

## Research Article

# Short-Term Load Forecasting Method Based on Deep Reinforcement Learning for Smart Grid

Wei Guo <sup>1</sup>, Kai Zhang <sup>2</sup>, Xinjie Wei <sup>2</sup> and Mei Liu <sup>3</sup>

<sup>1</sup>Market Service Center, State Grid Hebei Electric Power Co., Ltd., Shijiazhuang 050021, China

<sup>2</sup>State Grid Hebei Electric Power Co., Ltd., Shijiazhuang 050021, China

<sup>3</sup>Beijing Tsingsoft Technology Co., Ltd., Beijing 100085, China

Correspondence should be addressed to Wei Guo; [yxzx\\_guow@he.sgcc.com.cn](mailto:yxzx_guow@he.sgcc.com.cn)

Received 9 October 2021; Revised 7 November 2021; Accepted 11 November 2021; Published 29 November 2021

Academic Editor: Sang-Bing Tsai

Copyright © 2021 Wei Guo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Short-term load forecasting is an important part to support the planning and operation of power grid, but the current load forecasting methods have the problem of poor adaptive ability of model parameters, which are difficult to ensure the demand for efficient and accurate power grid load forecasting. To solve this problem, a short-term load forecasting method for smart grid is proposed based on multilayer network model. This method uses the integrated empirical mode decomposition (IEMD) method to realize the orderly and reliable load state data and provides high-quality data support for the prediction network model. The enhanced network inception module is used to adaptively adjust the parameters of the deep neural network (DNN) prediction model to improve the fitting and tracking ability of the prediction network. At the same time, the introduction of hybrid particle swarm optimization algorithm further enhances the dynamic optimization ability of deep reinforcement learning model parameters and can realize the accurate prediction of short-term load of smart grid. The simulation results show that the mean absolute percentage error  $e_{MAPE}$  and root-mean-square error  $e_{RMSE}$  of the performance indexes of the prediction model are 10.01% and 2.156 MW, respectively, showing excellent curve fitting ability and load forecasting ability.

## 1. Introduction

Load forecasting is a key link to ensure the stable and controllable state of power system [1, 2]. Accurate load forecasting can realize the formulation of unit combination, appropriate rotating standby, and maintenance plans, which have a great impact on the planning and operation of power system to a certain extent [3–5], and can ensure that the production and consumption of electric energy are always in a balanced and stable state [6]. Therefore, the importance of load forecasting in energy distribution and management of modern power system is self-evident.

At present, the power load forecasting can be classified into long-term analysis, medium-term analysis, and short-term analysis according to the forecasting time scale [7]. Medium- and long-term forecasting is mainly used for load forecasting in the next few weeks to several years. It is often used for long-term planning such as the optimization of grid structure and the layout of power supply, so as to improve

the economy of power grid planning and construction. Short-term load forecasting is a part of the daily operation of power grid, and its forecasting accuracy directly affects the stability of power grid planning and operation [8]. Therefore, this study realizes accurate prediction and analysis of power grid short-term load.

Traditional load forecasting methods are mainly regression analysis and time-series methods [9, 10]. According to the timeline, this kind of method sorts the load historical data, constructs the corresponding mathematical structure of load forecasting, and deduces the analysis formula. The traditional load forecasting method is based on the historical state data of power grid, and the data state model is constructed to realize data linear fitting for load forecasting research and analysis. However, the current smart grid has a large number of distributed energy and random access of intelligent terminals, which have time and space uncertainty, so the power grid state data also have high uncertainty and strong nonlinearity, which have a great impact on the

construction and initialization of traditional mode models. [11].

The emergence of deep learning technology provides a new solution for power grid load forecasting and avoids the disadvantages of traditional methods of directly constructing forecasting and analysis models based on data [12]. The power state data can be continuously trained and learned through the multilayer structure network model to build a multilayer prediction network model with superior fitting ability, so as to achieve accurate and efficient load analysis and prediction [13]. It should be noted that due to the deep and complex characteristics of the multilayer network structure, its network parameters are difficult to track the data to achieve adaptive dynamic adjustment [14], and there is a problem of multistep power load prediction error.

Aiming at the above problems, this study proposes a new forecasting method based on deep learning network and reinforcement learning network. The main innovations are as follows:

- (1) Aiming to realize the reliable data support of the prediction network, this study preprocesses the collected power state data based on the integrated empirical mode decomposition (IEMD) method, so as to improve the order and reliability of the network model training data
- (2) Aiming to support the prediction network for accurate analysis, the reinforcement learning inception network model is used to adjust the parameters of the deep neural network (DNN), and the hybrid particle swarm optimization algorithm is integrated into the network to achieve accurate and efficient short-term load forecasting of smart grid

## 2. Related Work

Accurate short-term load forecasting can realize stable and controllable operation of the power grid [15], and the average error of short-term load can save tens or even millions of dollars for every one percentage point reduction [16].

Traditional load forecasting methods are considered from the aspect of mathematical modeling, such as regression analysis method and time-series method. The traditional forecasting method directly builds the mathematical model of power data, which has high requirements for the quality of collected power state data and can realize power load forecasting with small data volume. However, for the complex and diverse power state data, the analysis error of the prediction model is large. Moreover, for the traditional methods, there are some difficulties in the construction and initialization of the model, which are also difficult to ensure the prediction accuracy [17–19].

Due to the random grid connection of new energy and the complex change in load, the change in power load presents high uncertainty, and the power grid state data present strong nonlinearity, which poses a higher challenge to the short-term accurate prediction of power load [20–22].

Deep learning and reinforcement learning can extract the characteristics of load state data through multilayer network

structure and continuously improve the prediction network model through training and learning process to realize efficient power grid load forecasting. Reference [23] considers the uncertainty of temperature and load, and realizes medium-term load forecasting based on quantile regression neural network. Reference [24] uses automatic coding to compress the load history data and constructs a prediction model based on multilayer gated recurrent unit (GRU) to analyze and predict the daily variation of power load. Reference [25] uses long short-term memory (LSTM) model to predict load power based on temperature history data and power history data. Reference [26] combines LSTM network and RNN network to realize effective prediction of industrial power load based on multilayer hybrid deep learning network. However, it should also be pointed out that although the current multilayer network analysis method solves the problem of direct mathematical modeling, it still has certain requirements for the data quality of the training set. A reliable and complete load data set is an important guarantee for accurate prediction [27]. Moreover, due to too many layers of network structure, there is a certain lag in the adjustment of network parameters, and the mismatch of model parameters will also lead to a certain error in the prediction network.

To solve this problem, this study selects IEMD method to optimize power load data, so as to provide reliable data support for forecasting network model. Based on the parameter adaptive multilayer network structure, the efficient and accurate prediction of short-term load of smart grid is realized, which provides a reliable reference for power grid planning and operation.

## 3. Multilayer Network Prediction Method

For the complex scenario of smart grid, this study uses the combination of empirical mode decomposition method and deep reinforcement learning to build a complete power grid state data analysis architecture of reliable preprocessing accurate prediction analysis, so as to realize the accurate prediction of short-term load of smart grid.

*3.1. Overall Framework.* Power state data have the characteristics of redundancy and incompleteness. It is difficult to achieve accurate short-term load forecasting by training and learning it directly. This study combines data preprocessing technology with load forecasting technology and proposes a deep learning load forecasting model architecture supported by complete data sets. Figure 1 shows the short-term analysis architecture of power load proposed in this study.

As shown in Figure 1, to achieve accurate power load forecasting, this study first uses the data preprocessing method to improve the data quality and second processes the data based on IEMD method to improve the reliability and integrity of training and learning data. At the same time, the multilayer network training model adopts the combination of reinforcement learning network and deep learning network. The concept module realizes dynamic data adjustment to the DNN prediction network to improve the fitting ability of the prediction model.

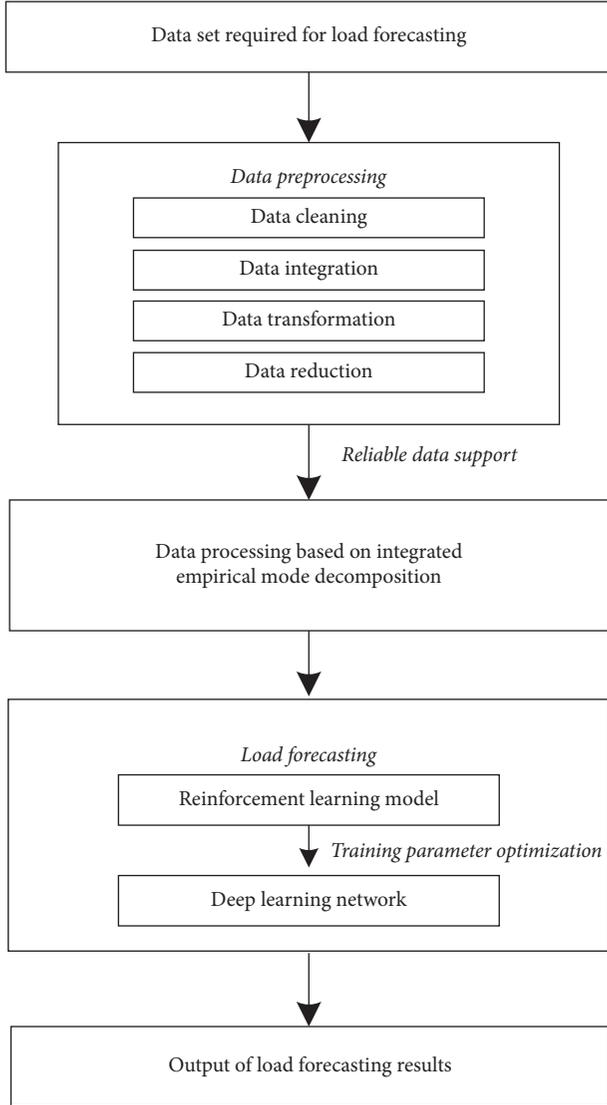


FIGURE 1: Smart grid power load forecasting analysis architecture.

**3.2. Load Data Preprocessing.** Power data acquisition has incomplete characteristics, heavy data storage tasks, inconsistent storage formats, and certain errors in data storage and is very vulnerable to noise interference. Therefore, data preprocessing is necessary to use data in practical tasks. In the actual data processing process, it mainly includes four steps: data cleaning, data integration, data specification, and data transformation. Among them, data transformation is the most important step in preprocessing. Transforming nonstandard data into standardized data can directly affect the prediction accuracy of the model.

There are two methods of data transformation, which can make different features have the same value range: normalization and standardization. Normalization is to rescale the data within the original range and scale it according to a certain proportion. The specific interval is adjusted according to actual needs to scale the data between  $[0, 1]$  and  $[-1, 1]$ . Suppose there is a sample of power grid state data, its normalization calculation formula is as follows:

$$\hat{\varphi}^{(n)} = \frac{\varphi^{(n)} - \min(\varphi^{(n)})}{\max(\varphi^{(n)}) - \min(\varphi^{(n)})}, \quad (1)$$

where  $\hat{\varphi}^{(n)}$  is the normalized power grid state data and  $\min(\varphi^{(n)})$  and  $\max(\varphi^{(n)})$  are the minimum and maximum values of characteristic  $\varphi$  on all samples, respectively.

Data standardization is a step of data preprocessing to make it in a specific range. The purpose is that the data features can have better convergence in back propagation, and each one-dimensional feature is adjusted to mean 0 and variance 1. Suppose there are  $N$  samples  $\{\varphi^{(n)}\}_{n=1}^N$ , for each one-dimensional characteristic  $\varphi$ , the mean and variance are calculated first. The mean formula and the variance calculation method are shown in the following formulas:

$$\mu = \frac{1}{N} \sum_{n=1}^N \varphi^{(n)}, \quad (2)$$

$$\sigma^2 = \frac{1}{N} \sum_{n=1}^N (\varphi^{(n)} - \mu)^2. \quad (3)$$

In the formula,  $\sigma$  cannot be 0. If the variance is  $\mu = 0$ , it means that this one-dimensional feature is indistinguishable from the task and can be deleted directly without being included in the feature dimension.

The power grid state data have time characteristics. The traditional method uses the wavelet analysis method to realize data processing, but there is the problem of complex and redundant calculation. The empirical mode decomposition (EMD) method can avoid the problem of specifying the basis function by wavelet method based on the time characteristics of the data itself and realize the reliable preprocessing of power grid state data.

EMD is used to analyze and process nonstationary non-linear signals with large fluctuation. Any signal can be decomposed into several intrinsic model functions (IMFs) to realize orderly and reliable data processing. However, due to the strong uncertainty of the current power grid load characteristics, there is a non-negligible reorganization error in the sorted data.

The IEMD method is used to decompose the signals with different variation characteristics, so as to realize the reliable support of power data. IEMD method overcomes the shortcomings of traditional EMD by adding adaptive white noise and calculating unique signal residual, so that the reconstructed signal is almost the same as the original signal. IEMD method not only overcomes the existing mode mixing phenomenon, but also reduces the reconstruction error by increasing the decomposition times.

$J_\gamma(\cdot)$  is defined as the calculation operator of the  $\gamma$  th mode component obtained through EMD,  $w(t)$  is Gaussian white noise conforming to normal distribution, and then the  $p$  th IMF of the original net load curve  $F(t)$  to be decomposed through IEMD is IMF $_p$ . The main contents of IEMD algorithm are as follows:

*Step 1.* Define  $\delta_0$  as the amplitude constant of Gaussian white noise,  $i = 1, 2, \dots, H$ , and conduct  $H$  experiments on

the original net load curve  $F(t) + \delta_0 w_i(t)$  with IEMD to obtain the first eigenmode function  $\text{IMF}_{i,1}$ . A component  $\text{IMF}_1$  obtained by IEMD decomposition is the mean of all  $\text{IMF}_{i,1}$  in  $H$  experiments.

$$\text{IMF}_1 = \frac{1}{H} \sum_{i=1}^H \text{IMF}_{i,1}. \quad (4)$$

*Step 2.* In the first stage, calculate the residual sequence  $\tau_1(t)$  for the first time as follows:

$$\tau_1(t) = F(t) - \text{IMF}_1. \quad (5)$$

*Step 3.* Define  $\delta_1$  as the amplitude constant of Gaussian white noise.  $J_1$  is the IEMD decomposition operator. At the same time, the sequence  $\tau_1(t) + \delta_1 J_1(w_i(t))$  is subjected to  $H$  EMD until its first IMF is obtained. Then, the second component  $\text{IMF}_2$  of IEMD can be calculated as follows:

$$\text{IMF}_2 = \frac{1}{H} \sum_{i=1}^H J_1(\tau_1(t) + \delta_1 J_1 w_i(t)). \quad (6)$$

*Step 4.* For each of the remaining stages  $p$ , repeat Step 3 and calculate the  $p + 1$  modal component as follows:

$$\begin{aligned} \tau_p(t) &= \tau_{p-1}(t) - \bar{I}_p, \\ \text{IMF}_{p+1} &= \frac{1}{M} \sum_{i=1}^M J_1(\tau_p(t) + \delta_p J_p w_i(t)). \end{aligned} \quad (7)$$

*Step 5.* Perform Step 4 until the obtained residual signal no longer executes any IMF, and the standard condition is that the IMF cannot be extracted from the residual.

Then, the final residual signal is as follows:

$$\tau(t) = F(t) - \sum_{p=1}^P \bar{I}_p. \quad (8)$$

Therefore, the original net load signal sequence  $F(t)$  is finally decomposed into

$$F(t) = \sum_{p=1}^P \bar{I}_p + \tau(t). \quad (9)$$

IEMD decomposition method can solve the traditional EMD model aliasing problem by adding standard normal white noise and has more self-adaptability in power grid short-term load forecasting.

**3.3. Multilayer Network Prediction Method.** To achieve an accurate and efficient short-term load forecasting method, this study combines reinforcement learning and deep learning to form a deep reinforcement learning model to provide excellent fitting and tracking ability. Among them, the reinforcement learning model is constructed by the inception module to improve the training parameters of in-

depth learning. The deep learning network is constructed by deep cyclic neural network.

The core of the reinforcement learning model is the policy gradient agent. In each time step  $t$ , the agent (agent) interacts with the environment  $s_t$  to determine the action of transferring a state to the DNN model. Then, the DNN model uses the newly selected state to complete the task. As shown in Figure 2, this study uses the inception network model as the main network of reinforcement learning model. Based on the currently selected model performance, it will be regarded as a reward for updating agent parameters. Next, we introduce three key points of agent, including environment, action, and reward function.

**Environment:** the previous state  $e_{t-1}$  and the current input  $x_t$  are combined to form the environment representation of the agent, as shown in the following formula:

$$z_t = e_{t-1} \oplus x_t. \quad (10)$$

**Action:** after observing the environment  $z_t$ , the agent should determine which state is best for the DNN unit. Formally, a state set  $z_p$  will be constructed, which retains the recently obtained  $p$  state and sets the maximum size  $p$  in advance. The agent takes measures by sampling the optimal state in  $z_p$  from the multinomial distribution  $\theta_p(p|z_t)$ , as shown in the following formula:

$$\begin{aligned} \eta &= \text{softmax}(\text{MLP}(z_t)), \\ \theta_p(p|z_t) &= \Pr(P = p|z_t) = \prod_{i=1}^p \eta_i^{(p=i)}, \end{aligned} \quad (11)$$

where  $[p = i]$ . If  $p = i$ , it evaluates to 1, otherwise it is 0. MLP represents a multilayer perceptron that transforms  $z_t$  into a vector with dimension  $k$ , and softmax function is used to transform the vector into probability distribution  $\eta$ .  $\eta_i$  is the  $i$  th element in  $\eta$ . Then, the  $z_{1-k}$  is transferred to the DNN unit for further calculation.

**Reward function:** the appropriate reward function can guide the agent to select a series of best skip actions to train a better prediction model. Through the interaction between reward and environment, the agent is encouraged to select the best skip to improve the performance of the model. The reward function is shown in the following formula:

$$J = \log(\hat{y}_T | e_T). \quad (12)$$

This model adopts the simplified inception network structure. In the inception module, the hidden state vectors of each time step are connected to construct a hidden state matrix, and a set of optimized features are generated from the current and nearby hidden state vectors.

For DNN training model, firstly, the parameters of the network need to be selected, including the number of input/output layer nodes and hidden layers. The specific DNN training network is shown in Figure 3.

The input layer network parameters of DNN model are closely related to the predicted power load data, which mainly affect the number of nodes in the input layer. The parameters of the hidden layer generally select the trial and error method. By constantly changing the parameters and

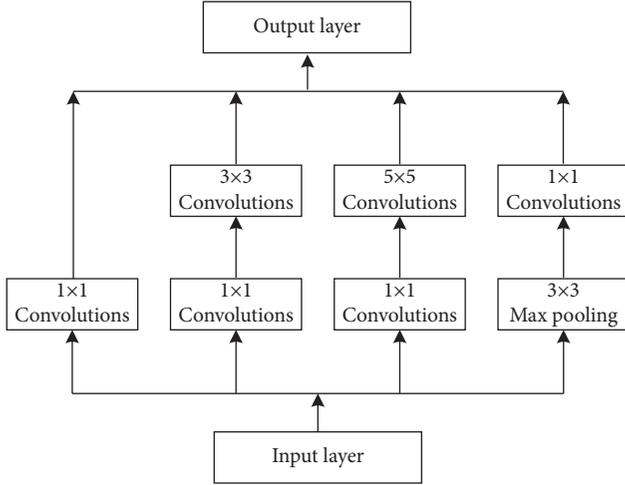


FIGURE 2: Inception module of reinforcement learning model.

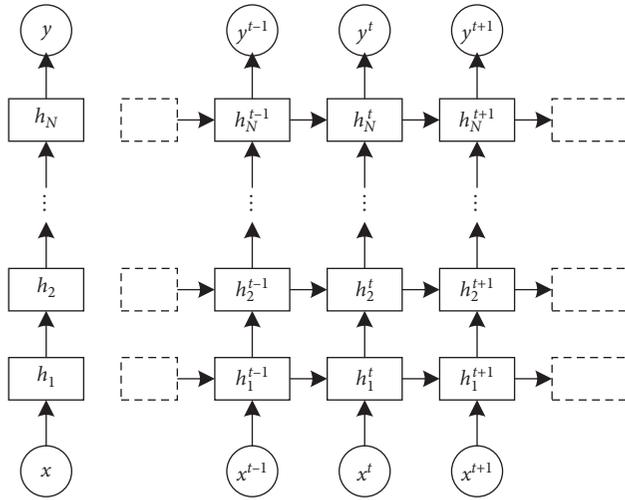


FIGURE 3: DNN training network model.

comparing the analysis results, the number of layers and nodes of the hidden layer can be determined. The specific network parameters are given in the experimental part of the next section and will not be repeated here. Then, the model is trained, including forward propagation and directional propagation training of DNN network.

DNN maps the input sequence of  $x$  value to the corresponding sequential output  $y$ . The learning process is carried out from  $t = 1$  to  $t = \nu$ . For the time step  $t$ , the network neuron parameters of the  $s^{\text{th}}$  layer update their shared state by the following equation:

$$\begin{aligned} \alpha_s^{(t)} &= \beta_s + J_s \cdot h_s^{(t-1)} + x^{(t)}, \\ h_s^{(t)} &= \kappa(\alpha_s^{(t)}), \\ y_{\text{forecast}}^{(t)} &= \beta_N + J_N \cdot h_s^{(t-1)} + h_s^{(t)}, \end{aligned} \quad (13)$$

where  $x^{(t)}$  is the input data of step  $t$ ;  $y_{\text{forecast}}^{(t)}$  is the corresponding prediction result;  $h_s^{(t)}$  is the sharing state of the volume layer network at time  $t$ ;  $\kappa$  is the excitation function, and tanh function is generally used;  $\alpha_s^{(t)}$  represents the input characteristics of the  $s$  layer at the step cycle; and  $J_s$

and  $J_N$  represent the model state coefficients of the network at the  $s^{\text{th}}$  layer.

The input features are screened by mutual information theory. Mutual information comes from the concept of entropy in information theory, which is used to represent the amount of information shared among multiple variables, so it is often used as a variable selection tool. The mutual information entropy of input data  $X$  and prediction data  $Y$  can be defined by the following formula:

$$H(A, B) = \sum_{i=1}^n \sum_{j=1}^m p(a_i, b_j) \ln \left( \frac{p(a_{i+1}, b_{j+1})}{p(a_{i+1})p(b_{j+1})} \right), \quad (14)$$

where  $n$  and  $m$  are the data collective quantity of the data  $A$  to be analyzed and the prediction result  $B$ . The probability that the value of  $A$  of the data to be analyzed is  $a$  can be defined as  $p(a)$ . The probability that the prediction result  $Y$  is  $b$  can be defined as  $p(b)$  accordingly;  $p(a_i, b_j)$  defines the formula for the joint probability calculation of data sets  $A$  and  $B$ .

For the deep reinforcement learning model, the essence of the model is multicore parameter network model, and its parameter adaptability is very important for the accurate analysis of the prediction model. Therefore, this study uses chaotic particle swarm optimization algorithm to realize the adaptive adjustment of model configuration.

Correction formula is as follows:

$$v_{k+1} = \delta v_k + \alpha_1 \beta_1 p_{b,k} + \alpha_2 \beta_2 p_{g,k}, \quad (15)$$

where  $v_k$  is the velocity vector of the particle;  $p_{b,k}$  is the optimal solution for the particle's current self;  $p_{g,k}$  is the optimal solution for population history;  $\beta_1$  and  $\beta_2$  are pseudorandom numbers between 0 and 1;  $\delta$  is inertia weight; and  $\alpha_1$  and  $\alpha_2$  are acceleration constants.

The steps of chaotic particle swarm optimization are as follows:

- (1) Initialize the population and set the built-in parameters of the model
- (2) Search for individual optimization and global optimization
- (3) Use equation (15) to update particle velocity and position and weight
- (4) Calculate the objective function of each particle and retain the best performance particles in the population
- (5) the best particles in the population by chaos to update the local best points and the global best points
- (6) when the global optimal solution meets the requirements

Figure 4 shows the main flowchart of the multilayer network power load forecasting method proposed in this study.

## 4. Experiment and Analysis

The experimental simulation is carried out in the high-performance working machine, and the simulation operation scenario is shown in Table 1.

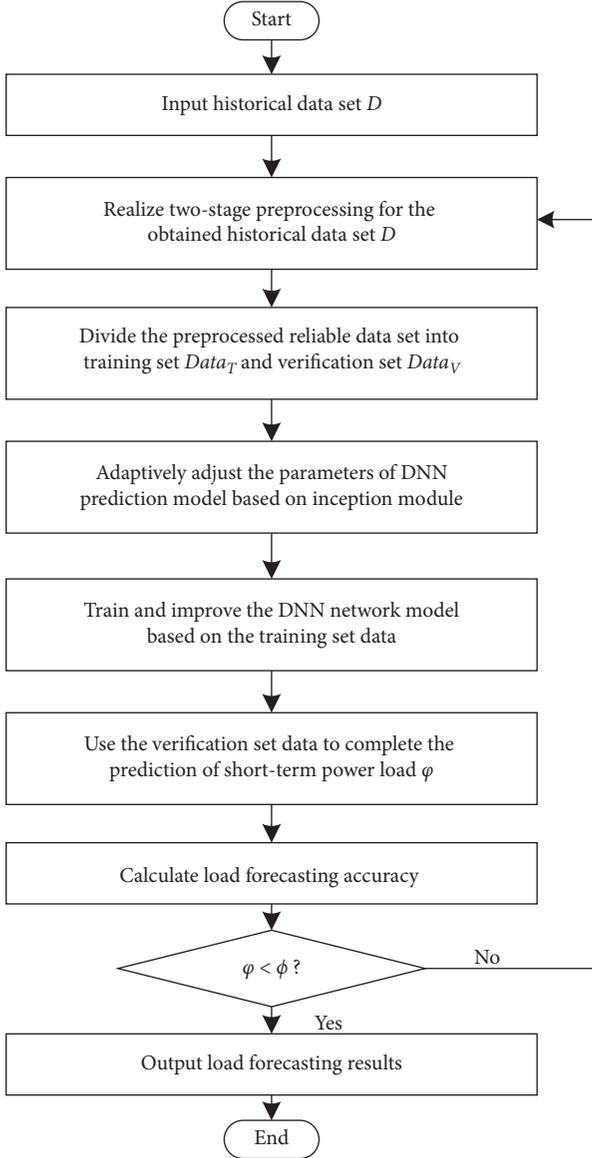


FIGURE 4: Multilayer network power load forecasting method.

TABLE 1: Running scenarios of simulation experiment.

Configuration	Parameter
Operating system	Ubuntu Linux 18.04
GPU	NVIDIA GeForce RTX 3050
Memory	64 GB
Software programming	Python 3.8.0
Algorithm framework	PyTorch 1.5

Meanwhile, Table 2 shows the main parameter configuration of the short-term load forecasting network proposed in this study.

**4.1. Experimental Dataset.** The dataset containing 53,154 real-time load demands which come from 2,187 electrical users in an urban power grid company is used to verify the effectiveness of our proposed method. The sampling frequency

TABLE 2: Parameter configuration of forecasting network.

Configuration	Parameter
(a) DNN network	
Number of hidden layers	20
Number of neurons in layer 1–5 network	8192
Number of neurons in layer 6–10 network	4096
Number of neurons in layer 11–15 network	2048
Number of neurons in layer 16–20 network	1080
(b) Chaotic particle swarm optimization	
Population number	15
Maximum number of iterations	35
Acceleration constant $\alpha_1$	0.26
Acceleration constant $\alpha_2$	0.48

of load is collected every hour from August 3, 2020, to August 23, 2020, during which the 20-day load curve of each load is recorded. Figure 5 shows an example of short-term power consumption of some loads in the simulation data.

Because the simulation data set used is large, the experimental data are scaled into training set and test set in proportion. The purpose of training set data application is to train the model to a better level. The main purpose of prediction set application is to carry out validation test and evaluate the performance of the model. Divided by training set: test set = 8 : 2.

**4.2. Forecasting Network Evaluation Index.** Aiming to visually verify and evaluate the advantages of the proposed method, two evaluation indexes for evaluating the prediction model of time series are selected, namely mean absolute percentage error  $e_{MAPE}$  and root-mean-square error  $e_{RMSE}$ . The calculation formula is as follows:

$$e_{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - y_i^*}{y_i} \right| \times 100\%, \quad (16)$$

$$e_{RMSE} = \sqrt{\frac{\sum_{i=1}^n (y_i - y_i^*)^2}{n}}, \quad (17)$$

where  $y_i$  and  $y_i^*$  are the actual value and the predicted value of load demand at the  $i$ -th time, respectively;  $n$  is the number of all samples as the test set.

As can be seen from the calculation formulas (16) and (17),  $e_{MAPE}$  value represents the percentage of the deviation of the prediction result, while  $e_{RMSE}$  value represents the absolute value of the actual deviation of the prediction result. The clearer the  $e_{MAPE}$  and  $e_{RMSE}$  values, the better the prediction effect.

### 4.3. Analysis of Prediction Results

**4.3.1. Data Reliability Analysis.** To prove that the proposed method can achieve accurate load forecasting analysis, the traditional EMD method is used as the comparison method to process and analyze the sample data, respectively. Table 3 shows the processing results of the proposed method and the comparison method.

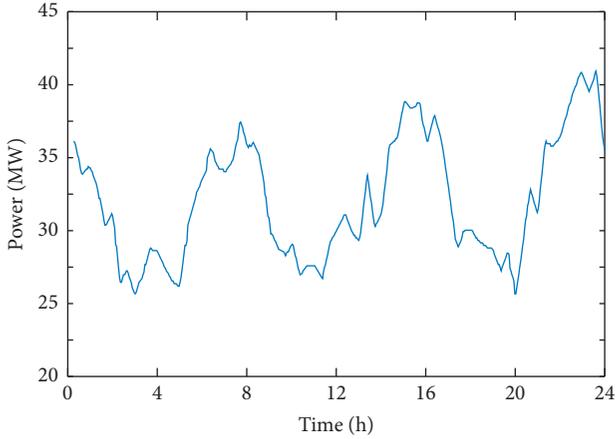


FIGURE 5: Example of partial simulation dataset.

TABLE 3: IMF analysis performance under different methods.

Component	$e_{MAPE}$ (%)	$e_{RMSE}$ (MW)
(a) IEMD method proposed in this study		
IMF 1	12.21	1.281
IMF 2	11.59	0.957
IMF 3	11.15	1.064
IMF 4	9.47	0.737
IMF 5	10.11	0.901
IMF 6	6.57	0.799
IMF 7	4.97	0.613
IMF 8	8.01	0.771
R	0.059	0.011
(b) Traditional EMD method		
IMF 1	15.25	2.149
IMF 2	10.51	1.578
IMF 3	12.54	1.746
IMF 4	10.87	0.994
IMF 5	12.57	1.024
Vb	0.501	0.084

As shown in Table 3, in terms of the number of IMFs, the IEMD data processing method can obtain 8 IMF components, which have more IMFs than that in the traditional EMD method, which proves that the data processing method can obtain the frequency components more effectively. At the same time, the frequency and amplitude of the proposed method for IMF 1-imf 3 are kept between 11–12.5% and 1.0–1.2 mw, respectively, which are higher than that of the other components, while the frequency and amplitude of each component of EMD method have little difference. The residual R value also shows that the proposed method has better data processing ability than the proposed traditional EMD method and can provide more reliable and complete data support for the subsequent prediction network.

**4.3.2. Load Forecasting Analysis.** In this study, reference [25] and reference [26] are used as comparison methods to realize the prediction and simulation experimental verification of the experimental data set, respectively. All methods realize the performance analysis under the same experimental scenario. Figure 6 shows the qualitative analysis results.

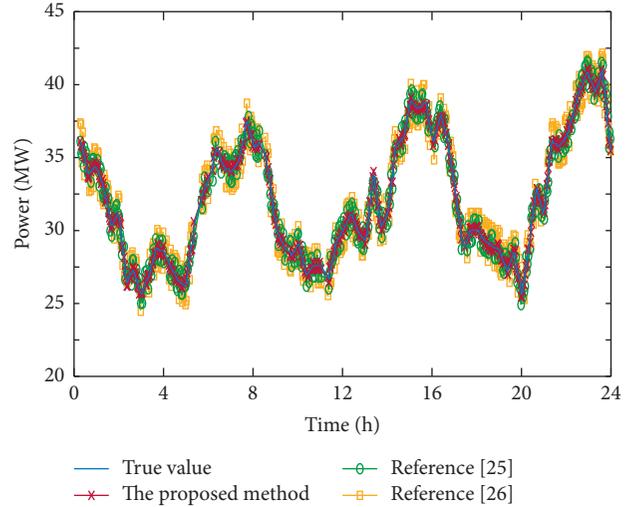


FIGURE 6: Qualitative evaluation of different prediction methods.

TABLE 4: Quantitative analysis results of different prediction methods.

Method	$e_{MAPE}$ (%)	$e_{RMSE}$ (MW)
The proposed method	10.01	2.156
Reference [25]	21.94	3.815
Reference [26]	25.17	4.671

As shown in Figure 6, the method proposed can better track the change curve than the comparison method and has excellent curve fitting ability. The reason is that in the early stage of network model training, this study has obtained reliable and complete data support through IEMD data processing method. More importantly, due to the introduction of inception module of reinforcement learning model, DNN prediction network can realize more rapid and accurate parameter adaptive change. Combined with the application of hybrid particle swarm optimization algorithm, the adaptability of prediction network is further strengthened, and its intuitive performance is more rapid and has accurate load curve tracking ability. On the contrary, the network models in references [25, 26] do not involve the adaptive adjustment of prediction network parameters, so it is difficult to achieve real-time and effective load analysis.

To more intuitively analyze the advantages of the prediction performance of the proposed method, this study makes a quantitative analysis. Table 4 shows the prediction performance evaluation indexes of different prediction methods.

As shown in Table 4, the mean absolute percentage error  $e_{MAPE}$  and root-mean-square error  $e_{RMSE}$  of the method proposed are 10.01% and 2.156 MW, respectively, 11.93% and 1.659 MW lower than reference [25], and 15.16% and 2.515 MW lower than reference [26], showing better load forecasting performance.

## 5. Conclusion

In view of the low performance of the current short-term load forecasting methods, this study combines reinforcement learning model with deep learning to propose an

efficient and accurate deep reinforcement learning load forecasting method. This method is mainly divided into two parts: one is to use the IEMD method to orderly process the load side power data, which can provide reliable and complete analysis data support for the prediction network model; secondly, the inception module is used to adjust the dynamic parameters of the DNN for load forecasting network, which can improve the tracking and fitting ability of the forecasting network. The simulation results confirm that the proposed method can track and analyze the power load energy consumption in the actual complex scenario efficiently and accurately.

There are still a few data errors in the existing load data. In this study, the data are eliminated manually. However, if the data screening process is ignored, it will have a great impact on the subsequent model prediction. Therefore, it can be combined with blockchain technology to realize distributed storage of power data, improve data transmission quality, and support short-term load forecasting of smart grid.

### Data Availability

The data used to support the findings of the study are included within the article.

### Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this study.

### Acknowledgments

This research was supported by the Research Projects of State Grid Corporation of China under Grant no. 1300-202013387A-0-0-00.

### References

- [1] Y. Li, D. Han, and Z. Yan, "Long-term system load forecasting based on data-driven linear clustering method," *Journal of Modern Power Systems and Clean Energy*, vol. 6, no. 2, pp. 306–316, 2018.
- [2] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Applied Energy*, vol. 235, no. 1, pp. 10–20, 2019.
- [3] W. Zhang, G. Mu, G. Yan, and J. An, "A power load forecast approach based on spatial-temporal clustering of load data," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 23, pp. 1–12, 2018.
- [4] D. Liu, L. Zeng, C. Li, K. Ma, Y. Chen, and Y. Cao, "A distributed short-term load forecasting method based on local weather information," *IEEE Systems Journal*, vol. 12, no. 1, pp. 208–215, 2018.
- [5] H. P. H. Anh, "Fuzzy load forecast with optimized parametric adjustment using jaya optimization algorithm," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, pp. 875–892, 2020.
- [6] L. Bartolucci, S. Cordiner, V. Mulone, and M. Santarelli, "Hybrid renewable energy systems: influence of short term forecasting on model predictive control performance," *Energy*, vol. 172, no. 1, pp. 997–1004, 2019.
- [7] Y. Wang, Q. Chen, M. Sun, C. Kang, and Q. Xia, "An ensemble forecasting method for the aggregated load with subprofiles," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3906–3908, 2018.
- [8] P. Zeng, M. Jin, and M. F. Elahe, "Short-term power load forecasting based on cross multi-model and second decision mechanism," *IEEE Access*, vol. 8, no. 1, pp. 184061–184072, 2020.
- [9] Y. Wang, Q. Chen, N. Zhang, and Y. Wang, "Conditional residual modeling for probabilistic load forecasting," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7327–7330, 2018.
- [10] X. Tang, Y. Dai, T. Wang, and Y. Chen, "Short-term power load forecasting based on multi-layer bidirectional recurrent neural network," *IET Generation, Transmission & Distribution*, vol. 13, no. 17, pp. 3847–3854, 2019.
- [11] X. Xiao, W. J. Xie, Y. Y. Zhou, W. Zhao, X. Liu, and C. Zhang, "Prediction and analysis of energy demand of high energy density AC/DC park based on spatial static load forecasting method," *Journal of Engineering*, vol. 16, no. 1, pp. 3388–3391, 2019.
- [12] S. H. Rafi, N. Al-Masood, S. R. Deeba, and E. Hossain, "A short-term load forecasting method using integrated CNN and LSTM network," *IEEE Access*, vol. 9, no. 1, pp. 32436–32448, 2021.
- [13] Y. Y. Li, S. Zhang, R. X. Hu, and N. Lu, "A meta-learning based distribution system load forecasting model selection framework," *Applied Energy*, vol. 294, no. 1, pp. 1–13, 2021.
- [14] M. Nadeem, M. Altaf, and A. Ahmad, "Short-term load forecasting using hybrid neural network," *International Journal of Applied Metaheuristic Computing*, vol. 12, no. 1, pp. 142–156, 2021.
- [15] J. Luo, T. Hong, and S. C. Fang, "Robust regression models for load forecasting [J]," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5397–5404, 2018.
- [16] S. S. Reddy, "Bat algorithm-based back propagation approach for short-term load forecasting considering weather factors," *Electrical Engineering*, vol. 100, no. 3, pp. 1297–1303, 2018.
- [17] B. Li, M. Z. Lu, and Y. Y. Zhang, "A weekend load forecasting model based on semi-parametric regression analysis considering weather and load interaction," *Energies*, vol. 12, no. 20, pp. 1–19, 2019.
- [18] C. Zhang and R. Li, "A novel closed-loop clustering algorithm for hierarchical load forecasting," *IEEE Transactions on Smart Grid*, vol. 12, no. 1, pp. 432–441, 2021.
- [19] J. C. Fang, D. Q. Shen, X. Y. Li, and H. Li, "An efficient power load forecasting model based on the optimized combination," *Modern Physics Letters B*, vol. 34, no. 16, pp. 1–13, 2020.
- [20] D. W. Wang and Z. W. Sun, "Big data analysis and parallel load forecasting of electric power user side," *Proceedings of the CSEE*, vol. 35, no. 3, pp. 527–537, 2015.
- [21] N. Mohan, K. P. Soman, and S. Sachin Kumar, "A data-driven strategy for short-term electric load forecasting using dynamic mode decomposition model," *Applied Energy*, vol. 232, no. 1, pp. 229–244, 2018.
- [22] P. Wang, L. Li, and C. Hu, "Short term load forecasting method of smart grid in cloud computing environment," *Science Technology and Engineering*, vol. 18, no. 7, pp. 153–158, 2018.
- [23] D. Gan, Y. Wang, S. Yang, and C. Kang, "Embedding based quantile regression neural network for probabilistic load forecasting," *Journal of Modern Power Systems and Clean Energy*, vol. 6, no. 2, pp. 244–254, 2018.

- [24] K. Kang, H. B. Zhang, C. K. Zhang, and B. Carl, "Short-term electrical load forecasting method based on stacked auto-encoding and GRU neural network," *Evolutionary Intelligence*, vol. 12, no. 3, pp. 385–394, 2019.
- [25] K. Yudiantaka, J. S. Kim, and H. Song, "Dual deep learning networks based load forecasting with partial real-time information and its application to system marginal price prediction," *Energies*, vol. 13, no. 1, pp. 1–17, 2019.
- [26] S. Motepe, A. N. Hasan, B. Twala, and R. Stopforth, "Effective load forecasting for large power consuming industrial customers using long short-term memory recurrent neural networks," *Journal of Intelligent and Fuzzy Systems*, vol. 37, no. 6, pp. 8219–8235, 2019.
- [27] R. C. Zheng, J. Gu, Z. J. Jin, H. Q. Peng, and Y. Zhu, "Load forecasting under data corruption based on anomaly detection and combined robust regression," *International Transactions on Electrical Energy Systems*, vol. 30, no. 7, pp. 1–16, 2020.