 24th International Conference on Neural Information Processing Systems. Red Hook, NY, USA, 12-15 Dec., pp 2546-2554

- [33] Wang Z, Zoghi M, Hutter F, et al. (2013) Bayesian optimization in high-dimensions via random embeddings. Proceedings of the Twenty-third International Joint Conference on Artificial Intelligence. Beijing, 2013 in China, pp: 1778-1784
- [34] Snoek J, Larochelle H, Adams R P (2012) Practical Bayesian optimization of machine learning algorithms. Proceedings of the 25th International Conference on Neural Information Processing Systems, Volume 2, 3-6 Dec., Lake Tahoe, NV, USA, pp 2951-2959
- [35] Srinivas N, Krause A, Kakade S, et al. (2010) Gaussian process optimization in the bandit setting: No regret and experimental design. In: Proceedings of the 27th International Conference on Machine Learning. Haifa, Israel, pp: 1015-1022
- [36] DKASC details. 2008. URL: <https://dkasolarcentre.com.au/locations/alice-springs>
- [37] Cover T, Hart P (1967). Nearest-neighbor pattern classification. IEEE Transactions on Information Theory, 13(1): 21-27
- [38] Liu F T, Ting K M, Zhou Z H (2008) Isolation forest. Proceedings of the 8th IEEE International Conference on Data Mining, Pisa, Italy, pp: 413-422

Biographies



Ning Zhou, graduated with a bachelor's degree in Thermal Power from Shanghai Jiao Tong University in 1992, and graduated as a postgraduate from North China Electric Power University in the major of Electrical Engineering in 2012. Has been working at the Electric Power Research Institute of State Grid Henan Electric Power Company since 1999. The main research direction is in the field of safe operation of distribution networks and new energy consumption technology.



Bowen Shang received the B.S. degree from the School of Electrical Engineering, Northeast Electric Power University, Jilin, China, in 2018, and the M.S. degree from the School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, China, in 2021. She currently works with the State Grid Henan Electric Power Research Institute, Zhengzhou,

China. Her current research interests include novel distribution networks, image recognition, and power electronic fault diagnosis technology.



Mingming Xu, graduated with a bachelor's degree in Electrical Engineering and Automation from Xi'an Jiaotong University in 2007, and graduated with a doctorate in Electrical Theory and New Technology from the Institute of Electrical Engineering of China Electric Power Research Institute in 2015.

He has been working at the Electric Power Research Institute of State Grid Henan Electric Power Company since 2015. At present, he mainly engages in research work in the aspects of distribution network fault handling technology and the development and grid connection of distributed new energy.



Lei Peng, graduated with a Bachelor's degree in Power System and Automation from Zhengzhou University of Technology in 2000; and graduated with a Master's degree in Electrical Engineering from Zhengzhou University in 2011. Since 2012, he has been working in the Equipment Department of State Grid Henan Electric Power Company. At

present, he is mainly engaged in distribution network operation and maintenance technology and equipment management work.



Yafei Zhang received the B.S. degree from the School of Electrical and Information Engineering, Tianjin University in 2018, and the M.S. degree from Department of Electrical Engineering, Tsinghua University in 2021. His research interests include renewable energy generation and smart grids.

(Editor Zedong Zhang)