SCIENCE CHINA
Technological Sciences

• Article •

https://doi.org/10.1007/s11431-019-9547-4

Short-term power load forecasting using integrated methods based on long short-term memory

ZHANG WenJie[1], QIN Jian[2], MEI Feng[1], FU JunJie[2], DAI Bo[1] & YU WenWu[2*]

[1]*State Grid Zhejiang Electric Power Corporation Information and Telecommunication Branch, Hangzhou 310007, China;*
[2]*Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, School of Mathematics, Southeast University, Nanjing 210096, China*

Received April 1, 2019; accepted June 24, 2019; published online January 2, 2020

The development of power system informatization, the massive access of distributed power supply and electric vehicles have increased the complexity of power consumption in the distribution network, which puts forward higher requirements for the accuracy and stability of load forecasting. In this paper, an integrated network architecture which consists of the self-organized mapping, chaotic time series, intelligent optimization algorithm and long short-term memory (LSTM) is proposed to extend the load forecasting length, decrease artificial debugging, and improve the prediction precision for the short-term power load forecasting. Compared with LSTM prediction, the algorithm in this paper improves the prediction accuracy by 61.87% in terms of root mean square error (RMSE), and reduces the prediction error by 50% in the 40-fold forecast window under some circumstances.

long short-term memory, chaotic time series, intelligent optimization, integrated network architecture

Citation: Zhang W J, Qin J, Mei F, et al. Short-term power load forecasting using integrated methods based on long short-term memory. Sci China Tech Sci, 2020, 63, https://doi.org/10.1007/s11431-019-9547-4

1 Introduction

As an important part of the energy management system, power load prediction quality has a direct impact on the analysis results of the subsequent safety check in the power grid, which is of great significance to the dynamic state estimation, load scheduling and power generation cost reduction of the power grid [1].

Due to the increasement of spatial load, the development of electric vehicles, and the promotion of electric power market, the sources and volume of load data increase significantly as well as the difficulty of load forecasting. Traditional forecast methods (including time series, the grey system, trend extrapolation, regression analysis, etc.), support vector machine (SVM) regression [2] and artificial neural network [3], which are all shallow model prediction methods that are difficult to extract deep characteristics of load sequence, curb the generalization of model performance and hinder the further improvement of prediction accuracy when it comes to the problems of modern electric power system load forecasting.

In recent years, the restricted Boltzmann machine (RBM) has been demonstrated that it has a great performance in costumers electricity load forecasting [4]. Some scholars have applied the deep belief network (DBN) to short-term load forecasting, achieving high prediction accuracy, and thus verifying the advantages of deep neural network model [5]. A combination of self organizing map (SOM) and recurrent neural networks (RNN) was proposed for prediction in ref. [6] which verified the feasibility of integrated learning method. An integrated learning method combining k-means clustering and convolutional neural network (CNN) was adopted for load forecasting in ref. [7], which significantly reduced the prediction error compared with the single CNN method.

*Corresponding author (email: wwyu@seu.edu.cn)

However, the temporal correlation of time series data has not been considered in the above methods since there are no memory units inside them. Therefore, the LSTM model in deep learning is naturally used for power load forecasting to construct model which takes the time-related factors of power load in account by connecting the last time information to the current time task. The standard LSTM and the LSTM-based sequence to sequence (S2S) methods were used for building energy load in ref. [8] on consumption dataset of residential load. They were trained and tested with one hour and one minute resolution. A combination of Autoencoder and LSTM was investigated in ref. [9] for forecasting renewable energy power plants, and the predictive performance was compared with the state of art approaches such as Artificial Neural Network, LSTM and DBN. The traditional grid total load or substation load forecasting methods are not applicable to single household load forecasting, so the authors in ref. [10] have proposed a method based on LSTM and Recurrent Neural Network (RNN) for short-term load forecasting of single household users, which has solved the problem of random fluctuation of load and great uncertainty. They have also predicted the short-term intelligent metering level user load based on LSTM deep learning method, and the results showed that LSTM is better than the traditional time series and kalman filter load prediction methods [11].

In this paper, three new integrated forecasting methods based on LSTM are proposed to handle the occasional failure of prediction in LSTM forecasting, the complex manually debugging problems, and also to improve the accuracy of short-term load forecasting. The rest of this paper is arranged as follows: Sect. 2 presents the preliminary knowledge of SOM, LSTM and chaotic time prediction. The three flaws in LSTM stepwise prediction are analyzed in Sect. 3, and the three novel LSTM integration methods to deal with the problems are proposed in Sect. 4. Numerical simulation and analysis are performed in Sect. 5, and Sect. 6 gives the conclusion and future work.

2 Preliminaries

In this section, preliminaries on self-organizing map and a deep network with long and short memory are presented. In addition, the chaotic time series prediction method is also introduced.

2.1 Self-organizing map

The network structure of self-organizing map (SOM) is composed of two layers: the input layer and the competition layer (or output layer), which contain input neurons and competition neurons respectively. The competition neurons are organized as a one-dimensional linear array or a two-dimensional planar array as shown in Figure 1 where the network is fully connected, that is, each input node is connected to all output nodes.

Let $\mathcal{X} = (X_1, X_2, \cdots, X_N)$ be the input samples, and there are n input neurons in each sample, i.e., $X_r = (x_{r1}, x_{r2}, \cdots, x_{rn})^{\mathrm{T}}$, $r = 1, 2, \cdots, N$. Denote $W = (w_1, w_2, \cdots, w_p)$ as the weights of connections between input neurons and competition neurons, where $w_j = (w_{1j}, w_{2j}, \cdots, w_{nj})^{\mathrm{T}}$ is the weight vector corresponding to the jth output competition unit. It also represents the center of the corresponding jth class, for each $j = 1, 2, \cdots, p$, where p denotes the number of competition neurons and clusters.

The SOM finds the winner neuron unit and adjusts weights in the tth iteration respectively by [12]

$$i(x) = \arg\min_j \left(\sum_{k=1}^{n} (|x_{rk} - w_{kj}|)^q \right)^{\frac{1}{q}}, \quad j = 1, 2, \cdots, p, \tag{1}$$

$$w_j(t+1) = \begin{cases} w_j(t) + \eta(t)[x - w_j(t)], & j \in NE_i(t), \\ w_j(t), & \text{otherwise}, \end{cases} \tag{2}$$

where $i(x)$ represents the subscript of the winning neuron when the input is x, that is, the subscript of the neuron whose weight value is closest to the input x in the competition layer and q is the parameter in the Mingkowski distance. Particularly $q = 2$ is the Euclidean distance, $\eta(t) = \frac{1}{t}$ is the step size which varies with t and $NE_i(t) = \{j | \, \|c_j - c_i\| \leqslant d(t), \ j = 1, \cdots, p\}$ is the neighborhood function of winner competition neuron i, where c_j is the location of jth competition neuron in corresponding array and $d(t)$ is the neighborhood diameter which decreases over time [13].

The SOM trains with eqs. (1) and (2) until its weights are consistent with input samples to achieve the best match. Then the clustering centers $\{w_j\}$ are obtained.

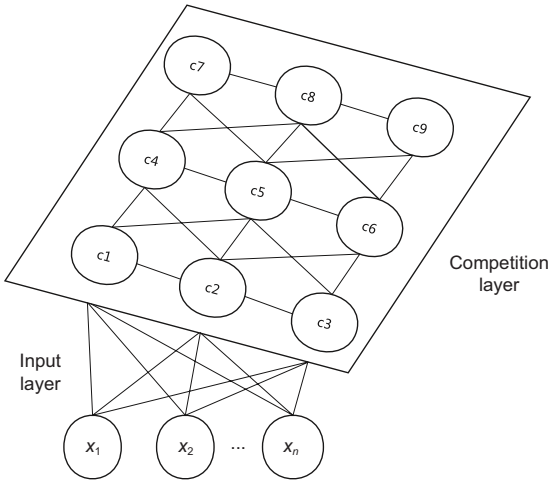

Figure 1 Self-organizing map structure.

2.2 Long short-term memory network

The Long short-term memory network (LSTM) is a kind of cyclic neural network. Its hidden units are composed of memory unit, input gate, forgetting gate and output gate, as shown in the Figure 2.

The corresponding functions of each gate structure and memory unit are as follows [14]:

$$f_t = \sigma(w_{fx} \circ x_t + w_{fh} \circ h_{t-1} + b_f), \tag{3}$$

$$i_t = \sigma(w_{ix} \circ x_t + w_{ih} \circ h_{t-1} + b_i), \tag{4}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(w_{cx} \circ x_t + w_{ch} \circ h_{t-1} + b_c), \tag{5}$$

$$o_t = \sigma(w_{ox} \circ x_t + w_{oh} \circ h_{t-1} + b_o), \tag{6}$$

where f_t, i_t, c_t, o_t represent the vector values of the LSTM network node at t in the forgotten gate, input gate, memory unit and output gate, respectively, w_{fx}, w_{ix}, w_{cx}, w_{ox} represent the weights between input unit and the corresponding gate or memory unit, w_{fh}, w_{ih}, w_{ch}, w_{oh} represent the weights between the hidden unit of last moment and the corresponding gate or memory unit, b_f, b_i, b_c, b_o represent the bias of the corresponding gate or memory unit, x_t is the input unit at time t, h_{t-1} is the hidden unit at $t-1$, and σ is the sigmoid function.

The forward calculation of LSTM at t produces $h_t = o_t \circ \tanh(c_t)$ and c_t, then LSTM is trained and learnt by the back propagation through time algorithm (BPTT) [14]. This forward calculation and back propagation process continues until the maximum number of iteration epochs or the given training precision is reached.

2.3 Chaotic time series prediction

Suppose the time series are $x(t)$, $t = 1, 2, \cdots, T$, and R^m is the reconstructed phase space where m is embedded dimension, then $x(t)$ are reconstructed in R^m as

$$Y(t) = (x(t), x(t+\tau), \cdots, x(t+(m-1)\tau)), \tag{7}$$

where τ is delay time, $t = 1, 2, \cdots, N$, $N = T - (m-1)\tau$.

Takens delay embedding theorem [15] shows that an equivalent phase space which reflects various dynamic behaviors of the original system can be reconstructed when the delay time τ and the embedding dimension m are given appropriate values respectively. There exists a map f in R^m to predict the system with the equation:

$$Y(t+k) = f(Y(t)), \tag{8}$$

where k is the prediction steps.

Take $\{Y(1), Y(2), \cdots, Y(N-k)\}$ and $\{Y(k+1), Y(k+2), \cdots, Y(N)\}$ as the fitting input set and output set respectively, then the corresponding development rule eq. (8) is found where f is approximated by a proper fitting function. These points from $T+1$ to $T+k$ steps are predicted by eq. (8) where the last position component of $Y(N-i)$ can be taken as the predicted value $x(T+i)$ for each $i = 1, 2, \cdots, k$. The above process is shown in Figure 3.

3 Problem statement

The LSTM can be used for power load prediction in two ways: centralized prediction and stepwise prediction. The former one is to predict the values of all steps at once, and the latter one is to predict the value of each step until the final step. Centralized prediction requires $H \cdot (2T + H)$ neurons at least, where H is the neuron dimension of hidden layer, while only $h \cdot (2 + h)$ neurons are needed for the stepwise prediction since $h \cdot (2 + h)$ neurons are needed for each one of T models in the stepwise prediction, where h is the neuron dimension of the hidden layer, and the parameters in each model are fine tuned by transfering weights of the previous step neurons. Note that the memory features between sequences are

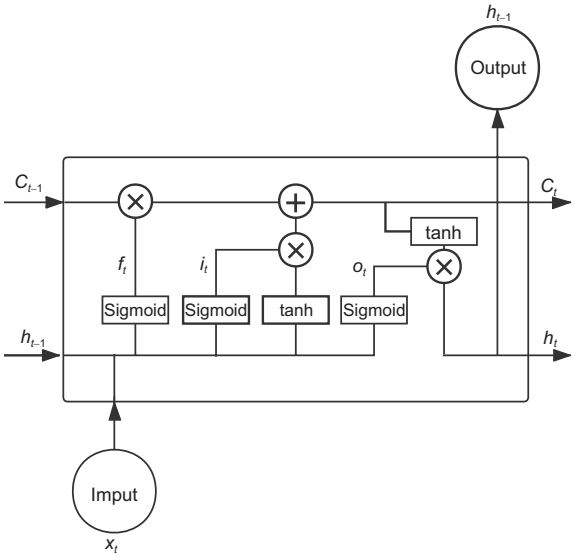

Figure 2 LSTM structure.

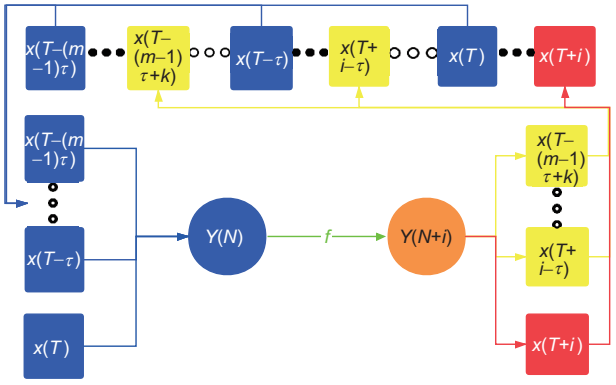

Figure 3 (Color online) Chaotic time series prediction.

reduced in stepwise prediction, which means $h \leqslant H$. The number of neurons required by the stepwise prediction model is obviously smaller than centralized prediction model, so it has a relatively low time complexity and space complexity. At the same time, the smaller number of neurons indicates that the LSTM model has fewer parameters, which means this kind of model is easier to conduct tuning and training. However, there are also some limitations in the stepwise prediction of power load by LSTM which are detailed below.

3.1 Complex prediction model

In the power market environment, the state of urban development, the user's energy-using behavior and their response to incentive policies all increase the complexity of the power load curve. Meanwhile, the large-scale distributed power access and widespread use of electric vehicles increase power load volatility as well [16].

As shown in Figure 4, the power load curve fluctuates violently so that the LSTM requires a large number of neurons to record the fluctuation characteristics of each curve, which makes the complexity of the prediction model increase rapidly with the growth of the load sequence.

For the above reasons, using LSTM to predict step by step is not suitable for large-scale power which demands large number of neurons.

3.2 Short forecast period

Usually, the power load is affected by external factors such as politics, economy and climate as well as internal factors of the power system, which makes the power load curve behave irregular and unpredictable for the changes of future power load in a long period of time.

At the same time, it is easy to figure out that the mapping between output and input in LSTM is approximated by a compounded function consists of a group of simple linear and nonlinear functions in eq. (3) to eq. (6). This strong nonlinear and laminated mapping relationship generates rich dynamics that contain many fixed points, which implies that the prediction values are easy to converge into these fixed points as shown in Figure 5.

As a result, the long-time prediction can easily fail.

3.3 Tedious manual debugging

The predictive performance and learning ability of LSTM depend on the parameter settings of the network as shown in Figure 6, where the test RMSE varies dramatically as the weights change. So it brings great difficulty to manual debugging.

Therefore, the adjustment of parameters in LSTM is a huge and time-consuming work as for large-scale power load forecasting.

4 LSTM integrated forecasting algorithms

In order to solve the three issues of complex prediction model, short forecast period and tedious manual debugging in LSTM prediction, this paper puts forward three new integration algorithms which are based on the self-organizing feature map, chaotic time series prediction and intelligent

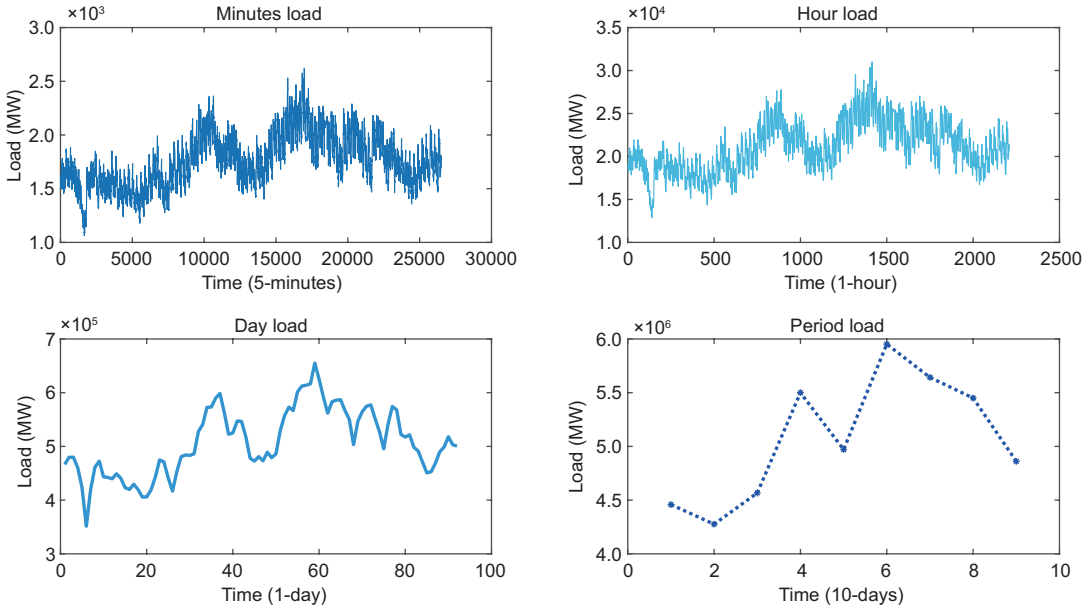

Figure 4 (Color online) Load curve.

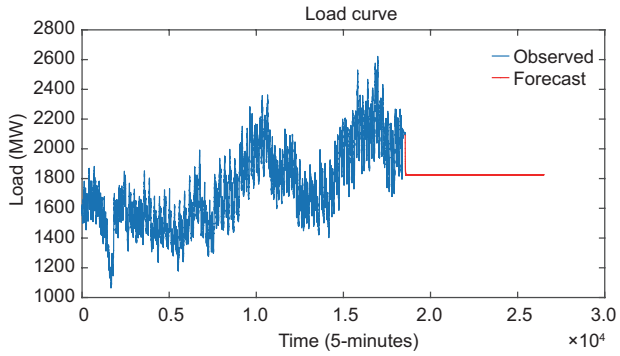

Figure 5 (Color online) Persistence forecasting.

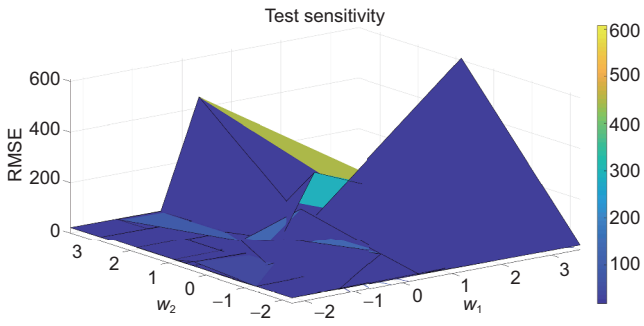

Figure 6 (Color online) Test sensitivity.

optimization respectively to improve the forecasting performance. The algorithm architecture diagram is shown in Figure 7 and the integration algorithms are described as below.

4.1 LSTM prediction based on self-organizing feature mapping

There are some significant differences in electricity consumption during different time period, where the power load is relatively low in the dawn time frame (00:00–06:00) and morning time frame (06:00–12:00), and relatively high in the afternoon time frame (12:00–18:00) and evening time frame (18:00–24:00).

In order to make full use of the power load correlation in the same time frame, the power load data samples are divided according to this four time frames to reduce the fluctuation of the power load curve as shown in Figure 8. Then the accuracy of load prediction can be improved in different time periods.

In essence, segmented forecasting is an artificial equal classification method for prediction, that is, the number of samples is the same in each category. Although this method reduces the fluctuation of load curve, the overall complexity of load curve has not been effectively decomposed, which means the complexity of model training and the number of needed neurons have not been reduced significantly.

Since power load forecasting is also affected by factors such as precipitation, temperature, air pressure, humidity, sunshine, wind speed and holidays, the self-organizing feature map clustering is used to capture the similarity from days in the same class, so as to reduce the complexity of load curve. The classified load forecasting algorithm is proposed in Algorithm 1.

4.2 LSTM prediction based on chaotic time series

Power load curve presents a multi-level and self-similar dynamic evolution process, which makes the changes of power load data between the previous and the next moment, and the same time on different dates have certain regularity. At the same time, the actual power system is a chaotic system where the generated load data have inherent randomness. The multidimensional chaotic characteristics of the original dynamic system can be recovered from the actual power load data by means of phase space reconstruction, and the chaotic attractor characteristics can help avoid the phenomenon of LSTM

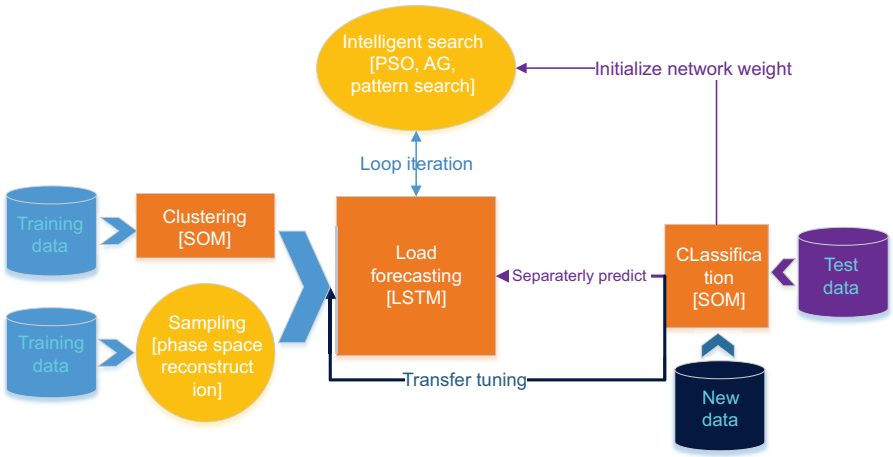

Figure 7 (Color online) Algorithm frame.

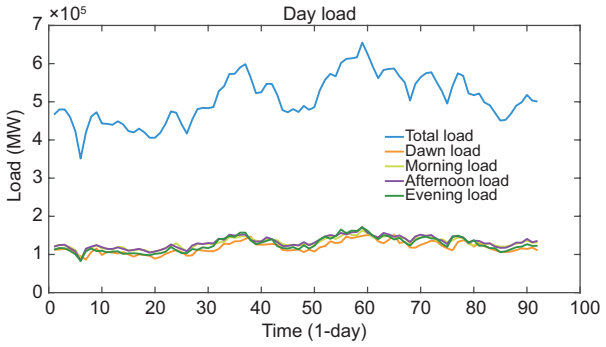

Figure 8 Sectional load curve by day.

Algorithm 1 Classified load forecasting algorithm

Initialize W, $NE_i(0)$, $\eta_{\min}$, and $\eta(0)$, where $i = 1, \cdots, p$
Normalize X
$t \leftarrow 0$
Repeat
  Find the winner neuron unit j^* with eq. (1)
  Adjust w_j through eq. (2), where $j \in NE_{j^*}(t)$
  $NE_{j^*}(t) = NE_{j^*}(t) - 1$
  $t \leftarrow t + 1$
  $\eta(t) = \frac{1}{t}$
Until $\eta(t) < \eta_{\min}$
Get p types of samples Y from trained SOM
Put 70% of Y for trained in LSTM
Test 30% of Y in trained LSTM

prediction errors accumulating to the fixed point, thus extending the prediction interval.

The chaotic characteristics of power load can be quantitatively characterized by calculating the fractal dimension and the maximum Lyapunov exponent of power load data. The box dimension of power load is 1.68 which is calculated by $\lim_{\varepsilon \to 0} \frac{\log N(\varepsilon)}{\log(1/\varepsilon)}$ and the correlation dimension of power load is 5.2 which is calculated by G-P algorithm [17]. This two fractal dimensions indicate that the power load curve is non-smooth, non-continuous, self-similar and multi-level. The maximum Lyapunov exponent of the power load data series is 0.00052 which is calculated by the small data volume method [18]. The positive maximum Lyapunov exponent indicates that the system has chaotic effect, and the evolution orbit of power load time series will converge to the chaotic attractor in the high-dimensional phase space. The above analysis indicates that the time series of power load are chaotic time series, thus the phase space reconstruction method can be used to process the power load samples so that their orbits present the chaotic characteristics which are easy for LSTM to capture.

We obtain the embedding dimension $m = 11$ and the delay time of $\tau = 62$ by C-C method [19]. The power load data points are reconstructed by eq. (7), and LSTM is used to approximate the function eq. (8).

Finally, LSTM is trained continuously through track points until the specified number of steps is reached. The corresponding algorithm is shown in Algorithm 2.

4.3 LSTM prediction based on intelligent optimization

Due to the complex mapping between the LSTM prediction precision and initial weights, it is hard to get gradient information directly for optimization. The manual debugging is quite tedious, and it does not make full use of the initial weight information among each set. Therefore, an intelligent optimization is put forward to achieve faster search for better local optimal weights in LSTM. Specifically, the LSTM weight W is taken as a single agent, and the information interaction between different agents is conducted by the individual location and its search direction. Dominant agents are retained through population evolution, and new agents are explored and searched according to some pattern directions.

Firstly, the genetic algorithm [20] is used to carry out evolutionary calculation to obtain dominant groups from a group of random initial weights. Then a certain proportion of dominant groups are searched thorugh pattern search [21] to avoid the concentration of dominant groups in a local area. The scattered population acquired by pattern search and the most adaptable individual in dominant population generated from genetic algorithm are taken as part of initial particles, and then the particle swarm optimization algorithm in refs. [22, 23] is used to search the optimal individual with these particles moving. Finally, pattern search explores or verifies the optimal individual obtained by particle swarm search. The above processes are described by Algorithm 3.

5 Numerical simulation

In this section, simulations on real power load data set are

Algorithm 2 Chaotic load forecasting algorithm

Initialize W, maxstep
Normalize X
Compute m and τ
Let Y = the reconstructed X by eq. (7)
$t \leftarrow 0$
Repeat
  $t \leftarrow t + 1$
  Put 70% of Y for trained in LSTM
  Test 30% of Y for the t steps by trained LSTM
  Get forcasting result $R(t)$ in the 6 step
Until t ==maxstep
Let $R = \sum_{t=1}^{\text{maxstep}} \frac{R(t)}{\text{maxstep}}$

Algorithm 3 Intelligent load forecasting algorithm

1: Initialize a group of weights W_I
2: Normalize X
3: Let W_I as the initial population
4: Search for the dominant groups W_D by genetic algorithm
5: Sort W_D by fitness
6: Select part of top in sorted W_D as initial points W_{D1} in pattern search
7: Find the better scattered population W_S using pattern search
8: Select W_S and the most adaptable individual in W_D as part of initial particles
9: Obtain the optimal individual W_O through particle swarm algorithm
10: Search for a better individual starts from W_O in pattern search

performed to illustrate the effectiveness of the proposed methods. The 26496 power load data samples used in this paper are collected every five minutes from June to August, 2011 in Nanjing, China. All data are standardized for training and testing, among which 70% are selected as training samples, and the remaining 30% are used for testing.

5.1 Classified forecasting

The factors that affect power load, including daily precipitation, daily high temperature, daily low temperature, daily average temperature, air pressure, relative humidity, sunshine time, wind speed, and holiday labels, are used to cluster the 92 days' sample data set with SOM to obtain four groups of classified sample sets. The number of sample sets and the spatial distribution of four cluster points are shown in Figure 9. The distribution of the four clustering points are surrounded by most sample points, which indicates that SOM has a good clustering effect.

Figure 10 shows the weights distribution of the nine factors affecting power load, where the light color indicates the larger weight, and the dark color indicates the small weight. It shows that the four cluster groups are divided in accordance with the dry humidity, rain, temperature and other factor. Specifically, the weight of the upper left corner and lower right corner denote dry humidity factor, and the lower left corner and upper right corner denote temperature influence

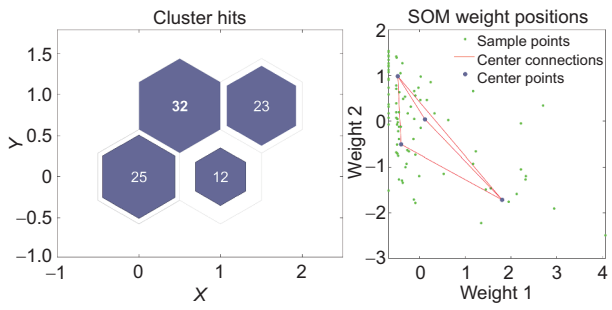

Figure 9 Cluster sample sets and spatial distribution.

factor. For the upper left corner weight, the light color shows partial dry climate and the dark color shows partial wet climate. For the lower right corner weight, the light color indicates more rains and the dark color shows the less rains. For both of the lower left corner and upper right corner weight, the light color shows temperature has strong influence, and the dark color shows weak influence. Moreover, the holiday factor is taken as other factors which are characterized by the combination of these four weights.

In order to illustrate the effectiveness of the proposed algorithm, the LSTM prediction based on artificial time segment (dawn, morning, afternoon and evening) is selected for comparison with that based on SOM classification. The LSTM network parameters are set as in Table 1.

Figures 11 and 12 are the results and residual graphs of segmented forecasting and classified forecasting respectively.

Table 2 shows that the segmented forecast accuracy are 1.7% and 14.7% in the morning and afternoon sessions respectively. The afternoon time accuracy is considerably improved, but the overall accuracy improvement is not obvious. Because the segmented forecasting merely reduces the volatility instead of the complexity of load curve, and the

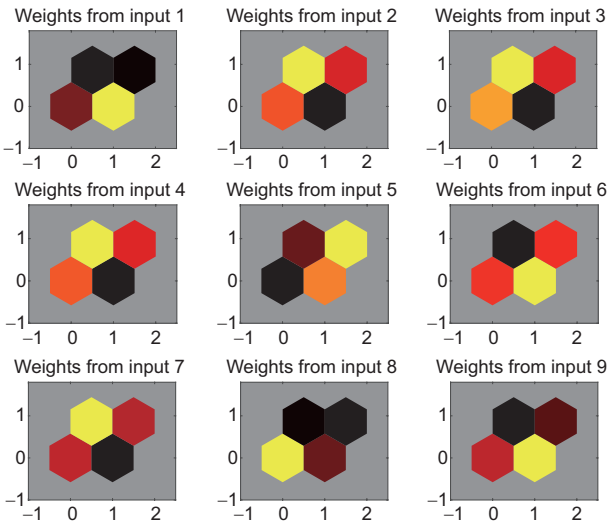

Figure 10 The weight distribution of nine factors.

Table 1 LSTM parameters

Network parameters	Artificial time segment	SOM classification
Input neuron dimension	1	1
Output neuron dimension	1	1
Hidden neuron dimension	25	15
Max epochs	200	200
Initial learn rate	0.05	0.05
Learn Rate Drop Period	100	100
Learn Rate Drop Factor	0.2	0.2

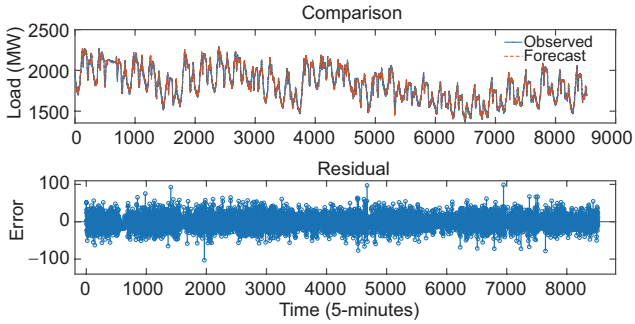

Figure 11 Segmented forecasting.

splicing of load curve from every morning and night time respectively increases the complexity compared to the origin load curve in some extents.

In comparison, the classified forecast accuracy in three categories are improved by 2.9%, 7.3% and 6.6% respectively, the overall accuracy is improved by 2% and the number of neurons required is reduced by 60% at the same time. This is because the classified forecasting extracts three local fluctuation characteristics of the whole load curve, as shown in the Figure 13.

Table 2 Forecasting precision

Forecasting methods	Overall RMSE reduction rate	Group RMSE reduction rate
Segmented	0.4%	(1.7%, –10.1%, 14.7%, –3.1%)
Classified	2%	(2.9%, –3.6%, 7.3%, 6.6%)
Hierarchical	–40%	(–57.5%, 0.2%, –438.7%, –41.5%)
K-Means	–51%	(–2.1%, 3.5%, –356.6%, 4.8%)
Mixed Gaussian	–4%	(–3.3%, –63.5%, 7.4%, 0.3%)

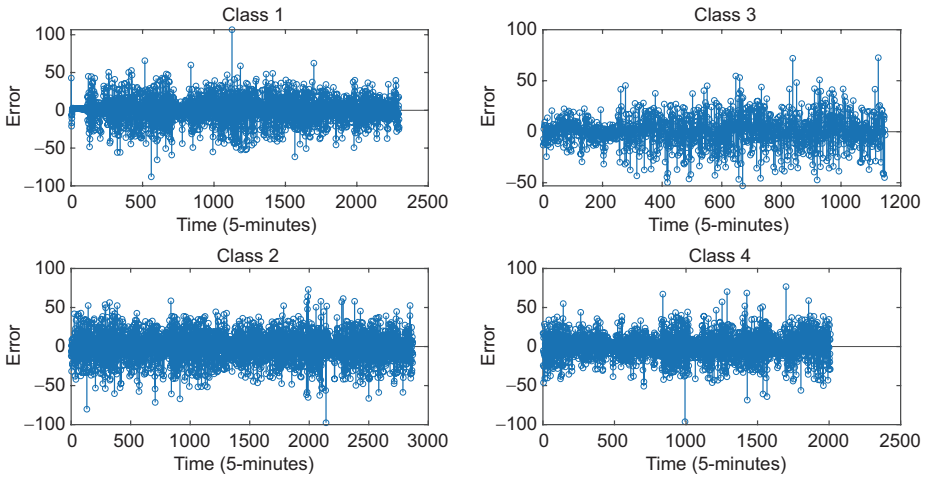

Figure 12 Classified forecasting.

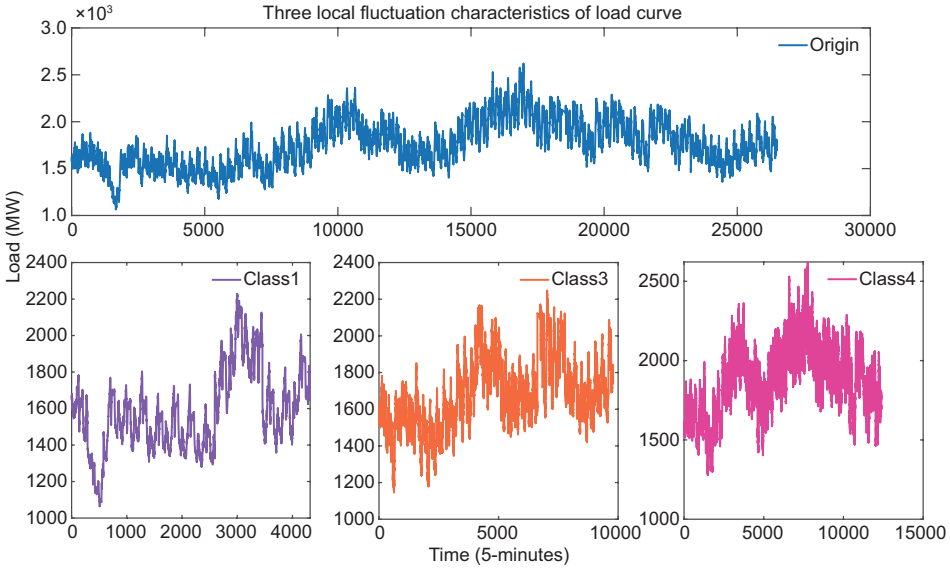

Figure 13 Three local fluctuation characteristics of load curve.

In order to highlight the advantages of SOM in clustering, three common clustering algorithms, including hierarchical clustering, k-means clustering and mixed gaussian distribution clustering, are selected to test on LSTM. Table 2 shows that this three common clustering algorithms do not work well on LSTM, because the difference of sample size distributions in the formed clusters leads to the remarkable difference in forecasting performance, so that the overall forecasting performance is poor. Conversely, the quantity distributions of samples in SOM clusters are relatively uniform to ensure the small difference between forecasting performance, which means the integrated algorithm that combined SOM with LSTM has the best performance for forecasting.

5.2 Multidimensional forecasting

According to Algorithm 2, set the maximum forecast step number to 20 for LSTM continuous forecasting, and the forecasting result is shown in Figure 14. It can be seen that the LSTM continuous forecasting no longer gets stuck at the fixed point.

Compared with LSTM direct forecasting, the forecasting accuracy of Algorithm 2 improves by 17.1% and 55.3% in the first 20 steps and 798 steps respectively. Figures 15 and 16 indicate that the characteristics of chaotic attractors in power load sequences are learnt by LSTM training process.

5.3 Intelligent forecasting

Twenty groups of LSTM weight parameters are generated, among which 80% are random weights between $(-1, 1)$ and 20% are weights near the zero point. The forecast average RMSE with these initial weight is 45.457. The related parameters of genetic algorithm, particle swarm optimization and pattern search used in Algorithm 3 are set as shown in Table 3, and 6-cores CPU are used for parallel calculation.

Figure 17 presents the result of genetic evolution calculation on the initial weights of the twenty groups. The bottom subgraph shows the evolution of the population, where the red lines indicate mutation children, the blue lines indicate crossover children, and the black lines indicate elite individuals in each generation. The top subgraph shows the optimal individual prediction error is up to 17.4677, and the average prediction error of the final population which is concentrated at 34.0582 which has the 25.08% prediction accuracy improvement compared to the original initial weights.

The first 25% of the dominant individuals in the population are used to explore for more optimized points by pattern search, and the results are shown in Table 4. The average

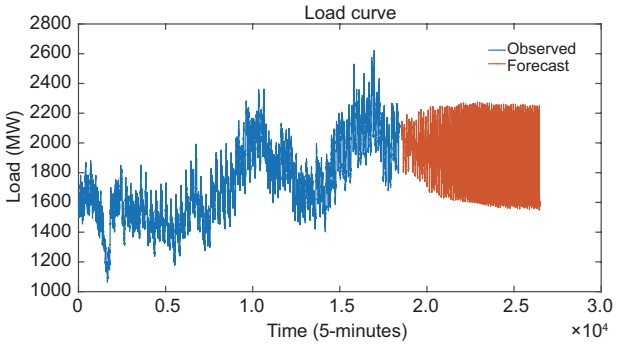

Figure 14 Multidimensional forecast result for entire time.

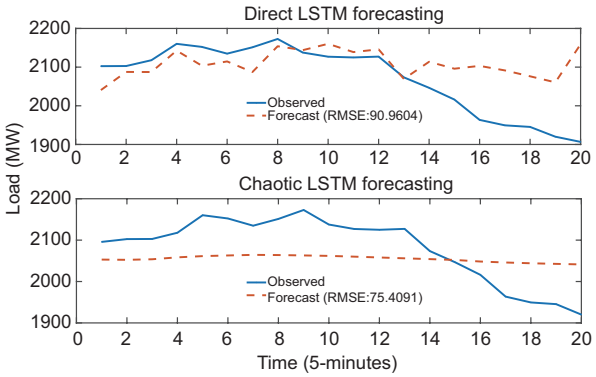

Figure 15 Multidimensional forecast result for short time.

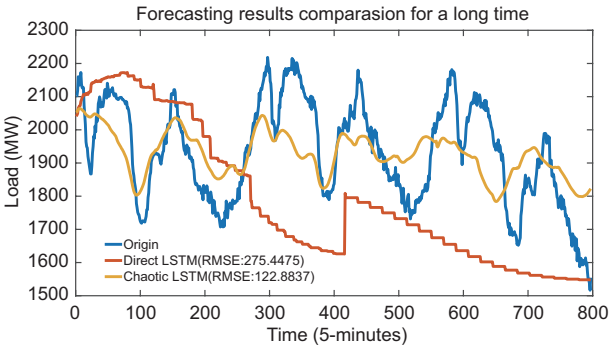

Figure 16 Multidimensional forecast result for long time.

Table 3 Intelligent optimization parameters

Algorithm	Parameter	Value
Genetic	Population size	20
	Crossover fraction	0.8
	Migration fraction	0.2
	Migration interval	20
	Elite count	0.05*population size
Particle swarm	Swarm size	20
	Min neighbors fraction	0.25
	Self adjustment weight	1.49
	Social adjustment weight	1.49
Pattern search	Initial mesh size	0.2
	Mesh contraction factor	0.5
	Mesh expansion factor	2

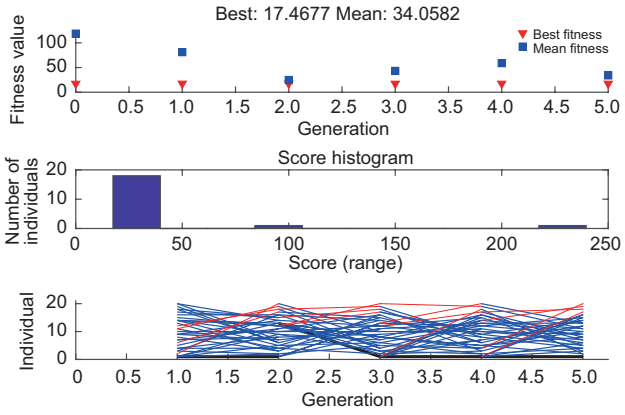

Figure 17 Genetic algorithm result.

Table 4 Genetic and pattern search result

No.	Genetic algorithm	Pattern search
1	17.4677	17.4458
2	17.5547	17.4628
3	17.5583	17.4742
4	17.7735	17.5956
5	17.8134	17.5727
Average	17.6335	17.5102
Distance	24.4429	25.849

RMSE of the dominant individuals is 17.5102 after pattern search and the average spacing of the dominant population is 25.8495, which means the average prediction accuracy of the population is improved and the dominant individuals are pulled apart to avoid falling into the identical local optimum.

The dominant individuals obtained by genetic algorithm and pattern search are taken as the initial particle swarm, and the particle swarm algorithm is used to further reduce the prediction error to 17.3652.

Figure 18 shows that this algorithm converges to the locally optimal particle, and there is a lower valley where the prediction error is 17.3331 exists within the range of radius 1.6 of the optimal particle detected by pattern search shown in Figure 19.

Finally, the initial weights with the RMSE of 17.3331 has been found by Algorithm 3, improving the prediction accuracy by 61.87% on average compared to initial weights generated randomly in the LSTM load forecasting.

6 Conclusion and future work

The LSTM is now commonly used in short-term load forecasting of large-scale electric power grid. Compared with traditional forecasting, LSTM has better prediction performance. This paper attempts to solve the problems of short forecasting cycle and inconvenient debugging in LSTM

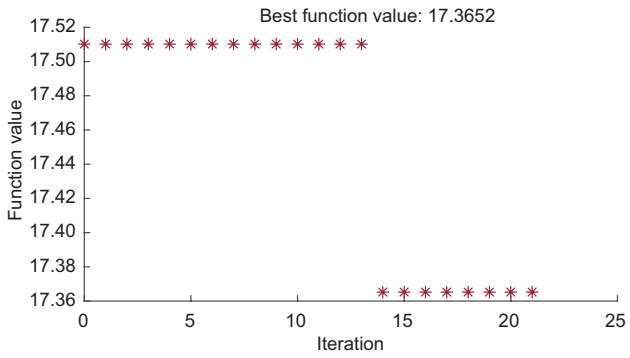

Figure 18 Particle swarm optimization.

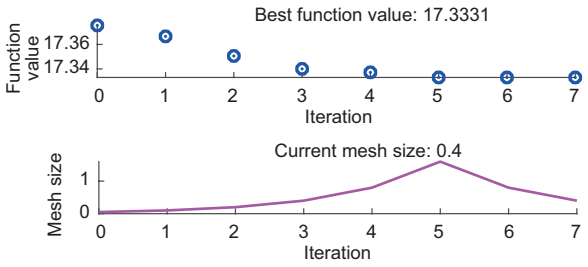

Figure 19 Pattern search.

stepwise prediction and to improve the accuracy of LSTM forecasting. Three integrated algorithms are proposed, that is, the LSTM combined with SOM, chaotic time series and intelligent optimization algorithm respectively, which can effectively address the above three problems. In the future, the characteristics of power load will be further employed to improve the forecast accuracy, and the learning algorithm will be accelerated by combining the methods of meta-learning and transfer learning.

This work was supported by the National Natural Science Foundation of China (Grant No. 61673107), the National Ten Thousand Talent Program for Young Top-notch Talents (Grant No. W2070082), the General Joint Fund of the Equipment Advance Research Program of Ministry of Education (Grant No. 6141A020223), and the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence (Grant No. BM2017002).

1 Amjady N. Short-term hourly load forecasting using time-series modeling with peak load estimation capability. IEEE Trans Power Syst, 2001, 16: 798–805
2 Fan G F, Peng L L, Hong W C, et al. Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression. Neurocomputing, 2016, 173: 958–970
3 Hu R, Wen S, Zeng Z, et al. A short-term power load forecasting model based on the generalized regression neural network with decreasing step fruit fly optimization algorithm. Neurocomputing, 2017, 221: 24–31
4 Ryu S, Noh J, Kim H. Deep neural network based demand side short term load forecasting. In: Proceedings of the 2016 IEEE International Conference on Smart Grid Communications. IEEE, 2016. 10: 308

5 Dedinec A, Filiposka S, Dedinec A, et al. Deep belief network based electricity load forecasting: An analysis of Macedonian case. Energy, 2016, 115: 1688–1700

6 Cherif A, Cardot H, Boné R. SOM time series clustering and prediction with recurrent neural networks. Neurocomputing, 2011, 74: 1936–1944

7 Dong X, Qian L, Huang L. Short-term load forecasting in smart grid: A combined cnn and k-means clustering approach. In: Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE, 2017. 119

8 Marino D L, Amarasinghe K, Manic M. Building energy load forecasting using deep neural networks. In: Proceedings of the Industrial Electronics Society, IECON 2016-42nd Annual Conference. IEEE, 2016. 7046

9 Gensler S A, Henze J, Sick B, et al. Deep learning for solar power forecasting: an approach using autoencoder and lstm neural networks. In: Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2016. 002858

10 Kong W C, Dong Z Y, Jia Y W, et al. Short-term residential load forecasting based on LSTM recurrent neural network. IEEE Trans Smart Grid, 2019, 10: 841–851

11 Kong W C, Dong Z Y, Hill D J, et al. Short-term residential load forecasting based on resident behaviour learning. IEEE Trans Power Syst, 2018, 33: 1087–1088

12 Kohonen T. The self-organizing map. Proc IEEE, 1990, 78: 1464–1480

13 Vesanto J, Alhoniemi E. Clustering of the self-organizing map. IEEE Trans Neural Netw, 2000, 11: 586–600

14 Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput, 1997, 9: 1735–1780

15 Takens F. Detecting strange attractors in turbulence. In: Dynamical Systems and Turbulence, Warwick 1980. Berlin, Heidelberg: Springer, 1981. 366–381

16 Liu J, Gao H, Ma Z, et al. Review and prospect of active distribution system planning. J Mod Power Syst Clean Energy, 2015, 3: 457–467

17 Grassberger P, Procaccia I. Measuring the strangeness of strange attractors. Phys D-Nonlinear Phenomena, 1983, 9: 189–208

18 Rosenstein M T, Collins J J, De Luca C J. A practical method for calculating largest Lyapunov exponents from small data sets. Phys D-Nonlinear Phenomena, 1993, 65: 117–134

19 Kim H S, Eykholt R, Salas J D. Nonlinear dynamics, delay times, and embedding windows. Phys D-Nonlinear Phenomena, 1999, 127: 48–60

20 DeJong K A. Analysis of behavior of a class of genetic adaptive system. Dissertation for Dcotoral Degree. Ann Arbor: University of Michigan, 1975

21 Audet C, Dennis Jr J E. Analysis of generalized pattern search. doi: 10.1109/ICIP.2003.1247183

22 Kenedy J, Eberhar R C. Particle swarm optimization. In: Proceedings of the 1995 IEEE International Conference on Neural Network. IEEE, 1995. 4: 1942

23 Kenedy J, Eberhart R C. A new optimizer using particle swarm. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science. Nagoya, 1995