IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 13 December 2022, accepted 9 January 2023, date of publication 13 January 2023, date of current version 19 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3236663

RESEARCH ARTICLE

Short-Term Load Forecasting and Associated Weather Variables Prediction Using ResNet-LSTM Based Deep Learning

XINFANG CHEN[1], WEIRAN CHEN[2], (Student Member, IEEE), VENKATA DINAVAHI[2], (Fellow, IEEE), YIQING LIU[1], AND JILIN FENG[1]

[1]College of Information Engineering, Institute of Disaster Prevention, Sanhe, Hebei 065201, China
[2]Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada

Corresponding author: Weiran Chen (weiran4@ualberta.ca)

This work was supported by the Natural Science and Engineering Research Council of Canada (NSERC). The work of Xinfang Chen was supported by the China Scholarship Council (CSC).

ABSTRACT Short-term load forecasting is mainly utilized in control centers to explore the changing patterns of consumer loads and predict the load value at a certain time in the future. It is one of the key technologies for the smart grid implementation. The load parameters are affected by multi-dimensional factors. To sufficiently exploit the time series characteristics in load data and improve the accuracy of load forecasting, a hybrid model based on Residual Neural network (ResNet) and Long Short-Term Memory (LSTM) is proposed in this paper. First, the data with multiple feature parameters is reconstructed and input into ResNeT network for feature extraction. Second, the extracted feature vector is used as the input of LSTM for short-term load forecasting. Lastly, a practical example is used to compare this method with other models, which verifies the feasibility and superiority of input parameter feature extraction, and shows that the proposed combined method has higher prediction accuracy. In addition, this paper also carries out prediction experiments on the variables in the weather influencing factors.

INDEX TERMS Long short-term memory (LSTM), residual neural network (ResNet), ResNet-LSTM, short-term load forecasting, time-series features.

I. INTRODUCTION

Load forecasting is mainly used to explore the changes in a regular pattern and influencing factors of consumer load in a smart grid, and take necessary control actions. The accuracy of short-term load forecasting can provide the basis for many control center functions such as planning, dispatching, load frequency control, and economic operation, and it is of great significance to ensure dynamic balance, and the stable and reliable operation of smart grid, stable and reliable operation of smart grid [1]. With the increase in power demand, how to improve the prediction accuracy is an urgent problem to be solved [2]. The changes of various factors will affect the load, such as regional differences, socio-economic activities, natural climate, price, and other factors [3]. As a result, the load data has the characteristics of randomness, volatility, periodicity, and diversity. The key problem of load forecasting research is how to mine the internal law of load change from historical load data and find an accurate forecasting method [4].

To improve the speed and accuracy of load forecasting, scholars have put forward many methods. Each method has its specific advantages and disadvantages. Some are suitable for linear data prediction, while others are suitable for classified prediction [5]. Short-term load forecasting technology can be divided into four categories: statistical technology, artificial intelligence technology, knowledge-based expert system and hybrid technology, which are arranged in chronological order, as shown in Table 1.

The associate editor coordinating the review of this manuscript and approving it for publication was Emilio Barocio.

Statistical approaches require an explicit mathematical model which gives the relationship between load and several input factors. Classical statistical include multiple regression analysis, exponential smoothing, iterative reweighted least square, adaptive load forecasting, and stochastic time series. In [34], the peak demand of a typical growth system with genetic dynamic load characteristics is estimated. In [35], the regression-based daily peak load forecasting method with the transformation technique was presented; there is an obvious seasonal load change characterized as a nonlinear relationship between temperatures and loads. A new trend removal technique was based on optimal smoothing [36].

The expression computational intelligence is commonly used to refer to the fields of fuzzy systems, artificial neural networks (ANN), evolutionary computation, and swarm intelligence. Of these fields, neural networks are the subtype that is most often applied in load forecasting. In [37], an artificial neural network (ANN) method is applied to forecast the short-term load for a large power system. Adaptive Fuzzy Clustering model based on recursive Gustafson-Kessel algorithm and recursive weighted least-square is used to improve region division [38]. An attenuated radial basis function (RBF) neural network was used to train 24-hour power load forecasting. In [39], a modified deep residual network is formulated to improve the forecast results.

Expert systems are the result of advancements in Artificial Intelligence in the last two decades. These are rule-based methods, which make decisions based on the experience of experts. They are regarded as a supplementary method. A generalized technique for short-term load forecasting was tested using data from four diverse sites [40]. In [41], a knowledge-based expert system was implemented to support the choice of the most suitable load forecasting model for medium/long-term power system planning. A rule-based method was put forward in [42], which brought the prior expert knowledge of load curve into the statistical model.

Statistical methods and traditional machine learning methods can not take into account the high volatility, uncertainty, and time correlation of load data at the same time, so that the prediction accuracy is far from efficient, and there is still room for improvement [43]. Single methods often come with several types of disadvantages including low computational efficiency, high computational complexity, and high error percentage. Over the years, researchers have been working on building hybrid load forecasting models to obtain better accuracy with minimum error rate [44]. A composite load model was developed for predicting hourly electric loads 1-24 h ahead [45]. In [46], a hybrid demand model to enhance load modeling in distribution applications was proposed, which was conducted a state-space model and an ANN model. In [47], the merged particle swarm optimization with fuzzy neural networks is proposed. A neural network was proposed that combined elements of a convolutional neural network (CNN) and a long short memory network (LSTM) in [48].

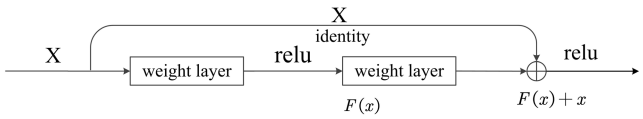

FIGURE 1. Residual module structure.

In addition to these, weather factors are crucial for load forecasting. Over the years, operational numerical weather prediction (NWP) models have been developed to improve the accuracy, reliability, and resolution of predictions [49]. At the same time, some scholars also proposed to use the deep learning model for weather forecasting [50]. In this paper, we continue to study the prediction of weather variables using a hybrid method based on Residual neural network (ResNet) and LSTM, so that when there are anomalies in the power load data or part of the weather variable factor data is lacking, the ResNet LSTM model can be used to predict the weather variables, such as dry-bulb temperature (drybt) and humidity, and then the power can be predicted.

The remainder of this paper is arranged as follows. The deep learning principle of ResNet and LSTM is introduced in Section II. In Section III, the combined model based on the ResNet-LSTM network for short-term load forecasting is presented. We also discuss the evaluation indices. Case studies are given. and discussed in Section IV, and the conclusions are presented in Section V.

II. PRINCIPLE OF DEEP LEARNING

A. RESIDUAL NETWORK

ResNet was proposed in [51], which solves the problem of degradation of deep neural networks, i.e., shallow networks are directly stacked into deep networks, which is difficult to make full use of the powerful feature extraction ability of deep networks, and the accuracy will also decline. ResNet has three features: ultra-deep network layer (breaking through 1000 layers), residual block, and accelerated training with the Batch Normalization algorithm. These features not only solve the problem of degradation gradient but also solve the problems of vanishing gradient and exploding gradient. For the vanishing gradient, i.e., when the error gradient of each layer is less than 1, the deeper the network is during backpropagation, the closer the gradient is to 0. Similarly, exploding gradient means that if the gradient error of each layer is greater than 1; the deeper the network, the bigger the gradient.

1) RESIDUAL BLOCK

To solve the degradation problem in deep networks, ResNet proposes a residual block. The residual block is composed of multiple cascaded convolution layers and a shortcut connection, also known as residual mapping and identity mapping. After accumulating their outputs, the output of the residual block is obtained through the *Relu* activation function. As shown in Figure 1, where the weight layer is convolution operation, X is the input, $F(x)$ is the residual mapping, and

TABLE 1. Four categories of load forecasting techniques.

Methods	Salient characteristics	Advantages	Drawbacks
Statistical Techniques [6]–[12]	Produce good results for linear time series.	The principle is simple; fast computation.	High data stability and a clear mathematical model are needed.
Artificial Intelligence (AI) Techniques [13]–[21]	Makes the machine competent for some complex work that needs human intelligence to complete.	Computational intelligence is a relatively new research field.	The data set is required to have high accuracy and integrity.
Knowledge-Based Expert Systems [22]–[25]	The rule-based methods take decisions based on the experience of experts.	Can be used as a supplementary method.	The relationship between rules is not transparent; inefficient search strategy; no learning ability.
Hybrid Techniques [26]–[33]	These approaches overcome some drawbacks of the original methods.	Can effectively carry out data mining and improve the accuracy of load forecasting.	Complex model and many parameters.

$H(x)$ is the output. The mapping relationship between the three is:

$$H(x) = F(x) + X. \tag{1}$$

In the residual network, the input X is directly short-circuited to the output of the network. At this time, the network will no longer directly learn the optimal mapping function but instead learn its residual, which is shown in equation (2):

$$F(x) = H(x) - X. \tag{2}$$

If the network tends to be optimal, continue to deepen the network. If residual mapping becomes 0, i.e., $F(x) = 0$, then $H(x) = X$; in theory, the network will always be in the optimal state, and the performance of the network will not decrease with the increase of depth. If $F(x) \neq 0$, but $F(x)$ is close to 0, then X approximates the actual mapping $H(x)$. In this way, the gradient degradation caused by network layer stacking is solved.

ResNet has five basic network structures with different layers, namely ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152. ResNet18 and ResNet34 are the residual blocks of two-layer convolution, and Resnet50, Resnet101, and Resnet152 are the residual blocks of three-layer convolution. There are also some differences in the implementation of the different residual blocks, which will not be discussed here.

2) BATCH NORMALIZATION ALGORITHM

This algorithm refers to batch standardization processing, i.e., the feature map of a batch of data meets the distribution law with a mean value of 0 and variance of 1. This operation is carried out between each full connection and excitation function so that the variation range of input X in the hidden layer will not be too large, and the input value will pass through the sensitive part of the activation function, to accelerate the convergence of the network and improve the accuracy.

Figure 2 illustrates the calculation process of mean $\mu_{\mathcal{B}}$ and variance $\sigma^2_{\mathcal{B}}$ of the batch normalization algorithm with

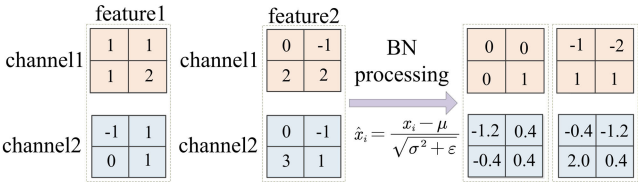

FIGURE 2. Mean $\mu_{\mathcal{B}}$ and variance $\sigma^2_{\mathcal{B}}$ of the batch normalization algorithm. Red and blue represent Channel 1 and Channel 2 respectively.

batch size 2. After the convolution and pooling operations of Image 1 and Image 2, the characteristic matrices Feature 1 and Feature 2 are obtained:

$$\begin{cases} x^{(1)} = \{1, 1, 1, 2, 0, -1, 2, 2\} \\ x^{(2)} = \{-1, 1, 0, 1, 0, -1, 3, 1\} \end{cases}, \tag{3}$$

$$\begin{cases} \mu_1 = \frac{1}{m}\sum_{i=1}^{m} x_i^{(1)} = 1 \\ \mu_2 = \frac{1}{m}\sum_{i=1}^{m} x_i^{(2)} = 0.5 \end{cases} \Longrightarrow \mu = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}, \tag{4}$$

$$\begin{cases} \sigma_1^2 = \frac{1}{m}\sum_{i=1}^{m} \left(x_i^{(1)} - \mu_1\right)^2 = 1 \\ \sigma_2^2 = \frac{1}{m}\sum_{i=1}^{m} \left(x_i^{(2)} - \mu_1\right)^2 = 1.5 \end{cases} \Longrightarrow \sigma^2 = \begin{bmatrix} 1 \\ 1.5 \end{bmatrix}, \tag{5}$$

$$\hat{x}_i = \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}} + \epsilon}}, \tag{6}$$

$$y_i = \gamma \hat{x}_i + \beta, \tag{7}$$

where $x^{(1)}$ represents the data in Channel 1 of all features of the batch. Similarly, $x^{(2)}$ represents the data in Channel 2 of all features of the batch. According to (4) and (5), we calculate the mean $\mu_{\mathcal{B}}$ variance $\sigma^2_{\mathcal{B}}$ of $x^{(1)}$ and $x^{(2)}$ respectively to obtain two vectors. Then we calculate the standard deviation of each channel according to (6) (ϵ is a small constant that prevents the denominator from being zero). There are also two parameters in (7): γ is used to adjust the variance of the numerical distribution, and β is used to adjust the position of the numerical mean. These two parameters are learned in the back-propagation process, which means that the neural

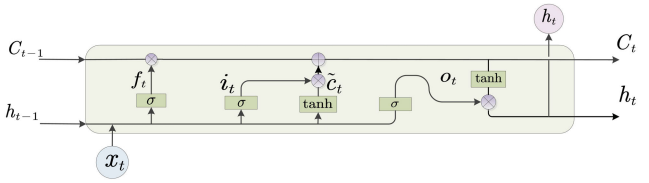

FIGURE 3. LSTM basic unit structure.

network will choose the most suitable distribution along with the training process. The default value of γ is 1, and the default value of β is 0.

B. LSTM NETWORK

The original intention of the RNN is to learn the long-term dependencies in time-series problems. The practice has proved that RNN has a good performance in dealing with this problem. At the same time, a large number of experiments show that the standard RNN will lead to a vanishing gradient and exploding gradient in the training process because of its iteration. To solve this problem, Hochreiter proposed LSTM [52], which is an improved RNN. LSTM adds a long-term memory function unit, which carries forward the transmission of data information [20]. The basic network unit is shown in Figure 3.

The basic unit of LSTM includes the forgetting gate, input gate, and output gate [53]. The input x_t, the state memory unit $C_{(t-1)}$ and the intermediate output $h_{(t-1)}$ determine jointly the forgetting part of the state memory unit. In the input gate, it determines the reserved vector in the state memory unit after sigmoid and *tanh* functions. The intermediate output h_t is determined by the updated C_t and output o_t, and the calculation is shown in (8)-(13).

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \tag{8}$$
$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \tag{9}$$
$$\tilde{c}_t = \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right), \tag{10}$$
$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \tag{11}$$
$$C_t = C_{t-1} \cdot f_t + i_t \cdot \tilde{c}_t, \tag{12}$$
$$h_t = o_t \cdot \tanh\left(C_t\right), \tag{13}$$

where f_t, i_t, $\tilde{c}_t$, o_t, C_t, and h_t as the state of forgetting gate, input gate, input node, output gate, intermediate output, and state unit respectively. W_f, U_f, b_f, W_i, U_i, b_i, W_o, U_o, and b_o are the door training parameters, *tanh* as the activation function.

The forget gate is used to forget selectively the unit state at the last time and correct the parameters, the input gate is used to update the state of the information, and the output gate is used to read, output, and correct the parameters. LSTM adopts the "Gates" structure to increase the transmission and exchange of information, this solves the problem of "gradient vanishing/explosion" in model training and can learn the long short-term dependence information of time series, which can be applied in many scenarios.

III. ResNet-LSTM MODEL

A. ResNet-LSTM STRUCTURE

The ResNet-LSTM model proposed in this paper consists of two parts: the ResNet is regarded as the pre-feature extraction unit and the LSTM is regarded as the time series feature learning unit. The model structure is shown in Figure 4.

As shown in Figure 4 (a), the Resnet18 network is used in the model, which consists of 5 parts, Conv1, and Conv2_x, Conv3_x, Conv4_x, and Conv5_x, respectively. Conv1 performs a convolution operation and a max-pooling operation. Each of the remaining parts consists of two residual blocks. The Resnet18 network is used in the model, and each residual block is composed of a two-layer convolution network. The features matrices of the two branches of the residual block are added and then output through the rule activation function. We input the data with the structure of (None, 56, 1) into the ResNet, where 56 is the number of feature parameters and the number of channels is 1. The number of convolution kernels in each part is 32, 32, 64, 128, and 256 respectively. The size of the convolution kernel is 3. The maximum pool is used to reduce the complexity of the model. The pool size is also 3 and the step size is 2. After ResNet extracts the features and outputs a three-dimensional vector (None, 2, 256) to the LSTM network.

In the residual block, when the shape of the input features matrix is consistent with that of the output features matrix, the main branch and the short connected branch can be added directly, i.e., the solid line residual structure. At this time, the stride of the convolution operation is 1, as shown in Figure 4 (b).

When the shape of the input features matrix is inconsistent with the shape of the output features matrix, i.e., there is a dashed residual block. At this time, a convolution operation with a strde of 2 is performed on the short connection branch, so that the shape of the features matrix of the short connection branch is consistent with that of the main branch, and then it is added, as shown in Figure 4 (c).

Through experiments, it was found that adding an LSTM network helps to improve the prediction ability of the model. The final model includes two layers of the LSTM network, and the number of units in each layer is $1,024, 256$. Dropout is used between each LSTM network layer to prevent the overfitting of the model. Finally, the predicted load value is output through two layers of Dense. The number of neuron nodes in the first layer is 64 and the number of neuron nodes in the second layer is 1.

B. EVALUATION INDICES

In this paper, Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE), and Absolute Percentage Error (APE) are used as evaluation indices [8] to evaluate the prediction level of the

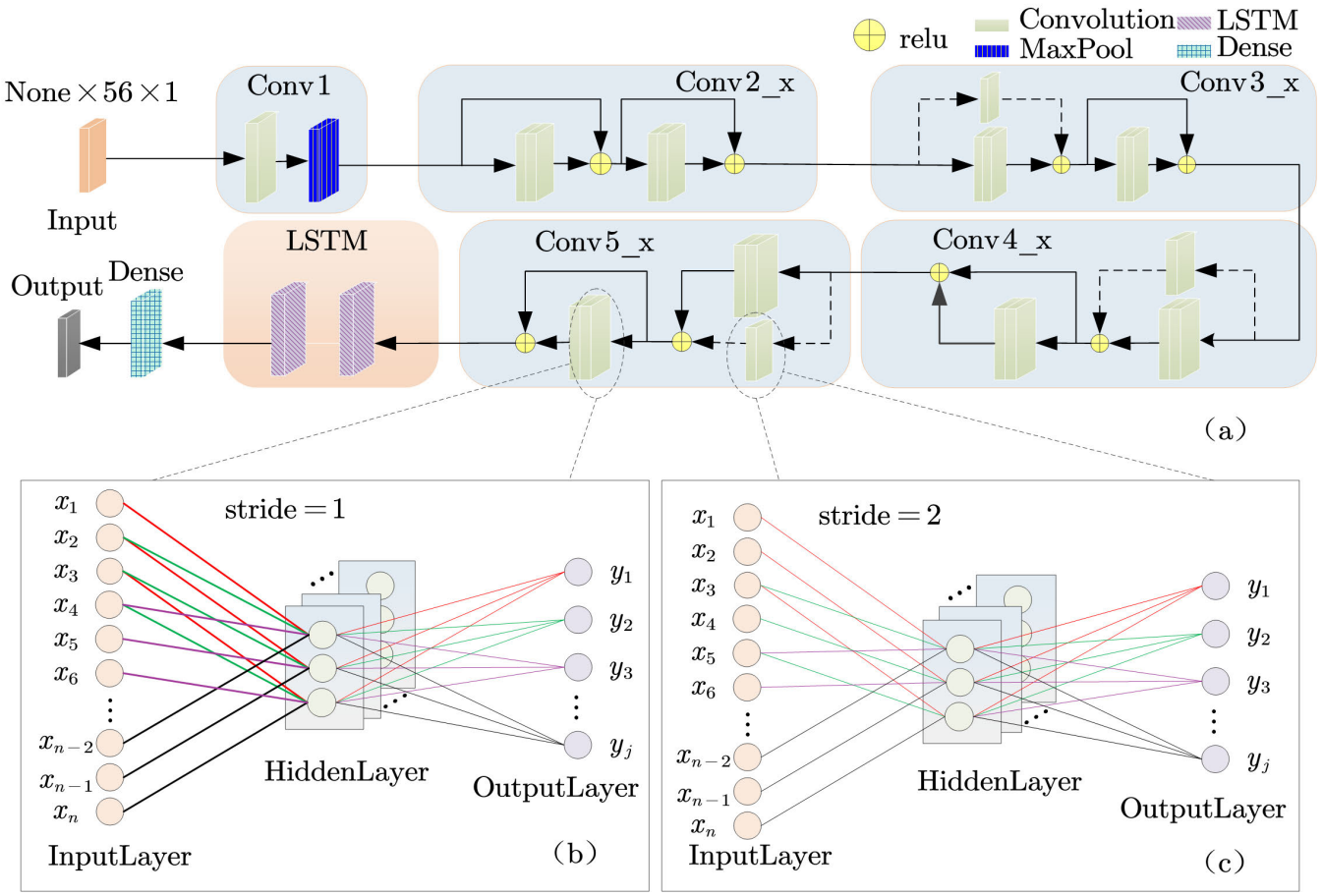

FIGURE 4. ResNet-LSTM model structure: (a) processing flow of ResNet-LSTM model; (b) CNN structure with step size of 1; (c) CNN structure with step size of 2.

model. These indices are calculated (14), (15), (16), and (17):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{N}}, \tag{14}$$

$$MAPE = \sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \times \frac{100}{N}, \tag{15}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right|, \tag{16}$$

$$APE = \sum_{i=1}^{n}\left|\hat{y}_i - y_i\right|, \tag{17}$$

where y_i is the actual load value of sampling point i, $\hat{y}_i$ is the load forecast value of sampling point i, and N is the number of sample points.

IV. EXAMPLE ANALYSIS

This paper used the publicly available online load data set published in Queensland, Australia from Jan. 1, 2006, to Dec. 31, 2010 [54]. The sampling interval is 30 minutes, including 48 sampling points every day, and the data set has a total of 87649 rows of data. It contains six feature parameters: load, dry-bulb temperature (drybt), dew point temperature (dewbt), wet-bulb temperature (wetbt), humidity, and price. The data for the first four years is the training set, the data from January 1 to November 30, 2010 is the validation set and the last month is the test set. The feature parameters are shown in Table 2.

TABLE 2. Feature parameters.

Type	Name	Detailed description
Date	Data time	Sampling interval 30 mins
Weather factors	dry-bulb temperature ($^\circ C$)	the true thermodynamic temperature of the measured air
	dew point temperature ($^\circ C$)	the temperature at which water vapor and water reach equilibrium
	wet-bulb temperature ($^\circ C$)	the thermodynamic adiabatic saturation temperature
	humidity (RH)	indicates the degree of atmospheric dryness in the air
Economic factor	price ($\$/MWh$)	price of $1kWh$
Power load	load (MW)	power generation per unit time under rated conditions

A. DATA PROCESSING

1) CORRELATION ANALYSIS

The analysis of the data set shows that the load changes periodically in a week, and the other parameters also have

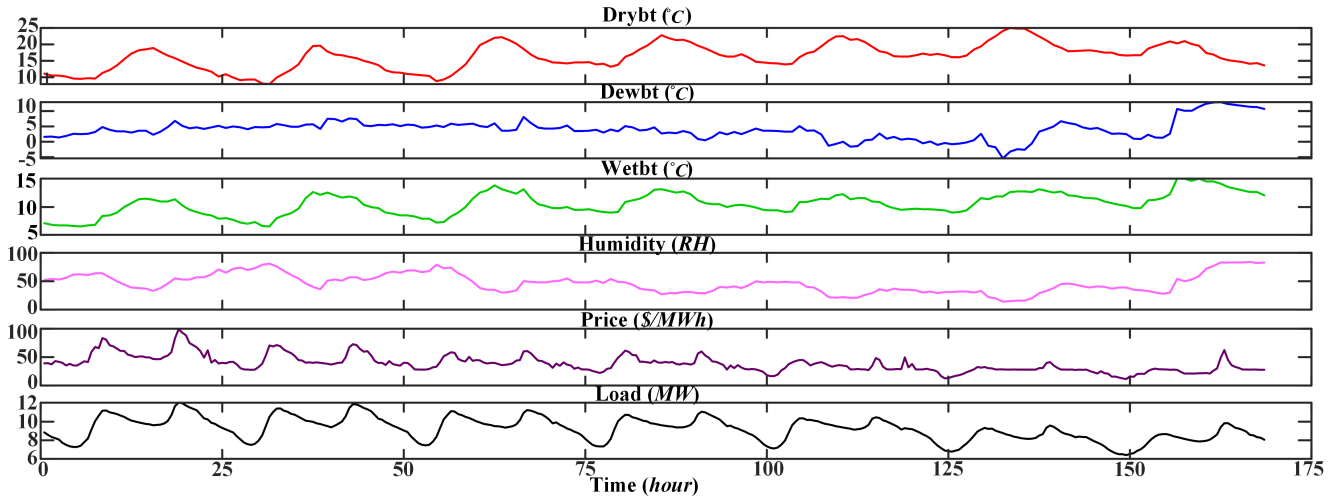

FIGURE 5. Curves of feature parameters in a week (drybt: dry-bulb temperature; dewbt: dew point temperature; wetbt: wet-bulb temperature).

this trend. Figure 5 shows the change process of feature parameters in a week.

The Pearson Correlation Coefficient method was used to analyze the correlation of the feature parameters [55], and the correlation thermodynamic diagram is shown in Figure 6.

It can be seen from Figure 5 and Figure 6 that the load has a strong positive correlation with dry-bulb temperature, wet-bulb temperature, and price, and a strong negative correlation with dew point temperature, and humidity. The work and rest habits of residents also affect the change of load. Therefore, the information on holidays and weekdays are also used as feature parameters to participate in the training. Besides, weather factors also have the features of time-series and periodicity, and there is a strong correlation between the variables. In this paper, the two weather variables of drybt and humidity are selected, predicted, and analyzed.

2) ABNORMAL DATA PROCESSING

Load data and meteorological data may have abnormal values due to communication errors or data loss. In the process of calculation and analysis, abnormal values distort the results and affect the prediction accuracy, which needs to be eliminated. In this paper, the box diagram is used to analyze and correct the outliers in the data set.

3) RESTRUCTURING THE FEATURE PARAMETERS

To couple the characteristic information of data and accurately mine the temporal characteristics between data, we construct the load data at any time into time-series data with multiple feature parameters. After reconstruction, each load value has 56 feature parameters, including weather, price, holiday, weekdays, sampling time, and load values of the previous 48 sampling points. The data reconstruction process is shown in Figure 7. The size of load vector at each sampling point is $8 + n$, and the n position in $t_0 \sim t_{n-1}$ sampling point is filled with 0. Starting from the t_1 sampling point, fill the load value of the previous sampling point into the feature vector in turn. The feature vector of load at t_n sampling point are $Weather_{t_n}$, $Price_{t_n}$, $Holiday_{t_n}$, $Weekday_{t_n}$, t_n, $Load_{t_0}$, $Load_{t_1}$, …, and $Load_{t_{n-1}}$.

Similarly, the training data can be changed to regenerate the hybrid ResNet-LSTM model for predicting the weather variable factors (drybt, dewbt, wetbt, and humidity). Taking the predicted variable drybt as an example, the data is reconstructed considering its time series characteristics, combined with its historical data. The data reconstruction process is shown in Figure 8. To be consistent with the size of the load input vector, the size of the drybt vector is also $8 + n$, i.e., 8 relevant features including dewbt, wetbt, humidity, load, price, holiday, weekdays, and sampling time, and the previous n_{th} drybt sampling points. The blank positions of the vector can be set to 0, as needed.

In this paper, the value of n is 48. After that, the feature vector of the sampling points is reconstructed in this way. Among them, weather includes four variables: drybt, dewbt, wetbt, and humidity. At time $t_0 \sim t_{47}$, there is 0 in the eigenvector, so it does not participate in the training of the model.

4) DATA NORMALIZATION

Different feature parameters have different properties and orders of magnitude. No standardized training will weaken the impact of the lower order of magnitude data. In the experiment, Min-Max Scaling is used to linearly transform the data x, and the data size is constrained between [0, 1].

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{18}$$

where x^* is the post value after normalization; $x_{\max}$ is the maximum value in the sample data; $x_{\min}$ is the minimum value in the sample.

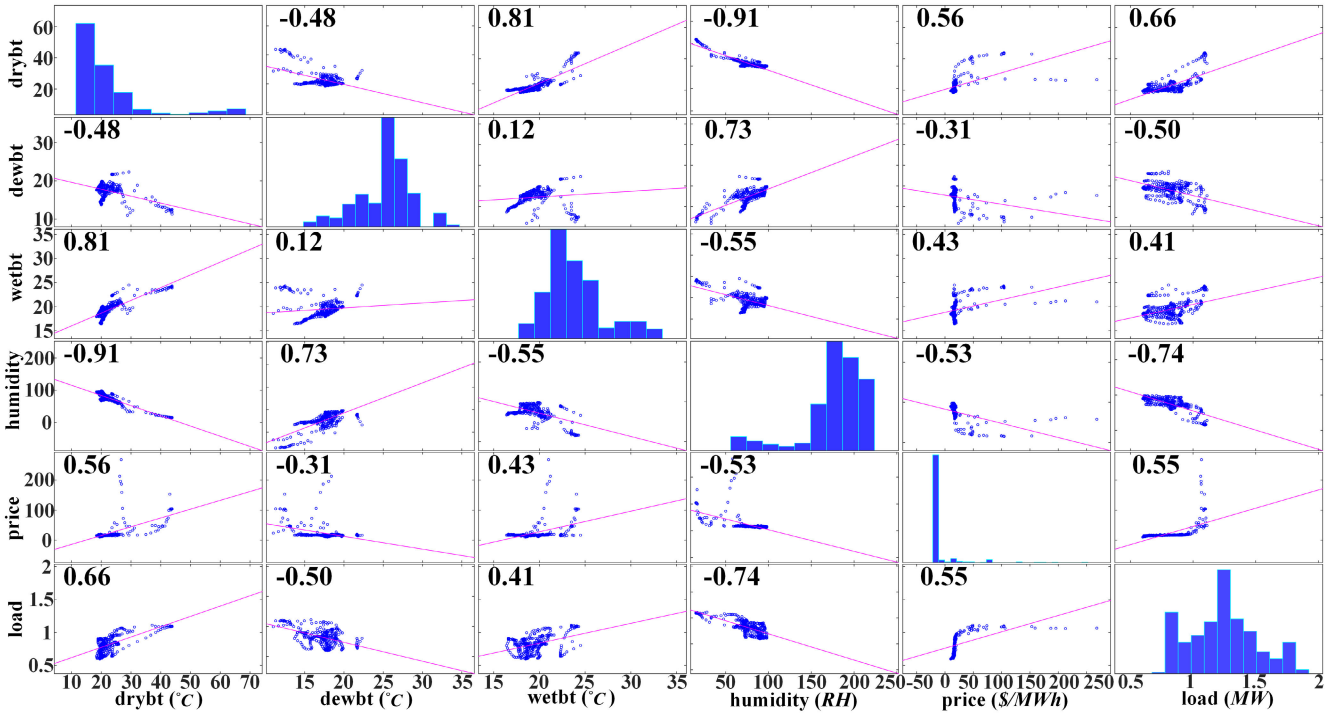

FIGURE 6. Matrix diagram of characteristic parameter correlation.

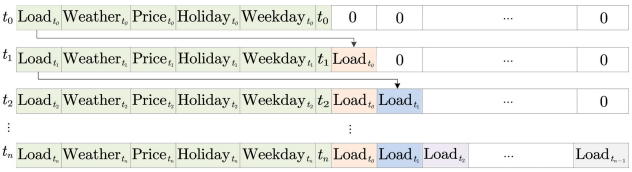

FIGURE 7. Data reconstruction process of power forecasting.

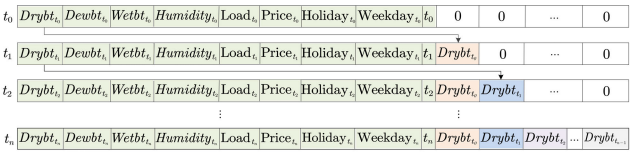

FIGURE 8. Data reconstruction process of drybt forecasting..

B. EXPERIMENTAL CONFIGURATION

In the experiment, a computer workstation with Intel (R) core (TM) i7-9750h CPU and NVIDIA Quadro GPU was used, which was built with TensorFlow-GPU2.6.2, Keras2.6.0, CUDA11.6, and cuDNN8.3.2 combined development environment. The software environment is the Keras framework based on TensorFlow. Keras provides a concise and consistent programming interface and has the characteristics of modularization. At the same time, it supports the free combination of models, which helps users quickly understand the neural network architecture and reduces the repeated writing of code. The MSE is used as the loss function and trained by the Adam optimizer [56].

TABLE 3. Model error comparison for data on Dec. 1, 2010.

Model	MAPE	RMSE	MAE
MLR	0.645	67.980	50.862
LSTM	0.798	88.504	59.246
CNN	0.696	83.036	61.442
ResNet	0.692	70.137	58.416
CNN-LSTM	0.939	101.998	77.085
ResNet-LSTM	0.607	61.095	49.910

C. ANALYSIS OF EXPERIMENTAL RESULTS

In the experiment, LSTM is used as the baseline model and the control variable method is used to optimize the parameters. To illustrate the positive role of the proposed model in short-term load forecasting, the model is compared with multiple linear regression (MLR), CNN, LSTM, CNN-LSTM, and ResNet methods used in short-term load forecasting [57]. Each model predicts from Dec. 1, 2010, Dec. 1 to Dec. 2, 2010, and Dec. 1 to Dec. 7, 2010.

Table 3 shows the evaluation indicators of each model predicted on Dec. 1, 2010. Compared with other models, the three evaluation indicators of the ResNet-LSTM model are the lowest. MAPE decreased respectively by 3.80%, 30.56%, 12.79%, 12.28%, and 35.36%. RMSE decreased respectively by 10.13%, 30.97%, 26.42%, 12.89%, and 40.10%.

Table 4 shows the evaluation indicators of each model predicted on Dec. 1 to Dec. 2, 2010. MAPE decreased respectively by 19.35%, 25.16%, 20.58%, 12.96%, and 36.77%. RMSE decreased respectively by 25.34%, 31.69%, 28.46%, 15.38%, and 38.86%.

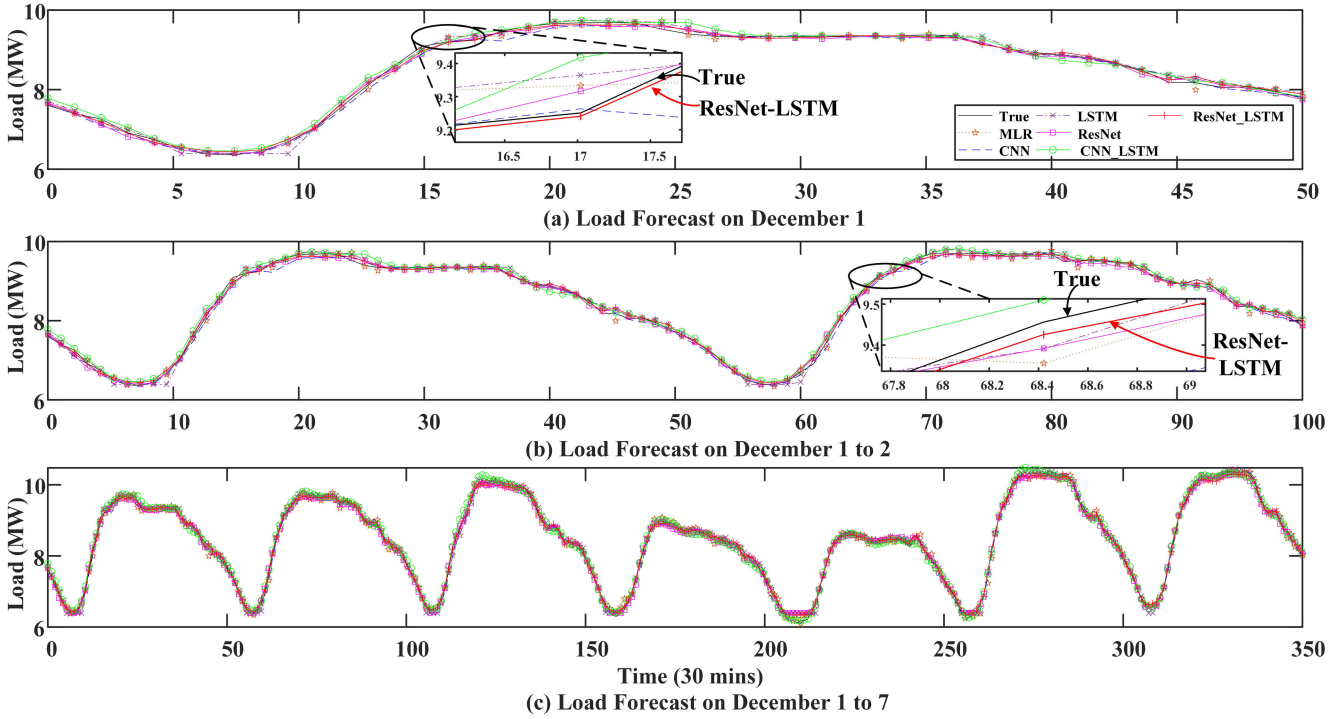

FIGURE 9. Comparison of single network models for short-term load forecasting.

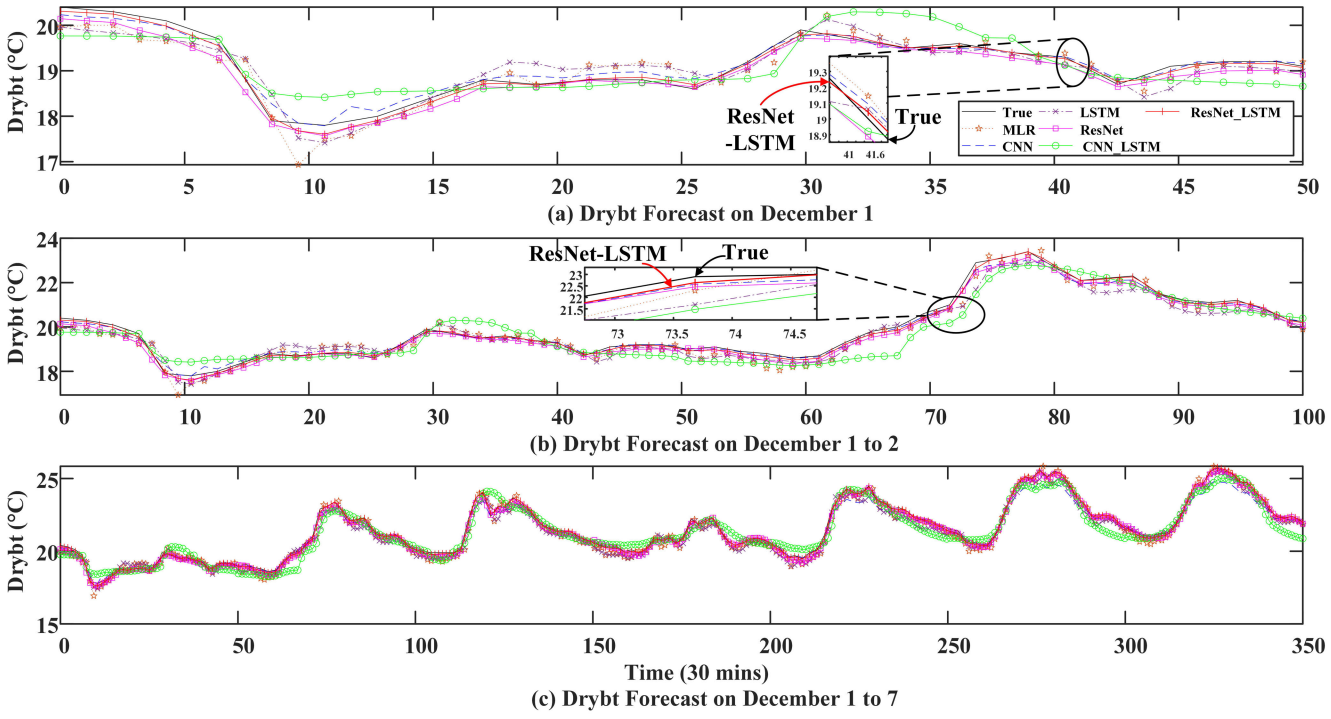

FIGURE 10. Comparison of single network models for dry-bulb temperature prediction.

Table 5 shows the evaluation indicators of each model predicted on Dec. 1 to Dec. 7, 2010. MAPE decreased respectively by 22.36%, 27.37%, 26.75%, 20.72%, and 28.07%. RMSE decreased respectively 25.35%, 29.61%, 27.24%, 20.99%, and 29.50%.

It can be seen from Table 3, Table 4, and Table 5 that when the data volume is small, the simple and well-designed MLR method is consistent with the evaluation indicators of the ResNet-LSTM model proposed in this paper. With the increase of data, the evaluation index of the MLR model

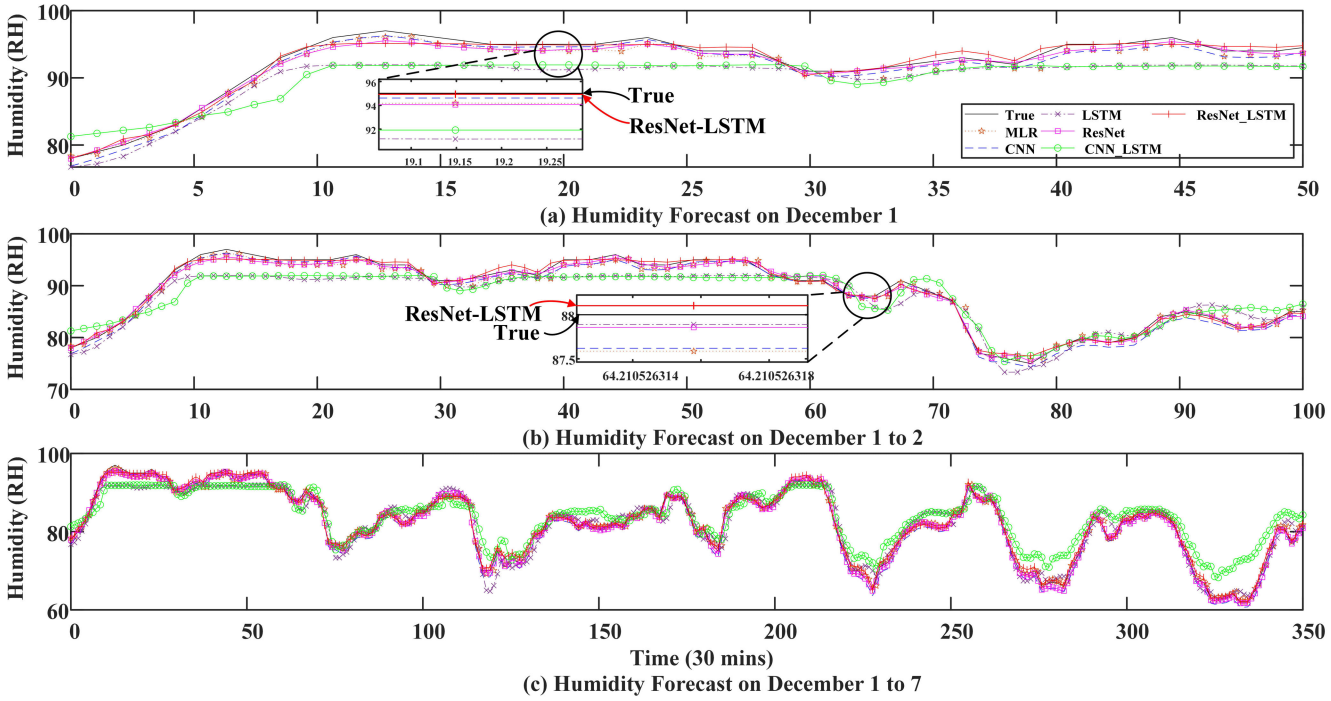

FIGURE 11. Comparison of single network models for humidity prediction.

TABLE 4. Model error comparison on Dec. 1 to Dec. 2, 2010.

Model	MAPE	RMSE	MAE
MLR	0.708	78.431	60.426
LSTM	0.763	85.714	62.684
CNN	0.719	82.949	61.442
ResNet	0.656	69.198	55.997
CNN-LSTM	0.903	95.769	74.246
ResNet-LSTM	0.571	58.554	47.031

TABLE 5. Model error comparison on Dec. 1 to Dec. 7, 2010.

Model	MAPE	RMSE	MAE
MLR	0.769	84.307	69.938
LSTM	0.822	89.396	65.982
CNN	0.815	86.494	67.747
ResNet	0.753	79.653	62.488
CNN-LSTM	0.830	89.264	69.501
ResNet-LSTM	0.597	62.933	49.301

becomes larger, because linear regression is based on the assumption that the data changes linearly.

The evaluation indexes of other neural network models are larger than those of the ResNet-LSTM model. Hence, the efficiency of the ResNet-LSTM model is higher in comparison to other methods.

Table 8 shows the load APE of different models at each sampling point on Dec. 1, 2010. To make the data understandable, the APE in the table is reduced by 100 times. It can be seen from Figure 12 that the APE values of the six models for load forecasting at 48 sampling points on Dec. 1, 2010, are 24.41, 27.56, 31.35, 28.04, 37, and 23.96 respectively.

The diagram of load forecasting for each model on Dec. 1, 2010, Dec. 1 to Dec. 2, 2010, and Dec. 1 to Dec. 7, 2010, is shown in Figure 9 (a), (b), and (c), respectively.

It can be seen that in the smoothing stage of load change, the prediction of each model is accurate and there is little difference. In the area with severe load fluctuation, the prediction results of ResNet are relatively accurate. The

TABLE 6. Parameters and FLOPs of different model.

Model	LSTM	CNN	ResNet	CNN-LSTM	ResNet-LSTM
Parameters	19009	16449	1063743	27137	7622527
FLOPs	82504	32772	2188403	94698	24208637

TABLE 7. Running time (sec) of different models forecast load.

Model	Dec. 1	Dec. 1 to Dec. 2, 2010	Dec. 1 to Dec. 7, 2010
MLR	4.5938	4.5	7.0469
LSTM	7.1094	8.6875	19.422
CNN	6.6562	8.0156	15.672
ResNet	7.8125	9.625	20.25
CNN-LSTM	7.4844	9.9375	25.562
ResNet-LSTM	10.391	13.734	24.594

ResNet-LSTM model has a better ability to capture the load change trend than the ResNet model. From the perspective of the combined model, the predicted change curve of model ResNet-LSTM is closer to the real value than that of the CNN-LSTM model.

TABLE 8. Absolute percentage error of different deep learning models on load forecasting over a 24-hour period.

Time ($hours$)	Actual (kW)	MLR		CNN model		LSTM model		ResNet model		CNN-LSTM model		ResNet-LSTM model	
		Predicted	APE	Predicted	APE	Predicted	APE	Predicted	APE	Predicted	APE	Predicted	APE
0	7605.3	7651.726	0.46426	7617.7085	0.12408	7710.519	1.0522	7667.963	0.6266	7782.955	1.7765	7643.067	0.3777
0.5	7397.18	7394.815	0.02365	7386.228	0.10952	7464.776	0.676	7437.541	0.4036	7558.808	1.6163	7415.254	0.1807
1	7257.96	7171.928	0.86032	7092.9375	1.65023	7172.485	0.8548	7119.155	1.388	7296.047	0.3809	7187.612	0.7035
1.5	6914.44	7010.635	0.96195	6910.961	0.03479	6991.648	0.7721	6814.926	0.9951	7046.855	1.3241	6899.826	0.1461
2	6744.45	6704.956	0.39494	6625.7563	1.18694	6749.052	0.046	6672.253	0.722	6766.164	0.2171	6671.34	0.7311
2.5	6496.76	6557.198	0.60438	6514.988	0.18228	6392.162	1.046	6488.832	0.0793	6554.406	0.5765	6552.815	0.5605
3	6380.44	6375.541	0.04899	6437.507	0.57067	6392.468	0.1203	6412.101	0.3166	6466.849	0.8641	6454.128	0.7369
3.5	6360.47	6400.0475	0.39577	6372.725	0.12255	6392.533	0.3206	6399.471	0.39	6456.474	0.96	6438.988	0.7852
4	6453.71	6418.352	0.35358	6424.889	0.28821	6392.529	0.6118	6431.107	0.226	6532.929	0.7922	6488.5	0.3479
4.5	6685.78	6638.552	0.47228	6645.6055	0.40175	6392.25	2.9353	6690.455	0.0468	6727.797	0.4202	6757.196	0.7142
5	7018.36	7050.39	0.3203	6990.146	0.28214	7011.065	0.073	7052.871	0.3451	7146.985	1.2863	7033.718	0.1536
5.5	7567.88	7555.636	0.12244	7486.045	0.81835	7650.71	0.8283	7658.255	0.9038	7742.416	1.7454	7652.007	0.8413
6	8120.03	8001.264	1.18766	8030.157	0.89873	8173.181	0.5315	8234.296	1.1427	8296.879	1.7685	8175.833	0.558
6.5	8611.67	8576.494	0.35176	8462.811	1.48859	8565.987	0.4568	8515.215	0.9645	8622.85	0.1118	8486.214	1.2546
7	9087.19	8999.931	0.87259	8885.853	2.01337	9009.313	0.7788	8899.723	1.8747	8937.397	1.4979	8966.944	1.2025
7.5	9205.01	9316.12	1.1111	9206.556	0.01546	9318.919	1.1391	9206.962	0.0195	9222.052	0.1704	9190.412	0.146
8	9250.63	9333.733	0.83103	9263.093	0.12463	9364.729	1.141	9316.444	0.6581	9417.819	1.6719	9240.789	0.0984
8.5	9467.37	9346.985	1.20385	9223.728	2.43642	9412.62	0.5475	9442.885	0.2449	9495.105	0.2773	9448.26	0.1911
9	9595.48	9557.666	0.37814	9401.458	1.94022	9577.668	0.1781	9493.08	1.024	9588.115	0.0736	9539.754	0.5573
9.5	9662.11	9679.905	0.17795	9569.333	0.92777	9716.453	0.5434	9607.543	0.5457	9696.715	0.346	9621.099	0.4101
10	9677.35	9678.076	0.00726	9609.537	0.67813	9749.689	0.7234	9638.646	0.387	9739.792	0.6244	9633.61	0.4374
10.5	9693.99	9634.746	0.59244	9586.301	1.07689	9729.864	0.3587	9591.963	1.0203	9720.099	0.2611	9618.615	0.7538
11	9674	9680.871	0.06871	9614.795	0.59205	9718.054	0.4405	9581.038	0.9296	9678.421	0.0442	9649.414	0.2459
11.5	9520.06	9727.943	2.07883	9629.073	1.09013	9689.054	1.6899	9583.526	0.6347	9685.242	1.6518	9616.27	0.9621
12	9399.9	9374.264	0.25636	9484.799	0.84899	9542.084	1.4218	9518.64	1.1874	9684.001	2.841	9492.432	0.9253
12.5	9341.91	9265.648	0.76262	9363.433	0.21523	9388.519	0.4661	9420.082	0.7817	9519.651	1.7774	9390.242	0.4833
13	9321.29	9314.726	0.06564	9343.899	0.22609	9316.597	0.0469	9316.193	0.051	9396.992	0.757	9304.797	0.1649
13.5	9342.38	9344.88	0.025	9339.511	0.02869	9310.147	0.3223	9288.899	0.5348	9343.952	0.0157	9290.201	0.5218
14	9334.95	9295.714	0.39236	9363.113	0.28163	9349.114	0.1416	9301.133	0.3382	9346.701	0.1175	9304.683	0.3027
14.5	9327.12	9373.538	0.46418	9344.95	0.1783	9363.042	0.3592	9276.223	0.509	9353.532	0.2641	9314.05	0.1307
15	9333.91	9307.505	0.26405	9336.819	0.02909	9355.642	0.2173	9317.211	0.167	9353.964	0.2005	9364.314	0.304
15.5	9328.46	9362.751	0.34291	9328.264	0.00196	9356.425	0.2797	9346.575	0.1812	9356.564	0.281	9356.945	0.2849
16	9356.4	9278.707	0.77693	9287.926	0.68474	9343.352	0.1305	9319.954	0.3645	9338.176	0.1822	9355.555	0.0084
16.5	9377.76	9405.057	0.27297	9326.953	0.50807	9345.346	0.3241	9294.002	0.8376	9343.434	0.3433	9317.219	0.6054
17	9356.08	9357.938	0.01858	9325.47	0.3061	9357.707	0.0163	9299.005	0.5708	9342.097	0.1398	9308.138	0.4794
17.5	9142.33	9202.822	0.60492	9178.226	0.35896	9337.955	1.9563	9186.935	0.446	9281.814	1.3948	9136.036	0.0629
18	9026.38	8994.708	0.31672	8975.486	0.50894	9079.108	0.5273	8992.372	0.3401	9047.359	0.2098	9024.712	0.0167
18.5	8924.31	8911.574	0.12736	8894.676	0.29634	8938.068	0.1376	8837.705	0.866	8849.366	0.7494	8868.152	0.5616
19	8864.62	8900.072	0.35452	8841.012	0.23608	8882.6	0.1798	8843.169	0.2145	8736.178	1.2844	8939.977	0.7536
19.5	8800.34	8753.721	0.46619	8761.9795	0.38361	8820.567	0.2023	8775.494	0.2485	8675.98	1.2436	8839.774	0.3943
20	8667.82	8654.509	0.13311	8638.25	0.2957	8688.63	0.2081	8656.48	0.1134	8630.271	0.3755	8682.206	0.1439
20.5	8485.33	8513.198	0.27868	8498.275	0.12945	8502.223	0.1689	8382.44	1.0289	8475.26	0.1007	8449.39	0.3594
21	8184.69	8325.506	1.40816	348.16	1.6347	8378.172	1.9348	8244.02	0.5933	8357.73	1.7304	8245.76	0.6107
21.5	8174.3	7997.145	1.77155	8180.0845	0.05784	8287.279	1.1298	8274.138	0.9984	8220.028	0.4573	8333.957	1.5966
22	8055.39	8091.79	0.364	8032.6772	0.22713	8186.938	1.3155	8109.825	0.5443	8180.392	1.25	8131.832	0.7644
22.5	8067.28	8032.909	0.34371	7965.0938	1.02186	8074.243	0.0696	8025.371	0.4191	8049.019	0.1826	8071.51	0.0423
23	7956.62	8023.728	0.67108	7899.996	0.56624	8023.846	0.6723	7905.456	0.5116	7938.13	0.1849	7981.408	0.2479
23.5	7782.86	7787.669	0.04809	7818.7124	0.35852	7908.109	1.2525	7752.447	0.3041	7828.746	0.4589	7892.456	1.096

D. WEATHER VARIABLES PREDICTION

Figure 10 (a), (b), and (c) show the fitting curve between the predicted and actual values of drybt by different models on Dec. 1, 2010, Dec. 1 to Dec. 2, 2010, and Dec. 1 to 7, 2010. It can be seen that there are great differences in the prediction results between the models. Among them, the prediction results of ResNet and ResNet-LSTM models are relatively accurate. The predicted change curve of the ResNet-LSTM model is closer to the actual value. It can be seen from Figure 12 that the APE values of the six models for drybt forecasting at 48 sampling points on Dec. 1, 2010, are 10.80, 4.24, 11.77, 7.54, 15.71, and 3.08, respectively.

Figure 11 (a), (b), and (c) shows the fitting comparison between the predicted and actual values of humidity variables between different models. It can be seen that the predicted values of different models are different to varying degrees. When the humidity fluctuates greatly, the ResNet-LSTM model can better capture the changing trend, and the model has a good prediction effect. It can be seen from Figure 12 that the APE values of the six models for humidity forecasting at 48 sampling points on Dec. 1, 2010, are 37.37, 33.46 122.64, 26.88, 133.18, and 24.01, respectively.

Figure 12 shows the APE values of load, dry bulb temperature, and temperature for the six models on Dec. 1, 2010. It can be seen visually that ResNet_LSTM has good accuracy in feature extraction and relationship processing of time series data.

E. COMPUTATIONAL COMPLEXITY ANALYSIS

To describe a deep learning model, in addition to accuracy, the number of floating point operations (FLOPs) and the number of parameters are normally used to illustrate the complexity.

In the ResNet-LSTM model, the number of parameters of a convolution kernel is given as:

$$parameter_k = K^2 * C_{in} + 1, \tag{19}$$

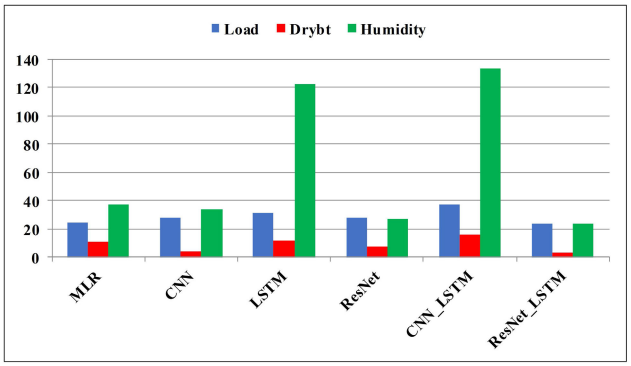

FIGURE 12. Histogram of absolute percentage error for various models.

Similarly, the number of parameters of a convolution layer is given as:

$$parameter_l = K^2 * C_{in} * C_{out} + C_{out}, \tag{20}$$

We can calculate the value of FLOPs according to (21):

$$FLOP_s = 2HW\left(C_{in}K^2 + 1\right)C_{out}, \tag{21}$$

where H, W, and C_{in} are the height, width, and number of channels of the input feature map, K is the kernel width, and C_{out} is the number of output channels [58].

After programming, FLOPs and parameters of different neural network models are shown in Table 6. In addition, the operation time required for load forecasting by different models on Dec. 1, 2010, Dec. 1 to Dec. 2, 2010, and Dec. 1 to Dec. 7, 2010 is shown in Table 7; the running time unit in the table in seconds.

It can be seen from Table 6 and Table 7 that although the training parameters and FLOPs of the ResNet-LSTM model are larger than those of other models, the time required for forecasting loads on different days is within 30 seconds. Considering the accuracy, the ResNet-LSTM model proposed in this paper is an effective short-term load forecasting method.

V. CONCLUSION

With the increasing requirements of power system short-term load forecasting accuracy, this paper proposed a combined model based on ResNet-LSTM, which uses the feature expression ability of ResNet to extract effective features and processes the temporal relationship through the LSTM network. The conclusions are as follows:

1) Considering the time series feature of load and taking into account the characteristics of historical data, the load data of Queensland, Australia was reconstructed. The feature parameters include date factors, weather factors, economic factors, and the historical load data of the previous 48 sampling times.

2) Compared with other machine learning models, the ResNet-LSTM model proposed in this paper gives full play to the feature extraction advantages of ResNet. At the same time, it gives full play to the ability of the LSTM network to better fit the timing and complex nonlinear relationship, and further excavates the potential timing feature expression of power load data. Experiments show that compared with other short-term load forecasting methods, the model has better forecasting accuracy.

3) The weather variables also have the features of time series and periodicity. This paper also makes an experimental comparative analysis of the two variables of dry-bulb temperature and humidity, which further shows efficacy of ResNet-LSTM model in processing time series weather data.

To sum up, this paper not only proposes a short-term load forecasting combination model for multi-dimensional input characteristic parameters but also reconstructed the data, which provides ideas and references for researchers to further explore how to improve the accuracy of load forecasting for various smart grid applications.

REFERENCES

[1] 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, pp. 184061–184072, 2020.

[2] Y. Wang, Q. Chen, N. Zhang, and Y. Wang, "Conditional residual modeling for probabilistic load forecasting," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 7327–7330, Nov. 2018.

[3] Y. Wang, Q. Xia, and C. Kang, "Secondary forecasting based on deviation analysis for short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 26, no. 2, pp. 500–507, May 2011.

[4] W. Charytoniuk, M. S. Chen, and P. Van Olinda, "Nonparametric regression based short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 725–730, Aug. 1998.

[5] H. Xiaoyan, L. Bingjie, S. Jing, L. Hua, and L. Guojing, "A novel forecasting method for short-term load based on TCN-GRU model," in *Proc. IEEE Int. Conf. Energy Internet (ICEI)*, Sep. 2021, pp. 79–83.

[6] A. D. Papalexopoulos and T. C. Hesterberg, "A regression-based approach to short-term system load forecasting," *IEEE Trans. Power Syst.*, vol. 5, no. 4, pp. 1535–1547, Nov. 1990.

[7] G. A. N. Mbamalu and M. E. El-Hawary, "Load forecasting via suboptimal seasonal autoregressive models and iteratively reweighted least squares estimation," *IEEE Trans. Power Syst.*, vol. 8, no. 1, pp. 343–348, Feb. 1993.

[8] C. Guan, P. B. Luh, L. D. Michel, and Z. Chi, "Hybrid Kalman filters for very short-term load forecasting and prediction interval estimation," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3806–3817, Nov. 2013.

[9] C. C. Gaudes, I. Santamaria, J. Via, E. M. Gomez, and T. S. Paules, "Robust array beamforming with sidelobe control using support vector machines," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 574–584, Feb. 2007.

[10] Y. Chakhchoukh, P. Panciatici, and L. Mili, "Electric load forecasting based on statistical robust methods," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 982–991, Oct. 2010.

[11] J. W. Taylor, "Short-term load forecasting with exponentially weighted methods," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 458–464, Feb. 2012.

[12] I. Goethals, K. Pelckmans, J. A. K. Suykens, and B. De Moor, "Subspace identification of Hammerstein systems using least squares support vector machines," *IEEE Trans. Autom. Control*, vol. 50, no. 10, pp. 1509–1519, Oct. 2005.

[13] A. G. Bakirtzis, J. B. Theocharis, S. J. Kiartzis, and K. J. Satsios, "Short term load forecasting using fuzzy neural networks," *IEEE Trans. Power Sys.*, vol. 10, no. 3, pp. 1518–1524, Aug. 1995.

[14] H. Mori and H. Kobayashi, "Optimal fuzzy inference for short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 11, no. 1, pp. 390–396, Feb. 1996.

[15] M.-Y. Chow and H. Tram, "Application of fuzzy logic technology for spatial load forecasting," *IEEE Trans. Power Syst.*, vol. 12, no. 3, pp. 1360–1366, Aug. 1997.
[16] E. Gonzalez, M. A. Jaramillo, and D. Carmona, "Monthly electric energy demand forecasting based on trend extraction," *IEEE Trans. Power Syst.*, vol. 21, no. 4, pp. 1946–1953, Nov. 2006.
[17] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Trans. Power Syst.*, vol. 16, no. 1, pp. 44–55, Feb. 2001.
[18] M. Delfanti, G. P. Granelli, P. Marannino, and M. Montagna, "Optimal capacitor placement using deterministic and genetic algorithms," *IEEE Trans. Power Syst.*, vol. 15, no. 3, pp. 1041–1046, Aug. 2000.
[19] M. Alamaniotis, A. Ikonomopoulos, and L. H. Tsoukalas, "Evolutionary multiobjective optimization of kernel-based very-short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 27, no. 3, pp. 1477–1484, Aug. 2012.
[20] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, Jan. 2019.
[21] X. Tang, Y. Dai, Q. Liu, X. Dang, and J. Xu, "Application of bidirectional recurrent neural network combined with deep belief network in short-term load forecasting," *IEEE Access*, vol. 7, pp. 160660–160670, 2019.
[22] D. Srinivasan, S. S. Tan, C. S. Cheng, and E. K. Chan, "Parallel neural network-fuzzy expert system strategy for short-term load forecasting: System implementation and performance evaluation," *IEEE Trans. Power Syst.*, vol. 14, no. 3, pp. 1100–1106, Aug. 1999.
[23] K.-H. Kim, J.-K. Park, K.-J. Hwang, and S.-H. Kim, "Implementation of hybrid short-term load forecasting system using artificial neural networks and fuzzy expert systems," *IEEE Trans. Power Syst.*, vol. 10, no. 3, pp. 1534–1539, Aug. 1995.
[24] M. S. Tsai, "Development of an object-oriented service restoration expert system with load variations," *IEEE Trans. Power Syst.*, vol. 23, no. 1, pp. 219–225, Feb. 2008.
[25] B. Goehry, Y. Goude, P. Massart, and J.-M. Poggi, "Aggregation of multi-scale experts for bottom-up load forecasting," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 1895–1904, May 2020.
[26] K. L. Ho, Y. Y. Hsu, C. F. Chen, T. E. Lee, C. C. Liang, T. S. Lai, and K. K. Chen, "Short term load forecasting of Taiwan power system using a knowledge-based expert system," *IEEE Trans. Power Sys.*, vol. 5, no. 4, pp. 1214–1221, Nov. 1990.
[27] A. Khotanzad, R. Afkhami-Rohani, and D. Maratukulam, "ANNSTLF-artificial neural network short-term load forecaster-generation three," *IEEE Trans. Power Syst.*, vol. 13, no. 4, pp. 1413–1422, Nov. 1998.
[28] C.-M. Huang, C.-J. Huang, and M.-L. Wang, "A particle swarm optimization to identifying the ARMAX model for short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 1126–1133, May 2005.
[29] G.-C. Liao and T.-P. Tsao, "Application of a fuzzy neural network combined with a chaos genetic algorithm and simulated annealing to short-term load forecasting," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 330–340, Jun. 2006.
[30] L. Wu and M. Shahidehpour, "A hybrid model for day-ahead price forecasting," *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1519–1530, Aug. 2010.
[31] S. Li, P. Wang, and L. Goel, "A novel wavelet-based ensemble method for short-term load forecasting with hybrid neural networks and feature selection," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 1788–1798, May 2016.
[32] T. A. Farrag and E. E. Elattar, "Optimized deep stacked long short-term memory network for long-term load forecasting," *IEEE Access*, vol. 9, pp. 68511–68522, 2021.
[33] K. Park, S. Yoon, and E. Hwang, "Hybrid load forecasting for mixed-use complex based on the characteristic load decomposition by pilot signals," *IEEE Access*, vol. 7, pp. 12297–12306, 2019.
[34] E. H. Barakt, M. A. Qayyum, M. N. Hamed, and S. A. AI-Rashe, "Short-term peak demand forecasting in fast developing utility with inherent dynamic load characteristics," *IEEE Trans. Power Sys.*, vol. 5, no. 3, pp. 813–824, Aug. 1990.
[35] T. Haida and S. Muto, "Regression based peak load forecasting using a transformation technique," *IEEE Trans. Power Syst.*, vol. 9, no. 4, pp. 1788–1794, Nov. 1994.
[36] D. G. Infield and D. C. Hill, "Optimal smoothing for trend removal in short term electricity demand forecasting," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 1115–1120, Aug. 1998.
[37] K. Y. Lee, Y. T. Cha, and J. H. Park, "Short-term load forecasting using an artificial neural network," *IEEE Trans. Power Syst.*, vol. 7, no. 1, pp. 124–132, Feb. 1992.
[38] G. Černe, D. Dovzan, and I. Skrjanc, "Short-term load forecasting by separating daily profiles and using a single fuzzy model across the entire domain," *IEEE Trans. Ind. Electron.*, vol. 65, no. 9, pp. 7406–7415, Sep. 2018.
[39] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3943–3952, Jul. 2019.
[40] S. Rahman and O. Hazim, "A generalized knowledge-based short-term load-forecasting technique," *IEEE Trans. Power Syst.*, vol. 8, no. 2, pp. 508–514, May 1993.
[41] M. S. Kandil, S. M. El-Debeiky, and N. E. Hasanien, "Long-term load forecasting for fast developing utility using a knowledge-based expert system," *IEEE Trans. Power Syst.*, vol. 17, no. 2, pp. 491–496, May 2002.
[42] S. Arora and J. W. Taylor, "Short-term forecasting of anomalous load using rule-based triple seasonal methods," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3235–3242, Aug. 2013.
[43] P. Liyun, Z. Wenjun, W. Sining, and H. Lu, "Short-term load forecasting based on DenseNet-LSTM fusion model," in *Proc. IEEE Int. Conf. Energy Internet (ICEI)*, Sep. 2021, pp. 84–89.
[44] A. A. Mamun, M. Sohel, N. Mohammad, M. S. H. Sunny, D. R. Dipta, and E. Hossain, "A comprehensive review of the load forecasting techniques using single and hybrid predictive models," *IEEE Access*, vol. 8, pp. 134911–134939, 2020.
[45] J. H. Park, Y. M. Park, and K. Y. Lee, "Composite modeling for adaptive short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 6, no. 2, pp. 450–457, May 1991.
[46] S. A. Villalba and C. A. Bel, "Hybrid demand model for load estimation and short term load forecasting in distribution electric systems," *IEEE Trans. Power Del.*, vol. 15, no. 2, pp. 764–769, Apr. 2000.
[47] G.-C. Liao, "A novel particle swarm optimization approach combined with fuzzy neural networks for short-term load forecasting," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Jun. 2007, pp. 1–6.
[48] H. H. Goh, B. He, H. Liu, D. Zhang, W. Dai, T. A. Kurniawan, and K. C. Goh, "Multi-convolution feature extraction and recurrent neural network dependent model for short-term load forecasting," *IEEE Access*, vol. 9, pp. 118528–118540, 2021.
[49] G. Mulder, F. J. van Leijen, J. Barkmeijer, S. de Haan, and R. F. Hanssen, "Estimating single-epoch integrated atmospheric refractivity from InSAR for assimilation in numerical weather models," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, May 2022, Art. no. 4108612.
[50] M. A. R. Suleman and S. Shridevi, "Short-term weather forecasting using spatial feature attention based LSTM model," *IEEE Access*, vol. 10, pp. 82456–82468, 2022.
[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jan. 2016, pp. 770–778.
[52] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[53] D. Chen, J. Zhang, and S. Jiang, "Forecasting the short-term metro ridership with seasonal and trend decomposition using loess and LSTM neural networks," *IEEE Access*, vol. 8, pp. 91181–91187, 2020.
[54] *Australia Load Data*. Accessed: Dec. 2010. [Online]. Available: https://github.com/weiran4/AustraliaData
[55] J. Wang and N. Zheng, "A novel fractal image compression scheme with block classification and sorting based on Pearson's correlation coefficient," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3690–3702, Sep. 2013.
[56] A. H. Khan, X. Cao, S. Li, V. N. Katsikis, and L. Liao, "BAS-ADAM: An ADAM based approach to improve the performance of beetle antennae search optimizer," *IEEE/CAA J. Automat. Sinica*, vol. 7, no. 2, pp. 461–471, Feb. 2020.
[57] N. Amral, C. S. Ozveren, and D. King, "Short term load forecasting using multiple linear regression," in *Proc. 42nd Int. Universities Power Eng. Conf.*, Sep. 2007, pp. 1192–1198.
[58] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," 2016, *arXiv:1611.06440*.

XINFANG CHEN received the B.Eng. degree from Liaoning Petrochemical University, in 2000, and the M.S. degree in engineering from Dalian Maritime University, in 2008. He is currently an Associate Professor and the Master's Supervisor at the Institute of Disaster Prevention. He is mainly engaged in the research of big data storage and analysis, data visualization, machine learning, data mining, and emergency information processing technology.

WEIRAN CHEN (Student Member, IEEE) received the B.Eng. degree in electrical engineering from Harbin Engineering University, Harbin, Heilongjiang, China, in 2018. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems, power electronic systems, and field programmable gate arrays.

VENKATA DINAVAHI (Fellow, IEEE) received the B.Eng. degree in electrical engineering from the Visvesvaraya National Institute of Technology (VNIT), Nagpur, India, in 1993, the M.Tech. degree in electrical engineering from the Indian Institute of Technology (IIT) Kanpur, India, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, ON, Canada, in 2000. He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems, power electronic systems, electromagnetic transients, device-level modeling, large-scale systems, and parallel and distributed computing. He is also a fellow of the Engineering Institute of Canada.

YIQING LIU received the B.S. degree from the Hebei College of Science and Technology, in 2021. He is currently a Graduate Student with the Institute of Disaster Prevention. His research interests include big data analysis, data processing, machine learning, deep learning, distributed systems, and data visualization. He is mainly working on the application of network models based on time series analysis in different fields.

JILIN FENG received the B.S. degree from the China University of Geosciences, in 1984. He is currently a Professor and the Master's Supervisor with the Institute of Disaster Prevention. His main research interests include algorithm design, data analysis of remote sensing data, and disaster information processing technology. He is the Chairperson of the GIS Association of Institution of Disaster Prevention.