9th Asia-Pacific Conference on Wind Engineering
Auckland, New Zealand
3-7 December 2017

Very short-term wind speed forecasting by a new hybrid method

Chuanjin Yu[1], Yongle Li[1], Xianghuo Yue[1], Mingjin Zhang[1]

[1] Department of Bridge Engineering
Southwest Jiaotong University, Chengdu, Sichuan, 610031, PR China

Abstract

On the basis of data mining technology, a new hybrid method of very short-term wind speed forecasting named WPD-DBSCAN-ENN (Wavelet packet decomposition, Density-based spatial clustering of applications with noise and Elman neural network) is proposed. Firstly, a raw wind speed series is decomposed into several sub series. Next, every sub series is processed by DBSCAN to select representative ones for the ENNs, whose structure is determined by the Gradient Boosted Regression Trees (GBRT). A key parameter in the DBSCAN is to be chosen through a new proposed method. Finally, all the sub series forecasting are conducted by ENNs and summed up as the final predictions. Experimental results show that the performance of the proposed hybrid method outperforms other methods including WPD-ENN and the single ENN.

Introduction

Bridges and trains are susceptive to the wind action, especially under strong ones frequently. The lighter materials for trains and the higher running speed make the problem more significant. So wind alarm systems are really needed, aiming to control trains speed in high wind situations. Wind forecast model is a vital part of the system[1]. The accurate wind speed predictions can help bridge managers make better traffic management. However, resulting from that wind speed is a non-stationary signal, it is difficult to get satisfactory forecast results.

In recent years, wind speed forecasting has been studied extensively. There are four major methods for wind speed forecasting, including statistical methods, physical methods, intelligent methods and hybrid model[2]. In order to utilize the features of different forecasting methods for more accurate predictions, the hybrid models are becoming the most popular.

In general, wind speed series is firstly decomposed to sub series with more steady through the wavelet decomposition or the empirical mode decomposition etc. Then they are forecasted respectively by the autoregressive integrated moving average mode or the artificial neural networks etc.[3], which are added as the final prediction.

An additional powerful way to reinforce hybrid methods is data mining. And there are many applications of data mining achieve, especially about clustering[4],[5]. It can help improving wind forecasting precision, by gathering representative data into several groups and building corresponding prediction models. But there will be a big error if a new input data is sorted into a fake cluster, which means an inappropriate model is used [6]. Another obvious drawback is that the cluster number is vital to the forecasts but difficult to determine. Meanwhile, with cluster number increasing, the number of models for prediction is also growing as well as the calculation cost.

Therefore, a new hybrid method named WPD-DBSCAN-ENN is proposed. Firstly, a raw wind speed series is decomposed by the WPD. Then, for each wind sub series, it is processed by DBSCAN to select representative ones, and only an Elman neural network(ENN) is built to make prediction, whose structure is determined by the Gradient Boosted Regression Trees (GBRT)[7] .Finally, all the sub forecasting results are summed up as the final prediction.

Proposed method

WPD

The decomposition of wind speed histories is beneficial to the improvement of prediction accuracy. The WPD algorithm is taken as a special format of the wavelet decomposition, and its decomposition process by the WPD for 3 spaces is shown as Figure 1. In this study, the number of decomposition space is 3, and Daubechies 6 is used as wavelet function.

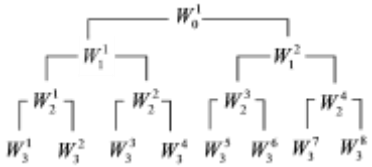


Figure 1. Node numbers of three spaces decomposition by WPD

GBRT

GBRT is an accurate and effective for regression problem, including linear or nonlinear ones, which is robust to outliers. The method GBRT can be summarized that a regression tree is built through studying from the residual errors of the temporary established model until the stopping condition is reached [7], [8]. So GBRT is used to determine the structure of neural network, through identifying the importance of input variables measured by one's contribution to the prediction accuracy.

Our study shows that it is not easy to set the threshold value of respective importance (*RI*) of the variable with different time lag for each sub wind series generated from the WPD. Since for different sub wind series，the biggest *RI* values of input variables are really different. To ensure that the process for choosing input variables for the ENN are automatic, the cumulative importance of the *j-th* input variable, abbreviated as *CI*, is defined as Eq. (1). Through this, all the input variables with high respective importance will be selected, at the same time the process determining the input layer structure of the Elman neural network are automatic. The threshold value of cumulative importance is set as 90% to avoid "dimensionality curse". So the number of *j* can be calculated as Eq. (2).

$$CI_j = \sum_{i=1}^{j} RI_j \qquad (1 \le j \le n) \tag{1}$$

$$j = \underset{j}{argmin}\,(CI_j \ge 90\%) \qquad (1 \le j \le n) \tag{2}$$

DBSCAN

DBSCAN is a density-based data clustering algorithm. It groups together points in dataset D with high density, and also identify outlier points. For wind speed forecast, herein DBSCAN is used to process the training dataset for ENN. The original training dataset is clustered into several groups by DBSCAN, including the outlier points and others. In fact, the outlier points in the original training dataset represent that do not frequently occur. So the outlier points will be eliminated. Meanwhile, the others are supposed as representative ones to strengthen the prediction ability of only an ENN for each decomposed series. More details about the algorithm can be found in the literature[9].

Two parameters are required for DBSCAN: 1. *Eps*： the maximum distance between two samples for them to be considered as in the same neighborhood; 2.*MinPts*： the number of samples in a neighborhood for a point to be considered as a core point. That finding proper values for these two parameters is based on the concept *k-Dist* proposed by the algorithm inventor. The definition of *k-Dist* for each point is shown as Eq. (3). Distances, a point to all points except itself measured by Euclidean distance, are calculated firstly and then sorted from small to large to get a distance array *dist(x)*. For each point, *k-Dist* is the *k-th* values in *dist(x)*. Meanwhile, *k* in the *k-Dist* equals to *MinPts*. Each *k-Dist* of points in the dataset is calculated. Then all the *k-Dist* is sorted from small to large and the corresponding change curve is plot. *Eps* is regarded as the *k-th* where the curve starts to change sharply.

$$k\text{-}Dist = \left\{ dist_{MinPts}(x) \middle| x \in D \right\} \tag{3}$$

To make parameters more proper in DBSCAN, the notable percentage and notable radius, are proposed. In Eq. (4), *f(k-Dist)* is the discrete probability density function. A *k-Dist* that makes the distribution probability not smaller than a threshold value, the *k-Dist* is regarded as notable radius ($Dist_{notable}$), and the threshold value is called as notable percentage ($P_{notable}$). Then the notable radius is taken as *Eps*.

$$Eps = \{ Dist_{notable} \left| \underset{k\text{-}Dist}{argmin} \sum f\left(k\text{-}Dist\right) \geq P_{notable} \right\} \tag{4}$$

It is obvious that the smaller the $P_{notable}$ is, the more outlier points in the dataset after clustering by DBSCAN. On the other hand, the larger the $P_{notable}$ is, the weaker the effect of the clustering to select representative training samples. It can be imagined that a proper $P_{notable}$ can effectively help to eliminate the outlier points and select representative training dataset. Three different $P_{notable}$, respectively equaling to 0.6, 0.7 and 0.8 will be investigated in the following study. Another parameter, *MinPts*, was suggested as 4[9]. If the value of *MinPts* is small, outlier points may not be found. So in the study, the value of *MinPts* will be properly enlarged to 10.

ENN

ENN is really suitable for wind speed forecast and reinforces the hybrid method greatly[10]. The topology structure of ENN can be divided into four layers namely input layer, hidden layer, connecting layer and output layer[11], as Figure. 2 shows.

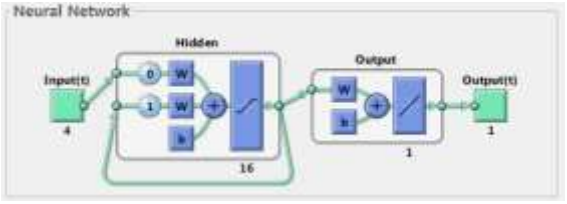


Figure. 2 A typical structure of Elman neural network used in the study in Matlab Neural Network Toolbox[12]

In the study, for each decomposed wind speed series, an ENN is built to make prediction. For each model, the number of neurons of inputting layer is determined by the GBRT as mentioned above. The number of neurons of hidden layer is determined by investigating the RMSE values for different number of hidden nodes. The number of neurons of output layer is one.

Framework of hybrid strategy

The proposed hybrid method WPD-DBSCAN-ENN is demonstrated in Figure. **3**. It can be included as follows:

Figure. 3 Framework of the proposed hybrid model WPD-DBSCAN-ENN

Step 1. Original wind speed series is decomposed by WPD into a range of appropriate and detailed sub-series.

Step 2. For each sub-series, GBRT is used to identify the importance of input variables. The cumulative importance is calculated and to be a basis for the structure of the input layer of Elman neural networks.

Step 3. For each sub-series, original training samples are built. Through DBSCAN, by defining the notable percentage and the notable radius, the original training samples is effectively processed. It means that the outliers are eliminated and a finite number of representative samples is selected for an ENN built in the next step.

Step 4. For each sub-series, an appropriate ENN is built and used for prediction. At last, the final forecast is obtained through summation of all predictions from each sub-series.

Evaluation Criteria

Four generally adopted error indexes are used to estimate the global and local errors of different forecast models, including the Mean absolute error (*MAE*), the Mean Absolute Percentage Error (*MAPE*), the Root Mean Square Error (*RMSE*) and the Maximum Absolute Error (*MaxAE*). The calculating formulas of these four error indexes are described as follows:

$$MAE = \frac{1}{N}\sum_{t=1}^{N}\left| X(t) - \hat{X}(t) \right| \tag{5}$$

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left| \frac{X(t) - \hat{X}(t)}{X(t)} \right| \tag{6}$$

$$RMSE = \sqrt{\frac{1}{N-1}\sum_{t=1}^{N}[X(t) - \hat{X}(t)]^2} \tag{7}$$

$$MaxAE = \max\left\{ \left| X(t) - \hat{X}(t) \right| \right\} \left(t \in [1, N]\right) \tag{8}$$

where, $X(t)$ is the raw wind speed data, $\hat{X}(t)$ is the forecasted wind speed data and N is the number of the wind speed samples of the $X(t)$ series.

Numerical examples

Due to the limited space, there is a wind speed time series $\{X_{1t}\}$ employed to validate the performance of the proposed method in the study (see Figure 4), which are all measured from Chinese Sichuan Province. It is 10min average over 16 days. The samples in the previous 15 days will be used to build forecasting models and there are a total of 143 samples in the last day utilized to verify the built models.

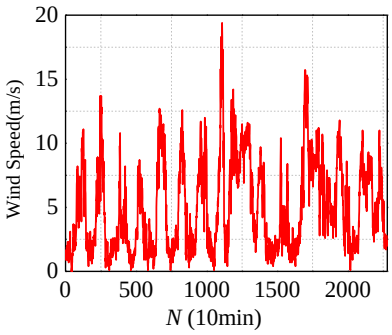
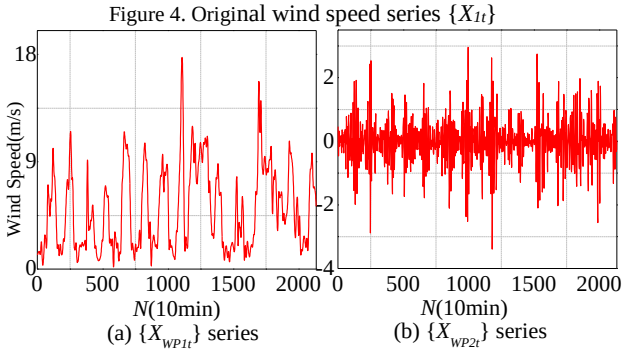


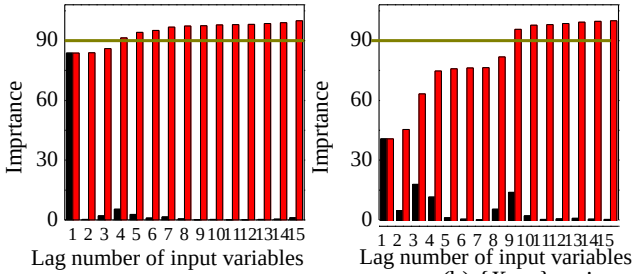
Figure 4. Original wind speed series $\{X_{1t}\}$



(a) $\{X_{WP1t}\}$ series (b) $\{X_{WP2t}\}$ series

Figure 5. Sub wind speed series $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$ generated by the WPD algorithm from $\{X1t\}$

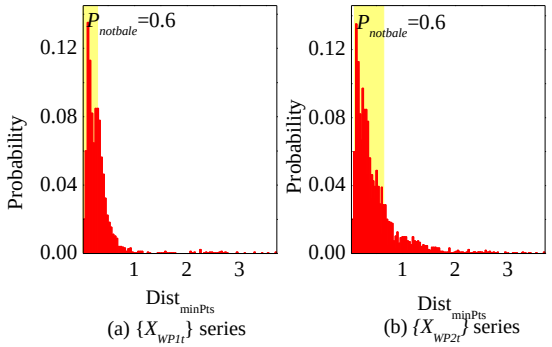
For $\{X_{1t}\}$, the training data is decomposed by WPD into 8 sub wind speed series, which are denoted from $\{X_{wp1t}\}$ to $\{X_{wp8t}\}$.Two of them $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$, is shown in Figure 5. The cumulative importance for all sub-series are calculated by GBRT. For instance, the details about *RI* and *CI* for each input variable in $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$ are illustrated in Figure 6. It can be seen that an input variable with small lag usually has high respective importance. Specifically, the input variable with only one lag is usually in the situation. Nevertheless, for different sub wind series, the biggest *RI* of input variables are really different. By the definition of the cumulative importance, all input variables with high *RI* will be selected.



(a) $\{X_{WP1t}\}$ series RI CI (b) $\{X_{WP2t}\}$ series

Figure 6. Cumulative importance of each input variables for sub wind series $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$ in $\{X_{1t}\}$ calculated by the GBRT

Then DBSCAN is used to select representative samples training samples for Elman neural networks. For $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$, the discrete probability density function of *k-Dist* is shown in Figure 7. Next, 8 ENNs are built for each decomposed series and make forecasts. At last, the final wind speed forecast is got by summing all the predictions up. What' more, the standard ENN and that with the WPD (WPD-ENN) are also used to make predictions. Herein, all the prediction processes are one step ahead, while the multi-step ones will be further studied in the other study. The prediction results are shown in Figure 8.



(a) $\{X_{WP1t}\}$ series (b) $\{X_{WP2t}\}$ series

Figure 7. Discrete probability density function of *k-Dist* of sub wind series $\{X_{wp1t}\}$ and $\{X_{wp2t}\}$ in $\{X_{1t}\}$

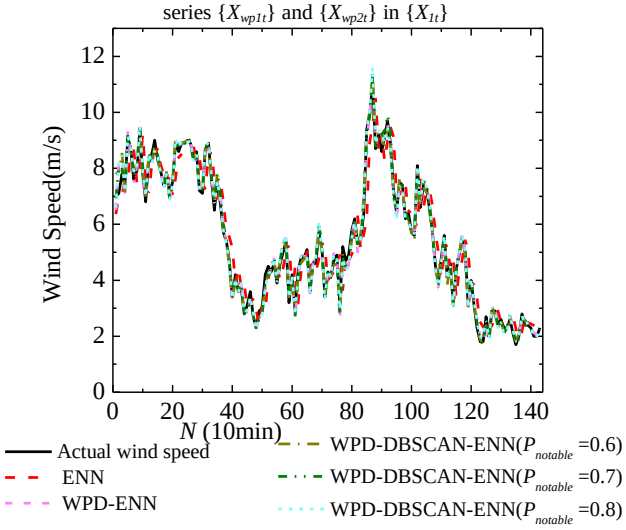


Figure 8. Original wind speed series $\{X_{1t}\}$ and Forecasting resulting by different models

Table 1 shows the estimated error results of all predictions from different models From Table 1, the performance of the hybrid model WPD-ENN has a substantial raise compared with that of the single Elman neural network model. The reason of the significant performance improvement is that the WPD decomposes the origin wind speed series into several sub series, which are more steady and easy to get accurate forecasts. When comparing the hybrid model WPD-ENN with WPD -DBSCAN-ENN, the performance of the latter is still markedly improved, regardless of the value of $P_{notable}$. It means the algorithm DBSCAN adopted in the study plays a useful role in wind speed forecasts, which can be used to select the representative training dataset for Elman neural networks. Different values of the parameter $P_{notable}$ in DBSCAN, can make different prediction performances. When $P_{notable}$ equals to 0.7, the suggested model can give the most accurate forecast under all the error criteria adopted in the study.

Error Indexes	Elman	WPD-Elman	WPD-DBSCAN- Elman		
			$P_{notable}$ =0.6	$P_{notable}$ =0.7	$P_{notable}$ =0.8
MAE	0.703	0.179	0.145	0.138	0.158
MAPE	15.0	3.7	2.9	2.7	3.2

RMSE	0.804	0.055	0.038	0.039	0.049
MaxAE	3.292	1.130	0.960	0.782	1.172

Table 1. Error indexes of forecasting results of different forecasting models in $\{X_{1t}\}$

Conclusions

A hybrid method of very short-term wind speed forecast is proposed based on WPD-DBSCAN-ENN, and an examples is conducted to verify the prediction accuracy of suggested methods. Limited to the space, more experiments can't be provided. But some conclusions can be still drawn on the basis of the above analysis：

1. through cumulative importance calculated by GBRT, it is convenient and automatic to select the number of neurons in the input layer of an Elman neural network.

2.the DBSCAN algorithm effectively makes positive impact on pre-processing the sub series decomposed from WPD to enhance wind speed forecast. The vital parameters, *Eps* can be easily determined by the proposed method in the study, including notable percentage and notable radius. What's more, it is believed that the proposed DBSCAN algorithm can be combined with other methods for making data pre-processing to improve accuracy of wind speed forecast.

3. the proposed hybrid model WPD-DBSCAN-ENN is characterized by convenience and automatic for wind speed prediction with high precision. It can be applied to forecast wind speed in wind alarm systems to ensure driving safety.

Acknowledgments

The authors are grateful for the financial supports from the National Natural Science Foundation of China (U1334201, 51525804), the Applied Basic Research Projects of the Ministry of Transport (2014319J13100) and the Sichuan Province Youth Science and Technology Innovation Team(2015TD0004).

References

[1] U. Hoppmann, S. Koenig, T. Tielkes, and G. Matschke, "A short-term strong wind prediction model for railway application: Design and verification," *J. Wind Eng. Ind. Aerodyn.*, vol. 90, no. 10, pp. 1127–1134, 2002.

[2] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated power," *Renew. Sustain. Energy Rev.*, vol. 13, no. 4, pp. 915–920, 2009.

[3] E. Cadenas and W. Rivera, "Wind speed forecasting in three different regions of Mexico, using a hybrid ARIMA-ANN model," *Renew. Energy*, vol. 35, no. 12, pp. 2732–2738, 2010.

[4] I. Colak, S. Sagiroglu, and M. Yesilbudak, "Data mining and wind power prediction: A literature review," *Renew. Energy*, vol. 46, pp. 241–247, 2012.

[5] J. Lorenzo, J. Méndez, M. Castrillón, and D. Hernández, "Short-Term Wind Power Forecast Based on Cluster Analysis and Artificial Neural Networks," in *International Work-Conference on Artificial Neural Networks*, 2011, pp. 191–198.

[6] S. Lhermitte, J. Verbesselt, W. W. Verstraeten, and P. Coppin, "A comparison of time series similarity measures for classification and change detection of ecosystem dynamics," *Remote Sens. Environ.*, vol. 115, no. 12, pp. 3129–3152, 2011.

[7] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001.

[8] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Front. Neurorobotics 7*, 2013.

[9] E. M, K. H. P, and S. J, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Kdd*, vol. 96, no. 34, pp. 226–231, 1996.

[10] R. Chandra, "Competition and collaboration in cooperative coevolution of elman recurrent neural networks for time-series prediction," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 26, no. 12, pp. 3123–3136, 2015.

[11] T. Koskela, M. Lehtokangas, J. Saarinen, and K. Kaski, "Time Series Prediction with Multilayer Perceptron, FIR and Elman Neural Networks," in *Proceedings of the World Congress on Neural Networks*, 1996, pp. 491–496.

[12] M. H. Beale, M. T. Hagan, and H. B. Demuth, "Neural Network Toolbox™ Reference," 2014.

.