IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 31, 2019, accepted August 16, 2019, date of publication August 21, 2019, date of current version August 30, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2936597

PV Power Prediction Based on LSTM With Adaptive Hyperparameter Adjustment

MINKANG CHAI[1], FEI XIA[1], (Member, IEEE), SHUOTAO HAO[2], DAOGANG PENG[1], CHENGGANG CUI[1], AND WEI LIU[3]

[1]College of Automation Engineering, Shanghai University of Electric Power, Shanghai 200090, China
[2]Fangshan Power Supply Company, Beijing Power Company, State Grid, Beijing 102488, China
[3]Haining Chinaust Plastics Piping System Company Ltd., Haining 314415, China

Corresponding author: Fei Xia (xiafeiblue@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFE0100900, and in part by the National Natural Science Foundation of China (NSFC) under Grant 51607111, Grant 71690234, and Grant 71871160.

ABSTRACT The randomness, volatility, and intermittence of solar power generation make it difficult to achieve the desired accuracy of PV output-power prediction. Therefore, the time learning weight (TLW) proposed in this paper is used to improve the time correlation of the LSTM network. The Fusion Activation Function (FAF) is used to resolve gradient disappearance. Learning Factor Adaptation (LFA) and Momentum Resistance Weight Estimation (MRWE) are used to accelerate weight convergence and improve global search capabilities. Finally, this paper synthesizes the improvement and proposes the AHPA-LSTM model to stabilize the convergence domain. Using actual data verification, the δ_{MAPE} indicator of the improved model is only 2.85% on a sunny day, 5.92% on a cloudy day, 7.71% on a rainy day, and only 5.8% on average. Therefore, the AHPA-LSTM model under full climate and climatic conditions has a good predictive effect which is generally applicable to the prediction of ultra-short-term PV power generation.

INDEX TERMS Photovoltaic output power, ultra-short-term prediction, long short term memory (LSTM), time weight decoupling, adaptive hyperparameter adjustment.

NOMENCLATURE

PV	photovoltaic
HEPV	hybrid energy-storage PV power generation system
DNN	deep neural network
LSTM	long short term memory neural network
Bi-LSTM	bidirectional LSTM
RNN	recurrent neural network
CELL	LSTM cell memory channel
ReLU	rectified linear unit
TLW	time learning weights
FAF	fusion activation function
SVR	support vector regression
MLE	maximum Lyapunov exponent
MR	momentum resistance method
MRWE	momentum resistance method with weight estimate
SDG	stochastic gradient descent mathod
Adam	adaptive moment estimation mathod
LFA	learning factor adaptation method
AHPA	adaptive hyperparameter adjustment
RE	clear-sky relative error
$\delta_{\mathbf{MAPE}}$	mean absolute percent error
$\delta_{\mathbf{QRER}}$	upper and lower quartile relative percentage error range
$\delta_{\mathbf{MBE}}$	mean bias error
$\delta_{\mathbf{RMSE}}$	root mean squared error
$\delta_{\mathbf{NRMSE}}$	normalized root mean squared error
$\delta_{\mathbf{THI}}$	time horizon-invariant metric
$\delta_{\mathbf{PVV}}$	photovoltaic variability metric
$\delta_{\mathbf{RATIO}}$	time horizon-invariant ratio
$\delta_{\mathbf{REUQ}}$	the relative error of the upper quartile
$\delta_{\mathbf{RELQ}}$	the relative error of the lower quartile

The associate editor coordinating the review of this article and approving it for publication was Zhiyi Li.

I. INTRODUCTION

Photovoltaic power generation prediction can provide a reference for power grid dispatching and power station maintenance, which is conducive to grid security and economical operation of PV power plants [1]. At the same time, the power grid dispatching center also needs to understand the power

generation capacity of the PV power plant and formulate a scheduling strategy to meet customer needs [2]. PV forecast can be classified as ultra-short-term, short-term, and medium-long-term, with respective time-frames of up to one hour, one day, and one month to one year [3]. Medium- and long-term (>24 hours) solar forecasts are useful for energy resource planning and scheduling. Whereas intraday short-term solar forecasts, i.e., half-hour, or even five-minute, are useful for load tracking and pre-scheduling, which reduce the need for frequency control in actual environment [4]. Moreover, excessive or insufficient PV power output will affect the safe and reliable operation of the grid, which limits the application of large-scale PV systems connected to the grid [5], [6]. Therefore, it is necessary to establish an accurate ultra-short-term model to predict the output power of the PV modules in a timely and accurate method.

PV output power predictions can be divided into direct and indirect forecasts. The indirect method is to predict solar irradiance based on the functional relationship between irradiance and PV output power. A number of indirect methods for predicting PV power are presented in reference [7], which generally use irradiance as a medium. The biggest disadvantage of these methods is that after many predictions, the overlay error hinders the improvement of PV prediction accuracy. Furthermore, the direct method is to predict the detailed historical data of the PV power plant by means of statistics or artificial intelligence. Wolff *et al.* [8] proposed to apply SVR to small-scale PV power generation, and use real-time climate data and future meteorological data for prediction. Although SVR has good portability compared with physical methods, the prediction accuracy needs to be improved. Marquez *et al.* [9] and Lorenz *et al.* [10] proposed direct prediction using remote sensing methods and numerical weather prediction models($\delta_{MAPE} \approx 10\%$ - 20%). Although satellite imaging predictions can achieve high prediction accuracy, this approach requires access to satellite cloud maps and large climate databases, greatly increasing the cost of microgrid power generation.

To reduce forecasting costs, Zhang *et al.* [11] used various surface measurements, i.e., meteorological data (such as surface temperature, global horizontal irradiance, and diffuse reflection horizontal irradiance) and astronomical dates (such as solar time and earth declination) as influencing factors ($\delta_{NRMSE} \approx 13\%$ - 29%). Vaz *et al.* [12] used a nonlinear autoregressive model with external inputs, including local meteorological data and observation data from neighboring PV systems as input ($\delta_{NRMSE} \approx 9\%$ - 25%), which was much more economical than the satellite data model. Espinosa-Gavira *et al.* [13] uses the snapshot of the radiation field collected by the wireless solar sensor as input to perform short-term PV prediction. The low cost and portability of the measurement equipment make him an appropriate choice for small and medium-sized PV power plants.

Compared with the experimental method, the probability space-time model of PV power generation prediction is relatively novel. Scolari *et al.* [14] used uncertainty estimation and power reduction strategies to eliminate the unstable state of short-term PV systems. Agoua *et al.* [15] provided a complete PV prediction model (0-6 hours) by establishing a probability density function and an input variable automatic selection technique. However, the uncertainty prediction depends on the probability distribution, and the output range has a larger room for improvement than the ordinary prediction method. Sheng *et al.* [16] proposed an innovative method using a weighted Gaussian process regression method, so that data samples with higher outliers have lower weights and smaller effects on inputs. However, compared to the Gaussian regression process, the nonlinearity of the neural network is superior. Ospina *et al.* [17] used the hybrid wavelet LSTM-DNN structural model to solve the nonlinear prediction problem of PV output ($\delta_{NRMSE} \approx 6.6\%$ - 11.7%). However, through experiments, it can be found that the stability of multi-step prediction needs to be improved. Therefore, based on the traditional model, an enhanced LSTM depth neural network is proposed for prediction.

The LSTM model proposed by Hochreiter and Schmidhuber [18] is a time-recurrent neural network, which is a special case of the RNN. Gers *et al.* [19] proposed the addition of a forgetting gate to the LSTM network to enhance its temporal correlation, which is improved by Graves and Schmidhuber [20], Kudugunta and Ferrara [21]. Greff *et al.* [22] compared the LSTM depth network of various structures. LSTM with forgetting gates and peepholes is proposed, which has better generalization capabilities. Srivastava and Lessmann [23] uses LSTM to accurately predict the PV data of 21 locations. A large number of experiments show that LSTM has good performance in PV prediction. However, the following four major challenges in solar ultra-short-term forecasting still exist:

- The LSTM network in the existing algorithm can only enhance the temporal correlation of the prediction model by increasing the input dimension. However, excessive input data dimensions can undermine the predictive power of traditional LSTM networks.
- The complex structure of deep neural network and excessive matrix operation seriously affect the training speed of prediction model. Especially for high latitude PV ultra-short-term data, the original LSTM network can not meet the speed requirements of ultra-short-term prediction.
- With the deep cycle of the neural network, the algorithm may fall into local extremum, and the update weight will gradually disappear. This will destroy the accuracy of PV prediction.
- Deep learning networks are very sensitive to hyperparameters. Once the hyperparameters are set inappropriately, the predicted output will produce high frequency oscillation. Ultra-short-term prediction model has very high requirements for the stability of prediction data, so it is necessary to propose an improved scheme to stabilize the output.

In order to solve the above challenges, Zheng *et al.* [24] proposed a short-term prediction method based on MLE according to the high correlation of PV data with time and season. Although deep neural networks are not used to improve the nonlinearity of the model, the model constructs a time series of PV power generation under different meteorological conditions, which enhances the attention of time factors. Han *et al.* [25] suggested using copula function to reduce the dimension of input data in LSTM network. Obviously, the output predicted by traditional LSTM networks with fixed hyperparameters is unstable ($\delta_{NRMSE} \approx 15\%$). Therefore, referring to the concept of time factor and the shortcomings of fixed parameter neural networks, an adaptive hyperparameter LSTM network is proposed in this paper. The algorithm improves the way in which time factors are associated with traditional models. It aims at obtaining the most accurate results with fewer iterations and at the same time stabilizing the predicted output results. The main contributions of this study are as follows:

1) The problem of excessive dimension of input data is solved by adding TLW, and a fast prediction model with super memory is established.
2) The global extremum is obtained by constructing FAF and MRWE, which solves the problem of gradient disappearance of deep neural network and ensures the accuracy of prediction results.
3) The training speed of deep neural network is accelerated by constructing LFA, and the high efficiency of prediction results is guaranteed.
4) Through the synthesis of adaptive hyperparameters and the model switching, an AHPA algorithm is proposed to ensure the convergence region of the predicted output and stabilize the prediction results.
5) Based on the actual data of China's power grid PV power plant in Zhejiang Province, the improved model was rigorously evaluated, which proved the versatility of the method under various weather conditions.

The structure of this paper is as follows: Section II proposes some improved algorithms based on the challenges of PV ultra-short-term prediction and applies them to the traditional LSTM model. Section III introduces the prediction-accuracy evaluation indicators used in this paper. In Section IV, the proposed LSTM method is verified by actual PV data. Section V summarizes the paper and suggests future research.

II. UPDATE OF PV ULTRA-SHORT-TERM FORECASTING MODEL

In order to solve the challenges of PV ultra-short-term forecasting, the main structure of this section is as follows: Section II-A introduces the overall framework of the AHPA-LSTM model. Section II-B proposes a TLW structure based on time correlation improvement and optimizes the LSTM single neuron. In Section II-C, to solve the problem of LSTM gradient disappearance, an improved FAF activation function is proposed. Section II-D proposes an optimized search algorithm LFA based on ultra-short-term prediction, which improves the global search ability and learning speed of the original network. Section II-E combines the above proposed algorithm and stabilizes the convergence ability of the model, and proposes the AHPA algorithm.

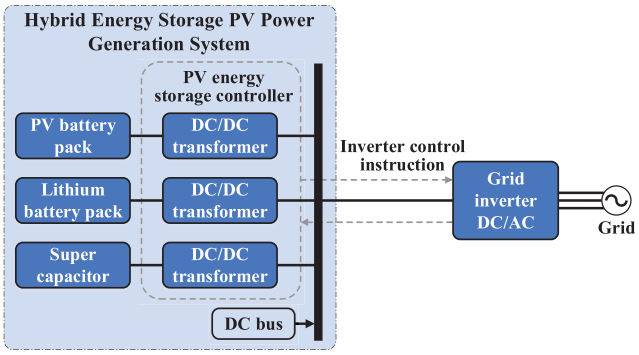


FIGURE 1. Topological structure of HEPV.

A. OVERVIEW OF THE AHPA-LSTM MODEL FRAMEWORK

In order to maximize the economic benefits of electricity, PV power plants generally use FIG. 1 for layout [26]. However, the high cost of lithium batteries limits the number of charge and discharge cycles. It is necessary to establish accurate prediction methods to track PV fluctuations and formulate a reasonable scheduling strategy to extend the life of lithium batteries. Nevertheless, the traditional LSTM network has not developed well in terms of stability, prediction accuracy and prediction speed. Therefore, this section is dedicated to establishing an improved LSTM network to accurately monitor the trend of ultra-short-term PV power generation.

The traditional LSTM has problems in PV ultra-short-term power prediction, such as slow learning speed, falling into local extrema, gradient disappearance, and lack of time correlation. For ultra-short-term power prediction, we need to improve the prediction accuracy as much as possible in a short period of time. On the other hand, LSTM deep neural networks usually have several hyperparameters, i.e., learning factors and penalty factors, in the machine learning process. The hyperparameter-values do not various with time, which limits generalization of the model and affects the improvement of prediction accuracy. Therefore, based on the basic LSTM neural network, this paper uses TLW to enhance the time factor of PV input data. FAF is used to improve the gradient disappearance of ultra-short-term prediction. MRWE is used to avoid the local minimum problem. The LFA is used to estimate the speed of weights, so as to accelerate the convergence of weights. Finally, based on the above improvements, an AHPA-LSTM model is proposed for ultra-short-term PV prediction, as shown in FIG. 2.

In this figure, the forecast input data are PV maximum output power, PV module back surface temperature, Bevel irradiance, Relative humidity and other meteorological factors at the current time, and time factors such as year, month

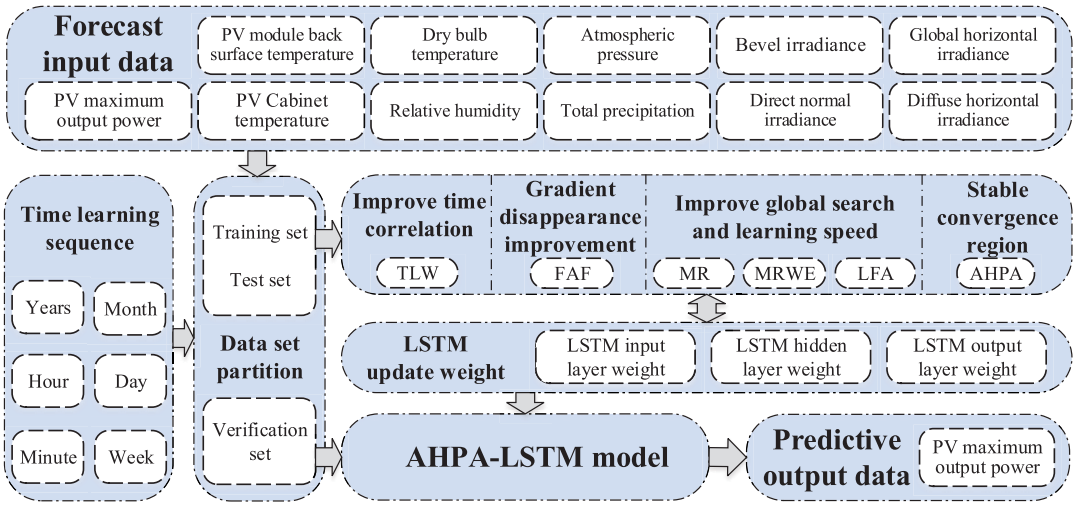


FIGURE 2. PV ultra-short-term forecasting model framework.

and day. The output data is the PV maximum output power of the next 5 minutes. The sampling frequency of ultra-short-term PV training data is 5 minutes.

B. DECOUPLING OF TIME CORRELATION

As a deep neural network, LSTM has more hidden layers than an original neural network, which exploits the long-term dependencies in depth with limited data samples. Each hidden layer contains a certain number of LSTM neuron units, and each contains a CELL that tries to store information in the previous iteration [27]. The main idea is to use special neurons to store and transmit information over a long period to

- obtain permanent memories,
- capture long-term dependencies in an easier manner,
- slow down the rate of information degradation,
- increase the preponderance of the calculation of deepness.

Specifically, the LSTM neural network adds three special gates to the RNN, i.e., input gate, forgetting gate, and output gate, to avoid dependency problems [28]. However, the LSTM network itself does not emphasize the attention to seasonal factors. Obviously, meteorological data has a high regularity within a certain time span, and the illumination radiation also has a specific trend. The original LSTM network can only integrate time information, i.e., year, month and day, with meteorological data and put it into the neural network for prediction. This general approach to all forecasting methods can only increase the correlation between input data and time factor. But most neural networks have high requirements for the dimensions of the input data. Blind overlaying of data is likely to destroy the accuracy of the original model. In addition, since the PV output changes with the amount of cloud in the local climate, it is highly random and there is no way to describe it in a quantitative way. Therefore, we propose a new method to improve the

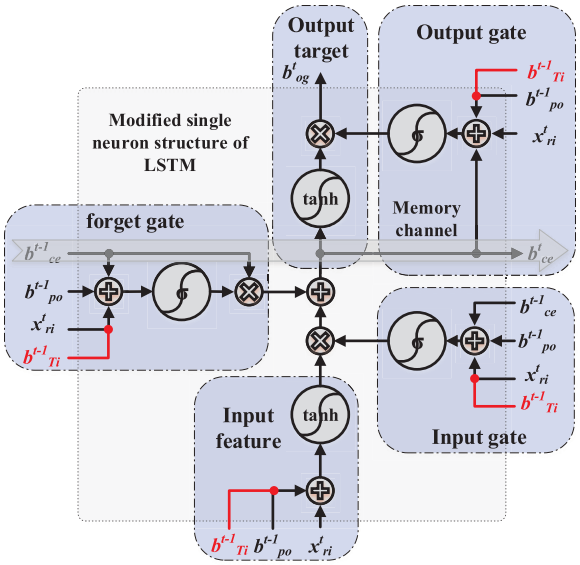


FIGURE 3. Structural variants of LSTM single neuron.

temporal correlation of predictive models to enhance the time correlation prediction of LSTM, as shown in FIG. 3.

In this figure, we can see 'tanh' denotes the activation function as $tanh(x)$, σ denotes the activation function as $sigmoid\ (x)$, '+' denotes the summation of the corresponding elements of two signal matrices, and '×' denotes the inner product of two signal matrices. b^t_{po} is the output matrix of other units , b^t_{ce} is the output matrix of other memory channels, b^t_{Ti} is the input matrix of time weights, x^t_{ri} is the input matrix of the unit, t is the current moment, and $t-1$ is the previous moment.

FIG. 3 adds the TLW to the traditional LSTM model, which enables the LSTM model to increase the consideration of time correlation while deep training. The original weights are divided into climate weights and time weights.

Therefore, TLW-LSTM is better able to distinguish between time information and climate information than ordinary LSTM. They use powerful nonlinear learning capabilities to solve quantitative representations of time and climate models. This structure makes LSTM more sensitive to partition modeling and detail mastery. The improved LSTM structure which added TLW has been marked red in FIG. 3, and the improved recursion equations are as follows, the input gate are

$$a_{ig}^{t} = x_{ri}^{t} \cdot w_{riig} + b_{ce}^{t-1} \cdot w_{ceig} + b_{po}^{t-1} \cdot w_{poig} + b_{Ti}^{t-1} \cdot w_{Tiig}, \quad (1)$$

$$b_{ig}^{t} = f\left(a_{ig}^{t}\right). \quad (2)$$

The improved forget gate are

$$a_{fg}^{t} = x_{ri}^{t} \cdot w_{rifg} + b_{ce}^{t-1} \cdot w_{cefg} + b_{po}^{t-1} \cdot w_{pofg} + b_{Ti}^{t-1} \cdot w_{Tifg}, \quad (3)$$

$$b_{fg}^{t} = f\left(a_{fg}^{t}\right). \quad (4)$$

The improved memory channel are

$$a_{ce}^{t} = x_{ri}^{t} \cdot w_{rice} + b_{po}^{t-1} \cdot w_{poce} + b_{Ti}^{t-1} \cdot w_{Tice}, \quad (5)$$

$$b_{ce}^{t} = b_{fg}^{t} \cdot b_{ce}^{t-1} + b_{ig}^{t} \cdot g\left(a_{ce}^{t}\right). \quad (6)$$

The improved output gate are

$$a_{og}^{t} = x_{ri}^{t} \cdot w_{riog} + b_{ce}^{t} \cdot w_{ceog} + b_{po}^{t-1} \cdot w_{poog} + b_{Ti}^{t-1} \cdot w_{Tiog}, \quad (7)$$

$$b_{og}^{t} = f\left(a_{og}^{t}\right). \quad (8)$$

The output target are

$$y_{po}^{t} = w_{popo} \cdot b_{po}^{t} = w_{popo} \cdot b_{og}^{t} \cdot g\left(b_{ce}^{t}\right), \quad (9)$$

where w_{Ti} is the time weight, w_{popo} is the output weight, y_{po}^{t} is the predicted value of the output, $f(x)$ and $g(x)$ are the activation function.

As can be seen from Eq. (1) - (9), the TLW-LSTM network is more sensitive to time factors than the original neural network. The decoupling of the original data can be performed by the above weight separation, so that the LSTM network can clearly identify the time factor and the meteorological factor. We all know that ultra-short-term PV forecasts are highly dependent on time and future weather data, so this improvement can effectively segment input data, reduce data dimensions, and improve the accuracy of LSTM predictions.

C. IMPROVEMENT OF GRADIENT DISAPPEARANCE

The activation function is an important part of the neural network, and its main function is to provide the learning ability of network nonlinear mapping. The ReLU, also known as a modified linear unit, is a commonly used activation function in artificial neural networks. It usually refers to the nonlinear function represented by the slope function and its variants, as follows:

$$f(n) = \begin{cases} x & x \geq 0 \\ 0 & x < 0. \end{cases} \quad (10)$$

We see from Eq. (10) that when $x < 0$, ReLU is hard saturated, and when $x > 0$, there is no saturation problem. At this point, ReLU can keep the gradient unchanged, thus alleviating the gradient disappearance problem [29]. However, with this training, part of the input will fall into the hard-saturated region (i.e., $x < 0$), resulting in weights that cannot be updated, which will destroy the training accuracy of the original model. In addition, the output of the ReLU function is greater than zero, so the output is not zero-mean, which will cause the latter layer of neurons to receive a non-zero mean output signal (zero drift) of the upper layer as input. Zero drift and hard saturation error will affect the network's convergence. Therefore, based on the one-way saturation and zero-drift problem of the activation function, this paper proposes to combine ReLU and hyperbolic sinusoidal functions to form FAF to ensure the stability of the prediction model. The specific compound equation and derivative equation are

$$f(n) = \begin{cases} x & x \geq 0 \\ \tanh(x) & x < 0, \end{cases} \quad (11)$$

$$f'(n) = \begin{cases} 1 & x \geq 0 \\ 1 - \tanh(x)^2 & x < 0. \end{cases} \quad (12)$$

It can be seen that when $x > 0$, the derivative maintains the gradient of the artificial neural network. When $x < 0$, it filters the different external inputs, which is adaptive. The FAF combines the advantages of the hyperbolic sine and ReLU functions, with soft saturation on the left side and none on the right side. The left soft saturation allows the FAF to be more robust to the input, while the right linear portion enables the mitigation of the gradient disappearance problem.

D. IMPROVEMENT IN GLOBAL OPTIMIZATION

It is not enough to simply add the FAF to improve the gradient disappearance of the AHPA-LSTM model. For the traditional LSTM network, the attraction of local minima cannot be avoided while predicting PV. Therefore, weight estimation and time-varying learning rate are used to enhance the global search ability of the model.

1) GET RID OF LOCAL EXTREMUM ATTRACTION

For error surfaces with multiple extremums, once the gradient method reaches an extremum point, it is assumed that the answer has been found. However, from a physical point of view, if a small ball is located on a rugged ramp, the rolling of the ball will not stop because of a small pit. This is because the quality of the ball itself keeps it in its original state of motion. Refer to the physics equation as follows:

$$F_{inertia} = m \cdot a. \quad (13)$$

For some narrow and long error surfaces, the error obtained by the gradient method will produce an oscillating search

between the valley walls, and never stop. However, from a physical point of view, if a small ball rolls off the edge of a narrow canyon. Due to the air resistance, the kinetic energy of the ball is lost. It is impossible for a ball to bounce back and forth between narrow rock walls without falling into the canyon. The aerodynamic equation is as follows:

$$F_{resistance} = \frac{1}{2} \cdot C \cdot \rho \cdot S \cdot V^2. \tag{14}$$

It can be seen that in physics, inertia and resistance are a pair of opposing forces, but they work together to bring the object back to equilibrium quickly. In order to ensure that the PV prediction model can quickly and stably transition from a non-equilibrium state to an equilibrium state, this paper proposes an MR optimization search algorithm based on analogical physics experience with reference to [30]. In order to find the extreme points of the error surface, the MR algorithm establishes an iterative factor similar to physical inertia and resistance during the iterative process. These two iterative factors promote and constrain each other, and the error of the weight update is no longer abrupt with time iteration, so the global optimal solution can be quickly found. The new weight gradient change rate is balanced by the combination of the momentum and resistance factors, and the weight update is as follows:

$$\begin{aligned} d_i &= (1 - \beta + \mu) \cdot d_{i-1} + \beta \cdot g(\theta_{i-1}) \\ &\quad - \mu \cdot g(\theta_{i-1})^2, \end{aligned} \tag{15}$$

$$\theta_i = \theta_{i-1} - \alpha \cdot d_i, \tag{16}$$

where r is the learning factor, d is the weight update rate, $g(\theta)$ is the gradient of the objective function at θ, β is the momentum factor, and μ is the resistance factor.

Due to the introduction of the momentum factor β, the gradient produces memory. So each calculation has a preorder gradient $g(\theta_{i-1})$ that involves the operation. The weight update speed d updates the gradient of this iteration by memorizing the update amount of a part of the previous iteration. Also, due to the addition of the resistance factor μ, each gradient has a certain positive attenuation. Referring to Eq. (14), the air resistance is proportional to the square of the relative wind speed. A forward attenuation factor $g(\theta_{i-1})^2$ is established, so that weight updating in the iteration process will not be so fast as to produce oscillation, and the stability of the update speed is guaranteed. The inertia component $\beta \cdot g(\theta_{i-1})$ and the resistance component $-\mu \cdot g(\theta_{i-1})^2$ are mutually constrained in the iterative process. The balance of the weight update is maintained at all times. The parameter $(1 - \beta + \mu)$ mainly acts on the current weight update direction, so as not to destroy the uniformity of the weight due to the increase of the inertia component or resistance component, and maintains the balance in the update direction.

According to repeated experiments, the inertia and resistance are the main and secondary component, respectively. Therefore, the momentum factor should generally be 10, or even 100 times the resistance factor to ensure the stability of the iteration. When the updated value of the gradient decrease coincides with the direction of the previous time-update value, the inertial component is enhanced, and the learning ability of the weight can be accelerated. Conversely, when the directions are inconsistent, the inertia component will be weakened, and the oscillation can be suppressed.

2) ACCELERATE WEIGHT CONVERGENCE

The momentum resistance method is improved from the Adam algorithm [31]. Each parameter update direction depends not only on the gradient of the current position but on the direction of the last parameter update. However, according to Eq. (15) - (16), since it is known that the i-th update weight must increase the previous time component $\alpha \cdot (1 - \beta + \mu) \cdot d_{i-1}$, the ($i$-1)-th update weight does not have to use the current position gradient $g(\theta_{i-1})$. We propose that the component of θ_{i-1} is first added to the (i-1)-th $\alpha \cdot (1 - \beta + \mu) \cdot d_{i-1}$, and then continue to the next update. Updating in this way will increase the (i-1)-th estimate at time i-th, and obtain the update compensation of $\alpha \cdot (1 - \beta + \mu) \cdot d_{i-1}$ by calculation to achieve equilibrium quickly. However, updating by this method requires solving a new problem, i.e., obtaining the d_{i-1} component in advance. Since the d_{i-1} component is unknown at time (i-1)-th and cannot be obtained by the algorithm. So it is necessary to design an equivalent estimated component to achieve, as follows:

$$\begin{aligned} g(\theta_i) &= g(\theta_{i-1} - \alpha \cdot d_i) = g(\theta_{i-1}) - g(\alpha \cdot d_i) \\ &= g(\theta_{i-1}) - \alpha \cdot g(d_i), \end{aligned} \tag{17}$$

$$g(\theta_{i-1}) - g(\theta_i) = \alpha \cdot g(d_i). \tag{18}$$

From Eq. (17) - (18), the equivalent estimated component can be derived as $[g(\theta_{i-2}) - g(\theta_{i-1})]$. However, the proportion of the estimated component in the whole will change continuously with the iteration, so it is necessary to add an estimation factor γ to control the weight of the estimated component:

$$\begin{aligned} d_i = (1 - \beta - \gamma) \cdot d_{i-1} + \beta \cdot g(\theta_{i-1}) - \mu \cdot g(\theta_{i-1})^2 \\ + \gamma \cdot [g(\theta_{i-2}) - g(\theta_{i-1})]. \end{aligned} \tag{19}$$

Thus, we have obtained an improved MRWE algorithm. The idea is to estimate the parameters first and then calculate the error from the estimates. If the gradient at the current moment is larger than the gradient at the previous moment, there is reason to believe that it will continue to grow larger, so it is necessary to add the imminent enlargement ahead of time, and vice versa.

3) IMPROVE THE LEARNING RATE

The MRWE weight estimation method uses a fixed learning factor, which may cause too many iterations and slow convergence of the deep learning network. At this point, the learning factor must be adaptively adjusted [32], [33]. Since the learning factor is always positive, $|g(\theta_{i-1})|$ is obtained, shown as

$$\begin{aligned} r_i = (1 - \beta - \gamma) \cdot r_{i-1} + \beta \cdot |g(\theta_{i-1})| \\ + \gamma \cdot [|g(\theta_{i-2})| - |g(\theta_{i-1})|]. \end{aligned} \tag{20}$$

where the learning factor is r, and the step size as the gradient falls is controlled by the product of the learning factor and the first derivative of the loss function.

However, due to the obvious difference between the initial and final stages of iterations, the learning efficiency is high at the initial stage of the iteration, so a high learning factor is required. At the final stage of the iteration, the weight has entered the optimal solution range. If the learning factor is high at this time, oscillation will occur and the algorithm's stability will be destroyed. To this end, a function y is designed to keep the learning rate decay immediately and to prevent a zero denominator:

$$y = INT\ (n/100) + 1, \tag{21}$$

$$\theta_i = \theta_{i-1} - \frac{\varepsilon}{r_i + y} \cdot d_i, \tag{22}$$

where ε is the initial learning rate, $INT\ (x)$ is the integer-valued function, and n is the current iteration number.

The role of LFA is to make the learning rate adaptive. If the current time gradient is large, then the learning rate attenuates faster. Conversely, if the current time gradient is small, the learning rate attenuates more slowly. The method is suitable for dealing with non-stationary timing targets and improves the convergence rate of nonlinear and time-varying prediction of ultra-short-term PV output power data.

E. LSTM OF ADAPTIVE HYPERPARAMETER ADJUSTMENT

After the improvement of Section II-C and Section II-D, the AHPA-LSTM model has increased the TLW, momentum factor β, resistance factor μ, and estimation factor γ. In the initial iteration, gradient adjustment requires less inertia, resistance constraints, and larger estimation constraints in order to quickly reach extreme values. In the mid-late iterations, inertia and estimation constraints are required to make the gradient exceed the local extrema, thus requiring greater inertia constraints. At the end of the iteration, after the global optimal solution has been found, a large resistance constraint is needed to stabilize the algorithm and prevent the oscillation from occurring. Based on this idea, we propose a adaptive hyperparameter model to adapt to this parameter change, as shown in Alg. 1 (line 7 and 9).

For deep learning networks, the premise that the gradient does not oscillate is that the learning rate monotonically decreases in the late iteration. Although the drag coefficient proposed in this paper is to prevent the occurrence of vibration, the second-order momentum of the drag coefficient does not guarantee the monotonicity of the learning rate. Due to the difference in data, the second-order momentum may have a large interference in the latter part of the iteration. Therefore, in order to make the algorithm run stably, this paper proposes to add SDG in the algorithm switching later in the algorithm, i.e., Alg. 1 (line 14 to 18). Once the learning rate occurs in the late stage of the iteration, the SDG algorithm is immediately switched to continue the search, which avoids the fluctuation of the prediction accuracy. Finally, we summarized the AHPA algorithm and proposed Alg. 1.

Algorithm 1 The Specific Description of AHPA Algorithm

Input: The gradient $g\ (\theta_{i-1})$ and $g\ (\theta_{i-2})$ of LSTM depth iteration at θ_{i-1} and θ_{i-2};
Updating speed d_{i-1} at $i-1$;
Learning speed r_{i-1} at $i-1$;
Number of current iterations n, maximum iteration N;
Initialize: Initial learning factor $\varepsilon \in [0, 1)$, resistance factor μ, momentum factor β, estimation factor γ;

1 **begin**
2 Initialize 1^{st}, 2^{nd} gradient vector: $d_0, d_1 \leftarrow 0$;
3 Initialize 1^{st}, 2^{nd} velocity vector: $r_0, r_1 \leftarrow 0$;
4 Initialize hyperparameter: $\beta \leftarrow 0.5$; $\mu \leftarrow 0.05$; $\gamma \leftarrow 0.1$;
5 **while** $n \leq N$ ***and*** $g\ (\theta_i)$ *not converged* **do**
6 **if** $n > 3N/4$ **then**
7 $\beta \leftarrow 0.1$; $\mu \leftarrow 0.15$; $\gamma \leftarrow 0.01$;
8 **else if** $N/4 \leq n \leq 3N/4$ **then**
9 $\beta \leftarrow -0.8 \cdot n/N + 0.7$; $\mu \leftarrow -n/5N$; $\gamma \leftarrow -0.18/N \cdot n + 0.145$;
10 **end**
11 Weight gradient update value: $\Delta d_i \leftarrow \beta \cdot g\ (\theta_{i-1}) - \mu \cdot g(\theta_{i-1})^2 + \gamma \cdot [g\ (\theta_{i-2}) - g\ (\theta_{i-1})]$ refer to Eq. (19);
12 Learning rate update value: $\Delta r_i \leftarrow \gamma \cdot [|g\ (\theta_{i-2})| - |g\ (\theta_{i-1})|] + \beta \cdot |g\ (\theta_{i-1})|$ refer to Eq. (20);
13 Weight gradient compensation: $d_i \leftarrow [(1 - \beta + \mu - \gamma) \cdot d_{i-1} + \Delta d_i] / (\beta - \mu + \gamma)$;
14 **if** $\frac{\varepsilon}{\|r_i - r_{i-1}\| + y} \cdot d_i \geq \theta_{i-1}$ ***and*** $n > 4N/5$ **then**
15 SDG algorithm switching: $r_i \leftarrow r_{i-1}$;
16 **else**
17 Learning rate compensation: $r_i \leftarrow \frac{(1-\beta-\gamma) \cdot r_{i-1} + \Delta r_i}{\beta + \gamma}$;
18 **end**
19 $y \leftarrow INT\ (n/100) + 1$ refer to Eq. (21);
20 $\theta_i \leftarrow \theta_{i-1} - \frac{\varepsilon}{r_i + y} \cdot d_i$ refer to Eq. (22);
21 **end**
22 **return** Gradient results θ_i
23 **end**

III. EVALUATION INDICATORS

In order to evaluate the pros and cons of the prediction model, we introduce the following evaluation indicators. The main indicator δ_{MAPE} has no positive and negative phase cancellation, which better reflects the mean of the relative error of prediction, defined as:

$$\delta_{MAPE} = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{|y_i - \widehat{y_i}|}{y_i} \times 100\%, \tag{23}$$

where n is the size of dataset; y_i is the true value of the i-th dataset; $\widehat{y_i}$ is the predicted value of the i-th dataset.

TABLE 1. The example of input and output data for forecast.

Parameter	Data											
Input	**5:30**	**5:35**	**5:40**	**5:45**	**5:50**	**5:55**	**6:00**	**6:05**	**6:10**	**6:15**	**6:20**	**6:25**
PV maximum output power (W)	2.76	2.97	3.19	3.33	3.47	3.63	3.77	3.90	4.05	4.23	4.83	5.45
PV module back surface temperature ($^\circ C$)	20.00	20.40	20.70	21.10	21.20	21.20	21.30	21.50	21.80	21.80	22.00	22.40
PV Cabinet temperature ($^\circ C$)	24.80	23.80	24.80	23.60	24.80	23.60	24.80	23.50	24.70	24.20	23.00	24.70
Dry bulb temperature ($^\circ C$)	19.70	19.80	19.80	19.50	19.60	19.40	19.30	19.50	19.70	19.60	19.60	19.90
Relative humidity ($\%RH$)	67.70	67.40	71.80	71.30	71.60	72.50	71.90	72.40	72.30	71.40	72.20	71.50
Atmospheric pressure (Pa)	996.10	996.20	996.20	996.30	996.30	996.30	996.30	996.30	996.20	996.20	996.20	996.20
Total precipitation (mm)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bevel irradiance (W/m^2)	31.40	34.50	37.60	40.70	43.80	47.00	50.50	53.70	55.80	60.40	71.00	83.80
Direct normal irradiance (W/m^2)	362.90	383.00	412.30	436.40	454.00	473.70	491.10	513.60	533.40	546.60	564.80	579.10
Global horizontal irradiance (W/m^2)	86.20	97.00	110.40	123.30	135.20	148.10	161.00	175.00	189.40	202.90	218.00	232.20
Diffuse horizontal irradiance (W/m^2)	35.30	38.00	40.90	43.50	45.70	47.70	49.80	51.40	53.20	55.40	57.30	58.90
Output	**5:35**	**5:40**	**5:45**	**5:50**	**5:55**	**6:00**	**6:05**	**6:10**	**6:15**	**6:20**	**6:25**	**6:30**
PV maximum output power (W)	2.97	3.19	3.33	3.47	3.63	3.77	3.90	4.05	4.23	4.83	5.45	6.20

The main indicator δ_{QRER} eliminates the influence of extreme values and embodies the range of error fluctuations in the relative error sequence (the smaller the value, the more stable the error), which is defined as follows:

$$\delta_{QRER} = \delta_{REUQ} - \delta_{RELQ}. \tag{24}$$

Due to the scheduling problem involving PV grid-connected, the PV forecast has a high degree of attention to the deviation δ_{MBE} of the output error, defined as:

$$\delta_{MBE} = \frac{1}{n} \cdot \sum_{i=1}^{n} (y_i - \widehat{y_i}). \tag{25}$$

The main indicator δ_{RMSE} measures the deviation between the predicted value and the actual PV output power, which can better reflect the absolute deviation of the error, defined as:

$$\delta_{RMSE} = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} (y_i - \widehat{y_i})^2}. \tag{26}$$

Referring to Marquez and Coimbra [34], we define the δ_{THI}. Meanwhile, we also define the ratio of i-th actual output PV power y_i to that of a clear-sky output PV power $y_{i_{clear}}$ as clear-sky relative error RE, so as to neglect the diurnal variability [35].

$$\delta_{THI} = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} \left(\frac{y_i - \widehat{y_i}}{y_{i_{clear}}} \right)^2} \tag{27}$$

$$RE = \frac{y_i - \widehat{y_i}}{y_{i_{clear}}} \tag{28}$$

The step-changes standard deviation of RE define as δ_{PVV}. Also, the δ_{RATIO} directly evaluates the variability effectively reduced by the forecasting models and normalizing it with respect to δ_{PVV}:

$$\delta_{PVV} = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} \left(\frac{y_i}{y_{i_{clear}}} - \frac{y_{i-1}}{y_{i-1_{clear}}} \right)^2}, \tag{29}$$

$$\delta_{RATIO} = 1 - \frac{\delta_{THI}}{\delta_{PVV}}. \tag{30}$$

TABLE 2. The structure of verification dataset.

	Feb.	May.	Aug.	Nov.	total
sunny days	no data	1-7	4-6	no data	10
cloudy days	2, 4-7	no data	2-3	2, 6-7	10
rainy days	1, 3	no data	1, 7	1, 3-5	8

IV. VERIFICATION OF PV OUTPUT POWER FORECAST

In order to verify the validity of the AHPA algorithm, the main structure of this section is as follows: Section IV-A introduces the source of the data. Section IV-B introduces the comparison between the AHPA-LSTM model and the traditional model, highlighting the validity of the proposed model. Section IV-C describes the prediction capabilities of the AHPA-LSTM model under different climatic conditions and demonstrates the universality of the model.

A. DETAILS OF TRAINING DATASET

The dataset we used is PV data from Zhejiang Province, China, with the total peak power ranging from 10 kW to 15 kW. At the same time, the data also includes the surface irradiance and global horizontal irradiance of the local solar sensor network. After the previous data processing and missing value supplements, we chose a database with relatively complete data and few missing values for training. The input and output data of the training are shown in TAB. 1.

The data relates to the period from 2015 to 2016, the sampling period is 06:00-18:00 every day, and the time resolution is 5 minutes. The location is on the southeast coast of China and belongs to the subtropical monsoon climate. The annual average temperature is between $13^\circ C$ and $20^\circ C$, and the average annual precipitation is between 800mm and 1500mm. The four seasons are distinct, with more rain in autumn and winter and complex clouds. To highlight the degree of data discrimination, we selected January, April, July, and October of the year (30 days per quarter) as four data sets to build the AHPA-LSTM model. Seven days after the selection of each data set is used as the verification data set, and the selection structure is as shown in TAB. 2.

To facilitate efficient training of the LSTM network and prevent overfitting of training, we randomly extracted 90% data from each dataset as training samples, and the

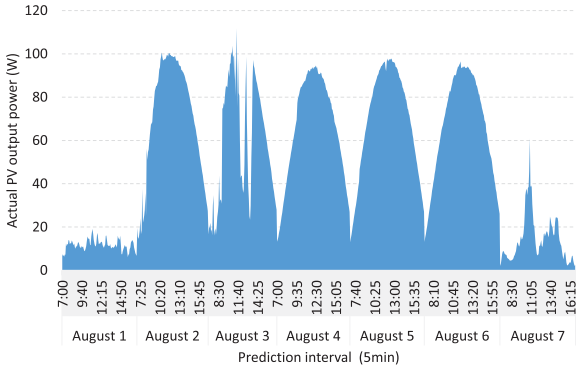


FIGURE 4. Examples of validation datasets.

remaining 10% as testing samples of the model, as shown in FIG. 4. Referring to Ospina *et al.* [17], we propose a network of 1 input layer, 1 output layer and 16 hidden layers according to the data characteristics of this paper. In order to satisfy the high-speed and efficient prediction requirements as much as possible, we select the maximum number of iterations as 10 times. At the same time, ITER-LSTM with 100 iterations was established as a comparison. By the way, we used an Intel Core i7-6700HQ @2.60 GHz CPU, and 24 GB memory as the computer operating environment and used MATLAB 2018a to build the ultra-short-term power prediction model of HEPV system.

B. MODEL PREDICTION EFFECT ANALYSIS

To fully illustrate the impact of the AHPA-LSTM model on ultra-short-term prediction of PV output power, this Section has established a variety of comparison models. Two of the traditional prediction models are LSTM and bidirectional LSTM. The others are as follows: Based on the TLW in Section II-B, here, we establish the Time-LSTM model that adds TLW. Based on the comparison of iterations, 100 iterations of the ITER-LSTM model were established. Based on the fusion activation function in Section II-C, we establish the FAF-LSTM model that increases robustness of the model. Based on the local extremum improvement in Section II-D.1, we establish the MR-LSTM model that suppresses weight oscillation. Based on the weight estimation improvement in Section II-D.2, we establish the MRWE-LSTM model that accelerates weight updating. Based on the improvements to accelerate the learning rate in Section II-D.3, we establish the LFA-LSTM model. Based on the stable convergence domain in Section II-E, the AHPA-LSTM model is established.

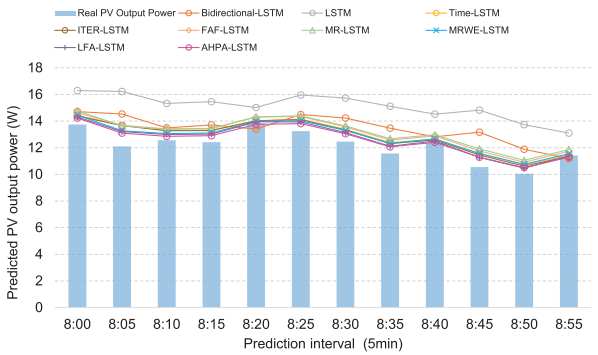
In the general climate classification (rainy, cloudy, sunny), solar radiation is weak in rainy days, which is the most difficult type of climate classification for predicting power generation. Therefore, we first select an hour of rainy day to verify the prediction accuracy of the AHPA-LSTM by image, as shown in FIG. 5. Secondly, we present the evaluation metrics for all validation datasets in tabular form to demonstrate the overall effectiveness of the model.

It can be seen that the output error of the original LSTM and the bidirectional LSTM network is large, and the maximum relative error is about 20%. The main reason for the large prediction error is that the model does not make a good distinction between the original inputs, so that the high dimension of the input data destroys the accuracy of the prediction. The maximum relative error of the Time-LSTM model with TLW added in the same time period is about 10%, which indicates that our improvement effectively improves the prediction effect. As shown in FIG. 5c and FIG. 5d, when the original data changes sharply at 12:40-12:55, the AHPA-LSTM model can effectively adapt to this fluctuation, showing better wave tracking capability. Compared with the other six models, the AHPA-LSTM model not only has the smallest error interval, but also the error average is closest to zero and the error standard deviation is the smallest (the error fluctuated in a small range). Therefore, it can be concluded that the model has better accuracy and stability.

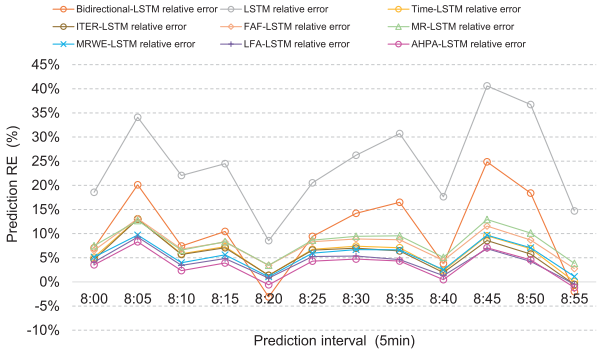
Nevertheless, FIG. 5 of the forecasted production for 1 hour in August was far from enough to fully evaluate the comprehensive forecasting ability of the AHPA-LSTM model throughout the year. Therefore, we use the evaluation indicators in Section III to compare the four verification datasets of the above nine models, and the results were shown in TAB. 3. Compared with the LSTM and Bi-LSTM networks, the Time-LSTM prediction model with TLW has greatly improved the performance of each indicator. From the training time and forecast running time, Time-LSTM has a certain gap compared with traditional LSTM. The training time of LSTM is about 30 minutes longer than the traditional mode, however, Time-LSTM has great advantages in other indicators. From the δ_{RATIO} indicator, Time-LSTM is 0.25 more than the traditional LSTM and 0.26 higher than the Bi-LSTM. From the four relative error indicators, the relative fluctuation range of Time-LSTM is 13.04% lower than the traditional LSTM by 12.04% compared to Bi-LSTM. And the total predicted running time of the AHPA-LSTM model is 30.86s, which meets the shortest time interval (5 minutes). It can be concluded that the LSTM network that distinguishes between time weight and climate weight has higher recognition ability. While reducing the input data dimension, the learning ability of traditional neural networks is enhanced.

In addition, compared with the training time and the predicted running time, except for the ITER-LSTM model, the time requirements of other models meet the requirements of ultra-short-term prediction, and the difference in accuracy of prediction is small. The ITER-LSTM model has a MAPE of 8.65% and a RATIO of 0.88. However, the model predicted training time and predicted running time were 307.65 minutes slower than AHPA-LSTM, exceeding the time requirement for ultra-short-term forecasting. Therefore, the ITER-LSTM model with deep iteration is not suitable for ultra-short-term PV prediction.

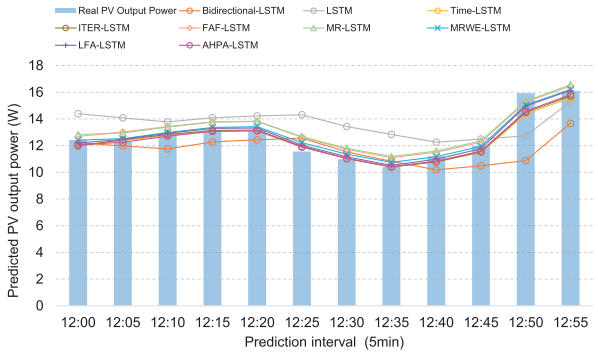
In order to compare AHPA-LSTM and the other models more intuitively, we present indicators of the above models in the form of statistical graphs. Compared with other parameter indexes, the δ_{THI} index and δ_{RATIO} index ignore diurnal variability, so these parameters can reflect realistic prediction in



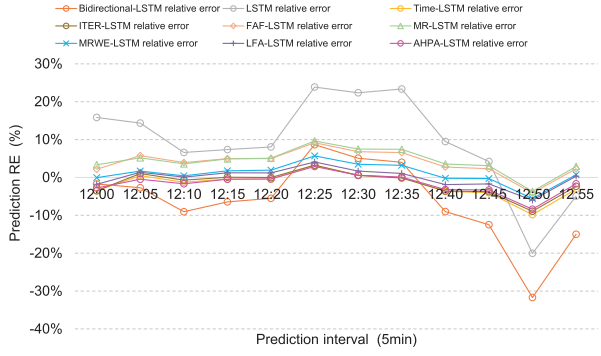
(a) PV output power prediction results at 8:00



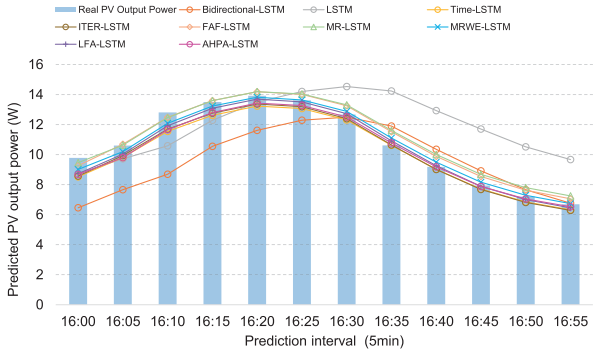
(b) Prediction error of PV output power at 8:00



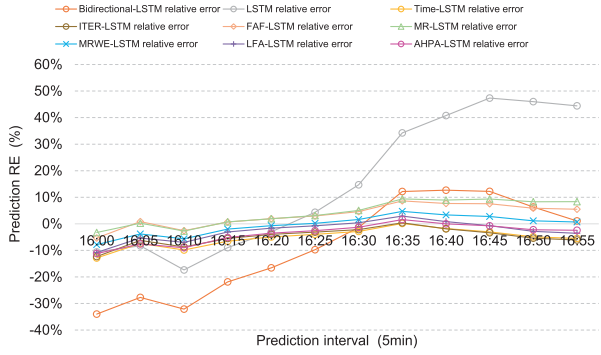
(c) PV output power prediction results at 12:00



(d) Prediction error of PV output power at 12:00



(e) PV output power prediction results at 16:00



(f) Prediction error of PV output power at 16:00

FIGURE 5. **Comparison of PV output prediction results for different models.**

PV forecasting. As can be seen in FIG. 6, the δ_{RATIO} index of the AHPA-LSTM was closer to 0.9, and the accuracy was the best. ITER-LSTM and Time-LSTM had similar prediction and evaluation indicators, but there was a huge gap in running time, which demonstrates that the Time-LSTM model had a better effect. Compared with Time-LSTM, FAF-LSTM, and AHPA-LSTM, it shows that adding the TLW and the fusion activation function had a good effect on accelerating the search for the optimal solution of a deep learning network and can alleviate the gradient-disappearance problem of LSTM. Compared with MR-LSTM, MRWE-LSTM, LFA-LSTM, and AHPA-LSTM, the simple momentum-resistance and weight-estimation methods demonstrate no obvious improvement in the accuracy of ultra-short-term load forecasting due to the fixed pre-learning factors. The AHPA-LSTM model combines the above advantages and solves the problems of local extrema, gradient disappearance, and a slow learning rate. It had a good performance in PV ultra-short-term power prediction.

In conclusion, δ_{MAPE} of AHPA-LSTM was 5.8%, δ_{QRER} was 5.56%, δ_{RMSE} was 1.63 W, δ_{THI} was 1.19E-01, and the δ_{RATIO} index was 0.91, which meet the accuracy requirements of PV ultra-short-term power prediction. Regarding running time, AHPA-LSTM was 50% slower than the non-improved LSTM model under the same iteration structure and cycle times. However, as regards overall running time, the model meets both the actual demand and the ultra-short prediction time interval (5 minutes). It can be seen that the AHPA-LSTM model had the best effect on improving the predicted output, so the adaptive hyperparameter adjustment AHPA-LSTM model had practical significance for ultra-short-term PV power prediction.

TABLE 3. Evaluation indexes for PV data forecast throughout the year.

		δ_{MAPE} (%)	δ_{QRPE} (%)	δ_{MBE} (W)	δ_{RMSE} (W)	δ_{THI}	δ_{RATIO}	δ_{REUQ} (%)	δ_{RELQ} (%)	Average of RE (%)	Mean square error of RE (%)	Training times (minutes)	Running time (s)
Bi-LSTM	**Jan.**	28.2390%	36.9603%	-0.8009	12.0639	0.5964	0.7791	30.2394%	-6.7209%	15.6574%	39.4128%	30.30	19.14
	Apr.	9.4708%	4.5751%	-0.9624	1.7892	0.5031	0.8346	4.8718%	0.2968%	6.8823%	19.4312%	31.20	19.01
	Jul.	11.8259%	7.2164%	0.7068	5.3213	0.4607	0.9247	3.1756%	-4.0408%	-2.8477%	22.9382%	33.40	18.77
	Oct.	30.2759%	37.4477%	0.9100	9.7122	0.1122	-0.0356	22.0302%	-15.4175%	11.3968%	51.4664%	29.80	17.14
	average	**19.9529%**	**21.5499%**	**-0.0366**	**7.2216**	**0.4181**	**0.6257**	**15.0793%**	**-6.4706%**	**7.7722%**	**33.3121%**	**31.18**	**18.52**
LSTM	**Jan.**	25.1315%	29.8487%	2.0035	12.1801	0.8456	0.6868	7.6105%	-22.2382%	-2.8603%	37.2107%	25.60	17.33
	Apr.	9.4041%	2.7777%	1.1178	1.8689	0.2228	0.9267	-0.7168%	-3.4945%	-7.0998%	25.2776%	23.50	17.23
	Jul.	14.5565%	12.3709%	-0.3563	5.2838	0.7609	0.8757	10.7073%	-1.6636%	6.6108%	28.5776%	22.10	16.80
	Oct.	30.1981%	37.2306%	0.9573	9.7135	0.1124	-0.0372	21.5771%	-15.6535%	10.8479%	51.5122%	25.30	16.90
	average	**19.8225%**	**20.5570%**	**0.9306**	**7.2616**	**0.4854**	**0.6130**	**9.7945%**	**-10.7624%**	**1.8746%**	**35.6445%**	**24.13**	**17.07**
Time-LSTM	**Jan.**	9.4260%	12.7446%	0.2007	3.6611	0.2212	0.9181	8.8539%	-3.8907%	3.8724%	14.3477%	68.50	29.83
	Apr.	5.8454%	1.9433%	0.1641	0.9267	0.2150	0.9293	0.9663%	-0.9770%	3.5872%	16.6463%	66.30	30.10
	Jul.	5.0859%	2.4199%	-0.0292	1.0055	0.2932	0.9521	1.3873%	-1.0326%	2.5887%	11.7875%	64.20	28.40
	Oct.	17.1951%	16.9438%	1.0866	2.7207	0.0341	0.6849	-2.2370%	-19.1808%	-12.9627%	24.1388%	61.30	29.31
	average	**9.3881%**	**8.5129%**	**0.3555**	**2.0785**	**0.1909**	**0.8711**	**2.2426%**	**-6.2703%**	**-0.7286%**	**16.7301%**	**65.08**	**29.41**
ITER-LSTM	**Jan.**	8.4241%	9.8103%	-0.4495	2.1549	0.2381	0.9118	10.1991%	0.3888%	6.9460%	10.8731%	355.30	27.40
	Apr.	4.8811%	4.4973%	0.7560	1.9028	0.1158	0.9619	1.7076%	-2.7897%	0.2936%	11.3035%	377.60	30.90
	Jul.	4.7053%	2.4226%	-0.0743	0.9408	0.3274	0.9465	1.7132%	-0.7094%	2.3436%	10.7889%	320.50	32.50
	Oct.	16.5882%	17.7079%	0.8790	2.5205	0.0323	0.7016	-0.4156%	-18.1235%	-11.2836%	24.7794%	390.40	29.80
	average	**8.6497%**	**8.6095%**	**0.2778**	**1.8797**	**0.1784**	**0.8805**	**3.3011%**	**-5.3085%**	**-0.4251%**	**14.4362%**	**360.95**	**30.15**
FAF-LSTM	**Jan.**	7.6295%	9.7931%	-0.2656	2.4673	0.2088	0.9226	9.0702%	-0.7229%	5.0232%	10.2056%	68.70	30.70
	Apr.	4.5722%	5.3179%	0.6067	1.9010	0.1455	0.9521	2.9921%	-2.3257%	2.0394%	8.5800%	74.50	30.82
	Jul.	4.4829%	3.1205%	-0.4413	1.3955	0.2010	0.9672	3.4919%	0.3714%	2.9620%	8.2794%	75.50	29.90
	Oct.	14.6200%	15.9107%	0.8168	2.9245	0.0344	0.6823	1.4153%	-14.4954%	-3.2216%	21.1496%	72.30	31.34
	average	**7.8261%**	**8.5355%**	**0.1791**	**2.1721**	**0.1475**	**0.8811**	**4.2424%**	**-4.2932%**	**1.7008%**	**12.0537%**	**72.75**	**30.69**
MR-LSTM	**Jan.**	6.4174%	7.3259%	0.5603	1.5884	0.1415	0.9476	0.5995%	-6.7264%	-3.3824%	9.2132%	69.90	31.34
	Apr.	3.8808%	2.7717%	0.5647	1.3052	0.0827	0.9728	0.4750%	-2.2967%	-1.7948%	8.9766%	73.60	31.20
	Jul.	4.3360%	3.1996%	-0.2660	1.2788	0.2168	0.9646	3.1055%	-0.0941%	2.7547%	8.4469%	74.50	29.96
	Oct.	13.3958%	13.7333%	0.5171	2.5066	0.0305	0.7185	2.8978%	-10.8356%	-4.3109%	21.3452%	68.80	30.89
	average	**7.0075%**	**6.7576%**	**0.3440**	**1.6697**	**0.1179**	**0.9009**	**1.7695%**	**-4.9882%**	**-1.6834%**	**11.9955%**	**71.70**	**30.85**
MRWE-LSTM	**Jan.**	4.9075%	5.6563%	-0.1069	1.7837	0.1462	0.9458	5.0877%	-0.5686%	3.2505%	7.3043%	65.60	30.89
	Apr.	3.5883%	3.2194%	0.6177	1.4900	0.0833	0.9726	0.8859%	-2.3335%	-0.9145%	7.6308%	61.30	30.27
	Jul.	3.8627%	1.3307%	-0.1351	0.9501	0.2353	0.9616	1.1838%	-0.1468%	1.2889%	8.3652%	64.50	29.75
	Oct.	13.0002%	13.9732%	0.5909	2.4470	0.0298	0.7252	1.8434%	-12.1299%	-5.6133%	20.4581%	62.50	31.60
	average	**6.3397%**	**6.0449%**	**0.2417**	**1.6677**	**0.1237**	**0.9013**	**2.2502%**	**-3.7947%**	**-0.4971%**	**10.9396%**	**63.48**	**30.63**
LFA-LSTM	**Jan.**	4.6318%	4.9256%	-0.1243	2.1570	0.2094	0.9224	4.0440%	-0.8817%	2.7314%	7.7852%	53.20	31.38
	Apr.	3.4786%	3.5708%	0.3386	1.3614	0.1042	0.9657	1.7711%	-1.7997%	0.2777%	7.3602%	50.10	30.60
	Jul.	3.5180%	2.1993%	-0.2411	1.2636	0.1723	0.9719	1.9424%	-0.2569%	1.5415%	6.8510%	48.40	29.83
	Oct.	12.3812%	12.8646%	0.1903	2.4006	0.0289	0.7329	5.9083%	-6.9563%	0.7990%	20.7361%	53.10	28.20
	average	**6.0024%**	**5.8901%**	**0.0409**	**1.7956**	**0.1287**	**0.8982**	**3.4165%**	**-2.4736%**	**1.3374%**	**10.6832%**	**51.20**	**30.00**
AHPA-LSTM	**Jan.**	4.4207%	4.2215%	0.0611	1.6225	0.1449	0.9463	1.4004%	-2.8211%	-1.1185%	7.8881%	55.90	31.60
	Apr.	3.2654%	3.2874%	0.6323	1.5379	0.0693	0.9772	1.0039%	-2.2835%	-0.5941%	6.8529%	51.60	27.55
	Jul.	3.3866%	2.0813%	0.0472	1.1020	0.2345	0.9617	0.9153%	-1.1660%	1.1592%	7.3032%	53.40	31.40
	Oct.	12.1193%	12.6648%	-0.0476	2.2460	0.0273	0.7480	7.5525%	-5.1123%	2.0878%	20.2583%	52.30	32.90
	average	**5.7980%**	**5.5637%**	**0.1732**	**1.6271**	**0.1190**	**0.9083**	**2.7180%**	**-2.8457%**	**0.3836%**	**10.5756%**	**53.30**	**30.86**

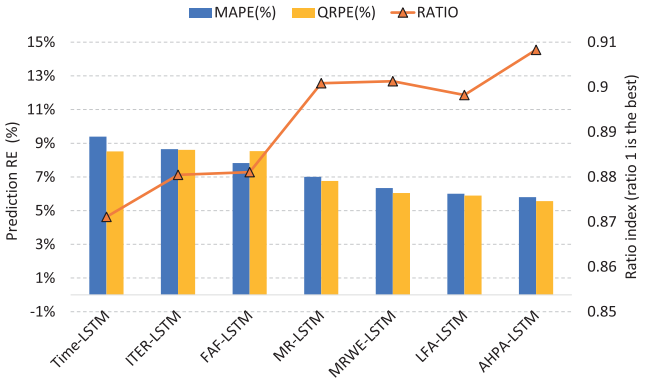


FIGURE 6. Effect evaluation index of LSTM model prediction after weight optimization.

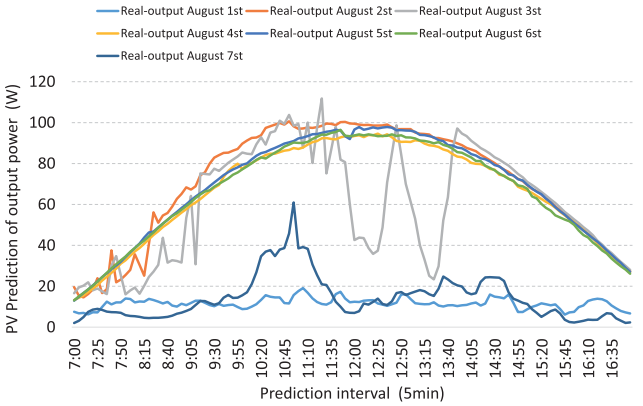
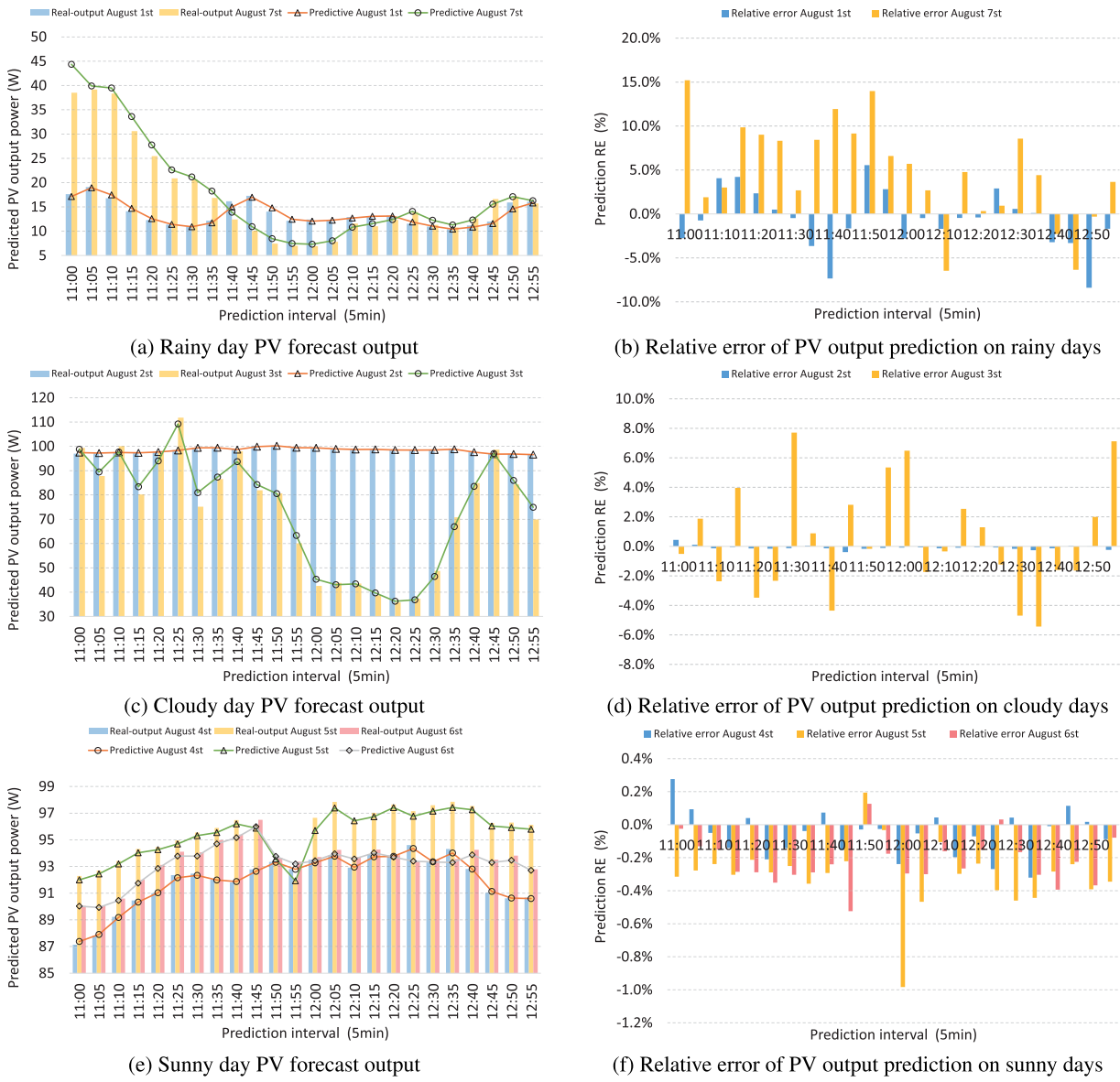


FIGURE 7. PV verification datasets under different meteorological conditions.

C. ANALYSIS OF VERIFICATION SAMPLES UNDER DIFFERENT METEOROLOGICAL CONDITIONS

Due to the dates of rainy days and sunny days, there were large differences in the PV output power curves, as shown in FIG. 7. Therefore, the prediction of PV output power under full climatic conditions was a great test for the comprehensive performance prediction of the model. Using the trained AHPA-LSTM model in the previous section, we predicted



(a) Rainy day PV forecast output

(b) Relative error of PV output prediction on rainy days

(c) Cloudy day PV forecast output

(d) Relative error of PV output prediction on cloudy days

(e) Sunny day PV forecast output

(f) Relative error of PV output prediction on sunny days

FIGURE 8. Validation samples from 10:00 to 11:55 under different meteorological conditions.

the PV output power under rainy, cloudy, and sunny conditions. Firstly, we can select 11:00 to 12:55 on sunny days (August 4-6), cloudy days (August 2, 3), and rainy days (August 1, 7) as verification data, and use images to display prediction results more intuitively, as shown in FIG. 8. Secondly, the overall forecast results of the four verification datasets for the whole year were displayed in tabular form.

It can be seen from the different weather conditions that the AHPA-LSTM model had high-accuracy and high stability under different climatic conditions. From different meteorological conditions, the relative error of sunny days was the best, and the relative error of rainy days was the worst. This demonstrates that the prediction error of the AHPA-LSTM model under clear-sky conditions was very small. On cloudy and rainy days, the error was stable at about 6% due to the randomness of the cloud. However, the verification data of the excerpts alone cannot be fully evaluated. The annual full-weather assessment indicators proposed by us were shown in TAB. 4.

It can be seen from FIG. 9 that for three climate types (rainy, cloudy, sunny), the relative error of the sunny days were in the range of 1%, the relative error of the cloudy days were in the range of 8%, and the relative error of the rainy days were in the range of 15%. And δ_{MAPE} under various weather conditions was basically kept at around 1%, δ_{RMSE} was kept at around 1 W, and the daily change of δ_{RATIO} basically stays around 0.9. Therefore, this model can accurately predict the PV power output under different climatic conditions, and the robustness of the model was good, which can meet the requirements of ultra-short-term PV output power prediction.

TABLE 4. Evaluation indexes of PV data forecast under different meteorological conditions.

		δ_{MAPE} (%)	δ_{QRPE} (%)	δ_{MBE} (W)	δ_{RMSE} (W)	δ_{THI}	δ_{RATIO}	δ_{REUQ} (%)	δ_{RELQ} (%)	Average of RE (%)	Mean square error of RE (%)
sunny days	Apr.	3.2654%	3.2874%	0.6323	1.5379	0.0693	0.9772	1.0039%	-2.2835%	-0.5941%	6.8529%
	Jul.	1.8870%	0.5912%	0.1226	0.3282	0.2473	0.9628	0.0722%	-0.5190%	0.3607%	5.5543%
	sunny total	**2.8541%**	**2.5787%**	**0.4802**	**1.3006**	**0.1471**	**0.9668**	**0.6506%**	**-1.9282%**	**-0.3092%**	**6.5073%**
cloudy days	Jan.	3.8788%	4.0537%	0.0164	1.8914	0.1588	0.9492	1.8861%	-2.1676%	0.1969%	6.9026%
	Jul.	2.8684%	1.7175%	0.1300	1.8935	0.2229	0.9678	0.2034%	-1.5141%	0.5309%	5.9237%
	Oct.	13.1558%	14.5006%	0.2107	3.3040	0.0381	0.7669	7.0954%	-7.4052%	2.1720%	21.9540%
	cloudy total	**5.9235%**	**4.2138%**	**0.0983**	**2.3303**	**0.1624**	**0.9621**	**1.9790%**	**-2.2348%**	**0.7918%**	**12.4105%**
rainy days	Jan.	5.7306%	5.7137%	0.1706	0.5504	0.1026	0.9048	0.4231%	-5.2905%	-4.3411%	9.1108%
	Jul.	6.4460%	8.0910%	-0.1672	0.7305	0.2241	0.9428	5.7181%	-2.3729%	3.1641%	10.2413%
	Oct.	10.1685%	9.5751%	-0.3659	1.0484	0.0173	0.5941	7.7913%	-1.7837%	4.6334%	17.1004%
	rainy total	**7.7147%**	**8.0432%**	**-0.1602**	**0.8342**	**0.1446**	**0.9401**	**5.1673%**	**-2.8759%**	**1.8638%**	**13.6209%**
AHPA-LSTM	**average**	**5.7980%**	**5.5637%**	**0.1732**	**1.6271**	**0.1190**	**0.9083**	**2.7180%**	**-2.8457%**	**0.3836%**	**10.5756%**

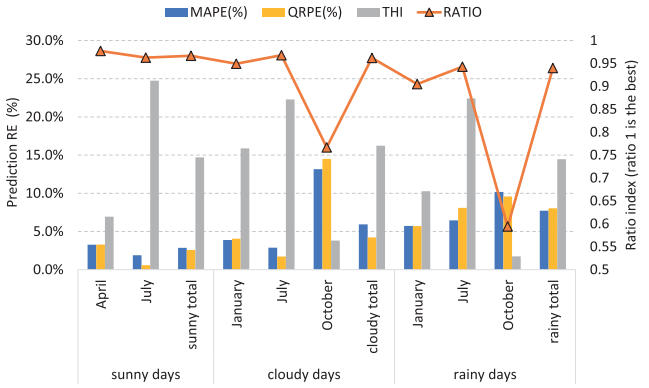


FIGURE 9. Effect evaluation index of LSTM model prediction under different meteorological conditions.

In summary, this paper proposes an AHPA-LSTM model based on the characteristics of nonlinearity, easy interference, and time variation of PV output power time series. This model had higher accuracy and was faster than previous models, which meets the requirements of ultra-short-term PV output power prediction.

V. CONCLUSION

We have proposed the AHPA-LSTM model for the prediction of ultra-short-term PV output power. The TLW is added to the model input data to enhance the time correlation of the PV output power. Then, the momentum resistance method is used to improve the LSTM weight update to global extremum. The weight estimation method is used to estimate the speed of the model, and the learning rate is accelerated. Since the momentum resistance method and weight estimation method introduce the new hyperparameters momentum factor β, resistance factor μ, and estimation factor γ, an adaptive hyperparameter adjustment scheme is proposed, thus obtaining the adaptive hyperparameter adjustment AHPA-LSTM model. Based on the actual PV power plant data and local meteorological data, we establish nine power-prediction models of Time-LSTM, MRWE-LSTM, AHPA-LSTM, etc., to compare with the PV output-power prediction effects. It can be seen from the prediction results that the AHPA-LSTM model can meet the high-accuracy application requirements of PV ultra-short-term power prediction under different weather conditions with fewer iterations, thereby improving the economic benefits of a PV grid connection.

In follow-up work, more real-time meteorological data can be added, such as video monitoring to identify the cloud amount above the PV module, thereby improving the prediction accuracy of the PV ultra-short-term prediction model under cloudy weather conditions. In future research, we will consider implementing the relevant algorithms on the embedded platform, so as to be closer to the actual demands of grid dispatching.

REFERENCES

[1] T. M. Razykov, C. S. Ferekides, D. Morel, E. Stefanakos, H. S. Ullal, and H. M. Upadhyaya, "Solar photovoltaic electricity: Current status and future prospects," *Sol. Energy*, vol. 85, no. 8, pp. 1580–1608, Aug. 2011.

[2] S. Sobri, S. Koohi-Kamali, and N. A. Rahim, "Solar photovoltaic generation forecasting methods: A review," *Energy Convers. Manage.*, vol. 156, pp. 459–497, Jan. 2018.

[3] R. H. Inman, H. T. Pedro, and C. F. Coimbra, "Solar forecasting methods for renewable energy integration," *Prog. Energy Combustion Sci.*, vol. 39, no. 6, pp. 535–576, 2013.

[4] F. Barbieri, S. Rajakaruna, and A. Ghosh, "Very short-term photovoltaic power forecasting with cloud modeling: A review," *Renew. Sustain. Energy Rev.*, vol. 75, pp. 242–263, Aug. 2017.

[5] C. Voyant, G. Notton, S. Kalogirou, M.-L. Nivet, C. Paoli, F. Motte, and A. Fouilloy, "Machine learning methods for solar radiation forecasting: A review," *Renew. Energy*, vol. 105, pp. 569–582, May 2017.

[6] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F. Martinez-de-Pison, and F. Antonanzas-Torres, "Review of photovoltaic power forecasting," *Sol. Energy*, vol. 136, pp. 78–111, Oct. 2016.

[7] G. M. Lohmann, "Irradiance variability quantification and small-scale averaging in space and time: A short review," *Atmosphere*, vol. 9, no. 7, p. 264, Jun. 2018.

[8] B. Wolff, J. Kühnert, E. Lorenz, O. Kramer, and D. Heinemann, "Comparing support vector regression for pv power forecasting to a physical modeling approach using measurement, numerical weather prediction, and cloud motion data," *Sol. Energy*, vol. 135, pp. 197–208, Oct. 2016.

[9] R. Marquez, H. T. C. Pedro, and C. F. M. Coimbra, "Hybrid solar forecasting method uses satellite imaging and ground telemetry as inputs to ANNs," *Sol. Energy*, vol. 92, pp. 176–188, Jun. 2013.

[10] E. Lorenz, J. Kühnert, D. Heinemann, K. P. Nielsen, J. Remund, and S. C. Müller, "Comparison of global horizontal irradiance forecasts based on numerical weather prediction models with different spatio-temporal resolutions," *Prog. Photovolt., Res. Appl.*, vol. 24, no. 12, pp. 1626–1640, Dec. 2016.

[11] X. Zhang, Y. Li, S. Lu, H. F. Hamann, B.-M. Hodge, and B. Lehman, "A solar time based analog ensemble method for regional solar power forecasting," *IEEE Trans. Sustain. Energy*, vol. 10, no. 1, pp. 268–279, Jan. 2019.

[12] A. G. R. Vaz, B. Elsinga, W. G. J. H. M. van Sark, and M. C. Brito, "An artificial neural network to assess the impact of neighbouring photovoltaic systems in power forecasting in Utrecht, The Netherlands," *Renew. Energy*, vol. 85, pp. 631–641, Jan. 2016.
[13] M. J. Espinosa-Gavira, A. Agüera-Pérez, J. J. G. de la Rosa, J. C. Palomares-Salas, and J. M. Sierra-Fernández, "An on-line low-cost irradiance monitoring network with sub-second sampling adapted to small-scale PV systems," *Sensors*, vol. 18, no. 10, p. 3405, Oct. 2018.
[14] E. Scolari, L. Reyes-Chamorro, F. Sossan, and M. Paolone, "A comprehensive assessment of the short-term uncertainty of grid-connected PV systems," *IEEE Trans. Sustain. Energy*, vol. 9, no. 3, pp. 1458–1467, Jul. 2018.
[15] X. G. Agoua, R. Girard, and G. Kariniotakis, "Probabilistic models for spatio-temporal photovoltaic power forecasting," *IEEE Trans. Sustain. Energy*, vol. 10, no. 2, pp. 780–789, Apr. 2018.
[16] H. Sheng, J. Xiao, Y. Cheng, Q. Ni, and S. Wang, "Short-term solar power forecasting based on weighted Gaussian process regression," *IEEE Trans. Ind. Electron.*, vol. 65, no. 1, pp. 300–308, Jan. 2018.
[17] J. Ospina, A. Newaz, and M. O. Faruque, "Forecasting of PV plant output using hybrid wavelet-based LSTM-DNN structure model," *IET Renew. Power Gener.*, vol. 13, no. 7, pp. 1087–1095, May 2019.
[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[19] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 1999.
[20] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, nos. 5–6, pp. 602–610, Jul./Aug. 2005.
[21] S. Kudugunta and E. Ferrara, "Deep neural networks for bot detection," *Inf. Sci.*, vol. 467, pp. 312–322, Oct. 2018.
[22] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[23] S. Srivastava and S. Lessmann, "A comparative study of LSTM neural networks in forecasting day-ahead global horizontal irradiance with satellite data," *Sol. Energy*, vol. 162, pp. 232–247, Mar. 2018.
[24] L. Zheng, Z. Liu, J. Shen, and C. Wu, "Very short-term maximum Lyapunov exponent forecasting tool for distributed photovoltaic output," *Appl. Energy*, vol. 229, pp. 1128–1139, Nov. 2018.
[25] S. Han, Y.-H. Qiao, J. Yan, Y.-Q. Liu, L. Li, and Z. Wang, "Mid-to-long term wind and photovoltaic power generation prediction based on copula function and long short term memory network," *Appl. Energy*, vol. 239, pp. 181–191, Apr. 2019.
[26] S. Luo, L. Qin, M. Hu, X. Hou, and S. Xie, "Stable operating area of photovoltaic cells feeding the interface converter in output current regulation mode," in *Proc. IEEE 8th Int. Power Electron. Motion Control Conf. (IPEMC-ECCE Asia)*, May 2016, pp. 185–192.
[27] B. Kanna and S. N. Singh, "Long term wind power forecast using adaptive wavelet neural network," in *Proc. IEEE Uttar Pradesh Sect. Int. Conf. Elect., Comput. Electron. Eng. (UPCON)*, Dec. 2016, pp. 671–676.
[28] S. Wang, X. Wang, S. Wang, and D. Wang, "Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting," *Int. J. Elect. Power Energy Syst.*, vol. 109, pp. 470–479, Jul. 2019.
[29] P. Petersen and F. Voigtlaender, "Optimal approximation of piecewise smooth functions using deep ReLU neural networks," *Neural Netw.*, vol. 108, pp. 296–330, Dec. 2018.
[30] W. He and Y. Liu, "To regularize or not: Revisiting SGD with simple algorithms and experimental studies," *Expert Syst. Appl.*, vol. 112, pp. 1–14, Dec. 2018.
[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[32] N. S. Keskar and R. Socher, "Improving generalization performance by switching from Adam to SGD," 2017, *arXiv:1712.07628*. [Online]. Available: https://arxiv.org/abs/1712.07628
[33] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of Adam and beyond," 2019, *arXiv:1904.09237*. [Online]. Available: https://arxiv.org/abs/1904.09237
[34] R. Marquez and C. F. M. Coimbra, "Proposed metric for evaluation of solar forecasting models," *J. Solar Energy Eng.*, vol. 135, no. 1, Feb. 2013, Art. no. 011016.
[35] C. F. Coimbra, J. Kleissl, and R. Marquez, "Overview of solar-forecasting methods and a metric for accuracy evaluation," in *Solar Energy Forecasting Resource Assessment*. 2013, pp. 171–194.

MINKANG CHAI received the B.Sc. degree from the Shanghai University of Electric Power, in 2017, where he is currently pursuing the master's degree. His main research interests include micro-grid energy storage systems and big data mining technologies.

FEI XIA received the B.Sc. degree from the Shenyang University of Technology, in 2000, the M.Sc. degree from the University of Poitiers, France, in 2003, and the Ph.D. degree from Tongji University, in 2017. He is currently an Associate Professor with the Shanghai University of Electric Power. His main research interests include machine vision, artificial intelligence, information security, and embedded system in clean energy system.

SHUOTAO HAO received the B.Sc. degree from East China Jiaotong University, in 2013, and the M.Sc. degree from the Shanghai University of Electric Power, in 2016. He is currently the President of a Distribution Engineering Room, Operation Maintenance, and Repair Department, Fangshan Power Supply Company, Beijing Power Company. His main research interests include distribution network fault control and distribution engineering management.

DAOGANG PENG received the B.Eng. and M.Eng. degrees in automation engineering from North China Electric Power University, China, in 2001 and 2004, respectively, the Ph.D. degree in control theory and control from Tongji University, China, in 2008. He is currently a Professor with the Shanghai University of Electric Power, Shanghai, China. His research interests include intelligent power generation automation, new energy microgrid, and energy internet.

CHENGGANG CUI received the B.Eng. degree in automation engineering from Jilin University, China, in 2004, and the Ph.D. degree in control theory and control from Zhejiang University, China, in 2010. He is currently a Lecturer with the Shanghai University of Electric Power, Shanghai, China. His research interests include the control and schedule of renewable energy systems and microgrid.

WEI LIU received the B.Sc. degree from the Shenyang University of Technology, in 2000. He is currently a Senior Engineer and a Senior Manager with Haining Chinaust Plastics Piping System Company Ltd. His main research interests include industrial system monitoring, data collection, and analysis.