



Research Article

A Short-Term Load Forecasting Model of LSTM Neural Network considering Demand Response

Xifeng Guo, Qiannan Zhao , Shoujin Wang, Dan Shan, and Wei Gong

Information & Control Engineering Faculty, Shenyang Jianzhu University, Shenyang 110168, China

Correspondence should be addressed to Qiannan Zhao; n18941643433@sina.cn

Received 11 January 2021; Revised 18 March 2021; Accepted 7 April 2021; Published 16 April 2021

Academic Editor: Qiuye Sun

Copyright © 2021 Xifeng 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.

As one of the key technologies for accelerating the construction of the ubiquitous Internet of Things, demand response (DR) not only guides users to participate in power market operations but also increases the randomness of grid operations and the difficulty of load forecasting. In order to solve the problem of rough feature engineering processing and low prediction accuracy, a short-term load forecasting model of LSTM neural network considering demand response is proposed. First of all, in view of the strong randomness and complexity of input features, the weighted method is used to process multiple input features to strengthen the contribution of effective features and tap the potential value of features. Secondly, an improved genetic algorithm (IGA) is used to obtain the best LSTM parameters; finally, the special gate structure of the LSTM model is used to selectively control the influence of input variables on the model parameters and perform load forecasting. The experimental results show that the research has high prediction accuracy and application value and provides a new way for the development of power load forecasting.

1. Introduction

In recent years, due to the shortage of fossil energy such as oil and the serious environmental problems caused by global carbon emissions, more and more experts and scholars focus on the direction of “Energy Internet [1–3].” People aim to establish a more flexible and stable form of hybrid power generation. As an important constraint for the safe and stable operation of power system, accurate power load forecasting plays an important role in the economic, safe, and reliable operation of the power system.

Demand response (DR) serves as important means to shift energy supply and uses across time to counter the indeterminate variations. It is of great significance to improve the overall energy efficiency of the energy system [4]. However, especially in the Regional Energy Internet System [5], the strong uncertainty of the user-side DR will increase the difficulty of load forecasting. Accurate load forecasting is the basis for realizing the economic operation and scientific management of the power system, as well as the inevitable choice for long-term strategic planning and operational decision-making. In fact, the fundamental requirement for

the implementation of DR programs is the load forecasting of its participants whether they are the group of residential consumers managed by aggregators, industrial consumers, or commercial consumers. DR capacity and DR events to be involved in these programs can be determined through load forecasting [6]. Instantly, the relevant technologies of artificial intelligence (AI) and machine learning (ML) have been widely used in renewable energy forecasting, load forecasting, power system fault diagnosis and optimal dispatching, power grid data visualization, and so forth [7]. At the same time, with the installation and application of a large number of measurement equipment such as smart meters [8] and weather monitoring [9], power grid companies have obtained unprecedented large amounts of data. It provides a lot of data support for power load forecasting based on AI [10]. On this basis, to improve the accuracy of load forecasting and the generalization ability of the load forecasting model has become an urgent problem to be solved.

Load forecasting is one of the most widely used areas of artificial intelligence technology in power systems. Scholars have carried out extensive research on the theory and methods of load forecasting. The methods for power load

forecasting mainly include traditional methods and artificial intelligence methods. With the development of the times, traditional methods [11–13] have gradually exposed problems such as poor model generalization ability, difficulty in determining model structure, and difficulty in selecting parameters. Artificial intelligence methods [14–18] have gradually occupied the stage of load forecasting. Ahmad et al. [19] used two ML algorithms to forecast regional short-term energy demand. In [20], deep recurrent neural network (DRNN) and deep convolutional neural network (DCNN) were presented for day-ahead multistep load forecasting in commercial buildings. Khan et al. used grey correlation based on random forest (RF) and mutual information (MI) for feature selection. Kernel principle component analysis (KPCA) was used for feature extraction and enhanced convolutional neural network (CNN) was used for classification [21]. The choice of model hyperparameters affects the entire prediction process. The essence of optimal model configuration is actually a combination problem. Bouktif et al. used metaheuristic-search-based algorithms to reduce the complexity of the search. This method can find the optimal LSTM hyperparameter set more accurately and quickly [22]. For the problem of hyperparameters, Santra and Lin have similar views. Santra and Lin proposed using GA to optimize the initialization parameters of LSTM. This method improved the robustness of short-term load forecasting [23]. Rong and de León [24] proposed a load estimation method suitable for complex power networks. In order to solve the problem of network unobservability, Rong and de León proposed a nonlinear power-temperature curve to predict the load that varies with temperature. The “effective temperature” was introduced in the paper to properly consider the impact on the power consumption of heating and cooling equipment. This method has been well verified in practice. Li et al. [25] proposed a short-term load forecasting method that considers demand response under an energy Internet environment. The grey correlation analysis method was used to process meteorological data to obtain similar daily characteristic variables input to the forecast model. The LSTM neural network model was used for load forecasting. This method separates the DR electricity price that causes users to participate in DR to indirectly consider the DR load. This method provides a new idea for load forecasting considering DR. The rapid development of AI provides more possibilities for load forecasting technology. At the same time, it also puts forward stricter requirements on the quality of load forecasting, which brings new challenges and opportunities to load forecasting technology.

In order to improve prediction accuracy and model generalization ability, a short-term load forecasting model of LSTM neural network considering DR is proposed in this paper. Based on characteristics of engineering processing, the weighted method [26] is used to deal with multiple input features. In order to strengthen the contribution of effective features and explore the potential value of effective features in depth, this paper defines the weights of input features. Then, IGA is used to select the optimal model parameters of LSTM [27]. Finally, the parameter-tuned LSTM model is used for load forecasting.

2. Feature Extraction

This paper uses the data of New South Wales, Australia, from 2006 to 2010 for experimental verification. The data from 2006 to 2008 is used as the training set and the data from 2009 to 2010 is used as the verification set. In this paper, the maximum temperature, the minimum temperature, the average temperature, the date, and the real-time electricity price of the area are extracted as the input features.

2.1. Extraction of Temperature Characteristics. The average temperature is the true average temperature of each day. The average temperature in winter is the lowest and the average temperature in summer is the highest. The load in these two quarters is higher than in other quarters due to the increase in air conditioner usage. Therefore, the feature of average temperature can reduce the abnormal high temperature and low temperature to a reasonable range to reduce temperature fluctuation and then combine it with the real temperature. In this paper, the average temperature weight is set to 0.6, and the true temperature weight is set to 0.4. The specific formula is as follows:

$$T = 0.6 * T_{\text{avg}} + 0.4 * T_{\text{true}}. \quad (1)$$

T is the input feature of temperature, T_{avg} is the average temperature, and T_{true} is the real temperature. Figure 1 shows the temperature characteristic curve after weighting treatment.

2.2. Extraction of Date Characteristics. Date type is another important influencing factor of short-term forecasting. The load on the user side on weekends will be greater than the load on workdays, and the load on the user side on holidays will also be greater than the load on workdays. This paper lists the date type as one of the factors that affect load forecasting. Considering that the parameters of the deep neural network need to be fed back, this paper expresses workdays as 1 and weekends and holidays as 2. Store the date feature in the fourth column of the feature vector.

2.3. Extraction of Real-Time Electricity Price Characteristics. The reform of the electricity market is an inevitable trend and requirement of national development. The marketization of the electricity prices is the top priority, and real-time electricity price is an important factor affecting load fluctuation. First, the maximum information coefficient (MIC) [28] is used to analyze the correlation between load and real-time point price. MIC is used to measure the degree of linear or nonlinear dependence between two variables. The greater the mutual information between the two variables is, the stronger the correlation is. MIC can overcome the shortcoming that mutual information is inconvenient to calculate continuous variables and can better reflect the degree of correlation between attributes. For a binary data set $D \in R^2$ divide D into a grid of x columns and y rows. For the divided grid G , calculate the probability of each unit in G

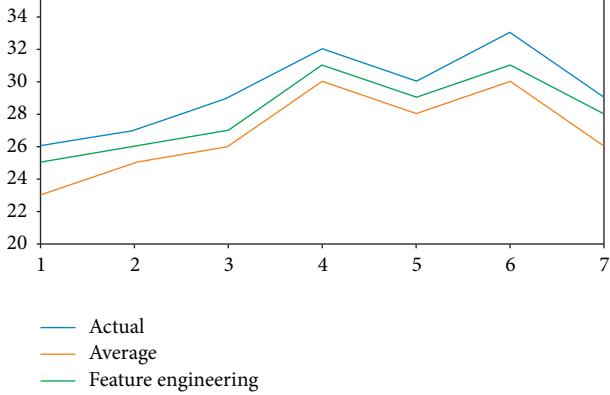


FIGURE 1: Feature engineering.

to obtain the probability distribution $D_{|G}$ of the binary data set D on the grid G . Save it as $I^*[D(x, y)]$:

$$I^*[D(x, y)] = \max I[D(x, y)]. \quad (2)$$

Standardize the obtained mutual information and find the maximum mutual information coefficient as follows:

$$M(D)_{x,y} = \frac{I^*[D(x, y)]}{\log \min\{x, y\}}, \quad (3)$$

$$\text{s.t. } xy < B(n), \quad (4)$$

$$F(D)_{\text{MIC}} = \max_{xy < B(n)} \{M(D)_{x,y}\}, \quad (5)$$

where n is the sample size; $B(n)$ is a function of the sample size, which represents the constraint of the total number of grids xy in grid G , which must be less than $B(n)$; and generally $B(n) = n^{0.6}$. In essence, MIC is a normalized maximum mutual information, with a value interval of $[0, 1]$. The statistical results show that $\text{MIC} > 0.4$ days, account for 908 days in total, accounting for 46.69%, of which $\text{MIC} > 0.6$ days account for 502 days, accounting for 26.43%.

Figure 2 is the curve of electricity price and load. It can be seen that the trend of electricity price and load is basically the same. Residents can check the real-time electricity price to appropriately reduce electricity consumption and reduce the load, which will play a role in peak clipping. When the load reaches a trough, the electricity price will also be reduced, so as to encourage users to increase the amount of electricity to fill the valley. It has been proved by experiments that the electricity price has a great correlation with the load. In this paper, the real-time price is put into the fifth column of the feature vector as the input feature.

3. Prediction Model

3.1. Improved Genetic Algorithm. Genetic algorithm (GA) is a global search optimization algorithm formed according to the selection and genetic law of “survival of the fittest” in nature. The initial population composed of feasible solutions is operated by three operators of selection, crossover, and mutation. Eliminate one by one according to the fitness value of the individual. Finally, the individuals with the best

fitness are left to form a new population, parallel calculation of multiple solution groups by genetic algorithm. Replace generations to achieve global convergence to obtain the optimal solution.

Hill climbing is a good local search algorithm. Firstly, a point is randomly selected as the initial point of iteration in the search space, and then a point is randomly generated in its neighborhood to calculate its function value. If the function value of the point is better than the current point, the current point is used to replace the initial point as a new initial point to continue searching in the neighborhood. Otherwise, the search process will be terminated if another point and initial point are randomly generated in the neighborhood until the better point is found or the better point is not found several times in a row. Because hill climbing is a method to optimize by randomly generating individuals in the neighborhood, it does not need to use a gradient. Hill climbing method can play the role of local optimization when genetic algorithm deals with complex problems. In the iterative process of this paper, the hill climbing method is introduced to optimize the individuals obtained by genetic algorithm, which greatly improves the efficiency of optimization.

3.2. LSTM Algorithm Mechanism. LSTM is an excellent variant of Recurrent Neural Network (RNN), which solves the problem of short memory and difficult training of RNN. It is very suitable for the classification and prediction of time series and is widely used in natural language processing. As shown in Figure 3, t is the time point; i_t is the input gate, which is mainly used to control how much information about the current state of the network needs to be saved in the internal state; f_t is the forget gate, which is mainly used to control how much information is in the past state and needs to be discarded; o_t is the output gate, mainly used to control how much information of the internal state at the current moment needs to be output to the external state; h_t is the internal state of the neuron at the current moment; c_t is the external state at the current moment; x_t is the external input at the current moment; and σ is the activation layer function. The calculation formula between each variable is as follows:

$$i_t = \sigma[W_{hi} \times h_{t-1} + W_{xi} \times x_t + W_{ci} \times c_{t-1} + b_i], \quad (6)$$

$$f_t = \sigma[W_{hf} \times h_{t-1} + W_{xf} \times x_t + W_{cf} \times c_{t-1} + b_f], \quad (7)$$

$$o_t = \sigma[W_{ho} \times h_{t-1} + W_{xo} \times x_t + W_{co} \times c_{t-1} + b_o], \quad (8)$$

$$c_t = f_t c_{t-1} + i_t \tanh[W_{hg} \times h_{t-1} + W_{xc} \times x_t + b_c], \quad (9)$$

$$h_t = o_t \tanh[c_t]. \quad (10)$$

Through the comprehensive application of these three control gates, the LSTM neural network can control the retention and discarding of the information transmitted in the neural network. It determines how much of the new state information at the current moment needs to be stored in the

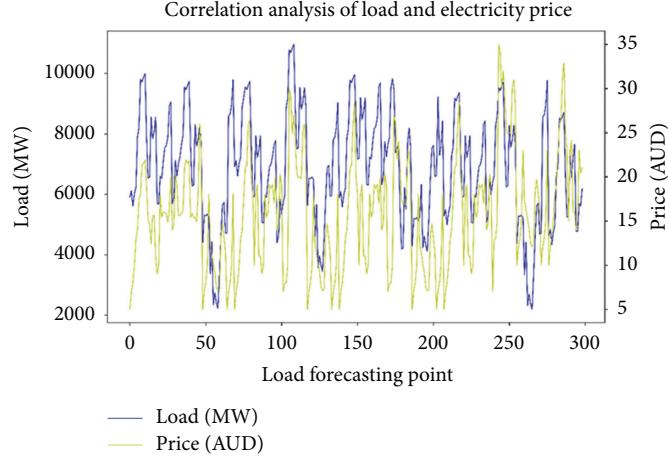


FIGURE 2: Curve of electricity price and load.

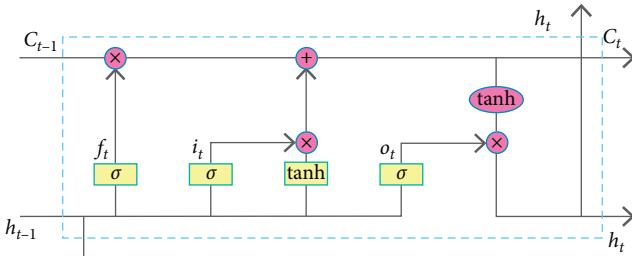


FIGURE 3: Model of LSTM.

memory unit. Compared with general neural networks, LSTM neural networks can learn dependencies with a relatively long span without the problems of gradient disappearance and gradient explosion [29].

3.3. Selection of LSTM Optimal Parameters. The initial values of the weighting matrices can affect the performance of the LSTM. Inspired by [23], an improved genetic algorithm (IGA) is used to assist in searching the proper initial values for the weighting matrices of the LSTM. Figure 4 shows the flow diagram of the proposed method.

Step 1. Initialize chromosome population.

A set of n chromosomes are randomly generated. Each chromosome w contains the values for all the weighting matrices in LSTM.

Step 2. Set fitness function.

For each chromosome in the current population, its value is used to initialize the weighting matrices in the LSTM. The LSTM predicts the output error of the training sample as the fitness function and calculates the fitness value of the individual in the initial population.

Step 3. Generate the new population through genetic operations.

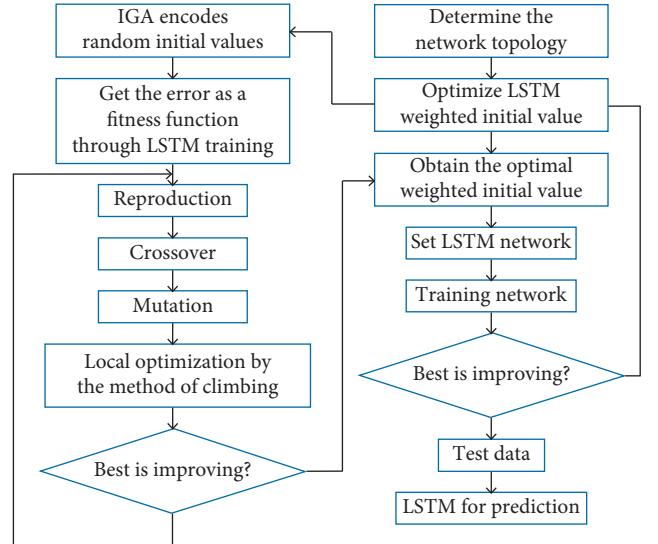


FIGURE 4: The flow diagram of the proposed method.

This step generates a new population that contains the same number of chromosomes as the current population. The roulette wheel selection is used so that a chromosome with a higher fitness value had a higher probability of being selected for genetic operations. The chromosomes of the new population are produced by reproduction, crossover, and mutation operations on the chromosomes selected from the current population.

Chromosomes with higher fitness in the current population will be selected for reproduction, and it will be added to the new population, until the reproduction ratio is reached. The elitism policy is used in this article to ensure that the best chromosome among the current chromosomes can be added to the new population group.

Repeatedly select two chromosomes of the current population as the parent chromosomes for the crossover operation to generate two offspring chromosomes and add them to the new population until the crossover ratio is

reached. Uniform crossover is adopted in this study. With uniform crossover, each value in the offspring chromosomes is independently chosen from the two values at the same corresponding position in the two parent chromosomes, as shown in Figure 5.

The mutation operation repeatedly selected a chromosome from the current population, modified the selected chromosome to generate a new chromosome, and added the mutant to the new population, until the mutation ratio has reached. With one-point mutation, a small random change is injected into the value of a randomly selected position in the selected chromosome to generate the mutant, as shown in Figure 6. Mutation operation makes the chromosome population more diverse.

Step 4. Select the best chromosome.

After one evolution, the fitness value is calculated for the retained chromosomes, and the best chromosome is retained. Compare the fitness value of the chromosome with the maximum fitness value and the best chromosome of the previous generation. The best chromosome is recorded. Finally, the chromosome with the best fitness value is used to initialize the weighting matrix in the LSTM, and the training data is fed to the LSTM to generate the fitness value of the chromosome.

Step 5. Stop criterion

If the fitness value of the best chromosome is already optimal, then the LSTM of the best chromosome is adopted, and the proposed method is terminated. The parameters of LSTM are initialized with the optimal weights to obtain the optimal network structure.

In this paper, the original real data set is divided into three training sets and a test set. Feature engineering processing is carried out by means of the weighted discretization method. The input features processed by the weighting method are more valuable, and then the IGA is used to select the optimal parameters of the LSTM network. The LSTM network after parameter tuning is used to train and predict real data. The specific process is shown in Figure 7.

4. Case Study

In order to evaluate the prediction performance, the mean absolute percentage error Y_{MAPE} and forecasting accuracy Y_{FA} are set according to the load forecasting indexes evaluated by State Grid Corporation of China. The specific formulas are as follows:

$$Y_{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{X_{act}(i) - X_{pred}(i)}{X_{act}(i)} \right|, \quad (11)$$

$$Y_{FA} = \left(1 - \frac{\left| X_{act}(i) - X_{pred}(i) \right|}{X_{act}(i)} \right) \times 100\%, \quad (12)$$

where n is the total number of predictions and $X_{act}(i)$ and $X_{pred}(i)$ are the real value and the predicted value of the load at time i respectively.

Parents	Offsprings
1 1 0 1 0 0 1 1 1 0	0 1 1 0 0 0 0 1 0 0
0 0 1 0 1 1 0 0 0 1	1 0 0 1 1 1 1 0 1 1

FIGURE 5: Uniform crossover.

Before mutation	After mutation
0 0 1 1 0 1 0 0 1 0	0 0 1 0 0 1 0 0 1 0

FIGURE 6: One-point mutation.

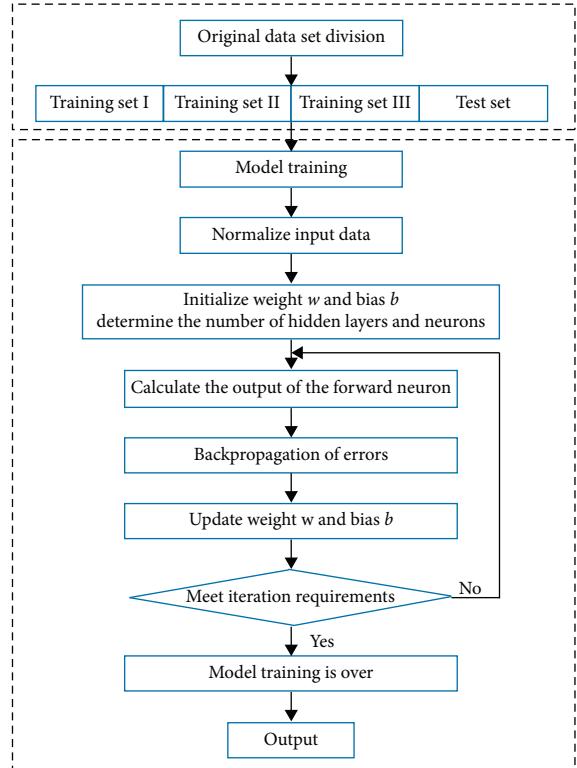


FIGURE 7: Proposed model.

4.1. Analysis of Experimental Results. After training and verification of the proposed model, this paper conducted day-ahead short-term load forecasting from 2009 to 2010 and calculated the Y_{MAPE} and the Y_{FA} of the parameter-tuned SVM model and RF model. The results are shown in Table 1. It can be seen from the table that whether it is a one-day or half a year, Y_{MAPE} of the proposed model is the smallest and Y_{FA} is the greatest and the goodness of fit is the best of all models, which shows that the method proposed in this paper is more accurate and the robustness is better.

Experimental verification shows that the comparison between the four methods and the real load curve is shown in Figure 8. It can be seen from the figure that the load curve predicted by the model proposed in this paper has the highest degree of fit with the real curve. The prediction results are more accurate than the other three methods. In

TABLE 1: Comparison of model prediction results.

Prediction model	One day		Half a year	
	Y_{MAPE}	$Y_{FA} (\%)$	Y_{MAPE}	$Y_{FA} (\%)$
Proposed	0.1067	89.31	0.0901	90.99
SVM	0.2025	79.75	0.1862	81.38
RF	0.1689	83.11	0.1259	87.41
GRU	0.1219	87.81	0.1015	89.85

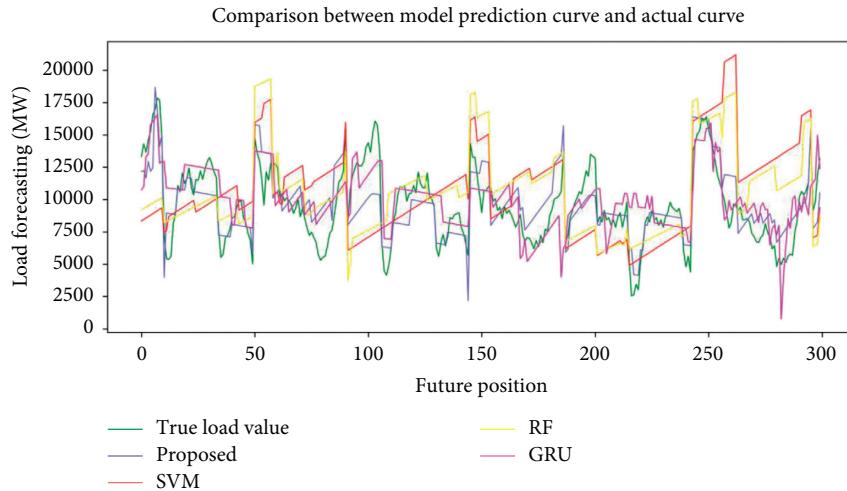


FIGURE 8: Model prediction comparison diagram.

order to further evaluate the performance of the model, this article compares the calculation speed of each model. The average calculation time of this model is 7.96 s, while the average calculation speeds of SVM, RF, and GRU are 11.14 S, 9.86 S, and 8.03 s, respectively. Through the comparison of average calculation time, it can also be seen that the model proposed in this article has certain advantages compared with other models.

5. Conclusions

In view of the phenomenon that user-side DR is highly uncertain and load prediction becomes more difficult, this paper selects regions with relatively perfect demand-side management as target regions and puts forward a short-term load prediction method that takes DR into consideration. In this paper, the weighted discretization method is used to process the input features and enhance the contribution degree of input features. Then, IGA is used to select the optimal model parameters of LSTM. After that, the LSTM prediction model is established for load prediction. Finally, the effectiveness and superiority of the proposed method are verified based on the actual data, and the following conclusions are drawn.

- (1) Considering a variety of load influencing factors, this paper uses the weighted discretization method to process input characteristics. And the maximum information coefficient method is used to

verify the real-time electricity price characteristics selected in this paper have a strong correlation with load.

- (2) In this paper, the IGA algorithm is used to select the optimal LSTM parameters.
- (3) The LSTM model has a special forgetting gate mechanism, which can automatically screen the input variables that are beneficial to the model during the training process. It cuts down the prediction time on the basis of improving the model performance and prediction accuracy.

Data Availability

The data used in the study are available at <http://www.aemo.com.au>.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by National Key Research and Development Plan (2019YFF0301505); Liaoning Natural Science Foundation (2019-ZD-0682); and Education Department of Liaoning Province Science and Technology Research Projects (LJJC201912).

References

- [1] R. Wang, Q. Sun, and W. Hu, "SoC-based droop coefficients stability region analysis of the battery for stand-alone supply systems with constant power loads," *IEEE Transactions on Power Electronics*, vol. 99, p. 1, 2021.
- [2] D. Ma, X. Hu, and H. Zhang, "A hierarchical event detection method based on spectral theory of multidimensional matrix for power system," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 99, pp. 1–14, 2019.
- [3] Q. Sun, R. Han, H. Zhang, J. Zhou, and J. M. Guerrero, "A multiagent-based consensus algorithm for distributed coordinated control of distributed generators in the energy internet," *IEEE Transactions on Smart Grid*, vol. 6, no. 6, pp. 3006–3019, 2015.
- [4] H. Xiao, W. Pei, and Z. Dong, "Bi-level planning for integrated energy systems incorporating demand response and energy storage under uncertain environments using novel metamodel," *CSEE Journal of Power and Energy Systems*, vol. 4, no. 2, 2018.
- [5] N. Li and K. Zheng, "Regional energy internet construction framework and key tasks," *IOP Conference Series: Earth and Environmental Science*, vol. 446, p. 22034, 2020.
- [6] M. Mansoor, F. Grimaccia, and S. Leva, "Comparison of echo state network and feed-forward neural networks in electrical load forecasting for demand response programs," *Mathematics and Computers in Simulation*, vol. 184, pp. 282–293, 2020.
- [7] J. Shi and J. Zhang, "Load forecasting based on multi-model by stacking ensemble learning," *Proceedings of the CSEE*, vol. 39, no. 14, pp. 4032–4041, 2019.
- [8] A. M. Alonso, F. J. Nogales, and C. Ruiz, "Hierarchical clustering for smart meter electricity loads based on quantile autocovariances," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 4522–4530, 2020.
- [9] T. Ahmad, W. A. Shah, and D. Zhang, "Efficient energy planning with decomposition-based evolutionary neural networks," *IEEE Access*, vol. 99, 2020.
- [10] X. Kong, C. Li, and F. Zheng, "A short-term load forecasting method based on EMD and characteristic correlation analysis," *Automation of Electric Power Systems*, vol. 43, no. 5, pp. 75–85, 2019.
- [11] K. Wan and R. Liu, "Application of interval time-series vector autoregressive model in short-term load forecasting," *Power System Technology*, vol. 36, no. 11, pp. 77–78, 2012.
- [12] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *Monash Econometrics & Business Stats Working Papers*, vol. 27, no. 1, pp. 134–141, 2010.
- [13] E. Paparoditis and T. Sapatinas, "Short-term load forecasting: the similar shape functional time-series predictor," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 3818–3825, 2013.
- [14] Z. Wu, X. Kong, and X. Dong, "Short-term spatial load forecasting based on partition and classification of power distribution information," *Proceedings of the CSU-EPSA*, vol. 31, no. 2, pp. 30–35, 2019.
- [15] L. Sehovac and K. Grolinger, "Deep learning for load forecasting: sequence to sequence recurrent neural networks with attention," *IEEE Access*, vol. 8, pp. 36411–36426, 2020.
- [16] 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.
- [17] G. Suo, L. Song, and Y. Dou, "Multi-dimensional short-term load forecasting based on XGBoost and fireworks algorithm," in *Proceedings-2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science*, pp. 245–248, DCABES, Wuhan, China, November 2019.
- [18] Z. Deng, B. Wang, Y. Xu, T. Xu, C. Liu, and Z. Zhu, "Multi-scale convolutional neural network with time-cognition for multi-step short-term load forecasting," *IEEE Access*, vol. 7, no. 99, pp. 88058–88071, 2019.
- [19] T. Ahmad, H. Chen, and Y. Huang, "Short-term energy prediction for district-level load management using machine learning based approaches," *Energy Procedia*, vol. 158, pp. 3331–3338, 2019.
- [20] M. Cai, M. Pipattanasomporn, and S. Rahman, "Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques," *Applied Energy*, vol. 236, pp. 1078–1088, 2019.
- [21] A. B. M. Khan, N. Javaid, and O. Nazeer, "Hourly electricity load forecasting in smart grid using deep learning techniques," *Innovative Mobile and Internet Services in Ubiquitous Computing*, vol. 19, pp. 185–196, 2020.
- [22] S. Bouktif, A. Fiaz, and A. Ouni, "Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting," *Energies*, vol. 13, 2020.
- [23] A. S. Santra and J.-L. Lin, "Integrating long short-term memory and genetic algorithm for short-term load forecasting," *Energies*, vol. 12, no. 11, p. 2040, 2019.
- [24] H. Rong and F. de León, "Load estimation of complex power networks from transformer measurements and forecasted loads," *Complexity*, vol. 2020, Article ID 2941809, , 2020.
- [25] C. Li, X. Kong, and S. Zhu, "Short-term load forecasting of regional power grid considering demand response in energy interconnection environment," *Automation of Electric Power Systems*, vol. 45, pp. 1–11, 2020.
- [26] Y. Xu, F. Zuo, and X. Zhu, "Research on load prediction based on improved GBDT algorithm," *Proceedings of the CSU-EPSA*, vol. 32, pp. 1–8, 2020.
- [27] F. Lv, Y. Liu, and Y. Qi, "Short-term load forecasting based on optimized learning machine using improved genetic algorithm," *Journal of North China Electric Power University*, vol. 45, no. 6, pp. 1–7, 2018.
- [28] P. Wen, J. Wang, and S. Yin, "Short-term load forecasting model based on Attention-LSTM in electricity market," *Power System Technology*, vol. 5, pp. 1745–1751, 2019.
- [29] G. B. Goh, N. O. Hodas, and A. Vishnu, "Deep learning for computational chemistry," *Journal of Computational Chemistry*, vol. 38, no. 16, pp. 1291–1307, 2017.