A Novel Prediction Algorithm for the Cross Temperature Estimation of Blast Furnace

Yan Jin¹, Sen Zhang¹ and YiXin Yin¹

¹School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China
E-mail: Jiny_ustb@163.com, zhangsen@ustb.edu.cn, yyx@ies.ustb.edu.cn

Corresponding author: Sen Zhang (zhangsen@ustb.edu.cn)

Abstract. In order to predict the distribution of gas flow, we need to get the temperature of each point in blast furnace throat in advance. In this paper, firstly, two intelligent modeling methods are used to establish a multiple-input multiple-output prediction model, one is extreme learning machine (ELM) algorithm and the other is online sequential extreme learning machine (OS-ELM) algorithm. And the model is a single-step prediction model of temperature in blast furnace, single-step prediction means the prediction of temperature in the next moment. We use autocorrelation analysis to determine input vector and output vector of the model. The result of autocorrelation analysis indicates that the method of temperature sequence prediction has a higher prediction accuracy and better prediction stability than the method of single point prediction. Next, based on real industrial data, we make a comparison between the multiple-input multiple-output model and least squares support vector machines (LS-SVM) model used in common. The experiment results show that OS-ELM model has a better forecast effect than the ELM model and LS-SVM model.

1. Introduction

Facing the increasingly serious environmental pollution problems, the blast furnace iron-making should bear the responsibility of energy-saving emission reduction and green production in the process of iron and steel production. The process of blast furnace iron-making is highly complex. Therefore, reasonably operating the blast furnace according to the change of energy consumption are the key to achieve low consumption and high yield in blast furnace iron-making [1]. Blast furnace gas flow is the main source of thermal energy and chemical energy in the process of blast furnace smelting. Therefore, study on distribution of gas flow in blast furnace can help diagnose the furnace condition and optimize the burden distribution system [2]. Beyond that, it will also help achieve the purpose of stabilizing blast furnace operation and reducing coke ratio and raising smelting level [3]. The cross temperature measurement device is widely used to monitor the top temperature in blast furnace. And coincidentally, the top temperature in blast furnace can reflect the distribution of gas flow [4]. Wu Min, et al proposed a new extension evaluation model based on the distribution characteristics of blast furnace gas flow to evaluate the distribution of gas flow [2]. This method provides qualitative analysis and quantitative indication for optimizing the burden distribution of blast furnaces. Using Bayesian techniques, particle swarm optimization algorithm, echo state network, expert knowledge methods, Zhang LiMin accurately predicted the change trend of blast furnace gas flow [3]. Zhou Ping, et al established the temperature estimation model of cross temperature
measuring center of blast furnace based on the two intelligent modeling methods of multi-output support vector regression machine (M-SVR) and random vector functional-link networks (RVFLNs) [4]. All above provides a reference for the establishment of ELM prediction model [5-6] and OS-ELM [7-8] prediction model in this paper. The ELM model and OS-ELM model are used to predict the temperature of seventeen points in blast furnace, through these temperature we can know the distribution pattern of gas flow and adjust burden distribution operation in advance.

2. Correlation Analysis

For the prediction of temperature in blast furnace, the traditional prediction method is to consider the 17 temperature points as an isolated point, and each temperature point is predicted separately, which will ignore the cross correlation between points. Therefore, it is necessary to analyze the autocorrelation between 17 temperature points on time series. Firstly, removing the noise from the original temperature data. Then taking out the data of any continuous 180 moments to calculate their correlation coefficient. The formula for calculating the correlation coefficient between the column vector X and Y can be expressed as:

\[
\rho_{xy} = \frac{\text{Cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E(X-\mu_x)(Y-\mu_y)}{\sigma_x \sigma_y}
\]

(1)

Where, \( \text{Cov} \) is covariance, \( E \) is mathematical expectation, \( \mu \) is mean value, \( \sigma \) is standard deviation.

\[
\text{Cov}(X,Y) = E(X-\mu_x)(Y-\mu_y)
\]

(2)

For \( X \), \( \mu_X = E(X) \), \( \sigma_X^2 = E(X^2) - E^2(X) \). Similarly, for \( Y \). Therefore,

\[
\rho_{xy} = \frac{E(XY)-E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}}
\]

(3)

Correlation coefficient between temperature sequence for 180 moments is shown in Fig.1.

**Figure 1.** Correlation coefficient between temperature sequence.

When considering all 17 temperature points as a whole, the correlation coefficient between predicted moment and the current moment is 1. The correlation coefficient between the predicted time and the first six moments is over 0.97. And then correlation coefficient decreases gradually, but it always stays over 0.86. This indicates that when the 17 temperature points are considered as a whole, the correlation of temperature sequence has been decreasing as time goes away from the predicted moment. But it always maintains a great correlation.
The correlation coefficient of a single temperature point at different times is calculated as follows:

$$\hat{R}(m) = \frac{1}{N} \sum_{n=1}^{N-m} x(n)x(n+m), m = 0,1,\ldots(N-1)$$  \hspace{1cm} (4)$$

$$\hat{\rho}(m) = \frac{\hat{R}(m)}{\hat{R}(0)}$$  \hspace{1cm} (5)$$

Where, $\hat{R}(m)$ is autocorrelation function, $\hat{\rho}(m)$ is autocorrelation coefficient. The result of correlation coefficient for one single temperature point in 180 moments is shown in Fig.2.

When the temperature point is taken into separate consideration, the autocorrelation coefficient is shown in Fig.2. Correlation coefficient between the predicted time and the current time is 1. The autocorrelation coefficient in previous moment is about 0.9 and the forward two times is 0.7, subsequently, the autocorrelation coefficient decreases to less than 0.5.

![Figure 2. Correlation coefficient of single temperature point.](image-url)

The comparison of the above two graphs can be seen that the correlation that considering the 17 temperature points as a whole is much greater than that of each temperature point considered alone.

3. Modeling of Temperature

In this chapter, the ELM algorithm and OS-ELM algorithm will be briefly introduced. And then the cross temperature prediction model of blast furnace based on OS-ELM algorithm will be introduced in detail.

3.1. ELM Theory

Extreme learning machine (ELM) is a single hidden layer feedforward neural network. The ELM algorithm is proposed by Huang [8-9] et al, which is a new simple and effective learning algorithm. The extreme learning machine algorithm does not need complex iteration in the training process of the model, and just need to set the number of neuron nodes in hidden layer. The weight matrix and the bias vector are randomly generated. After that, the output matrix of the hidden node is obtained. Through calculating the output matrix, we can gain the weight matrix in the hidden layer and the output layer, and fast calculate the global optimal solution.

For arbitrary sample set containing $N$ distinct samples $(x_i, t_i)$, where $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in R^n$, is n-dimensional feature vector of the $i$th sample, $t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \in R^m$, is m-dimensional target vector of the $i$th sample(output vector). The mathematical model of a single hidden layer neural network containing $L$ hidden nodes is expressed as:
\[
\sum_{i=1}^{k} \beta_i g_i(x_i) = \sum_{i=1}^{k} \beta_i g(\omega_i \cdot x_i + b_i) = t_k, \quad k = 1, 2, \ldots, N
\]  

(6)

Where, \( \omega_i = [\omega_{i1}, \omega_{i2}, \ldots, \omega_{iN}]^T \) is the weight vector connecting the \( i \)th hidden neuron and all the input neurons. \( \beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T \) is the weight vector connecting the \( i \)th hidden neuron and all the output neurons. \( b_i \) is the threshold of the \( i \)-th hidden neuron, \( \omega_i \cdot x_k \) is the inner product of \( \omega_i \) and \( x_k \), \( g(\omega_i \cdot x_k + b_i) \) is the output of the \( i \)-th hidden neuron.

The above \( N \) equations can be briefly expressed as:

\[
H \beta = T
\]  

(7)

Where,

\[
H = \begin{bmatrix}
g(\omega_{1} \cdot x_{1} + b_{1}) & \cdots & g(\omega_{L} \cdot x_{1} + b_{L}) \\
\vdots & \ddots & \vdots \\
g(\omega_{1} \cdot x_{N} + b_{1}) & \cdots & g(\omega_{L} \cdot x_{N} + b_{L})
\end{bmatrix}_{N \times L}
\]

(8)

\[
\beta = \begin{bmatrix}
\beta_{1}^T \\
\vdots \\
\beta_{L}^T
\end{bmatrix}_{L \times m}
\]

(9)

\[
T = \begin{bmatrix}
T_{1}^T \\
\vdots \\
T_{N}^T
\end{bmatrix}_{N \times m}
\]

(10)

\( H \) is known as the hidden layer output matrix of the neural network. The ELM algorithm is transformed into the following optimization problem:

\[
\min_{\beta} L_{ELM} = \frac{1}{2} \| \beta \|^2 + C \sum_{i=1}^{N} \| \epsilon_i \|^2
\]

subject to \( h(x_i) \beta = t_i - e_i \)

(11)

Where, \( \epsilon_i \) is the prediction error of the \( i \)-th sample, \( h(x_i) \) is the \( i \)-th row of the hidden output matrix \( H \). ELM algorithm uses the least square estimation theory, the output weight \( \beta \) is expressed as:

\[
\beta = H^T T = (H^T H)^{-1} H^T T
\]

(12)

Where, \( H^T \) is the generalized inverse of \( H \) matrix, \( H^+ = (H^T H)^{-1} H^T \).  

3.2. OS-ELM Theory

The concept of OS-ELM algorithm is based on ELM algorithm and OS-ELM is developed from a single hidden layer feedforward neural network with additional hidden nodes and RBF hidden nodes. The OS-ELM can train sequential data one-by-one or block-by-block [10-11]. For \( N_0 \) arbitrarily distinct samples \( (x_i, t_i) \), input vector is \( x_i = [x_{i1}, x_{i2}, \ldots, x_{in}] \in \mathbb{R}^n \), output vector is \( t_i = [t_{i1}, t_{i2}, \ldots, t_{in}] \in \mathbb{R}^m, i = 1, 2, \ldots, N \). If a single hidden layer feedforward neural network has \( L \) hidden nodes, then the \( N_0 \) samples can be estimated by \( 0 \) error. It indicates that it exist \( \beta_i, a_i \) and \( b_i \):
\[ f_L(x_j) = \sum_{i=1}^{L} \beta_i G(a_i, b_i, x_j) = t_j, \quad j = 1, \cdots, N \]  

(13)

\( a_i \) and \( b_i \) are the parameters of hidden nodes. \( \beta_i \) is output weight, \( G(a_i, b_i, x_j) \) is the output of the \( i \)-th hidden node with respect to input. When the hidden node is added:

\[ G(a_i, b_i, x_j) = g(a_i \cdot x_j + b_i), b_i \in \mathbb{R} \]  

(14)

Where, \( a_i \) is input weight vector, \( b_i \) is the bias of the \( i \)-th hidden node. When using RBF hidden node:

\[ G(a_i, b_i, x_j) = g(b_i \| x_j - a_i \|), b_i \in \mathbb{R}^+ \]  

(15)

\( a_i \) and \( b_i \) are the center and impact width of the \( i \)-th hidden node, \( \mathbb{R}^+ \) is set of all real numbers. There is no need for complex iteration in model training. But it is necessary to set hidden layer neuron nodes to facilitate the generation of weight matrix and offset vector. Using the idea of the basic extreme learning algorithm, we get the initial hidden layer output matrix \( H_0 \) and initial output weight that satisfy the equality \( \| H_0 \beta_0 - T \| = \min_{\beta_0} \| H_0 \beta_0 - T \| \).

\[
H_0 = \begin{bmatrix}
g(a_1 x_1 + b_1) & \cdots & g(a_L x_1 + b_L) \\
g(a_1 x_2 + b_1) & \cdots & g(a_L x_2 + b_L) \\
\vdots & \ddots & \vdots \\
g(a_1 x_N + b_1) & \cdots & g(a_L x_N + b_L)
g(a_1 x_1 + b_1) & \cdots & g(a_L x_1 + b_L)
\end{bmatrix}_{N \times L}  
\]  

(16)

Let \( \beta_0 = K_0^{-1} H_0^T T_0 \), \( K_0 = H_0^T H_0 \). When in the phase of online sequence update, data is in the learning process in the established model. When new data samples come into the model, the weight value will be updated according to the equation below:

\[
K_{k+1} = K_k + H_{k+1}^T H_{k+1} \\
\beta_{k+1}^{(k+1)} = \beta_k^{(k+1)} + K_{k+1}^{-1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta_k^{(k+1)}) 
\]  

(17)

Let \( P_{k+1} = K_{k+1}^{-1} \), it can be introduced by the Woodbury formula:

\[
P_{k+1} = P_k - P_k H_{k+1}^T (I + H_{k+1} P_k H_{k+1}^T)^{-1} H_{k+1} P_k \\
\beta_{k+1}^{(k+1)} = \beta_k^{(k+1)} + P_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta_k^{(k+1)}) 
\]  

(18)

Let \( k = k+1 \), go back to the online learning stage.

4. Experimental Results

In this paper, we uses the operational production data of a blast furnace in Wuhan Iron and Steel Company. Data sampling interval is one minute. Selecting 1500 samples that are removed outliers and denoising the noise, where, nine hundred samples is the training set for the model, the other three hundred samples is the testing data for the model. Because the value of different temperature measurement points is largely different, it will cause a obvious fluctuation in the forecast results and affect the accuracy of the prediction model. Therefore, the data is normalized in this paper. Then, the ELM and OS-ELM models are used to predict the single temperature and 17 temperature values, after that we obtain the experimental results.

Fig.3 shows the error comparison results of ELM model for single point prediction and temperature sequence prediction of TE3036; Fig.4 is the error comparison results of OS-ELM model for single point prediction and temperature sequence prediction of cross temperature measurement point TE3036.
According to Fig.5 and Fig.7, we can know that the prediction errors for temperature sequence prediction generally fall between [-0.05, 0.05], however, the prediction errors for single point prediction roughly fall between [-0.1, 0.1]. It can be found that the prediction error of OS-ELM algorithm is smaller than that of ELM algorithm. It shows that the OS-ELM prediction model has higher prediction accuracy.

**Figure 3.** Error of ELM model for single point prediction and temperature sequence prediction.

**Figure 4.** Error of OS-ELM model for single point prediction and temperature sequence prediction.

**Figure 5.** Comparison between the prediction value using the ELM prediction model and the real value.
Fig. 5 shows the comparison between the prediction value using the ELM prediction model and the real value of TE3036; Fig. 6 shows the comparison between the prediction value using the OS-ELM prediction model and the real value of TE3036. When the two different prediction models are used to predict the temperature, the method of temperature sequence prediction can better track the real value than that of single point prediction. And OS-ELM model has a higher prediction accuracy.

![Figure 6. Comparison between the prediction value using the OS-ELM prediction model and the real value.](image_url)

In order to evaluate the performance of the established model more comprehensively, in this paper, three evaluation formula are given below:

1) Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{k=1}^{n} (y(k) - \hat{y}(k))^2}$$  \hspace{1cm} (19)

2) Mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{n} \sum_{k=1}^{n} \left| \frac{y(k) - \hat{y}(k)}{y(k)} \right| \times 100$$ \hspace{1cm} (20)

3) Correlation coefficient between real target value series and prediction value series (CC):

$$CC = \frac{1}{\sigma(y)\sigma(\hat{y})} \sum_{k=1}^{n} (y(k) - \bar{y})(\hat{y}(k) - \bar{\hat{y}})$$ \hspace{1cm} (21)

Where, $\sigma(y)$ is the standard deviation of the actual target value sequence, $\sigma(\hat{y})$ is the standard deviation of the predicted value sequence. $\bar{y}$ is the mean value of the real values, $\bar{\hat{y}}$ is the mean value of the predicted values.

**Table 1.** A slightly more complex table with a narrow caption.

| model   | RMSE SPP | RMSE TSP | MAPE SPP | MAPE TSP | CC SPP | CC TSP |
|---------|----------|----------|----------|----------|--------|--------|
| ELM     | 0.0827   | 0.0593   | 4.5184   | 3.9350   | 0.7593 | 0.8024 |
| OS-ELM  | 0.0359   | 0.0297   | 3.2857   | 2.0503   | 0.8819 | 0.9254 |
In the above table, The SPP refers to Single Point Prediction. The TSP refers to Temperature Sequence Prediction. Tab.1 is the comparison of prediction results of ELM forecasting model and OS-ELM forecasting model for single point prediction and temperature sequence prediction. It can be found that the OS-ELM model can follow the tracks of unstable real values. And the root mean square error obtained by the method of temperature sequence prediction is minimum value, and the mean absolute percentage error is also minimum value, and correlation coefficient between real target value series and prediction value series is maximum value.

Fig.7 shows the prediction results of the three prediction models of ELM, OS-ELM and LS-SVM for TE3039. Fig.8 shows the three prediction results of the prediction models of ELM, OS-ELM and LS-SVM for TE3036. From Fig.7 and Fig.8, we can see that the ELM model, OS-ELM model and LS-SVM model can track the real value very well. However, the prediction effect of OS-ELM model is obviously better than that of ELM model and LS-SVM model, and the prediction effect of OS-ELM model is more stable.

5. Conclusions
In this paper, we analyse the correlation between temperature at different moments. The results show that thinking of all 17 temperature points as a whole has a greater relevance than thinking of each temperature point separately. Therefore, the traditional method of single point prediction is abandoned in the paper. We establish a multiple-input multiple-output prediction model to predict the temperature of 17 points in blast furnace. The ELM algorithm and OS-ELM algorithm are used in the model. Compared with the existing single point prediction model, the multiple-input multiple-output prediction model has higher prediction accuracy and better prediction stability.

Acknowledgments
This work is supported by National Natural Science Foundation (NNSF) of China under Grant No. 61673056 and 61673055.

References
[1] Lin Shi, WenLei Liu, JiangPeng Li, et al."Prediction model of gas utilization rate based on blast furnace throat gas distribution” Journal of Inner Mongolia University of Science and Technology, 2016, 35(2): 121-125.
[2] WU M, NIE Q, XU Y, et al."Method of gas flow distribution extension evaluation in a bell-less blast furnace and its application” Journal of Central South University (Science and Technology), 2010, 3: 033.
[3] LinMIN Zhang. "Research of Blast Furnace Flow Prediction Algorithm and its Application in Furnace Condition Analysis" YanShan University, 2015.

[4] Ping Zhou, Lei You, JiPing Liu, et al. "Centre Temperature Estimation of Blast Furnace Cross Temperature Measuring Based on M-SVR and RVFLNs" Journal of Northeastern University (Natural Science), 2017, 38(5): 614-619.

[5] G. B. Huang, Q. Y. Zhu, C. K. Siew. "Extreme learning machine: a new learning scheme of feedforward neural networks" IEEE International Joint Conference on, 2004, 2: 985-990.

[6] W. Zong, G. B. Huang, Y. Chen, “Weighted extreme learning machine for imbalance learning,” Neurocomputing, 2013, 101(3):229-242.

[7] H. Zhang, S. Zhang, Y. Yin, “Online Sequential ELM Algorithm with Forgetting Factor for Real Applications,” Neurocomputing, 2017, 261.

[8] Y. Lan, Y. C. Soh, G. B. Huang. "Ensemble of online sequential extreme learning machine" Neurocomputing, 2009, 72(13-15): 3391-3395.

[9] W. Xiao, J. Zhang, Y. Li, et al, “Class-specific cost regulation extreme learning machine for imbalanced classification,” Neurocomputing, 2017, 261.

[10] H. Zhang, S. Zhang, Y. Yin, “Online Sequential ELM Algorithm with Forgetting Factor for Real Applications,” Neurocomputing, 2017, 261.

[11] N. Y. Liang, G. B. Huang, P. Saratchandran, et al, “A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks,” IEEE Transactions on Neural Networks, 2006, 17(6):1411-1423.