2018 International Conference on Chemical, Physical and Biological (ICCPB2018)

Concentration Prediction of PM2.5 Based on OR-ELM Model

Dongpo Cheng
Shanghai University
(School of Economics)
Shanghai,China
dpc_calm@163.com

***Abstract*—Since the $PM_{2.5}$time series data often arrives in the form of data flow, and the potential distribution and the trend of the data change continuously with time, in this case, the online learning model with incremental update capability is more suitable for processing these non-stationary time series prediction problems. In this paper, the Online Recurrent Extreme Learning Machine (OR-ELM) is applied to the $PM_{2.5}$concentration prediction field for the first time. Compared with the existing PM2.5 concentration prediction model, the model improves the accuracy of PM2.5 concentration prediction by online learning. In order to verify the effect and superiority of the proposed model, this paper selects the hourly data of PM2.5 concentration in Beijing, Shanghai, Xi'an and Chongqing, and then conducts empirical research, and finally compares with benchmark models such as random forest (RF) ,gradient lifting tree (GBDT) and multi-layer perceptron model (MLP) in predictive performance . This model can provide a reference system for air quality warning due to its excellent predictive performance.**

Keywords*—$PM_{2.5}$, *Time series predicting, Online learning*; *Online Recurrent Extreme Learning Machine (OR-ELM)

I. INTRODUCTION

Accurate prediction of PM2.5 concentration has been a challenge due to the complexity of the factors of PM2.5 concentration. In recent years, researchers have made significant contributions to the prediction of PM2.5 concentration. The prediction models proposed in the research literature are roughly divided into two categories: the first category is the offline model, and the second category is the online model.

For off-line model research, it is divided into two sub-categories, a single model and a hybrid model. The single predictive models mainly include linear regression models, time series, gray models, Bayesian and other traditional methods, as well as support vector machines, neural networks and other algorithms-led artificial intelligence methods. A large number of literatures (Hu et al., 2015; Peng et al., 2014; Wang et al., 2017; Elbayoumi et al., 2014, etc.) have used linear models such as ARIMA and MLR to predict the concentration of air pollutants ($PM_{2.5}$ and PM_{10}). When the air pollutant concentration sequence is linear, ARIMA and MLR prediction results are more reliable and interpretable. But its limitation is that it relies too much on this linear mapping capability. In fact, the time series of contaminants are mostly non-linear, non-stationary and irregular sequences. In order to overcome the shortcomings of linear models, artificial intelligence algorithms such as ANN are widely used to predict particle concentration (Dai et al., 2017). However, artificial intelligence models also have their limitations. For example, neural network models are prone to fall into local optimum and overtraining, while support vector machines are sensitive to parameter selection. Since the linear model cannot accurately predict the extreme values of the data, and the nonlinear model can fit the nonlinear data well, some researchers try to predict the fine particle concentration by combining the linear model and the nonlinear model in order to obtain stronger prediction performance (Díaz- Robles et al., 2008). In order to improve the predictive performance of the model, more and more researchers have tried to use hybrid models in recent years to improve prediction performance (Lin et al., 2011; Perez, 2012; Antanasijević et al., 2013). As the idea of "decomposition and aggregation" develops, this idea is gradually applied to time series prediction (Yu et al., 2015). This method can make up for the shortcomings of deterministic models and statistical models. Researchers have proved that the "decomposition-collection" method is effective for time series prediction, and the accuracy of PM2.5 concentration prediction is greatly improved (Zhou et al., 2014; Yu et al. , 2016; Wang et al., 2017).

The above studies all use off-line processing. However, PM2.5 prediction is the same as real-life applications such as stock price forecasting, weather forecasting, and traffic flow forecasting. Time series data often arrives in the form of data streams, and the potential distribution of data and the trend of change is constantly changing over time. In this case, the online learning model with incremental update capability is more suitable for dealing with these non-stationary time series prediction problems. The Online Sequential Extreme Learning Machine (OS-ELM) is an emerging online learning algorithm proposed by Huang (2005). A Bueno (2017) uses OS-ELM to predict PM in the atmosphere and has better predictive performance than the ELM algorithm. However, OS-ELM has two fatal flaws. One is that the input weight cannot be adjusted, and the other is that the cyclic neural network cannot be trained. Jin et al. (2017) overcome the shortcomings of OS-ELM by adding the auto-encoding layer (ELM-AE) and regularization layer (LN) of the over-limit learning machine, and proposed an Online Recurrent Extreme Learning Machine (OR-ELM). This paper will use OR-ELM to predict PM2.5 particle concentration. This paper applies OR-ELM to the field of particle concentration prediction for the first time.

The rest of this paper is organized as follows: Section II gives a brief review of related theory. We will introduce the data and the evaluation method of the model in Section III. Performance evaluation of OR-ELM on its accuracy in timeseries prediction is given in Section IV. Conclusions based on the study are highlighted in Section V.

  DOI: 10.25236/iccpb.2018.028

II. MODELS

A. Random Forest

The random forest was proposed and extended by Breiman (2001), which is a classifier that uses multiple trees to train and predict samples. It combines two machine learning techniques including the "Bagging" idea and the random feature selection, in order to maintain the diversity of the sample tree sample set. The "Bagging" idea refers to the use of Boostrap resampling technique to extract multiple sample sets from the original sample, and to perform decision tree modeling for each sample set. Each prediction tree will give a prediction result. The process of random feature selection means that in each generation of the decision tree model, for each node, node splitting is performed by comparing the optimal segmentation from the randomly selected features. Existing literature prove the mathematical theory of random forest algorithm and that the random forest algorithm does not easily appear over-fitting phenomenon, and its generalization error is also smaller than the decision tree. It has high prediction accuracy and tolerance to outliers and noise data.

The Random Forest for Regression model is composed of regression tree growth related to the random vector θ. The dependent variable of the model is a numerical variable, and the training set is independently extracted from the distribution of the random vector Y and X. The mean squared generalization error of any numerical prediction $h(X)$ is $E_{X,Y}\big(Y - h(X)\big)^2$. The prediction result of the model is the mean of k regression trees$\{h(X, \theta_i, i = 1,2, \dots, k)\}$.

The main steps of the algorithm are as follows:

We note the original training set as $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, and then generate the random vector sequence $\theta_i(i = 1, \dots, k)$. Finally we apply sampling method of Booststrap, obtaining k subsamples noted as $T_i(i = 1,2, \dots, k)$;

A regression model $\{h(X, \theta_i), i = 1,2, \dots, k\}$ is established for each subsample set,where matrix X is a feature matrix and we assume that the parameter set $\{\theta_k\}$ is independently and identically distributed;

After the k-round training, the regression tree model sequence $\{h_1(X), h_2(X), \dots, h_k(X)\}$ is obtained. Its prediction result is the average summary of the k-round results given to any given new sample. The prediction results of the random forest model are:

$$f_r(x) = \frac{1}{k}\sum_{i=1}^{k} h_i(X)$$

B. Gradient tree

Jerome H. Friedman (1999) proposed a gradient-boosting decision tree that can be used for classification and regression. The essence of this algorithm is to generate predictive models by integrating multiple weak predictive models. GBDT is also an integrated learning method. The basic idea is: assuming the strong learner $f_{t-1}(x)$ obtained in the previous iteration, the loss function is $L(y, f_{t-1}(x))$.Then, the goal of our current round is to find a weak regression learner $h_t(x)$ of a decision tree model, in order to minimize the loss of this round $L\big(y, f_t(x)\big) = L(y, f_{t-1}(x) + h_t(x))$.

The main steps of the algorithm are as follows:

Let $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, K, L, $f(x)$ represent training set,maximum number of iterations , the loss function and the output strong learner.

Initialize the learner,

$$f_0(x) = \arg\min_c L(y_i, c)$$

For iterations$t = 1,2, \dots, K$,

a): Calculate a negative gradient for the samples$i = 1,2, \dots, m$,

$$r_{tj} = -\left[\frac{\partial L\big(y_i, f(x_i)\big)}{\partial f(x_i)}\right]_{f(x)=f_{t-1}(x)}$$

b): Based on $(x_i, r_{ti}), (i = 1,2, \dots, m)$, (i=1,2,...,m), we fit a CART regression tree to obtain the kth regression tree. It's corresponding leaf node area is $R_{tj}, j = 1,2, \dots, J$,where J is the number of leaf nodes of the regression tree k.

c): Calculate the best fit for the leaf area $j = 1,2, \dots, J$:

$$c_{tj} = \arg\min_c \sum_{x_i \in R_{tj}} L(y_i, f_{t-1}(x_i) + c)$$

d): Update learner:

$$f_t(x) = f_{t-1}(x) + \sum_{j=1}^{J} c_{tj} I(x \in R_{tj})$$

Obtain the final learner:

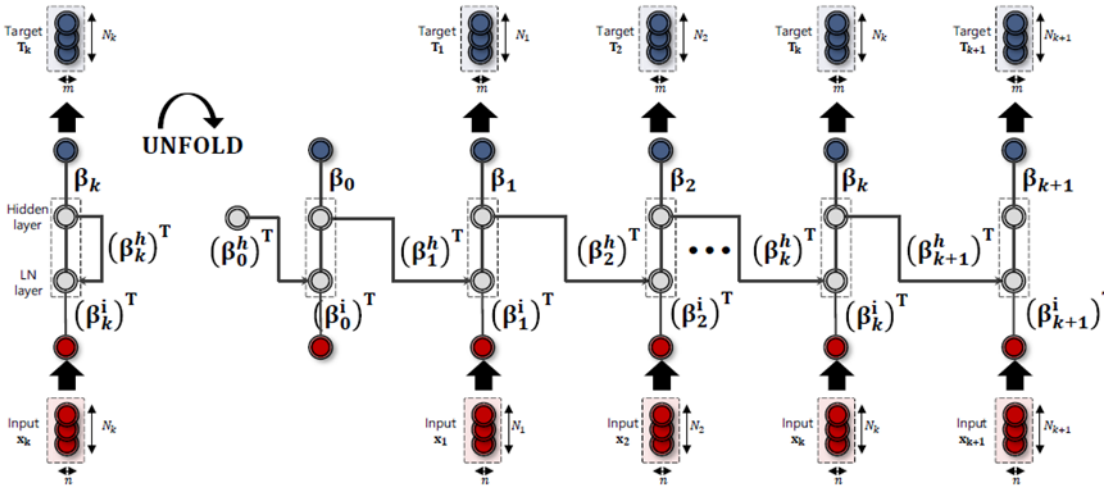
$$f(x) = f_K(x) = f_0(x) + \sum_{t=1}^{K} \sum_{j=1}^{J} c_{tj} I(x \in R_{tj})$$

C. Multilayer perceptron

Multilayer Perceptron Model (MLP) is the most widely used artificial neural network. It consists of input layer, hidden layer and output layer, that the signal is input from the input layer, passed to the output layer through the hidden layer, and output by the output layer. In the case of forward propagation, the input samples are passed in from the input layer, processed by the hidden layers, and passed to the output layer. If the actual output of the output layer does not match the expected output, it goes to the error back propagation phase. Error backpropagation is to forward the output error in some form through the hidden layer to the input layer layer by layer, and distribute the error to all the units, so as to obtain the error signal of each layer unit. The process of adjusting the weights of each layer to the propagation is repeated, and the weights are continuously corrected to make the actual output of the model close to the expected output.

D. Online Recurrent Extreme Learning Machine

OR-ELM is an improved algorithm of traditional OS-ELM, which uses auto-encoding technology and regularization technology to overcome the shortcomings of traditional OS-ELM models that cannot update input weights and train cyclic neural networks. The structure of the OR-ELM model is as follows:



The OR-ELM model consists of three network structures: a Recurrent Neural Network (RNN) for prediction, two auxiliary ELM-AEs that learn input weights and hidden layer weights. In RNN part ,the input layer, the hidden layer, and the output layer contain the number of neurons as n, L, and m, respectively. The input layer and the hidden layer are connected by the input weight $W \in W \in \mathbb{R}^{L\times n}$, and the hidden layer and the output layer connected by the output weight $\beta \in \mathbb{R}^{m\times L}$,and the hidden layer connect itself by weight $W \in \mathbb{R}^{L\times L}$.

The OR-ELM model consists of two phases: the initialization phase and the sequential learning phase.

Initialization:

We initialize the weight by $\beta_0 = 0, P_0 = \left(\frac{I}{C}\right)^{-1}$,where I, C, β_0 represents the identity matrix, regularization constant and initialization output weight respectively. The hidden layer output H_0 is randomly generated from the standard normal distribution. The weights $W^i, W^h$of the auxiliary ELM-AE are randomly generated by the standard normal distribution, and the output weightsβ_0^i, β_0^h and the corresponding auxiliary matrix P_0^i, P_0^h are generated by the equation.

Online sequential learning:

Update input weight: ELM-AE-I produces the (k+1)th input sample $x(k+1) \in \mathbb{R}^{n\times 1}$

The hidden layer output matrix is calculated as follows:

$H_{k+1}^i = g(\mathrm{norm}(W_{k+1}^i x(k+1))$

where,

$$\mathrm{norm}(x) = \frac{x - \mu^i}{\sqrt{\sigma^{i^2} + \epsilon}}$$

$$\mu^i = \frac{1}{L}\sum_{j=1}^{L} x_j$$

$$\sigma^i = \frac{1}{L}\sum_{j=1}^{L} (x_j - \mu^i)^2$$

Then calculate the output weight:

$$\beta_{k+1}^i = \beta_k^i + P_{k+1}^i {H_{k+1}^i}^T (x(k+1) - H_k^i \beta_k^i)$$

$$P_{k+1}^i = \frac{1}{\lambda} P_k^i - P_k^i {H_{k+1}^i}^T \left(\lambda^2 + \lambda H_{k+1}^i P_k^i {H_{k+1}^i}^T\right)^{-1} H_{k+1}^i P_k^i$$

Finally, Then the input weight of the OR-ELM model is

$$W_{k+1} = \beta_{k+1}^i$$

Update hidden layer weights:

ELM-AE-I propagates the (k)th hidden layer output $H_k \in \mathbb{R}^{L\times 1}$ to itself, so the output matrix is

$$H_{k+1}^h = g(\mathrm{norm}(W_{k+1}^h H_k))$$

Then calculate the output weightβ_{k+1}^h:

$$\beta_{k+1}^h = \beta_k^h + P_{k+1}^h {H_{k+1}^h}^T (H_k - H_k^h \beta_k^h)$$

$$P_{k+1}^h = \frac{1}{\lambda} P_k^h - P_k^h {H_{k+1}^h}^T \left(\lambda^2 + \lambda H_{k+1}^h P_k^h {H_{k+1}^h}^T\right)^{-1} H_{k+1}^h P_k^h$$

Finally the hidden layer weight of the OR-ELM model $V_{k+1} = \beta_{k+1}^h$.

Calculating the hidden layer output matrix:

$$H_{k+1} = g(\mathrm{norm}(W_{k+1} x(k+1) + V_{k+1} H_k))$$

Update output weight,

$$\beta_{k+1} = \beta_k + P_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1}\beta_k)$$

$$P_{k+1} = P_k - P_k H_{k+1}^T \left(I + H_{k+1} P_k H_{k+1}^T\right)^{-1} H_{k+1} P_k$$

Data

Data description

In this paper, four cities of Beijing, Shanghai, Chongqing and Xi'an are selected as research objects, which have different geographical locations and climatic conditions. The PM2.5 concentration data was captured from the China Environmental Monitoring Center (http://www.cnemc.cn/), which is the hourly data for August 1,2016, August 31, 2018. There was a small amount of missing data, and the data volumes of the four cities were 17777, 17801, 17809, and 17810, respectively. The average concentration of PM2.5 for the selected four cities from August 2016 to August 2018 is shown in Figure 1.

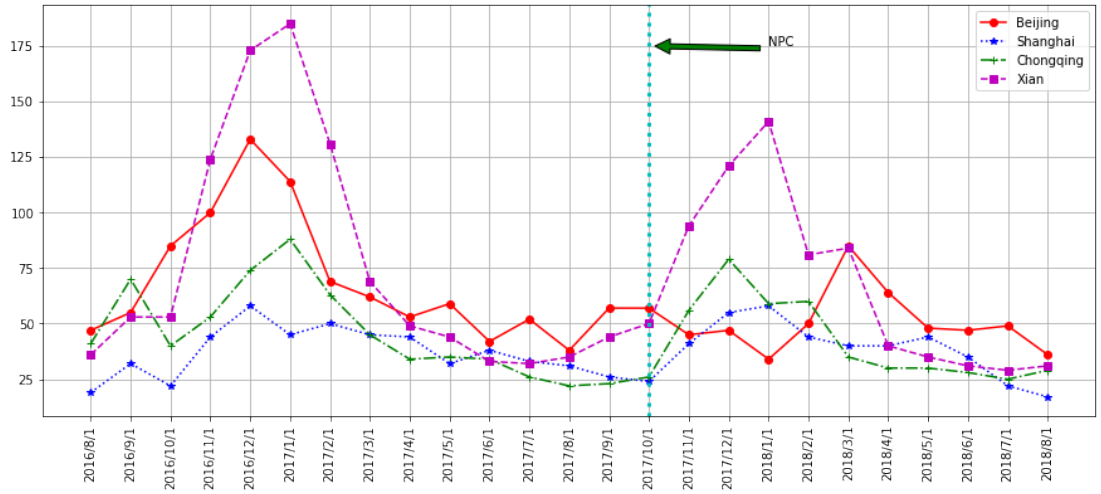


Fig. 1

From the figure, we can see that the PM2.5 concentration has similarities in different urban trends. The concentration of PM2.5 has a higher correlation with the season. When entering the winter heating period, the PM2.5 concentration accordingly increase. The concentration level of PM2.5 varies greatly in different cities, which is closely related to the geographical location and climatic conditions of the city. Overall, PM2.5 concentrations fluctuated significantly in northern cities compared to southern cities.

The traditional time series model uses the prediction of the current prediction time as the input value. We use the same method to sequence the machine learning model data. In this paper, the current time $PM_{2.5}$ concentration value is represented by x(t), and the data of the first hour before the current time (t) is used as the prediction feature of the RF, GBDT, and MLP models, and the predicted value is the $PM_{2.5}$ concentration value after p hours. That can be expressed as the following regression problem:

$$x(t+p) = f(x(t-1), x(t-2), \ldots, x(t-(d-1)))$$

The length of the interval is selected to have an impact on the prediction accuracy of the model, the training time, and the prediction needs outside the sample. The accuracy of the out-of-sample prediction is difficult to measure. Therefore, we comprehensively consider the prediction accuracy and training time, in the case of approximate accuracy, determine that the longer interval is the optimal time interval d.

E. Evaluation criteria

Model performance usually uses the following evaluation indicators MAPE, IA. The observation value is represented by y_t, $\hat{y}_t$represents the predicted value at time t, and $\bar{y}$represents the mean value of the observation result. MAPE are used to evaluate the relative error between$|y_i - \hat{y}_i|$ and $|y_i|$. The definition is as follows:

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{|y_i|} \times 100\%$$

IA is defined as:

$$\text{IA} = 1 - \frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N}(|\hat{y}_i - \bar{y}| + |y_i - \bar{y}|)^2}$$

IA is a dimensionless indicator that can be used to compare between different models. The value of IA is between 0 and 1. Specially for a perfect model IA is equal to 1 and MAPE is 0. MAPE is often used to quantify the extent to which the predicted value is close to the observed value. However, MAPE is sensitive to extreme values, and IA can summarize the similarity propensity between observed and predicted values.

III. Empirical Study

Based on the complexity of PM2.5 concentration factors, this paper chooses random forest algorithm (RF), gradient lifting algorithm (GBDT) and multi-layer perceptron (MLP), online loop overrun learning machine (OR-ELM). For RF, GBDT, and MLP, the data set is divided into training and test sets by 8/2. Different from the normal machine learning method to divide the data set, OR-ELM's prediction model has no such division because of its online learning characteristics. Such algorithms will predict each sample element in the sample stream. All experiments in this article were performed in a Python environment. In order to make the model have better prediction performance, this paper uses grid points to search the minimum leaf nodes of

random forests, the number of trees and the maximum number of random forest use features, and uses 50% cross-validation to train the training set. Establish a PM2.5 concentration prediction model. The parameter adjustment of the gradient lifting algorithm is similar to the random forest adjustment. For the parameters that need to be adjusted, the grid model search and cross-validation are used to find the optimal model parameters. In this paper, the perceptron model of the three-layer network is constructed, and the learning rate of the learning model and the grid point search are used to determine the optimal number of hidden layer neurons by the Adam optimization algorithm. By constructing a PM2.5 concentration prediction model, the predicted performance of each model in each city is as follows:

TABLE I: MAPE AND IA.

		RF	GBDT	MLP	OR-ELM
Beijing	MAPE(%)	7.91	8.06	9.86	10.47
	IA	0.9925	0.9923	0.9906	0.9869
Shanghai	MAPE(%)	7.67	7.76	8.77	10.05
	IA	0.9926	0.9923	0.9908	0.9889
Chongqing	MAPE(%)	5.65	6.02	6.86	6.54
	IA	0.9901	0.9889	0.9862	0.9916
Xian	MAPE(%)	7.25	7.90	8.97	7.88
	IA	0.9871	0.9844	0.9830	0.9955

Algorithm perspective: From the perspective of MAPE, the RF algorithm has better prediction performance than the GBDT, MLP, and OR-ELM algorithms in the prediction of PM2.5 concentration in four cities. From the perspective of IA, the model predictive performance is inconsistent with the results of MAPE. The performance of the OR-ELM model in Beijing and Shanghai is very small compared to other algorithms, and it is superior in Chongqing and Xi'an.The sensitivity of MAPE to extreme values, in general, the OR-ELM model performed best.

City perspective: From MAPE, the four machine learning algorithms in Chongqing PM2.5 concentration prediction performance are best taken into account the sensitivity of MAPE to extreme values, as seen from the IA of the four models, the eastern cities (Beijing, Shanghai The similarity between the predicted and true values of PM2.5 is higher than that of the central and western cities (Chongqing, Xi'an). The possible reasons are the climatic conditions and the location of the region. Compared with Chongqing and Xi'an in Beijing and Shanghai, factors of PM2.5 Concentration influencing are less complicated.

IV. CONCLUSION

This paper introduces the OR-ELM model for PM2.5 concentration prediction for the first time by introducing a new time series prediction algorithm. The model has higher PM2.5 concentration predictions in Beijing, Shanghai, Chongqing and Xi'an than previous forecasts. The OR-ELM model is an online loop learning algorithm. Compared with other algorithms, the offline learning algorithm has the characteristics of fast calculation speed, high prediction precision, and insensitivity to missing values and outliers. This model can provide a reference system for air quality warning due to its excellent predictive performance. The model has certain limitations, which is essentially a pure time series prediction method suitable for short-term prediction. In the future research, some covariate factors can be added to make more accurate off-sample prediction.

REFERENCES

[1] HU Yuxiao,DUAN Xianming,PM2.5 dispersion prediction based on Gaussian Plume model and Multiple Linear Regression model [J]. Journal of Arid Land Resources and Environment,2015,29(6)：86-92.

[2] PENG Si jun, SHEN Jia-chao,ZHU Xue,Forecast of PM2.5 Based on the ARIMA Model [J]. Safety and Environmental Engineering,2014, 21(6)：125-128.

[3] Wang Xu-zhi, Zeng, Pei,LIU Yong-hui, The Analysis and Forecast of PM2.5 in Shanghai [J]. Mathematics in Practice and Theory,2017,47(15)：210-217.

[4] Elbayoumi, M., Ramli, N.A., et al. Spatial and seasonal variation of particulate matter (PM 10 and PM 2.5) in Middle Eastern classrooms [J], Atmospheric Environment,2013,80(12): 389-397.

[5] DAI Lijie,Zhang Changjiang,MA Leiming,Dynamic forecasting model of short-term PM2.5 concentration based on learning machine[J]. Journal of Computer Applications,2017,37(11)：3057-3063.

[6] Díaz-Robles, L.A., Ortega, J.C., et al. A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: The case of Temuco, Chile [J], Atmospheric Environment,2003,42(35): 8331-8340.

[7] Lin, K.P., Pai, P.F., et al. Forecasting concentrations of air pollutants by logarithm support vector regression with immune algorithms [J], Applied Mathematics & Computation,2011, 217(12): 5138-5327.

[8] Perez, P., Combined model for PM10 forecasting in a large city. [J], Atmospheric Environment,2012,60(60): 271-276

[9] Antanasijević, D.Z., Pocajt, V.V., et al. PM(10) emission forecasting using artificial neural networks and genetic algorithm input variable optimization.. [J], Science of the Total Environment,2013, 443(3): 511-519.

[10] Zhou, Q., Jiang, H., et al. A hybrid model for PM 2.5 forecasting based on ensemble empirical mode decomposition and a general regression neural network [J], Science of the Total Environment,2014, 496(2): 264-274.

[11] Yu, L., Dai, W., et al. A novel decomposition ensemble model with extended extreme learning machine for crude oil price forecasting. [J], Engineering Applications of Artificial Intelligence,2016, 47: 110-121.

[12] Wang, D., Wei, S., et al. A novel hybrid model for air quality index forecasting based on two-phase decomposition technique and modified extreme learning machine. [J], Science of the Total Environment,2017, 580: 719-733.

[13] Wang, D., Wei, S., et al. Forecasting at Scale. [J], The American Statiscian ,2017, 72:1, 37-45.

[14] Breiman, L., Random Forests. [J], Machine Learing ,2001, 45(1): 5-32.

[15] Huang, G.B., Liang, N.Y., et al. On-Line Sequential Extreme Learning Machine [J], Computational Inteligence,2005,128(5): 232-237.

[16] Park, J.M., Kim, J.H., Online recurrent extreme learning machine and its application to time-series prediction [J], International Joint Conference on Neural Networks,2017: 1983-1990.