Revisit Neural Network based Load Forecasting

Yingshan Tao, Fei Zhao, Haoliang Yuan
Department of Electrical Engineering,
School of Automation
Guangdong University of Technology
Guangzhou, China
yings_tao@foxmail.com
503188839@qq.com

Chun Sing Lai
School of Automation
Guangdong University of Technology
Guangzhou China
School of Civil Engineering
University of Leeds, UK
c.s.lai@leeds.ac.uk

Zhao Xu
Department of Electrical Engineering,
Hong Kong Polytechnic University
Kolwoon, Hong Kong
zhao.xu@polyu.edu.hk

Wing Ng
School of Computer Science
South China University of Technology
Guangzhou, China
wing.ng@ieee.org

Rongwei Li, Xuecong Li
Department of Electrical Engineering,
School of Automation
Guangdong University of Technology
Guangzhou, China
674150948@qq.com

Loi Lei Lai
Department of Electrical Engineering
School of Automation
Guangdong U of Technology, China
Guangzhou, China
l.l.lai@gdut.edu.cn

Abstract— **The application of artificial neural network to load forecasting can overcome the problem of dynamic load change, and its ability to adapt to nonlinear relationships makes the prediction result satisfactory. This paper firstly reviews and introduces the concepts and basic principles of load prediction, discusses various methods for load forecasting, and then selects artificial neural network to establish a predictive model. In this paper, the European electric load is predicted with a BP neural network. From the prediction results, it is feasible to use BP neural network for load forecasting, and its accuracy can meet the needs of real-life engineering work. However, BP neural networks have the problem of slow convergence and easily falling into local minimum points. Therefore, this paper also uses three other neural networks for load forecasting, which are Radial Basis Network (RBF), Elman Network, and Long-Short Term Memory Network (LSTM). In the experiment, the four neural networks achieved expected prediction results, and the LSTM network had the best prediction effect. Scientific discussions are offered.**

Keywords—load forecasting, neural networks, back propagation, Elman Network, radial basis function, long-short term memory

I. INTRODUCTION

With the rapid development of the world economy, users' electricity consumption has been increasing. Therefore, research work on power load forecasting began in the 1950s and 1960s. Power load forecasting techniques are segmented into computational intelligence methods and traditional methods. Traditional forecasting methods, which have large computational complexity, complex computational processes and the unsatisfied accuracy of predictions, is difficult to meet the needs of evolving modern power companies [1]. With the growth of Internet technology and artificial intelligence technology, power load forecasting has gradually entered a new era of intelligence. Artificial intelligence technology shows great advantages in load forecasting [2]. In 1991, Park D.C et al. firstly applied artificial neural networks to power load forecasting and achieved satisfactory prediction results [3]. At the end of the twentieth century, Bell Labs scientist Vapnikt proposed the Support Vector Machine (SVM), which demonstrates its unique advantages in solving nonlinearity and local minimum problems, as well as minimizing the empirical risk and confidence interval for load forecasting [4]. Y. Chen et al. presented a similar day-based wavelet neural network method to predict the day-ahead load. The model uses wavelet decomposition and individual neural networks to determine the features of high-frequency and low-frequency loads [5]. Applying deep neural networks as deep learning to short-term load forecast is a relatively new topic because of their excellent scheduling capabilities. Two common deep neural network models, known as the convolutional neural network (CNN) and recurrent neural network (RNN), were presented and constructed in both direct multi-step and recursive manners [6]. H. Shi, M. Xu and R. Li proposed a new deep recursive neural network based on dataset, which distributes the load data of a group of customers into an input set. Compared with the latest technology in home load forecasting, the RMSE of this method is 19.5% smaller than ARIMA, 13.1% smaller than SVR, and 6.5% smaller than RNN [7]. E. Ceperic, V. Ceperic and A. Baric made two important improvements to the support vector regression (SVR) prediction method. One is the feature selection algorithm using automatic model input while the other is the technology based on particle swarm global optimization, which reduces operator interaction [8]. A. Khosravi and S. Nahavandi et al. presented the Interval Type-2 fuzzy logic system (IT2 FLS) for short-term load forecasting. The model is accurate and superior compared to the traditional type 1 Takagi-Sugeno-Kang (TSK) fuzzy logic system and feedforward neural network [9]. Throughout the research on load forecasting, the research direction mainly has the following three aspects, namely, factors affecting load, prediction algorithm and prediction modeling. Researchers have conducted most of the research on predictive algorithms, and various algorithms have emerged. The accuracy, complexity and flexibility of different algorithms are quite different. In order to compare the predictive accuracy from different algorithms, we proposed a systematic approach to

choose BP neural networking method, Radial Basis Network (RBF), Elman Network, and Long-Short Term Memory Network (LSTM) to make the comparison, which use European load as data samples to predict the load of one day.

The rest of the paper is organized as follows. Section II describes the formulation of BP neural network method, and the details of another three neural network methods, which are RBF, Elman Network, and LSTM. Simulation results and discussions are given in Section III. Section IV concludes the paper.

II. NEURAL NETWORK BASED FORECASTING METHODS

The power load is often nonlinear, and it is also affected by random factors such as weather and emergencies. Artificial neural networks (ANN) can deal with nonlinear problems very well. Therefore, as an initial review, four methods of BP, RBF, Elman and LSTM networks are selected for load forecasting, and the prediction accuracies of the four methods are compared.

A. BP Network Modeling

Assume that the dimension of the input space of the model is I, the dimension of the output space is P, the number of hidden layers is J, and the number of training samples is H.

The learning of the BP algorithm [10] involves two key steps. The first step is forward transfer [11]. The input information goes through the output layer from the hidden layer processing. The formulations for the hidden and output layers are given in the following Equations (1) and (2) respectively:

$$h_j = f_1\left(\sum_{j=1}^{J} w_{ij}x_i + b_j\right) \qquad (1)$$

$$y_p = f_2\left(\sum_{p=1}^{P} w_{jp}h_j + c_p\right) \qquad (2)$$

Where x_i is the input characteristics of the network. h_j, y_p represent the output of the hidden and output layer, respectively; w_{ij} and b_j are the weight and threshold from the input layer to the hidden layer, respectively; w_{jp} and c_p are the weight and threshold from the hidden layer to the output layer, respectively.

The second step is to compare the predicted value with the expected value. If the predicted value does not meet the requirements, it will enter the back-propagation process to adjust the weight and threshold of the network. The objective function is:

$$E_{av}(r) = \frac{1}{2}\sum_{h=1}^{H}\sum_{p=1}^{P}\left(t_{ph} - y_{ph}\right)^2 \qquad (3)$$

Where t_{ph}, y_{ph} represent the actual and predicted load of the p_{th} hour on day h, respectively; r is the number of iterations of BP network.

The gradient descent method [12] is applied to the training of BP networks. In each iteration, the weight and offset are optimized according to the objective function (3), thereby improving the accuracy of the load prediction.

$$w_{ij}(r+1) = w_{ij}(r) - \varepsilon\frac{\partial E_{av}(r)}{\partial w_{ij}} \qquad (4)$$

$$b_j(r+1) = b_j(r) - \varepsilon\frac{\partial E_{av}(r)}{\partial b_j} \qquad (5)$$

$$w_{jp}(r+1) = w_{jp}(r) - \varepsilon\frac{\partial E_{av}(r)}{\partial w_{jp}} \qquad (6)$$

$$c_p(r+1) = c_p(r) - \varepsilon\frac{\partial E_{av}(r)}{\partial b_p} \qquad (7)$$

Where ε is the learning rate, and its value ranges from (0, 1).

B. RBF Network Modeling

The RBF (Radial Basis Function) neural network [13] is a multilayer forward network consisting of three layers. Its structure is shown in Fig. 1 below:
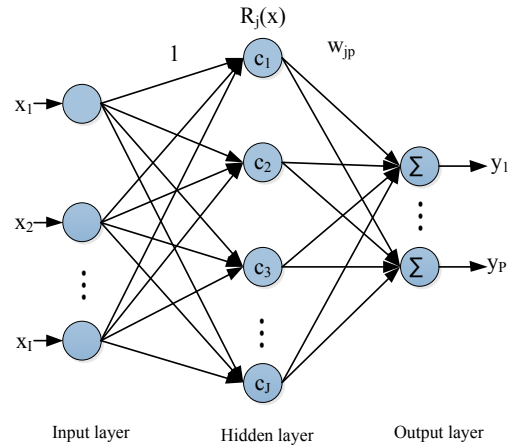

Fig. 1. Structure of RBF neural network

The transformation from the input layer to the hidden layer has a non-linear nature, mapping the input vector $x = (x_1, x_2, \cdots, x_i)$ directly into the hidden space without the weighted connection. The excitation function $R_j(x)$ of the hidden layer is a Gaussian function, which can be expressed by calculating the weight of the input from the center point of the function.

$$R_j(x) = \exp\left(-\frac{\|x - c_j\|^2}{2\sigma_j^2}\right) \qquad (8)$$

Where c_j is the center vector of the j_{th} hidden layer node; σ_j is the j_{th} diffusion constant of the radial basis function, determining the base width around the center point; $\|x - c_j\|$ is Euclidean norm;

The transformation from the hidden layer to the output layer is linear. The linear output is presented in the following expression.

$$y_p = \sum_{j=1}^{J} \omega_{jp} R_j(x), \ \text{p=(1, 2, ..., P)} \qquad (9)$$

Where w_{jp} is the weight from the hidden layer to the output layer; y_p is the output of the kth neuron; P is the number of neurons in the output layer.

C. Elman Network Modeling

Elman neural network was proposed by Elman in 1990 [14] which is a feedback network. Elman neural network is a common dynamic neural network and derives from the basic structure of an artificial neural network. It stores the internal state and has the function of mapping dynamic features so that the system can adapt to time-varying characteristics. Compared with the BP network, there is a receiving layer in addition to the input layer, the middle layer (hidden layer), and the output layer [15]. In the network, the output of the hidden layer can be fed back to itself through the receiving layer. Therefore, the receiving layer can be seen as a storage unit of the hidden layer, and the output of the previous instance is saved so that the network has the function of dynamic memory. The number of neurons in the receiving layer is the same as the number of neurons in the hidden layer. Its network topology is shown in Fig. 2 below:
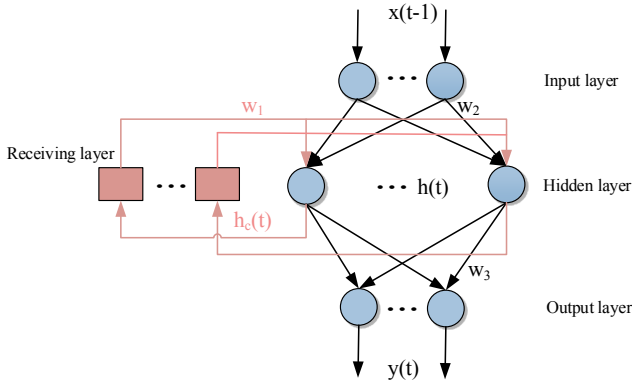

Fig. 2. Structure of Elman neural network

The nonlinear state-space expression [16] of the Elman neural network is shown below:

$$y(t) = g(w_3 h(t) + b_2) \qquad (10)$$

$$h(t) = f(w_1 h_c(t) + w_2(x(t-1)) + b_1) \qquad (11)$$

$$h_c(t) = h(t-1) \qquad (12)$$

Where t represents time, and y, h, x, h_c represent the one dimension output node vector, the m-dimensional hidden layer node unit vector, the n-dimensional input vector, and the m-dimensional feedback state vector, respectively. w_3, w_2, w_1 are the connection weight matrix of the hidden layer to the output layer, the input layer to the hidden layer, and the bearer layer to the hidden layer, respectively. $f(.)$ is the transfer function of the hidden layer neurons and adopts is a tansig function. $g(.)$ is the output layer transfer function and uses a purelin function. The b_1

and b_2 are the thresholds of the input layer and the hidden layer, respectively.

D. LSTM network models

Recurrent neural network (RNN) is a neural network which is good at processing nonlinear sequence data. The network consists of an input layer, a hidden layer, and an output layer. The activation function controls the output and the weights connect the layers. This allows the RNN to reflect relevant information in the sequence and make the model memory capable. However, RNN is prone to fall into the gradient problem and the explosion gradient problem when dealing with long-term dependency problems [17]. S. Hochreiter et al. proposed a new architecture for RNN to address the shortcomings of the basic RNN network and named it "LSTM" [18].

The LSTM storage unit consists of an input gate, an output gate, and a forgotten gate [19] that selectively reads, writes, and remembers events that occurred during the previous period. The structure is shown in Fig. 3 below:
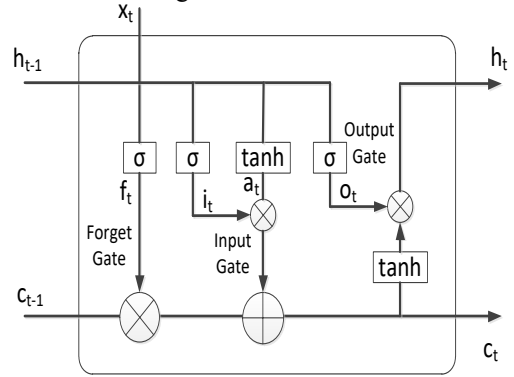

Fig. 3. Cell structure of the hidden layer

The output of the forgetting gate is expressed as Equation (13) below.

$$f_t = \sigma(w_f h_{t-1} + u_f x_t + b_f) \qquad (13)$$

Where f_t represents the probability of forgetting the state of the hidden layer cells at the last moment; w_f, u_f, and b_f represent the coefficients and biases of the linear relationship; σ denotes the sigmoid activation function [20, 21]; h_{t-1} denotes the output of the hidden layer at time t-1; x_t represent the input at time t.

The input gate consists of the following two parts:

$$i_t = \sigma(w_i h_{t-1} + u_i x_t + b_i) \qquad (14)$$

$$a_t = \tanh(w_a h_{t-1} + u_a x_t + b_a) \qquad (15)$$

Where w_i, u_i, b_i, w_a, u_a, b_a are the coefficients and biases of the linear relationship.

The result of the forgetting gate and the input gate will act on the cell state c_t:

$$c_t = c_{t-1} \otimes f_t \oplus i_t \otimes a_t \qquad (16)$$

The output gate consists of the following two parts:

$$o_t = \sigma(w_o h_{t-1} + u_o x_t + b_o) \qquad (17)$$

$$h_t = o_t \otimes \tanh(c_t) \qquad (18)$$

Where w_o, u_o, b_o represent the coefficients and biases of the linear relationship.

The final output is expressed as follows:

$$y_t = \sigma(vh_t + c) \tag{19}$$

Where v and c are the weight and threshold from the hidden layer to the output layer, respectively.

III. LOAD FORECASTING RESULTS

Our data set consists of the European Intelligent Technology Network (EUNITE) competitive load from Thursday, 1 January 1997 to Friday, 31 December 1998. We obtained half-hourly load data, average daily temperature data, weekday and holiday data. This work studies the forecasting performance for different methods with mean absolute percentage error (MAPE) [22].

We performed two simulation experiments with two different data sets. First, the load and temperature data from weekdays from 14 November 1998 to 28 November 1998 were used as a training set; the load and temperature data from 28 November 1998 to 29 November 1998 were used as a test set. In this case, the MAPE of different ANNs is shown in Table I below:

TABLE I.   MAPE OF THE FIRST SET OF SIMULATIONS

Type	MAPE (%)
BP	4.44
RBF	1.69
Elman	2.88
LSTM	1.55

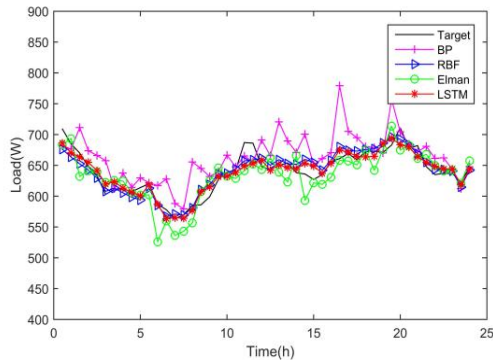The load prediction results of the four neural networks are shown in Fig. 4.

Fig. 4. Load forecasting results of four ANNs for the first simulation experiment

Second, the load data, temperature data and the weekday and holiday data from 1 January 1997 to 30 November 1998 were used as a training set; the load data, temperature data and the weekday and holiday data from 30 November 1998 to 31 December 1998 were used as a test set. In this case, the MAPE of different ANNs is shown in Table II.

TABLE II.   MAPE OF SECOND SET OF SIMULATIONS

Type	MAPE (%)
BP	6.42
RBF	3.69
Elman	5.30
LSTM	3.46

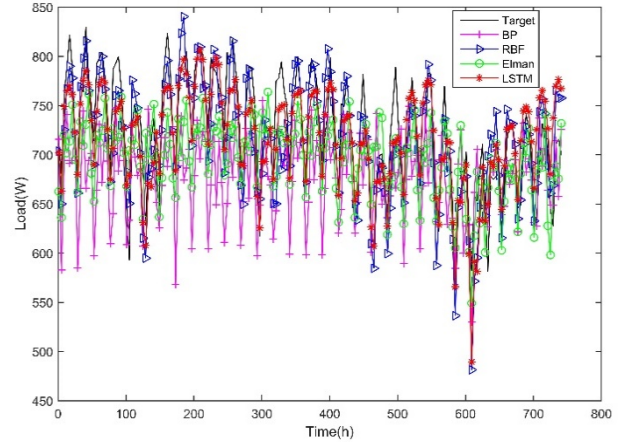The load prediction results of the four neural networks are shown in Fig. 5.

Fig. 5. Load forecasting results of four ANNs for the second simulation experiment

It can be seen from forecasting results that the average relative error of the BP network is the largest, and the prediction result of the LSTM network is the most accurate one. Because the activation function in the RBF neural network is a Gaussian function that makes the network have the best approximation property, so the prediction accuracy is higher than that of the BP network model. Although Elman Network is one of the earliest forms of RNN, having the capability to process sequence of inputs with internal state (memory), this study shows that RBF performs better than Elman Network. The reason is due to Elman Networks constitute drawbacks from BP ANN. RBF networks have a simple design, strong tolerance to input noise, and good generalization. LSTM can control the memory and forgetting the previous and current information. It has a long-term memory function, so it has a better capability in dealing with time-series data.

Since the first set of simulations only uses the data of the weekdays for load forecasting, and the second set of simulations uses all the data of the EUNITE competitive load, the prediction error of the second set of load prediction is generally higher. However, future work is needed to study if such networks could produce a general or specified result.

This paper predicts the load of one day and one month. Although the prediction result can meet the requirements, and if more accurate prediction results are needed, the prediction work can be refined in future work. For example, by dividing the working day and the rest day separately, it is also possible to predict hour by hour, that is, a neural network only predicts the load at one time point, and predicting the load of one day requires 24 neural networks, in the hope to improve the prediction accuracy greatly. The more accurate load-interval

forecasting always means a significant improvement in smart grid operation and control. We aim to further explore the load prediction intervals, the load pricing prediction, and long-term forecasting approach based on neural networks. Moreover, a reliable short-term load forecasting poses the challenges to the full utilization of renewable energies in the increasingly complex power market pricing strategies.

IV. CONCLUSION

In this paper, we compare four neural networks methods on load forecasting using European Intelligent Technology Network competitive load as data samples. Comparing the errors of the prediction results of each neural network, we can find that the average relative error of the LSTM neural network is the smallest, and the prediction result is the most accurate. In terms of training convergence speed, the training speed of the RBF network is the fastest, and the convergence speed of the BP network and the Elman network is almost is same, as they have very similar training function. The LSTM neural network converges faster than the BP network, and better the RBF network. The RBF network has the fastest training speed, which has no local minimum problem, and the prediction result is more accurate. After analysis, it can be seen that the time of neural network training is mainly related to the setting of the maximum training algebra and target error. Simulations show that it is possible to use the four discussed neural networks for load forecasting, and the prediction results depend on the application.

REFERENCES

[1] A. K. Singh, S. Khatoon, M. Muazzam et al., "Load forecasting techniques and methodologies: A review," 2nd International Conference on Power, Control and Embedded Systems, IEEE, 2012, pp. 1-10.

[2] A. Baliyan, K. Gaurav, S. K. Mishra, "A review of short term load forecasting using artificial neural network models," Procedia Computer Science, 2015, 48, pp. 121-125.

[3] D. C. Park, M. A. El-Sharkawi, R. J. Marks, L. E. Atlas and M. J. Damborg, "Electric load forecasting using an artificial neural network," IEEE Transactions on Power Systems, 1991, 6(2), pp. 442-449.

[4] D. Niu D, Y. Wang Y, D. D. Wu, "Power load forecasting using support vector machine and ant colony optimization," Expert Systems with Applications, 2010, 37(3), pp. 2531-2539.

[5] Y. Chen et al., "Short-term load forecasting: Similar day-based wavelet neural networks," IEEE Transactions on Power Systems, 2010, 25(1), pp. 322-330.

[6] M. Cai, M. Pipattanasomporn, S. Rahman, " Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques," Applied Energy, 2019, vol. 236, pp. 1078-1088.

[7] H. Shi, M. Xu and R. Li., "Deep learning for household load forecasting—A novel pooling deep RNN," IEEE Transactions on Smart Grid, 2018, 9(5), pp. 5271-5280.

[8] E. Ceperic, V. Ceperic and A. Baric, "A strategy for short-term load forecasting by support vector regression machines," IEEE Transactions on Power Systems, 2013, 28(4), pp. 4356-4364.

[9] A. Khosravi, S. Nahavandi, D. Creighton and D. Srinivasan, "Interval type-2 fuzzy logic systems for load forecasting: A comparative study," IEEE Transactions on Power Systems, 2012, 27(3), pp. 1274-1282.

[10] Z. Xu, R. Zhang and W. Jing, "When does online BP training converge?" IEEE Transactions on Neural Networks, Oct. 2009, 20(10), pp. 1529-1539.

[11] Ji-chao Li, Dan-ling Zhao, Bing-Feng Ge, Ke-Wei Yang, Ying-Wu Chen, "A link prediction method for heterogeneous networks based on BP neural network," Physica A: Statistical Mechanics and its Applications, 2018, 495, pp. 1-17.

[12] A. S. Bedi, P. Sarma and K. Rajawat, "Tracking moving agents via inexact online gradient descent algorithm," IEEE Journal of Selected Topics in Signal Processing, 2018, 12(1), pp. 202-217.

[13] A. H. Asgharnia, A. Jamali, R. Shahnazi, A. Maheri, "Load mitigation of a class of 5-MW wind turbine with RBF neural network based fractional-order PID controller," ISA Transactions, 2019.

[14] Q. Song, "On the weight convergence of Elman networks," IEEE Transactions on Neural Networks, 2010, 21(3), pp. 463-480.

[15] C. Lin and E. Boldbaatar, "Autolanding control using recurrent wavelet Elman neural network," IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2015, 45(9), pp. 1281-1291.

[16] Kun Xie, Hong Yi, Gangyi Hu, Leixin Li, and Zeyang Fan, "Short-term power load forecasting based on Elman neural network with particle swarm optimization," Neurocomputing, 2019. https://doi.org/10.1016/j.neucom.2019.02.063

[17] Mahdi Uz Zaman, Anisul Islam and Nahid Sultana, Short term load forecasting based on Internet of Things, thesis, BRAC University, 2018.

[18] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, 1997, 9(8), pp. 1735-1780.

[19] Ozal Yildirim, Ulas Baran Baloglu, Ru-San Tan, Edward J. Ciaccio and U. Rajendra Acharya, "A new approach for arrhythmia classification using deep coded features and LSTM networks," Computer Methods and Programs in Biomedicine, 2019, 176, pp. 121-133.

[20] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," IEEE Transactions on Smart Grid, 2019, 10(1), pp. 841-851.

[21] Baifu Huang, Danqi Wu, Chun Sing Lai, Xin Cun, Haoliang Yuan, Fangyuan Xu, Loi Lei Lai, Kim-Fung Tsang, "Load Forecasting based on Deep Long Short-term Memory with Consideration of Costing Correlated Factor," 2018 IEEE 16th International Conference on Industrial Informatics (INDIN).

[22] Arnaud de Myttenaere, Boris Golden, Bénédicte Le Grand, Fabrice Rossi, "Mean Absolute Percentage Error for regression models," Neurocomputing, 2016, 192, pp. 38-48.