Power System Load Forecasting Method Based on Recurrent Neural Network

Chuanjun Pang[1], *Tie* Bao[1*], and *Lei* He[1]

[1] NARI Group Co., Ltd. (State Grid Electric Power Research Institute Co., Ltd.), China, 100192

Abstract. Power system load forecasting plays an important role in the power dispatching operation. The development of the electricity market and the increasing integration of distributed generators have increased the complexity of power consumption model and put forward higher requirements for the accuracy and stability of load forecasting. A load forecasting method based on long-short term memory (LSTM) is proposed. This method uses deep recurrent neural network from the artificial intelligence field to establish a load forecasting model. Using the LSTM network to memorize the long-term dependence of the sequence data, the intrinsic variation of the load itself is identified from both the horizontal and vertical dimensions within a longer historical time period, while considering various influencing factors. Actual load data is used to verify the forecasting performance of different historical date windows and different network architectures.

1 Introduction

Power system load forecasting can estimate the power demand for a period in the future. The future unit commitment, generation plan, tie line exchange plan and organization of the electric power transactions can be arranged according to the load forecast result. Therefore, load forecasting is the basis to realize the safe, reliable and economical operation of power system, and it is also an important guarantee to improve the utilization ratio of power generation equipment and the efficiency of economic dispatch. Accurate load forecasting has great significance to optimal unit commitment, economic dispatch, optimal power flow and electricity market transaction, etc [1].

Power system load is affected by economic factors, weather, season, date, and other factors, so load variation has random components. With the development of power system, the types of user load and influencing factors are increasing. In the electric power market environment, the complexity of electricity consumption mode is increased by the users' behaviour, the response of users to the incentive policies and the urban development situation. The introduction of large-scale distributed generators and the widespread use of electric vehicles have increased the load fluctuation [2]. Peak-valley time-of-use price through its own economic leverage, urges users to constantly change the way they use electricity [3], and stimulates and encourages electric vehicles to change their charge-discharge behaviour [4].

At present, load forecasting methods are mainly divided into two categories: statistical methods and artificial neural network (ANN) based methods. Statistical methods use time series prediction to forecast [5], mainly including multiple linear regression (MLR), auto regression (AR) and autoregressive moving average (ARMA). The models of such methods are simple, but only a small number of influencing factors and sample data can be processed, and the stability of the original time series is highly demanded, because these methods are based on the assumption that the historical load, load influencing factors and future load are linearly related. In fact, the influencing factors and the load changes have strong non-linear characteristics. Artificial intelligence based methods include artificial neural network (ANN), support vector machine, fuzzy logic and other modern forecasting methods [6,7,8]. Artificial neural network has been successfully applied in speech recognition, natural language processing, computer vision and other fields [9,10,11]. Many researchers have tried to apply this method to load forecasting and new energy power forecasting and achieved good results [12,13,14]. ANN simulates brain mechanisms, learns implicit rules from data and forecasts the future data. ANN has a strong nonlinear modelling ability and can learn the nonlinear characteristics of the load from the historical load data. ANN includes a variety of different network architectures. At present, feedforward neural network (FNN) is widely used in power system load forecasting [14]. But FNN can't consider the dependence between the training samples, can't model the sequence data, and can't establish the load forecasting model from a longer historical time scale.

In this paper, an improved recurrent neural network (RNN) of long-short term memory unit is used to establish a load forecasting model. Using the deep recurrent neural network's characteristics of remembering historical information, based on many historical load data,

[*] Corresponding author: bt0504@126.com

the inherent rules of load variation can be identified from a longer historical time range, and the complex nonlinear relationship between the load influencing factors and the load can be learned. Actual load data are used to verify the model. The experimental results show that the method can identify the load variation rules to a certain extent and make a more accurate forecast of the future load.

2 Recurrent neural network

2.1 Ordinary recurrent neural network

Recurrent neural network (RNN) is a kind of neural network which can process sequence data [15]. The network architecture is shown in Figure 1.

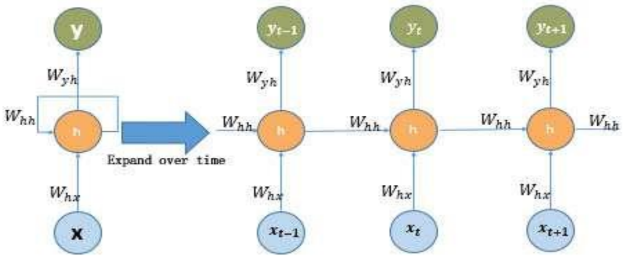

Figure 1. Recurrent neural network.

Compared with FNN, each neuron in RNN hidden layer can be recycled through self-link. Given the input data $X = \{x_1, x_2, x_3, \dots x_T, \}$, RNN uses formula (1) and formula (2) to calculate hidden state $h = \{h_1, h_2, h_3, \dots h_T, \}$ and output $y = \{y_1, y_2, y_3, \dots y_T, \}$.

$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \quad (1)$$

$$y_t = g(W_{yh}h_t + b_y) \quad (2)$$

W_{hx} , W_{hh} , W_{yh} represent input weight, hidden weight and output weight, respectively. f , g represent hidden layer activation function and output layer activation function, respectively. RNN uses hidden state to remember all previous input information. The output y_t at the current time is not only affected by the current input x_t, but also by the hidden state h_t at the previous time. RNN uses back-propagation through time to train model [12]. In the training process, RNN has difficulty in learning long-distance dependence because of the disappearance of gradient.

2.2 Long-short term memory

Long-Short Term Memory (LSTM) is an improvement on ordinary RNN. The long-term dependence problem, which makes the model not suitable on learning sequence data because of gradient disappearance, can be solved through introducing memory cell into each neuron in hidden layer, and using forget gate, input gate and output gate to control the state of memory cell [15]. The architecture of LSTM neuron is shown in Figure 2:

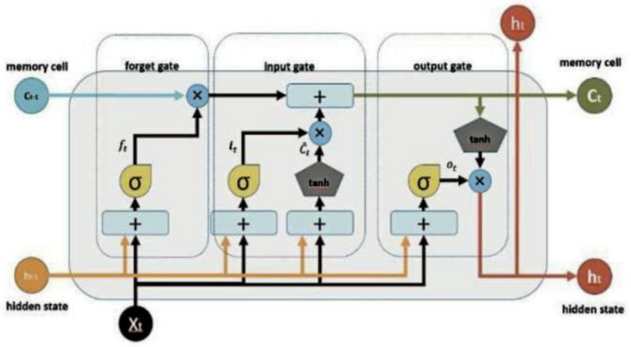

Figure 2. The architecture of LSTM.

Memory cell and the hidden state can memorize the historical information of the sequence data together. The information in the memory cell is controlled by three gate units.

Forget gate deletes the information in the memory cell according to the hidden state of the previous moment h_{t-1} and the input of the current moment x_t. Forget gate is calculated as follows:

$$f_t = \sigma(W_f[h_{t-1}, x_t]) + b_f \quad (3)$$

Input gate adds information to the memory cell according to the hidden state of the previous moment h_{t-1} and the input of the current moment x_t .The functions are shown in formula (4) and formula (5). i_t represents the information that needs to be memorized; $\hat{C}_t$ represents the candidate memory cell to update next memory cell.

$$i_t = \sigma(W_i[h_{t-1}, x_t]) + b_i \quad (4)$$

$$\hat{C}_t = \tanh(W_c[h_{t-1}, x_t]) + b_c \quad (5)$$

After calculating the forget gate and the output gate, formula (6) is used to update the memory cell.

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \quad (6)$$

Output gate decides the current hidden state h_t according to the hidden state of the previous moment h_{t-1}, the input of the current moment x_t and updated memory cell C_t , as shown in formula (7) and formula (8).

$$o_t = \sigma(W_o[h_{t-1}, x_t]) + b_o \quad (7)$$

$$h_t = o_t * \tanh(C_t) \quad (8)$$

In the formulas, W_f represents forget gate weight; W_i , W_c represent input gate weights; W_o represents output weight; b_f , b_i , b_c , b_o represent corresponding offset vectors.

3 Load forecasting based on LSTM

3.1 Basic theory

Power system load is periodic because of the influence of people's production and living patterns, and it also has great uncertainty because of the influence of random factors such as weather changes, major events and so on. The load l_t^d at a certain time of a day is affected by the type of the day (days of the week, whether it is a holiday or not) and the current weather factors (temperature, humidity, etc.). It is also related to the load l_{t-n}^d at the past time of the day and the load l_t^{d-w} at the same time of the past few days, where t is the moment, d is the date, and w is the historical date window. LSTM has the advantage of learning the rules of sequence data from a long-time range. It can recognize the load variation regularity of the forecasting day horizontally, identify the load variation regularity at the same time in the historical date window vertically, and consider the influence of related factors on the load of the forecasting day.

3.2 Forecasting model

Based on LSTM network, the forecast day load curve is forecasted using the load curves of w days before forecast day and the influencing factors of forecast day. The LSTM network structure is shown in Figure 3.

$$\hat{L}^d = \{\hat{l}_0^d, \hat{l}_1^d \ldots \hat{l}_T^d\} \tag{9}$$

$\hat{L}^d$is the forecasted load for forest day, and T is called step size, T=24|96|288|1440.

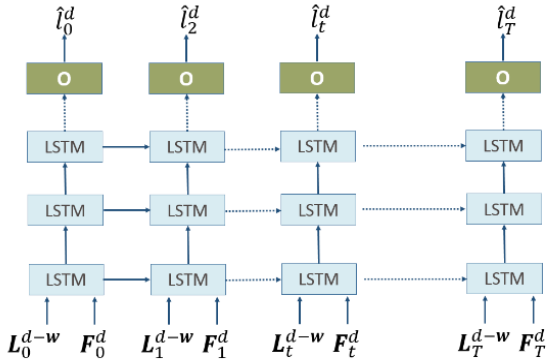

Figure 3. LSTM neural network model.

$$L_t^{d-w} = \{l_t^{d-1}, l_t^{d-2} \ldots l_t^{d-w}\} \tag{10}$$

l_t^{d-w} represents the load at time T of w days before forecast day. w is the historical date window.

$$\mathrm{F}_t^d = \{\mathrm{month}_t^d, \mathrm{is\_holiday}_t^d, \ldots, \mathrm{temp}_t^d\} \tag{11}$$

It includes load influencing factors, such as date types, holidays, weather and so on.

The input of the load forecasting model at time T is $\mathrm{input} = \{\mathrm{L}_t^{d-w},\ \mathrm{F}_t^d\}$ and the output is $output = \hat{l}_t^d$.

3.3 Model training

Model training includes three stages: data pre-processing, model training and model evaluation.

3.3.1 Data preprocessing

Data preprocessing mainly includes two steps: Vectorization and normalization. Neural network is based on linear algebra theory, and cannot be trained directly on the original data. So it is necessary to convert the original data into vector before training. Data vectorization concatenates the load data in the history date window L_t^{d-w} with the influence factor data of the forecast day F_t^d, and transforms the concatenated data into vector.

The neural network is trained by the back-propagation algorithm based on gradient descent. If the data is too large or too small, it will be difficult to find the optimal solution in the training process. Therefore, normalization of data into a standard interval is conducive to the solution of the model. The min-max scaling normalization is used to normalize each element of the vector into the interval [0-1], as shown in formula 12 [16]:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{12}$$

where $\mathrm{X_{max}}$ is the maximum value, $\mathrm{X_{min}}$ is the minimum value, X is the original value and $\mathrm{X_{norm}}$ is the normalized value.

3.3.2. Model training.

Back-propagation through time (BPTT) [17,18] algorithm is used to train the LSTM network for load forecasting. The training objective is to adjust the network parameters so that the network output is as close to the real values as possible. The training process is as follows:

1) Initialize network parameters.

2) Input the training data to the network and calculate the network output.

3) Calculate the difference between the network forecast value and the actual value, and use loss function to measure the error.

4) According to the error, calculate the gradients of network parameters over loss functions in two directions of network level and time.

5) Adjust network parameters according to the gradient.

6) Repeat steps 2-5.

Steps 2-3 constitute the forward reasoning process, and steps 4-5 constitute the backward propagation process.

The loss functions include mean square error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE) and so on. In this paper, MSE is used as the loss function, which is defined as follows [18]:

$$\mathrm{Loss(W, b\,)} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i=1}}^{\mathrm{N}}\left(\hat{\mathrm{L}}_\mathrm{i} - \mathrm{L_i}\right)^2 \tag{13}$$

where N represents total number of training samples, $\hat{L}_\mathrm{i}$ represents forecasted load, and L_i represents actual load.

3.3.3 Model evaluation.

In the process of model training, all data are used to input into the network to adjust its parameters, which is called an epoch. If the number of epoch is too small the network will be under-fitting; if the number of epoch is too large the network will be over-fitting. In order to determine the appropriate epoch and select the optimal model, the model should be evaluated during each epoch. The following steps are used to determine the optimal epoch.

1) Divide historical load data into training set, validation set and test set.

2) Input the training set into the network for training. After an epoch, the validation set is put into the network model, and the model is evaluated by formula (13).

3) If the performance of the current model exceeds those of all the previous epochs, keep the model as the optimal model.

4) Repeat steps 2-3 until the exit condition is satisfied. (The performance of continuous N epochs can't be improved any more.)

5) Output the optimal model and use test set to test the model.

4 Example analysis

4.1. Experimental data

The hourly historical system load data of Electric Reliability Council of Texas (ERCOT) control area from 2003 to 2016 are used to validate the proposed model [18]. Data from 2003 to 2014 are used as training set, data from 2015 as validation set and data from 2016 as test set. Validation set is used to verify the optimal number of training epoch, the optimal network structure and parameters. Test set is used to test the forecast accuracy of the model. While the historical weather information is not available, the weather factors are not taken into account in the model training process. Only the type of date (working day, the days of week), the same time load in the historical window period and maximum and minimum daily load in the historical window period are used.

4.1.1 Evaluation indices

The common model evaluation indices include mean square error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE). The definition of MSE is shown in formula (13). MAE [18] and MAPE [1] are defined as in formula (14) and (15).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{L}_i - L_i\right| \tag{14}$$

$$\mathrm{MAPE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\left|\frac{\hat{\mathrm{L}}_\mathrm{i}-\mathrm{L}_\mathrm{i}}{\mathrm{L}_\mathrm{i}}\right| \tag{15}$$

N represents total number of training samples, $\hat{L}_i$ represents forecasted load, and L_i represents actual load.

4.2. Network structure

In LSTM network, the number of hidden layers and the number of LSTM neurons in each hidden layer affect the accuracy of load forecasting. Using the hourly historical load data of 2015 as validation set, with fixed historical date window $w = 7$, the number of hidden layer neurons is selected layer by layer through enumeration method [1] to determine the optimal network structure. First, the optimal number of hidden neurons in the first layer is determined and fixed; then a hidden layer is added to determine the optimal number of hidden neurons in the second layer. The process is continued until the forecast accuracy is no longer improved.

The forecasting performance under different network structures is shown in Table 1. When the number of hidden neurons in each layer is selected, options are 5-40 with interval 5 and 8 levels. The number of hidden layers is set to 1, 2 and 3 in turns. As shown in Table 1, when the hidden layer is 1, the number of hidden neurons in the layer is 5, the minimum MAPE is 4.6%; when the hidden layer is 2, the number of hidden neurons in each layer is 20, the minimum MAPE is 4.72%; when the hidden layer is 3, the number of hidden neurons in each layer is 10, the minimum MAPE is 4.58%. Therefore, when $w = 7$, the number of hidden layer is 2 and the number of hidden neutrons in each layer is 10, the forecasting performance of the network is better.

Table 1. Forecasting performance of LSTM model with different structures

Hidden layer	Hidden cells	MAE/MW	MAPE/%	Optimal epoch
1	5	1772.27	4.6	500
	10	1800.32	4.72	38
	15	1792.25	4.74	53
	20	1844.75	4.94	40
	25	1815.24	4.79	30
	30	1878.15	4.99	24
	35	1808.85	4.80	35
	40	1852.92	4.80	16
2	5	1813.37	4.75	172
	10	1905.45	4.99	57
	15	1822.30	4.77	38
	20	1801.05	4.72	42
	25	1798.51	4.71	35
	30	1991.26	5.11	20
	35	1800.72	4.83	16
	40	1831.83	4.75	16
3	5	1782.06	4.59	180
	10	1779.85	4.58	65
	15	1824.71	4.74	48
	20	1785.96	4.65	55
	25	2002.88	5.14	20
	30	1790.58	4.65	52
	35	1915.15	5.00	20
	40	1891.02	4.87	39

4.3. Historical date window

Set the network layer as 1, 2, 3 and the number of hidden cells as 5, 20, 10, respectively. The historical window w takes 1-60 days. The hourly historical load data of 2015 are used as validation set.

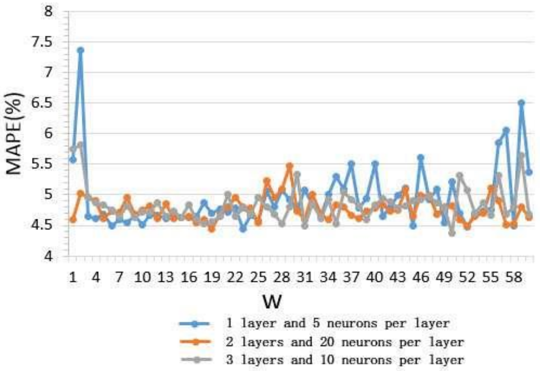

Figure 4. The change of MAPE with w under different LSTM network structures.

Figure 4 shows the curves of MAPE changing with w under different network structures. As shown in Figure 4, with the increase of w, MAPE of different network structures are almost the same. There is no obvious downward or upward trend, but the change of MAPE becomes unstable when w increases. Especially when the network has 1 layer and there are 5 neurons in the layer, the larger w causes MAPE to change violently. Under three network structures, when the value of w is in [7,30], MAPE is relatively low and stable; when the hidden layer is 1, the number of hidden neurons in the layer is 5 and w is 23, the forecasting performance is the best and MAPE is 4.4%.

4.4. Result analysis

In order to verify the validity and scientificity of the proposed method, the optimal forecasting model (hidden lay is 1, the number of neurons is 5 and w is 23) and the test set (hourly load of year 2016) are used to test the forecasting performance of the model.

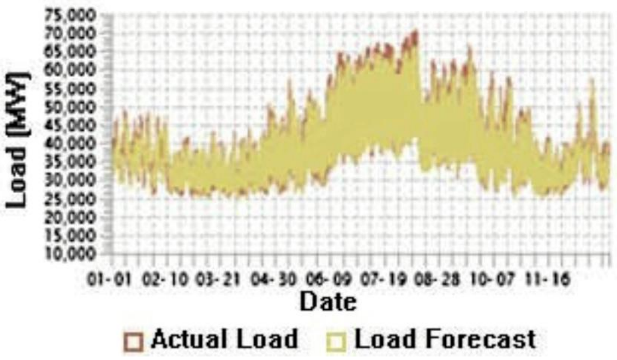

Figure 5. Actual load vs. load forecast of year 2016.

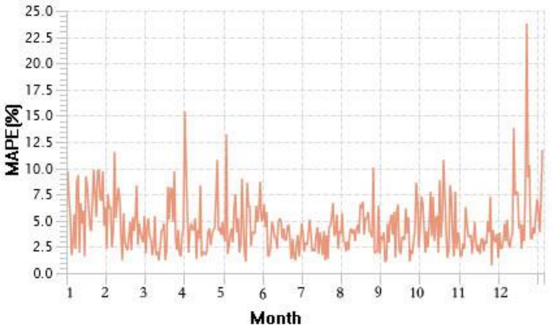

Figure 6. Daily forecasting MAPE.

The forecast results are shown in Figure 5, and the daily MAPE is shown in Figure 6, with average MAPE 4.45%.

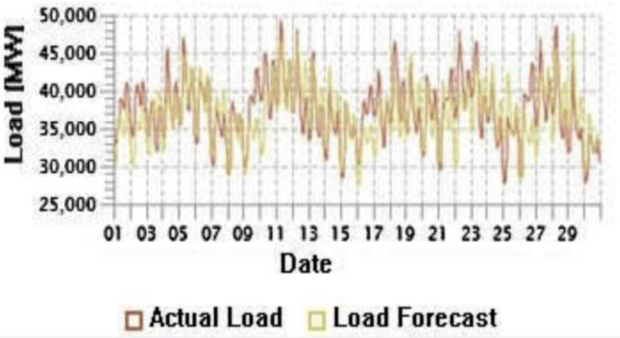

Figure 7. Actual load vs. load forecast of Jan, 2016.

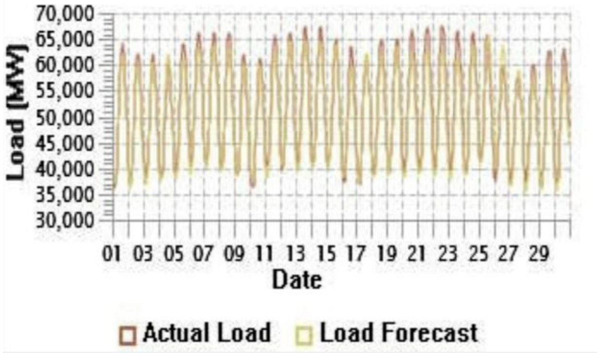

Figure 8. Actual load vs. load forecast of July, 2016.

Figure 7 and Figure 8 show the comparison between actual load and load forecast in 2016, January and July, respectively. The analysis shows that system load changes in January have great randomness, because the model does not consider weather, holidays (except Saturday and Sunday), special events and other factors. So the forecasting accuracy is low and only based on load itself, and MAPE is 6.66%. The forecasting accuracy of July is higher except the daily peak load and valley load, and MAPE is 3.51%.

According to the forecasting results, the forecasting model used in this paper can learn the variation of the historical load and accurately forecast the future load.

5 Conclusion

In this paper, recurrent neural network (RNN), from the field of artificial intelligence, is improved with long-short term memory (LSTM) unit to build load forecasting model, taking the advantage that LSTM can learn the long-term dependence of sequence data to identify the variation rules of load itself. The actual load data are used to verify the model and the influence on forecasting performance from different neural network structures and different historical time windows. The experimental results show that the LSMT network can forecast the future load accurately.

In the LSTM network training, the influence of historical weather factors, major events and other random factors on the load is not considered. The next step is to consider more affecting factors to validate the model and further improve the forecasting accuracy of the model.

Artificial intelligence technology, represented by deep neural networks, has made remarkable achievements in many fields. In the future, recurrent neural network (RNN) will be applied to more business areas related to power system dispatching operation, such as wind and photovoltaic power forecasting, voice dispatching log, dispatching robot and so on, to improve the intelligent level of power system dispatching operation.

This work is supported by the Science and Technology Project <Research on the technology of new energy brought into the medium- and long-term power balance considering the fluctuation characteristics of output time series (4000-201940193A-O-O-OO)> from the State Grid Corporation of China.

References

1. KONG Xiangyu, ZHENG Feng, E Zhijun, et al, "Short-term Load Forecasting Based on Deep Belief Network[J]", Automation of Electric Power Systems, 2018, 42(5), pp. 133-139.
2. LIUJ, GAO H,ZHAO M A, et al, "Review and prospect of active distribution system planning[J]", Journal of Modern Power Systems and Clean Energy, 2015, 3(4), pp. 457-467.
3. YAO Jiangang, FU Qiang, YE Lun, et al, "Substation Capacity Planning Method Considering Influence of Peak-valley Time-of-use Power Price[J]", Automation of Electric Power Systems, 2017, 41(13), pp. 53-61.
4. LIU H, ZENG P,GUO J, et al, "An optimization strategy of controlled electric vehicle charging considering demand side response and regional wind and photovoltaic[J]", Journal of Modern Power Systems and Clean Energy, 2015, 3(2), pp.232-239.
5. LI Bo, MEN Deyue, YAN Yaqin, et al, "Architecture and Key Technologies for Auxiliary Decision-making System of Integrated Dispatch Scheduling[J]", Automation of Electric Power Systems, 2015, 39(1), pp.137-140.
6. WU Qianhong, GAO Jun, HOU Guangsong, et al. "Short-term Load Forecasting Support Vector Machine Algorithm Based on Multi-source Heterogeneous Fusion of Load Factors[J]", Automation of Electric Power Systems, 2016, 40(15), pp. 67-72.
7. WU Xiaoyu, HE Jinghan, ZHANG Pei, et al, "Power System Short-term Load Forecasting Based on Improved Random Forest with Grey Relation Projection[J]", Automation of Electric Power Systems, 2015, 39(12), pp. 50-55.
8. XIAO Bai, NIE Peng, MU Gang, et al, "A Spatial Load Forecasting Method Based on Multilevel Clustering Analysis and Support Vector Machine[J]", Automation of Electric Power Systems, 2015, 39(12), pp. 56-61.
9. Lipton, Zachary C., Berkowitz, John, and Elkan, Charles, "A critical review of recurrent neural networks for sequence learning", arXiv preprint arXiv:1506.00019, 2015.
10. Justin Bayer, Daan Wierstra, Julian Togelius, and J¨urgen Schmidhuber, "Evolv-ing memory cell structures for sequence learning", Artificial Neural Networks–ICANN 2009, pp. 755–764, Springer, 2009.
11. Yoshua Bengio, Patrice Simard, and Paolo Frasconi' " Learning long-term depen-dencies with gradient descent is difficult", Neural Networks,IEEE Transactions on, 5(2), pp. 157–166, 1994.
12. XU Longbo,WANG Wei,ZHANG Tao, et al, "Ultra-short-term Wind Power Prediction Based on Neural Network and Mean Impact Value[J]", Automation of Electric Power Systems, 2017, 41(21), pp. 40-45.
13. LIU Ruiye, HUANG Lei, "Wind Power Forecasting Based on Dynamic Neural Networks[J]", Automation of Electric Power Systems, 2012, 36(11), pp. 19-22.
14. HERNANDEZ L, BALADRON C, AGUIAR J M, et al, "Artificial neural network for short-term load forecasting in distribution systems[J]", Energies, 2014 ,7(3), pp. 1576-1598.
15. Bengio Y, Boulanger-Lewandowski N, Pascanu R, "Advances in Optimizing Recurrent Networks", IEEE International Conference on Acoustics, 2012, pp. 8624–8.
16. Adam Gibson, Josh Patterson, "Deep Learning : A Practitioner's Approach [M]", Boston :O'Reilly Media, 2017.
17. Wim De Mulder, Steven Bethard, and Marie-Francine Moens, "A survey on the application of recurrent neural networks to statistical language modeling", Computer Speech & Language, 30(1), pp. 61–98, 2015.
18. "Hourly Load Data Archives [EB/OL]", [2018-06-12]. http://www.ercot.com/gridinfo/load/load_hist/.