

## Research Article

# Knowledge Mining Based on Environmental Simulation Applied to Wind Farm Power Forecasting

Dongxiao Niu,<sup>1</sup> Ling Ji,<sup>1</sup> Qingguo Ma,<sup>2</sup> and Wei Li<sup>3</sup>

<sup>1</sup> *Research Institute of Technology Economics Forecasting and Assessment, School of Economics and Management, North China Electric Power University, Beijing 102206, China*

<sup>2</sup> *School of Management, Zhejiang University, Hangzhou 310058, China*

<sup>3</sup> *MOE Key Laboratory of Regional Energy Systems Optimization, S&C Academy of Energy and Environmental Research, North China Electric Power University, Beijing 102206, China*

Correspondence should be addressed to Dongxiao Niu; niudongxiao617@126.com

Received 8 June 2013; Accepted 17 August 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Dongxiao Niu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Considering the inherent variability and uncertainty of wind power generation, in this study, a self-organizing map (SOM) combined with rough set theory clustering technique (RST) is proposed to extract the relative knowledge and to choose the most similar history situation and efficient data for wind power forecasting with numerical weather prediction (NWP). Through integrating the SOM and RST methods to cluster the historical data into several classes, the approach could find the similar days and excavate the hidden rules. According to the data reprocessing, the selected samples will improve the forecast accuracy echo state network (ESN) trained by the class of the forecasting day that is adopted to forecast the wind power output accordingly. The developed methods are applied to a case of power forecasting in a wind farm located in northwest of China with wind power data from April 1, 2008, to May 6, 2009. In order to verify its effectiveness, the performance of the proposed method is compared with the traditional backpropagation neural network (BP). The results demonstrated that knowledge mining led to a promising improvement in the performance for wind farm power forecasting.

## 1. Introduction

With the increasing resources constraints and environment pressure, wind power generation as an important application of renewable energy gains more and more concern. China has enormous potential in the wind energy utilization, with rich wind energy resource mainly in grass land or gobi of northwest, north, and northeast China, as well as coastal area and islands in east and southeast China. Due to the fast developed wind power technology and the distribution characteristics of wind resource, the trend of wind power generation development is concentrated with the large scale. According to statistics, by the end of 2012, the capacity of grid-connected wind generation has been 62660 MW and generated 100.8 billion kWh clean electricity, about 2% of the total electricity supply in China during 2012 [1].

Moreover, in wind resources system, various uncertainties exist in a number of system components as well as their

interrelationships, such as the random characteristics of natural processes (e.g., climate change) and weather conditions, the errors in estimated modeling parameters, and the complexities of system operation. Along with a large amount of wind power integrated into the power system, the intrinsic intermittency and uncertainties of wind resources will cause a great impact on the stability of the whole power system. Thus, precise short-term forecast of wind power farm will be necessary and bring lots of benefits, including cutting large spinning reserve, reducing the cost of wind power generation, and improving the safety and reliability of power system operation.

Previously, the physical method and the statistical approach are the main way for wind power forecasting, and physical conditions such as geography, topography, temperature, and pressure are employed to calculate the power generation capacity of wind farm [2, 3]. These methods require accurate relative meteorological data and difficult

modeling processes. In addition, they usually need to obtain the relationship between abundant historical data of wind power and other variances and are used for long-term wind power forecast, based on a large data base. Therefore, the emphasis of the methods is to build time series or dynamic model based on the experience gained from historical data.

Considering many uncertain factors and disturbances of wind power generating systems, the variation of wind power is largely affected by the complex random factors (e.g., wind speed); significant errors can be easily generated by time series model and regression algorithm, and expert system needs a mass of knowledge and experience with worse maintainability. As a popular simulation method, neural network is an alternative for handling random nonlinear complex mapping without indentifying the transform rules and has been applied in many fields (e.g., environmental modeling and system analysis) [4–6]. In this approach, the historical data of wind power and impact factors (e.g., uncertain parameters) are taken as the input variables. Moreover, the forecasting model is built after learning and training intelligently to avoid the errors or even minimize the impact of random factors on the simulation process. These advantages could lead the neural network to be a suitable method for the complex modeling problem.

In addition, knowledge mining can extract the connotative, unknown, and valuable knowledge or rules from the large-scale database, which is an area of the most extensive application value [7]. Knowledge mining has been widely applied, such as fuzzy time series [8], customer relationship management [9], and agricultural production prediction [10]. Climate factors, for example, wind speed, wind direction, temperature, and humidity, have a great impact on wind farm power changes [11]. Therefore, considering the characters of wind power forecasting, knowledge mining based on environmental simulation is proposed to build a data and knowledge base including history wind power and relative climate factors and to extract the similar historical situations though knowledge mining. Meanwhile, the new sample set would be generated through the historical situations to forecast the wind power with a similar characteristic in future. Moreover, classification is an important topic in data mining research. Given a set of data records, the classification problem is concerned with the discovery of classification rules that can allow records to be correctly classified.

Based on the above mentioned, in this paper, the objective is to propose an improved self-organization mapping neural network to classify the historical data and extract the useful rules for a precise wind power forecasting. A subset of the samples with the similar weather condition is used for training the neural network. The remaining of this paper is organized as follows. Section 2 describes the self-organization mapping based on rough set theory after a brief introduction of relative theory. Detailed structure of echo state network is proposed in Section 3. In Section 4, a case study is carried out to demonstrate the effectiveness of the proposed approach, and the performance is compared with the traditional neural network. The results analysis and discussion are also demonstrated. Finally, some remarks and conclusions of this study are presented in Section 5.

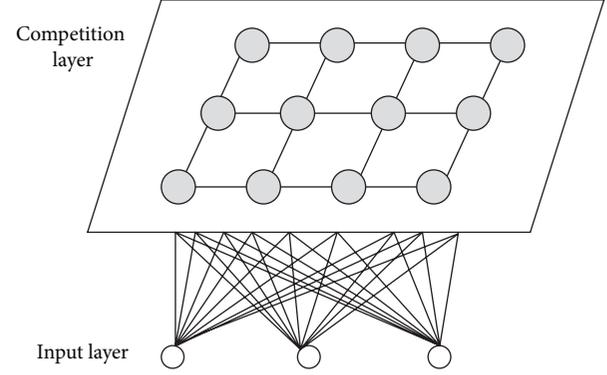


FIGURE 1: The schematic diagram of the SOM neural network.

## 2. Knowledge Mining Based on Environmental Simulation

*2.1. Self-Organization Mapping Theory.* Self-organization mapping (SOM), firstly proposed by Kohonen in 1981, is a neural network with unsupervised learning [12]. It has certain topological structure which is adjusted through the input information, and the pattern recognition is completed by the synergy among multiple neurons [13, 14]. The special idea of SOM is that there is no need to initialize the cluster center or guidance information and that the weight information of neurons is self-adaptively adjusted by input data.

Figure 1 illustrates the schematic diagram of the SOM neural network. Generally, there are two layers in SOM neural network, the input layer and the competition layer. The number of neurons in input layer is equal to the dimension of samples. Every neuron in the input layer connects the neurons in the competitive layer with variable weight values. The neurons in competitive layer will compete for the opportunity to respond to the input pattern, and the weight with the closest match to the presented input pattern is the winner neuron or the best matching unit (BMU). There also exist partial connections between the neurons in the competitive layers. The two-dimension form is the most common form of the neurons' arrangement in competition. The basic principle of SOM is described as follows.

(1) *Competition Process.* Let  $P$  be the input samples with  $m$  dimensions  $P = [p_1, p_2, \dots, p_m]^T$ , where  $m$  is the number of the input neurons. The number of the neurons in competition layer with two dimensions is  $N$  ( $N = n * n$ ). The connection weight between the input layer and the competition layer is denoted as  $W = [W_1^T, W_2^T, \dots, W_N^T]^T$ , where  $W_i^T = [w_{i1}, w_{i2}, \dots, w_{im}]$ .

Calculate the inner product  $n$  of input vector and the connection weight:

$$n = [n_1, n_2, \dots, n_N] = WP = [W_1^T P, W_2^T P, \dots, W_N^T P]^T. \quad (1)$$

Select the winner neuron in the competition process or the best matching unit (BMU). The basic rule is that the larger the inner product, the closer the neuron to the input vector,

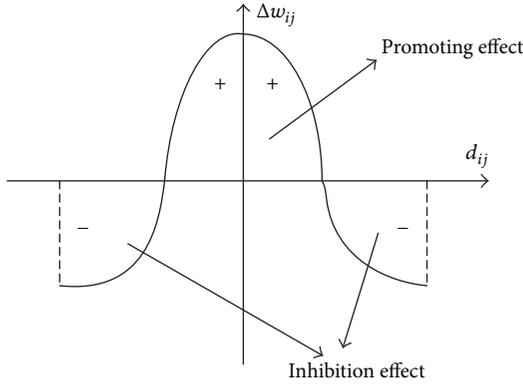


FIGURE 2: The effect of a neuron to its neighbors.

which indicates that the neuron matches to the presented input pattern. The winning formula is as follows:

$$a = \text{compet}(n) = \begin{cases} 1, & i = i^*, \\ 0, & i \neq i^*, \end{cases} \quad (2)$$

where  $n_i^* \geq n_i$ , for all  $i$ , and  $i^* \leq i$ , for all  $n_i = n_i^*$ . Finally, only one neuron wins; thus, the result is as similar as  $a = [0, \dots, 1, \dots, 0]$ , and  $i^*$  is the winner neuron.

(2) *Learning Process.* SOM neural network is arranged according to the two-dimensional structure; each neuron has a promoting effect to the neighboring neurons and on the contrast an inhibition effect to those far away, as shown in Figure 2.

In Figure 2  $d_{ij}$  denotes the distance between the neuron and its neighbor, and  $\Delta w_{ij}$  is the change of the connection weight. It is indicated that the neurons within the certain scope are promoted with weights increasing, while the neurons out of the scope are inhibited, and the weight would be reduced.

In SOM neural network, the weight is adjusted according to Kohonen learning rules. The main idea is to make the winner neuron and its neighbors closer to the input sample by modifying the weights. The formula is expressed as

$$w_i(t) = w_i(t-1) + \alpha(t)(P_i(t) - w_i(t-1)), \quad (3)$$

$$\alpha(t) = \alpha(0) * \left(1 - \frac{t}{T}\right), \quad (4)$$

$$N(t) = N(0) * \left(1 - \frac{t}{T}\right), \quad (5)$$

where  $w_i(t)$  and  $w_i(t-1)$  represent the weight vector of neuron  $i$  at time  $t$  and  $t-1$ , respectively.  $P(t)$  is the input sample at time  $t$ . And  $\alpha(t)$  means the learning rate at time  $t$ , ranging in  $[0, 1]$ . At the beginning, the learning rate is the largest, and the value of  $\alpha(t)$  is decreasing with the training. Since the weight shock of the neurons may occur during the training process, the learning rate need to be reduced gradually. Training neighborhood  $N(t)$  represents the neurons around the winner neuron whose weights will be adjusted. In the initial training, the scope of neighborhood is the largest,

which provides more neurons learning opportunity. With the increase of training, each neuron represents its own categories, and its neighborhood will be less.

Train the SOM neural network with all the input samples several times; after the procedure mentioned above, the neurons in competitive layer represent each cluster center, which achieves the clustering effect. As a result, the trained network can be utilized for pattern recognition. The algorithm itself can be summarized as follows.

*Step 1.* Initialize the SOM neural network. All the weight values  $w_{ij}$  are initialized to be random in  $[0, 1]$ .

*Step 2.* A training vector  $P_k$  is picked randomly from the training set.

*Step 3.* Calculate the Euclidean distance between input vector  $P_k$  and the weight vector  $W_i$ ;  $d_{ik} = \|P_k - W_i\|$ .  $\min\{d_{ik}\}$  is the winning node and becomes the BMU.

*Step 4.* According to (3), the values of weight vectors for the winning neuron are updated.

*Step 5.* Choose the new input vector and repeat Steps 3 and 4 until stop criterion is satisfied, for example, reaching sufficiently large iterations.

In fact, during the competitive procedure, there may be several neurons that are closely matched with the input vector, thus judging only one of them as the winner seems improper. Considering this issue, the concept of rough set theory is introduced to solve this problem, and self-organized neural network based on rough set is proposed in this paper.

2.2. *Rough Set Theory.* Rough set theory (RST) is a useful tool for data mining and decision support. In particular, it is popular in dealing with the incomplete information, vague concept, and uncertain data [15]. Besides, combined with other data mining algorithm, it can produce more hybrid data mining algorithm [16, 17].

Suppose  $U$  is the nonempty universe with finite members;  $R$  is the equivalence relation in  $U$ ; thus, the knowledge base can be expressed as a relation system  $K = (U, R)$ .

For the subset  $P \subseteq R$  and  $P \neq \phi$ , the intersection of all the equivalence relation among  $P$  can be called  $P$ -indiscernibility relation, defined by  $\text{IND}(P)$ :

$$[x]_{\text{IND}(P)} = \bigcap_{R \in P} [x]_R, \quad (6)$$

where  $[x]_R$  represents the equivalence class containing  $x \in U$  in relation  $R$ .

Lower and upper approximations are the important concept of RST. They help to measure the description of uncertain knowledge. Suppose  $X$  is the subset of  $U$ ; then, the lower approximation  $\underline{R}X$  and upper approximation  $\overline{R}X$  are

$$\begin{aligned} \underline{R}X &= \{x \in U : [x]_R \subseteq X\}, \\ \overline{R}X &= \{x \in U : [x]_R \cap X \neq \phi\}. \end{aligned} \quad (7)$$

Meanwhile, the boundary region  $BN_R(X) = \overline{RX} - \underline{RX}$ , and it consists of those objects that cannot be classified with certainty as members of  $X$  with the knowledge in  $R$ .

If  $BN_R(X) \neq \phi$ , it indicates that  $\overline{RX} \neq \underline{RX}$ , and  $X$  cannot be expressed by the equivalence class of  $R$  precisely. Thus, the set  $X$  is called “rough” (or “roughly definable”); otherwise,  $X$  is crisp.

**2.3. SOM Neural Network Combined with RST.** RST can solve the uncertain or imprecise knowledge expression; thus, it could be employed to deal with the imprecise problem in learning the process of SOM neural network. The novel network still has two-layer structure, while the different of the traditional SOM is in the competitive layer. In the competitive layer, each neuron contains upper approximation and lower approximation. In order to judge which neuron wins, we can determine that the input vector belongs to the lower approximation of a neuron exactly or to the upper approximation of several neurons imprecisely. Through this process, the imprecise problem in judging the winner will be solved properly.

Besides, we can set different learning rates for these two different matching results. If the input vector belongs to the lower approximation, it will get greater learning rate  $\alpha_{low}$ ; otherwise, it will get a lower learning rate  $\alpha_{up}$ . The idea is that when the input vector belongs to a pattern exactly, it can accelerate the learning; when it belongs to a pattern imprecisely, it will reduce its learning effect. The key issue of the novel SOM neural network is how to determine the input vector that belongs to a certain neuron or a set of neurons.

After selecting the best match neuron, it still needs to choose some suboptimum neurons using the suboptimum match degree, calculated as

$$n_{jk} = \frac{n_k}{n_j}, \quad 1 \leq k \leq N, \quad k \neq j, \quad (8)$$

where  $n_{jk}$  is the key factor to determine the lower or upper approximation. Then, it would define the set of suboptimal neurons:

$$V = \{w_k \mid n_{ij} \geq b \text{ and } 1 \leq k \leq N, \text{ and } k \neq j\}, \quad (9)$$

where  $w_k$  is the weight of the  $k$ th neuron;  $b$  is the threshold. Set  $V$  is the collection of the suboptimum neurons with match degree higher than  $b$ . If set  $V$  is empty, it indicates that there are no other close match neurons expect the best match one, and the input vector belongs to its lower approximation. Otherwise, the input vector belongs to the upper approximation of best match and suboptimal neurons. The different learning processes of SOM neural network are expressed, respectively.

- (1) The neuron belongs to the lower approximation exactly:

$$w_i(t) = w_i(t-1) + \alpha_{low}(t)(P_i(t) - w_i(t-1)),$$

$$\alpha_{low}(t) = \alpha_{low}(0) * \left(1 - \frac{t}{T}\right). \quad (10)$$

- (2) The neuron belongs to the upper approximation imprecisely:

$$w_i(t) = w_i(t-1) + \alpha_{up}(t)(P_i(t) - w_i(t-1)),$$

$$\alpha_{up}(t) = \alpha_{up}(0) * \left(1 - \frac{t}{T}\right), \quad (11)$$

where  $T$  is the total training times;  $\alpha_{low}(t)$  and  $\alpha_{up}(t)$  are the learning rates of lower approximation and upper approximation.

The detailed procedure of the proposed SOM neural network is presented as follows.

*Step 1.* Initialize the network, set  $t = 0$ , and  $P(0)$  denotes the input sample vector; initial weight  $w_{ij}(0)$  is a little random number;  $\alpha_{low}(0)$  and  $\alpha_{up}(0)$  are set at 0.9 and 0.5, respectively.

*Step 2.* According to (1), calculate the inner product of the input sample and the neurons in the output layers.

*Step 3.* Select the best match output neuron.

*Step 4.* Select the suboptimal match neurons as a collection  $V$ .

*Step 5.* Adjust the weights according to (10)-(11).

*Step 6.* Turn to Step 2 and repeat the process until all the samples have been tested or the learning rates have been reduced to 0.

### 3. Echo State Network

Recurrent neural networks (RNNs) are very powerful tools for solving complex temporal machine learning tasks [18]. In 2001, a new approach to RNN design and training was respectively proposed under the names of liquid state machines and echo state networks. Its reservoir computing (RC) is an RNN technique that offers a solution to the many problems associated with typical RNN architectures which have prevented their widespread use [19–21].

The classic echo state network contains three layers: input layer, hidden layer, and output layer (shown in Figure 2). The hidden layer is also called dynamic reservoir. Among the traditional circulate network, the scale of neurons is controlled within 12, while there is an abundance of neurons in the reservoir of ESN, about 20 to 500, with good short-term memory. Suppose there are  $L$  units in the input layer,  $M$  units in the output layer, and  $N$  units in the hidden layer. Generally,  $W^{in}$  represents the connection weight matrix of input layer;  $W$  means the connection weight within the reservoir, which keeps 1%~5% sparsely connected. In addition, the spectral radius is usually less than 1. These ensure the reservoir with dynamic memory and certain stability.  $W^{out}$  and  $W^{back}$  denote the connection weight matrix of output layer and feedback. It should be noticed that  $W^{in}$ ,  $W$ , and  $W^{out}$  are decided randomly before the network is established, and once determined they would have not changed.  $W^{out}$  is finally gained by training. Therefore, the main goal of network

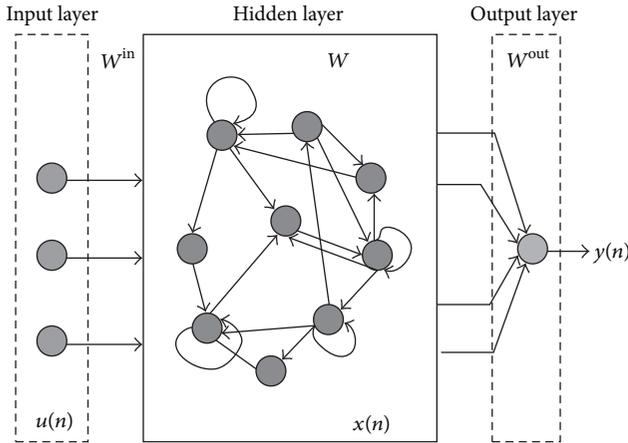


FIGURE 3: The schematic view of echo state network.

training is to determine the value of  $W^{\text{out}}$ . The schematic view of echo state network is presented in Figure 3.

The primary algorithm of echo state network is to inspire the reservoir by input information and to generate the state variables in the reservoir. Though linear regression between the state variables and desire output information, the connection weight of output layer can be determined. The state variables and the output are updated as follows:

$$\begin{aligned} x_{D_i}(n) &= f_{D_i} \left( W_{D_i}^{\text{in}} u(n) + W x_{D_i}(n-1) + W_{D_i}^{\text{back}} y_{D_i}(n-1) \right), \\ \widehat{y}_{D_i}(n) &= f_{D_i}^{\text{out}} \left( W_{D_i}^{\text{out}} \left( u_{D_i}(n), x_{D_i}(n) \right) \right), \\ W_{D_i}^{\text{out}} &= M_{D_i}^{-1} T_{D_i}, \end{aligned} \quad (12)$$

where  $T_{D_i} = [u_{D_i}(n) \ x_{D_i}(n)]$ ,  $M_{D_i} = [y_{D_i}(n)]$ .  $f_{D_i}$  and  $f_{D_i}^{\text{out}}$  are the activation functions of the reservoir and the output, and the most commonly used is a typical hyperbolic tangent function.

#### 4. Case Study and Results Analysis

A wind farm in northwest of China is considered as a case study to demonstrate the effectiveness of the proposed approach. In the study area, there are 66 wind turbines on the wind farm with total capacity 49.5 MW. The historical meteorological data and wind power data from April 1, 2008, to May 6, 2009, are taken as the database. The forecasting model is solved through Matlab on a single core of a 32-bit Lenovo workstation running on Windows7 with 2 dual-core 2.60 GHz CPU and 4.0 GB of RAM. We extract rules from the past information to forecast the wind power load. The main factors considered here are wind scale, temperature, and humidity. The feature vector  $V$  for knowledge mining is described as

$$\begin{aligned} V = & (HT_t, LT_t, W_t, HT_{t-1}, LT_{t-1}, W_{t-1}, \text{MAX } W_{t-1}, \\ & \text{MIN } W_{t-1}, \text{AVER } W_{t-1}), \end{aligned} \quad (13)$$

TABLE 1: The cluster results and effectiveness analysis.

Class number	Sample number	SSE
1	118	0.1845
2	48	0.2145
3	39	0.0954
4	30	0.1547
5	26	0.3458
6	75	0.5476
7	64	0.3547

where  $HT_t$  and  $LT_t$  are the highest and lowest temperatures on the  $t$ th day;  $HT_{t-1}$  and  $LT_{t-1}$  are the highest and lowest temperatures on the  $t-1$ th day;  $W_t$  and  $W_{t-1}$  are the wind scales on the  $t-1$ th day;  $\text{MAX } W_{t-1}$ ,  $\text{MIN } W_{t-1}$ , and  $\text{AVER } W_{t-1}$  are the maximum, minimum, and average wind speeds on the  $t-1$ th day.

In order to eliminate the dimension influence among different variables, data preprocessing is the first job need to be done.

The proposed RS-SOM neural network is employed to cluster the history information. To evaluate the compactness of the clustering results, the sum of squared error (SSE) is adopted in this study. The smaller the SSE, the better the effect. And it is calculated as

$$\begin{aligned} \text{SSE} &= \sum_k^K \sum_{x_i \in C_k}^n \|x_i - \mu_k\|^2, \\ \mu_{k,j} &= \frac{1}{N_k} \sum_{x_i \in C_k}^n x_{ij}, \end{aligned} \quad (14)$$

where  $C_k$  is the set of each cluster,  $\mu_k$  is the mean of the  $k$ th cluster, and  $N_k$  is the number of samples belonging to the  $k$ th cluster.

The cluster results are shown in Table 1. It can be seen that the 400 samples in the data base are divided into 7 classes. The SSE of class 3 is the smallest, while that of class 6 is the largest. This indicated that the samples in class 3 are the most closely similar and the shape of the curve can almost reflect the fluctuation of the wind output power on those days. Besides the average wind power output curve of each class is illustrated in Figure 4. Each curve has significant features and is obviously different from the others. For example, in class 1, the valley of most samples' power output is at about 10:00, while the peak time is at nearly 23:00. This is the most common situation with the largest number of class member 118. The class with the least samples is class 5, whose shape is extremely irregular.

Since the weather report may not be accurate, and the more the time lag, the worse the forecasting result, and this study only forecasted the wind power output on the next two days accordingly. The new input pattern can be discriminated using the trained RS-SOM neural network. The samples in the same class with the forecasting day are selected to train the ESN network for the forecasting day. In order to test the performance of ESN, BP neural network

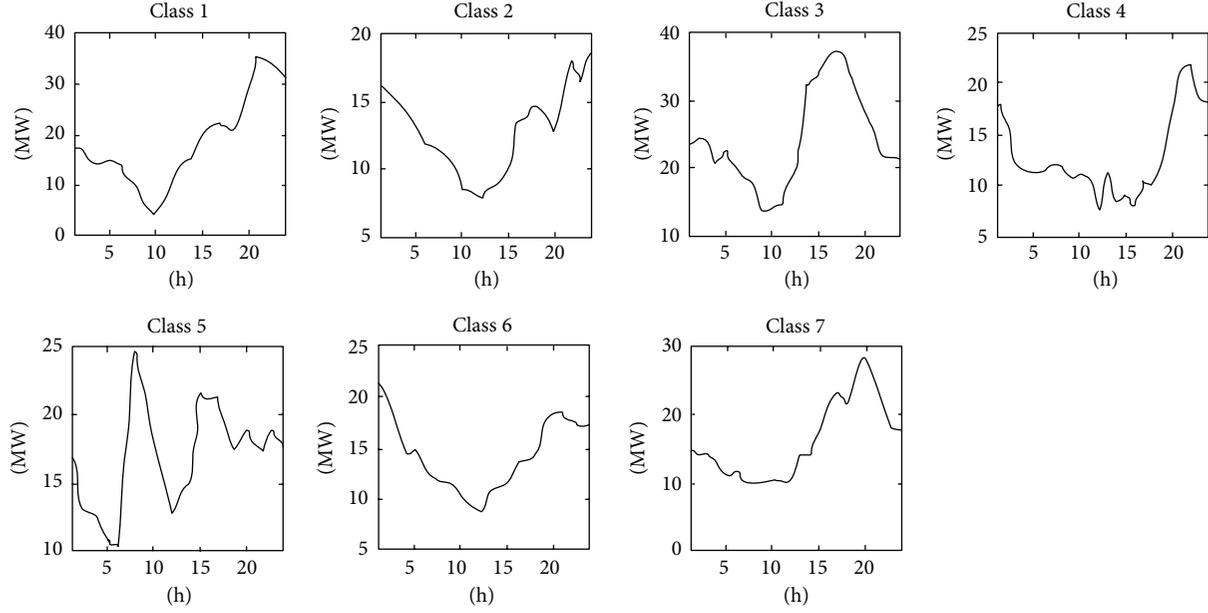


FIGURE 4: The wind farm output curves of different classes.

which has been widely used in load forecasting is also applied for the same task. Commonly used error evaluation indexes, including mean absolute percentage error (MAPE), mean absolute error (MAE), root mean square error (RMSE), and normal root mean square error (NRMSE), are used to discuss the performance of different forecasting methods:

$$\begin{aligned}
 \text{MAPE} &= \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right|, \\
 \text{MAE} &= \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|, \\
 \text{RMSE} &= \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}, \\
 \text{NRMSE} &= \sqrt{\frac{\sum (\hat{y}_i - y_i)^2}{\sum y_i^2}},
 \end{aligned} \tag{15}$$

where  $\hat{y}_i$  is the forecasting value,  $y_i$  is the actual value, and  $N$  is the number of samples.

Figure 5 shows the forecasting results of different methods and the actual wind power output. It indicates that the overall trend of forecasting power is in accordance with the actual situation. However, as the forecasting of peak and valley, the performance of ESN is obviously greater than BP model. The deviation of the latter is larger at the extremism values. During the period of 1:00 to 8:00, the performance of both methods is well, while, at night from 18:00 to 24:00, the deviation is larger. The error evaluation indexes for both methods are presented in Table 2, which shows that all of the indexes of ESN are lower than BP. The MAPE and MAE of ESN are 0.1366 and 1.7771 MW, respectively, lower than those

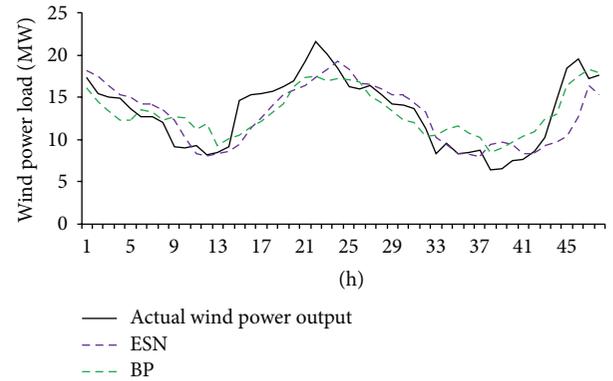


FIGURE 5: Forecasting results of different models and the actual wind power output.

TABLE 2: Error evaluation of different forecasting models.

Methods	MAPE	MAE (MW)	RMSE	NRMSE
ESN	0.1366	1.7771	2.2171	0.0019
BP	0.1709	1.9607	2.4320	0.0023

of BP 0.1709 and 1.9607 MW. The RMSE and NRMSE of ESN forecasting are 2.2171 and 0.0019, and those of BP are 2.4320 and 0.0023, respectively. As mentioned, the forecast accuracy for the study area obtained from ESN model is substantially more than that typically seen from the BP model. This extensive comparison reflects forecasts for all available 66 wind turbines. It is illustrated that the accuracy of the ESN forecasts was consistently better for the year and for each wind plant. It should be noted that accuracy would be even better if the data were adjusted for curtailments.

Compared with the results obtained from the two methods, it is indicated that the reason why accurate wind power forecasting of ESN is more than that of the BP model and the BP model has a higher degree of error than the ESN method is that (1) the wind farm is located in the northwest of China, the management system of meteorological measurements is imperfect, and meteorological data at the wind plant often are of poor quality and contribute to inaccuracy in traditional forecasting approaches; (2) wind direction readings from standard met towers might not even be applicable since surrounding terrain can affect this movement, and wind can ramp up or down quickly, that could lead to misleading information. All of the above problems with a very complex and important prediction process could directly affect the accuracy of the traditional forecasts. Data management also is vital to the accuracy of the forecasting method. Through integrating the SOM and RST methods to cluster the historical data in to several classes, the approach could find the similar days and excavate the hidden rules for increasing the forecast accuracy. In addition, it could reflect the uncertainties of the input parameters, avoid the data error, and provide valid data for ESN forecasting.

However, compared with other approaches, there is still much space for improvement of the proposed method. For example, there is no uniform way to determine the relative parameters (as the spectral radius of the connection weight within the reservoir) of the ESN network, which are mainly gained through massive experiments. Besides, other neural networks with universal approximation capability, for example, the radial basis function (RBF) network, will be studied and reformed further to improve the forecasting accuracy and calculation speed.

## 5. Conclusion

Wind power forecasting is an important tool for managing the inherent variability and uncertainty in wind power generation. Increasing the accuracy of forecasting can help to reduce the likelihood of an unexpected gap between scheduled and actual wind power generation, which can be extremely helpful for operators of power systems and wind power plants. In this study, we developed a database by using historical meteorological environment and power output data. Self-organizing map combining rough set theory as a knowledge mining technology is employed to discover and extract the rules. The classified samples are taken as the input of echo state network to train the structure of the network, respectively. Through integrating the SOM and RST methods to cluster the historical data in to several classes, the approach would provide valid data for ESN forecasting. The developed methods are applied to a case of power forecasting in a wind farm located in northwest of China with a wind power data from April 1, 2008, to May 6, 2009. The results demonstrated the successful use of the proposed method, which performs better than BP. The accuracy of prediction has been improved. However, the database is static in this study, which means the information of new samples will not be added into the knowledge pool automatically. And if

the database is small, it may not cover comprehensive situation. Thus, it will be proper for the wind farm with long operation period.

## Acknowledgments

This work was supported in part by the NSFC under Grant no. 71071052 and Grant no. 71201057, as well as “the Fundamental Research Funds for the Central Universities” under Grant no. 12QX23 and Grant no. 13X20. The authors would like to thank the anonymous reviewers and editors for their valuable comments, which greatly helped them to clarify and improve the contents of the paper.

## References

- [1] [http://www.gov.cn/gzdt/2013-04/09/content\\_2373121.htm](http://www.gov.cn/gzdt/2013-04/09/content_2373121.htm).
- [2] Y. Jiang, Z. Song, and A. Kusiak, “Very short-term wind speed forecasting with Bayesian structural break model,” *Renewable Energy*, vol. 50, pp. 637–647, 2013.
- [3] Y. H. Ma, N. B. Wang, F. C. Liu, G. T. Liu, and J. Wang, “A wind power forecast system for Jiuquan wind power base in Gansu province,” *Automation of Electric Power Systems*, vol. 33, no. 16, pp. 88–90, 2009.
- [4] F. Cassola and M. Burlando, “Wind speed and wind energy forecast through Kalman filter of Numerical Weather Prediction model output,” *Applied Energy*, vol. 99, pp. 154–166, 2012.
- [5] U. H. Ashraf, M. Paras, E. K. Mary, J. Meng, L. C. Chang, and T. Senjyu, “A new strategy for predicting short-term wind speed using soft computing models,” *Renewable and Sustainable Energy Reviews*, vol. 16, no. 7, pp. 4563–4573, 2012.
- [6] C. N. Yao, X. K. Gao, and Y. C. Yu, “Wind speed forecasting by wavelet neural networks: a comparative study,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 395815, 7 pages, 2013.
- [7] X. Y. Pan and L. C. Jiao, “A granular agent evolutionary algorithm for classification,” *Applied Soft Computing*, vol. 11, no. 3, pp. 3093–3105, 2011.
- [8] T. K. Liu, Y. P. Chen, and J. H. Chou, “Extracting fuzzy relations in fuzzy time series model based on approximation concepts,” *Expert Systems with Applications*, vol. 38, no. 9, pp. 11624–11629, 2011.
- [9] E. W. T. Ngai, L. Xiu, and D. C. K. Chau, “Application of data mining techniques in customer relationship management: a literature review and classification,” *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 2592–2602, 2009.
- [10] Y. L. Everingham, C. W. Smyth, and N. G. Inman-Bamber, “Ensemble data mining approaches to forecast regional sugarcane crop production,” *Agricultural and Forest Meteorology*, vol. 149, no. 3–4, pp. 689–696, 2009.
- [11] Z. H. Guo, Y. Dong, J. Z. Wang, and H. Y. Lu, “The forecasting procedure for long-term wind speed in the Zhangye area,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 684742, 17 pages, 2010.
- [12] Z. S. Wang, S. R. Bian, and Y. Liu, “The load characteristics classification and synthesis of substations in large area power grid,” *International Journal of Electrical Power & Energy System*, vol. 48, pp. 71–82, 2013.

- [13] L. Hernandez, C. Baladron, and J. M. Aguiar, "Classification and cluster of electricity demand patterns in industrial parks," *Energies*, vol. 5, no. 12, pp. 5215–5228, 2012.
- [14] V. Nourani, A. H. Baghanam, and J. Adamowski, "Using self-organizing maps and wavelet transforms for space-time pre-processing of satellite precipitation and runoff data in neural network based rainfall-runoff modeling," *Journal of Hydrology*, vol. 476, pp. 228–243, 2013.
- [15] S. I. Omurca, "An intelligent supplier evaluation, selection and development system," *Applied Soft Computing*, vol. 13, no. 1, pp. 690–697, 2013.
- [16] P. Georg, W. Richard, and N. Rene, "Dynamic rough clustering and its application," *Applied Soft Computing*, vol. 12, no. 10, pp. 3193–3207, 2012.
- [17] A. Ganivada, S. S. Ray, and S. K. Pal, "Fuzzy rough granular self-organizing map and fuzzy rough entropy," *Theoretical Computer Science*, vol. 466, pp. 37–63, 2012.
- [18] A. A. Ferreira, T. B. Ludermit, and R. B. A. Ronaldo, "An approach to reservoir computing design and training," *Expert Systems with Applications*, vol. 40, no. 10, pp. 4172–4182, 2013.
- [19] J. B. Butcher, D. Verstraeten, B. Schrauwen, C. R. Day, and P. W. Haycock, "Reservoir computing and extreme learning machines for non-linear time series data analysis," *Neural Networks*, vol. 38, pp. 76–89, 2013.
- [20] L. Ji, D. X. Niu, and H. Wu, "Daily peak load forecasting based on Bayesian framework and echo state network," *Power System Technology*, vol. 36, no. 11, pp. 82–86, 2012 (Chinese).
- [21] D. X. Niu, L. Ji, Y. L. Wang, and D. Liu, "Optimization of echo state network with wavelet analysis applied in short-term load forecasting," *Kybernets*, vol. 41, no. 10, pp. 1557–1570, 2012.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

