Applied Soft Computing Journal 89 (2020) 106070

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Adaptive LSSVM based iterative prediction method for NOx concentration prediction in coal-fired power plant considering system delay

Yongjie Zhai [a], Xuda Ding [b,*], Xiuzhang Jin [a], Lihui Zhao [a]

[a] School of Control and Computer Engineering, North China Electric Power University, Baoding, 071003, Hebei, China
[b] Department of Automation, School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 4 July 2018
Received in revised form 30 October 2019
Accepted 4 January 2020
Available online 23 January 2020

Keywords:
Time series prediction
Soft sensor
Least square support vector machine
Cyber–physical system
Transfer entropy

ABSTRACT

Obtaining accurate and real-time value of the pollution concentration is fundamental to effective and energy-saving operation for pollution controlling in coal-fired power plants. However, accurate measurements for NO*x* concentration cannot be guaranteed, due to the intrinsic hardware and software design in sensors. In this paper, a prediction method, including variables processing and model regression, is proposed for NO*x* concentration measurement. Specifically, a set of variables is firstly selected adaptively as an input set by using modified transfer entropy (TE), while the relationships among them are guaranteed to be as weak as possible. Then, the input set can cover features without introducing redundant information to the prediction model , and the system delay is reduced based on the TE and sequential displacement. After the variables are processed, a forgetting factor online least square support vector machine (FFOLSSVM) is constructed to predict NO*x* concentration timely and accurately. The proposed method is the first work that takes the system delay into consideration for NO*x* prediction model, without mechanism analysis. The simulation indicates that the computational time and prediction accuracy requirements are sufficiently guaranteed by the proposed model.

1. Introduction

With the increasing concerns about environmental pollution and energy conservation, the clean and efficient utilization of coal in power plants has become an important issue. For example, governmental authorities have set a series of strict regulations on pollutant emissions of power plants [1]. Among the pollutant reduction procedures, efficient denitration is extremely challenging for the power station. Currently, Selective Catalytic Reduction (SCR) is a popular flue gas denitration technology , for its environment-friendly reaction process, simple device structure, and high denitration efficiency (up to 90%) [2]. Even though SCR technology is satisfying, there are still some noticeable drawbacks in the denitration procedure, mainly introduced by the control system. The current Continuous Emission Monitoring System (CEMS) cannot monitor NO*x* concentration accurately, timely, and continuously. A control system cannot operate properly without accurate and in-time feedback. CEMS also needs frequent maintenance for its vulnerability and instability, incurring high costs [3]. Therefore, how to measure NO*x* accurately, timely, and continuously is a critical issue that needs to be tackled.

Fortunately, with the development of machine learning and big data technology, new methods are proposed and widely used in many fields [4–7]. Among those methods, a new method of measurement – soft sensor – has been put forth. Soft sensor methods have shown the potential to satisfy the requirement for measurement in practical application [8]. As a consequence, how to find suitable input variables and determine a proper model is crucial for excellent prediction performance.

As for selecting input variables, there are mainly two methods: filter-type and wrapper-type. Although the former reduces the number of variables based on mutual information, it ignores to avoid introducing redundant features into the soft sensor and thus lowers the prediction accuracy [9,10]. Jie et al. [11] proposed a wrapper-type method using Monte-Carlo uninformative variable elimination to remove redundant information. The wrapper-type selection considers the maximum accuracy as the goal in variable selection, it is quite difficult to implement in practical applications and easily leads to over-fit for its high computational complexity [12].

After the input variables are selected, a regression model is needed for prediction. The model used in the soft sensor is varied, such as artificial neural networks (ANNs) and support vector

* Corresponding author.
E-mail addresses: zhaiyongjie@ncepu.edu.cn (Y. Zhai), dingxuda@sjtu.edu.cn (X. Ding), jinxzsys@163.com (X. Jin), lihuizhao0704@126.com (L. Zhao).

https://doi.org/10.1016/j.asoc.2020.106070
1568-4946/

machines (SVMs). However, the ANN model in the work [13] for prediction does not have enough accuracy for practical application. The SVM may suffer from high computational complexity and over-fitting [14] . These issues can be addressed by using the Least Squares SVM (LSSVM) proposed in [15]. The LSSVM technique has been widely applied in power plants. Lv et al. [16, 17] constructed LSSVM models to predict the NO*x* emissions of a coal-fired boiler; also, in a later study, they used LSSVM models to predict the bed temperature of circulating fluidized bed boilers [18]. Furthermore, Gu et al. [19] used LSSVM to develop a boiler combustion model for operation optimization.

Note in NO*x* prediction, the main challenge is how to select a good input variable set. The input variables must be chosen from the process, which is related to the production of NO*x*. NO*x* is derived from the combustion process of fossil fuels and generated in the coal-fired boiler. Thus, the input variables are the measurement points of these processes, which can indicate the working condition of the furnace. State-of-the-art presented several NO*x* prediction methods based on the input mentioned above variables and showed us good results [16,17,20,21].

However, there are mainly three drawbacks in these work. First, the fact is overlooked that the working conditions are different from time to time (e.g., different operation habits of operators). Thus, the input variable set should be adjusted according to the different conditions. From the aspect of the prediction model, Zhou et al. [20] presented an efficient NO*x* emissions model based on support vector regression (SVR) and compared its performance with traditional modeling techniques, i.e., back-propagation (BPNN) and generalized regression (GRNN) neural network. The model achieved an accurate and fast computation result and covered a large range of working conditions. Lv et al. [16] considered real operation data of power plants are inclined to be concentrated in some local areas because of the operators' habits and control system design, proposed a soft fuzzy c-means cluster algorithm based LSSVM. Second, these work ignored that the denitration system has large time-delay characteristics, and the variables from different measurement points with the same time-tag could not indicate the same sequential process. The delay[1] can lead to low prediction accuracy. And the prediction accuracy in their work does not satisfy the requirement of the practical application; only about 70% of the testing samples have a relative error below 5%. For instance, Tan et al. [21] proposed a methodology combining the advanced extreme learning machine (ELM) and harmony search (HS). The model has excellent performance in both accuracy and speed. However, the whole process in their works is off-line without updates. In practice, we believe the initial training samples cannot cover all the conditions, which could lead to low accuracy. Hence, the soft sensor should have the update procedure to fit the measurement requirement. To design an on-line model, Lv et al. [17] proposed an update strategy of dividing the process variations into irreversible and reversible ones to improve the performance of the model in long-term prediction. Third, the computation speed of the model could be extremely slow when the parameters of the update strategy are not optimized. As mentioned above, the variation of the working conditions could not guarantee the rationality of the previous parameters. The model needs to search for new parameters during the update procedure, which is time-consuming. Also, even if the parameters are optimized, the LSSVM model will introduce more and more data while the on-line prediction progress is running, which will increase computational complexity and even cause the system to halt. Even so, the method in [17] is the most practical one for now.

[1] The definition and explanation of the delay in this paper are shown in 3.1.3.

In this paper, to develop a practical application-oriented NO*x* prediction method for long-term use, we solve the problems in input variables selection and on-line prediction model. First, the input variables are selected according to the cause–effect relationship analyzed by the real data instead of mechanism-based one. Thus, the selection method selects the variables which can cover a large proportion of features of the current operating characteristics. Compared with mechanism-based selection, the proposed method is more reasonable for a dynamic model. Along with the selection, time delays between variables are detected, and the sequences of variables are aligned to eliminate the influence of delays. Second, based on LSSVM, an on-line model is constructed to decrease the frequency of retraining and parameter-optimizing by adapting the iteration approach. Finally, the prediction based on real data shows that the proposed method can fully satisfy the requirement of the practical application.

For practical application, the novelties of the work are as follows:

1. To our best knowledge, this is the first time to identify time delay between variables and introduce the time delay to dissect and amend the practical prediction method by TE. This critical consideration improves the generalization ability of the method and satisfies the requirement of computational time for the practical application.
2. We propose an efficient prediction method including data preprocessing (input variable selection and sequential displacement) and on-line prediction model for long-term NO*x* concentration prediction, satisfying the requirement of computational speed, prediction accuracy, and stability for different systems.
3. The cause–effect relationship depicted by TE is proposed as the criteria, and the HITS algorithm is migrated into the process for input variable selection. The data-driven selection method provides a new idea to get the best possible solution for performance improvement.
4. We design a new iterative approach based on LSSVM, which only searches weights in the initial stage. The proposed approach reduces the time complexity of weighs-update by introducing iteration algorithm and guarantees the prediction speed and accuracy simultaneously.

The remainder of the paper is organized as follows. Section 2 introduces preliminaries related to the proposed soft sensor method. In Section 3, we present the approach in detail. Then Section 4 provides a simulation based on real data. Section 5 presents the results, and Section 6 presents the discussion. Finally, Section 7 concludes the paper.

2. Preliminary

In this section, we introduce some preliminaries about input variable selection and the LSSVM model.

2.1. Input variable processing

TE is a non-parametric measure that estimates the directed information flow among stochastic processes to detect cause–effect between variables [22,23]. Wu et al. [24] analyzed the influence of five nonlinear transformations on TE and introduced the TE for nonlinear transformation. Hu et al. [25] introduced normalized TE and normalized direct TE to consider random delays and mutual independence among variables.

2.1.1. Transfer entropy

We suppose a system $\boldsymbol{X}$, and it contains a series of possible events whose probabilities of occurrence is $p(x)$, and then the Shannon entropy reads:

$$H = -\sum_{x} p(x)\log_2 p(x). \tag{1}$$

The Shannon entropy shows the average uncertainty of the system $\boldsymbol{X}$ and quantifies the average number of bits needed to encode the system $\boldsymbol{X}$. Shannon entropy provides an absolute limit on the best possible average length of lossless encoding or compression of an information source. Then, Shannon entropy for two systems can be expressed as:

$$H_1 = -\sum_{x,y} p(x,y)\log_2 p(x,y). \tag{2}$$

TE is introduced based on Shannon entropy. We can suppose two systems which generates events: $\boldsymbol{X} = \{x_1, x_2, \ldots, x_n\}$ and $\boldsymbol{Y} = \{y_1, y_2, \ldots, y_n\}$ with the same length of n. The definition of TE from $\boldsymbol{X}$ to $\boldsymbol{Y}$ in [24] is shown as

$$T_{\boldsymbol{X}\to\boldsymbol{Y}} = \sum_{y_n, y_{n-1}, x_{n-1}} p(y_n, y_{n-1}, x_{n-1})\log_2 \frac{p(y_n|y_{n-1}, x_{n-1})}{p(y_n|y_{n-1})}. \tag{3}$$

TE from $\boldsymbol{X}$ to $\boldsymbol{Y}$ indicates the entropy change of $\boldsymbol{Y}$, which caused by $\boldsymbol{X}$. So, we can measure the cause–effect relationship by TE.

2.1.2. Hyperlink-induced topic search

However, the TE values of the variables cannot be directly used to indicate the authority of each variable in the whole system, nor choose variables for input set, for some variables could be dramatically influenced by others, and sometimes could be the same value. Even though they have a significant impact on the system's output, one of them is redundant. Such variables are not suitable for being the input variables of the prediction model, which lead to the poor performance of the soft sensor.

Thus, according to the graph theory and the authority measurement for Internet vertex, we use HITS to evaluate the variables. HITS, which is also known as Hubs and Authorities, is a link analysis algorithm that rates Web pages; it was developed by Jon Kleinberg [26]. HITS algorithm is easy to compute and can be migrated to evaluate vertex in system [27].

Assuming $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ is a directed graph, and then, the link matrix $\boldsymbol{A}$ of the graph $\boldsymbol{G}$ is a $n \times n$ asymmetric matrix.

$$\boldsymbol{A}(u,v) = \begin{cases} 1, & (u,v) \in \boldsymbol{E} \\ 0, & (u,v) \notin \boldsymbol{E} \end{cases}. \tag{4}$$

The authority $a(v)$, the hub $h(u)$, and link matrix $\boldsymbol{A}$ have the following relationships

$$\begin{aligned} a(v) &= \sum_{u} \boldsymbol{A}^{\mathrm{T}}(v,u)\cdot h(u) \\ h(u) &= \sum_{v} \boldsymbol{A}(v,u)\cdot a(u). \end{aligned} \tag{5}$$

Thus, the recursion formula is

$$\begin{aligned} \boldsymbol{a}_k &= \boldsymbol{A}^{\mathrm{T}}\boldsymbol{h}_{k-1} = \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{a}_{k-1}) = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})\boldsymbol{A}_{k-1} \\ \boldsymbol{h}_k &= \boldsymbol{A}\boldsymbol{a}_{k-1} = \boldsymbol{A}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{h}_{k-1}) = (\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})\boldsymbol{h}_{k-1} \end{aligned}. \tag{6}$$

When $k \to \infty$, the authority $\boldsymbol{a}$ converges to the main eigenvector of $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$. The hub $\boldsymbol{h}$ converges to the main eigenvector of $\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}$.

2.2. Prediction model

Assuming $\{x_k, y_k\}_{k=1}^{n}$ is training sample set, where $x_k \in \mathrm{R}^{\mathrm{d}}$ and $y_k \in \mathrm{R}$, d is the number of input variables. The LSSVM regression model can be provided as follows

$$f(x) = \mathbf{w}^{\mathrm{T}}\varphi(x_k) + b, \tag{7}$$

where $\varphi(\bullet)$ is a nonlinear function mapping input variables from the original space into a high feature space, $\mathbf{w}$ is a weight vector, and b is a bias.

The squared errors are used as the cost function according to the SRM principle. The LSSVM model can then be developed by using the following formulation:

$$\begin{aligned} &\min J(\mathbf{w}, e) = \mathbf{w}^{\mathrm{T}}\mathbf{w} + \frac{\gamma}{2}\sum_{k=1}^{n} e_k^2 \\ &\text{s.t. } f(x) = \mathbf{w}^{\mathrm{T}}\varphi(x_k) + b + e_k \end{aligned}, \tag{8}$$

where γ is a penalty parameter that balances model complexity and approximation accuracy, e_k is the kth error variable. Then, by using the Lagrangian method, the optimization problem can be converted into a group of linear equations:

$$\begin{aligned} L(\mathbf{w}, \gamma, e, \alpha, b) =& \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + \frac{\gamma}{2}\sum_{k=1}^{n} e_k^2 \\ &+ \sum_{k=1}^{n} \alpha_k\left(y_k - \mathbf{w}^{\mathrm{T}}\varphi(x_k) + b + e_k\right), \end{aligned} \tag{9}$$

where $\alpha_k (k = 1, 2, \ldots, n)$ is the Lagrangian multiplier vector. The solutions for optimality are obtained based on the KKT conditions. And we set $K(x, x_i) = \varphi(x)^{\mathrm{T}}\varphi(x_i)$.

Finally, the LSSVM regression model can be obtained as follows:

$$y(x) = \sum_{k=1}^{n} \alpha_k K(x, x_k) + b. \tag{10}$$

3. The proposed method

In this section, the modifications to the preliminary that forms the soft sensor method are represented. First, the variable processing approach is introduced.

3.1. Variable processing

As we mentioned above, the input variables are processed according to the cause–effect relationship and the time delay, which are measured by a modified TE method.

3.1.1. Modified TE

Because the soft sensor is built to predict the NOx concentration, the time-delay and non-Markovian characteristics in plant boiler system need to be considered when analyzing the cause–effect relationship between variables.

We calculate the TE between the current observation and the previous one to obtain the time delay between variables. Thus, the parameter τ is embedded in (3), and the following modifications considering the non-Markovian characteristics are also introduced into TE.

$$\begin{aligned} \boldsymbol{x}_n^{(k)} &= [x_n, x_{n-\tau}, \ldots, x_{n-(k-1)\tau}] \\ \boldsymbol{y}_n^{(l)} &= [y_n, y_{n-\tau}, \ldots, y_{n-(l-1)\tau}] \end{aligned}, \tag{11}$$

where k and l are related to the Markov property of system $\boldsymbol{X}$ and system $\boldsymbol{Y}$. Theoretically, k and l are set to 1 if the systems

are non-Markovian. Therefore, the embedded TE is defined as follows

$$T_{\boldsymbol{X}\to\boldsymbol{Y}} = \sum_{y_n,\boldsymbol{y}_{n-1}^{(l)},\boldsymbol{x}_{n-1}^{(k)}} p(y_n,\boldsymbol{y}_{n-1}^{(l)},\boldsymbol{x}_{n-1}^{(k)})\log_2\frac{p(y_n|\boldsymbol{y}_{n-1}^{(l)},\boldsymbol{x}_{n-1}^{(k)})}{p(y_n|\boldsymbol{y}_{n-1}^{(l)})}. \tag{12}$$

As it is shown in (12), the TE is practically calculated by binary data, and then, we change the variables of the system into 0–1 type. For a complex nonlinear system, we assume it contains a discrete sequence $\boldsymbol{X} = [x_1, x_2, \ldots x_n]$ with the length of n. There are two complementary events: whether the value fluctuation exceeds a certain range or not, as

$$\tilde{x}_j = \begin{cases} 1, \text{if } |x_i - x_{i-1}| \geq \delta, j = i \\ 0, \quad \text{otherwise} \end{cases}, \tag{13}$$

where x_i is the value of $\boldsymbol{X}$ at the time i, δ is the parameter of the fluctuation range, $\tilde{x}_j$ is the value of the new sequence at the time j. $\boldsymbol{X}$ can be binarized into a new sequence $\tilde{\boldsymbol{X}}$. As for two variables, they have to be normalized before binarized to eliminate numerical differences. Considering that there exists random delay between variables in the system, thus, $\boldsymbol{x}_n^{(k)}$ and $\boldsymbol{y}_n^{(l)}$ are modified as follows

$$\begin{aligned} \tilde{\boldsymbol{x}}_n^{(k)} &= \begin{cases} \cup_{i=n-k+1}^{n} x_i, \text{if} y_{n+1} = 1 \\ x_n, \quad \text{otherwise} \end{cases} \\ \tilde{\boldsymbol{y}}_n^{(l)} &= \begin{cases} \cup_{i=n-l+1}^{n} y_i, \text{if} y_{n+1} = 1 \\ y_n, \quad \text{otherwise} \end{cases} \end{aligned}, \tag{14}$$

where $\cup$ is the symbol of the union of binary variables. Overall the modified TE for complex nonlinear systems is shown as

$$T_{\boldsymbol{X}\to\boldsymbol{Y}} = \sum_{y_n,\tilde{\boldsymbol{y}}_{n-1}^{(l)},\tilde{\boldsymbol{x}}_{n-1}^{(k)}} p(y_n,\tilde{\boldsymbol{y}}_{n-1}^{(l)},\tilde{\boldsymbol{x}}_{n-1}^{(k)})\log_2\frac{p(y_n|\tilde{\boldsymbol{y}}_{n-1}^{(l)},\tilde{\boldsymbol{x}}_{n-1}^{(k)})}{p(y_n|\tilde{\boldsymbol{y}}_{n-1}^{(l)})}. \tag{15}$$

The results of TE of two variables are $\boldsymbol{T}_{t+l-1}^{\boldsymbol{X}\to\boldsymbol{Y}}$ and $\boldsymbol{T}_{t+l-1}^{\boldsymbol{Y}\to\boldsymbol{X}}$, where $t+l-1 = 1, 2, \ldots, n$. $\boldsymbol{T}_{t+l-1}^{\boldsymbol{X}\to\boldsymbol{Y}}$ indicates the TE from $\boldsymbol{X}$ to $\boldsymbol{Y}$ under the parameter t and l. The mutation value of $\boldsymbol{T}_{t+l-1}^{\boldsymbol{X}\to\boldsymbol{Y}}$ is the transfer entropy from $\boldsymbol{X}$ to $\boldsymbol{Y}$, the parameter $t+l-1$ is the time delay, and vice versa.

However, there could be unrelated variable pairs and the variables with the same value. Thus, we design an algorithm to find the relationships mentioned above. The algorithm 1[2] shows the procedure of finding the relationship between variables.

The algorithm shows the relationship between two variables. Further, if two variables have a cause–effect relationship, the algorithm also provides the cause–effect relation strength and the time delay from one to the other.

3.1.2. Input variable selection

According to the results calculated from Algorithm 1, the variables which unrelated to the output variable, and the variables which are the same as others are deleted from the candidates (where input variables are chosen from). Then we start to select input variables. According to the graph theory, the vertexes are the candidate variables; the initial value $\boldsymbol{a}_0$ is the TEs from candidate variables to the output variable.

Assuming $\boldsymbol{X}_i$, where $i = 1, 2, \ldots, n$, are the variables of system. $\boldsymbol{X}_n$ is the output variable, the others are candidates. A $n-1 \times n-1$ asymmetric matrix is constructed based on the results of Algorithm 1 and according to (4).

$$\boldsymbol{A}(i,j) = \begin{cases} C_{\boldsymbol{X}_i\to\boldsymbol{X}_j}, i \neq j, i < n, j < n \\ 0, \qquad i = j, i < n, j < n \end{cases}. \tag{16}$$

[2] 0 is assigned to indicate that there is no relationship nor time delay in that direction; the algorithm of drawing upper envelope is referred to [28].

Algorithm 1: Extract Relationships form TE Results

Input: $\boldsymbol{T}_{t+l-1}^{\boldsymbol{X}\to\boldsymbol{Y}}$, $\boldsymbol{T}_{t+l-1}^{\boldsymbol{Y}\to\boldsymbol{X}}$, parameter ε_1 and parameter ε_2.
Output: Cause–effect strength from $\boldsymbol{X}$ to $\boldsymbol{Y}$ $C_{\boldsymbol{X}\to\boldsymbol{Y}}$, Cause–effect strength from $\boldsymbol{Y}$ to $\boldsymbol{X}$ $C_{\boldsymbol{Y}\to\boldsymbol{X}}$, The time delay from $\boldsymbol{X}$ to $\boldsymbol{Y}$ $D_{\boldsymbol{X}\to\boldsymbol{Y}}$, The time delay from $\boldsymbol{Y}$ to $\boldsymbol{X}$ $D_{\boldsymbol{Y}\to\boldsymbol{X}}$, Same variable flag *Flag*.

```
if Pearson correlation coefficient of T^{X→Y}_{t+l-1} and T^{Y→X}_{t+l-1} > ε1 then
    Flag ← 1 (the same variables);
else
    Flag ← 0 (not the same variables);
end
if Flag = 0 then
    P^{X→Y} ← Pearson correlation coefficient of the upper envelope of
    T^{X→Y}_{t+l-1} and the line y = 1;
    P^{Y→X} ← Pearson correlation coefficient of the upper envelope of
    T^{Y→X}_{t+l-1} and the line y = 1;
    if P^{X→Y} > ε2 then
        C_{X→Y} ← 0, D_{X→Y} ← 0;
    else
        C_{X→Y} ← max(T^{X→Y}_{t+l-1}), D_{X→Y} ← t + l − 1 under max(T^{X→Y}_{t+l-1});
    end
    if P^{Y→X} > ε2 then
        C_{Y→X} ← 0, D_{Y→X} ← 0;
    else
        C_{Y→X} ← max(T^{Y→X}_{t+l-1}), D_{Y→X} ← t + l − 1 under max(T^{Y→X}_{t+l-1});
    end
end
return C_{X→Y},C_{Y→X},D_{X→Y},D_{Y→X},Flag
```

The initial authority $\boldsymbol{a}_0$ indicates the influence from the other variables to the output variable, which consisted as

$$\boldsymbol{a}_0(i) = C_{\boldsymbol{X}_i\to\boldsymbol{X}_n}, i < n. \tag{17}$$

According to (6), after the iteration, we can obtain the authority $\boldsymbol{a}$ and hub $\boldsymbol{h}$. During the iteration, to overcome numerical overflow, the vectors are divided by their maximum elements of each step.

The independent correlation index (ICI) can be obtained as

$$\boldsymbol{I} = \frac{\boldsymbol{a}_0}{\boldsymbol{h}} / \sum \frac{\boldsymbol{a}_0(i)}{\boldsymbol{h}(i)}, i < n. \tag{18}$$

The variable that has a high rank in the index indicates that the variable has a stronger cause–effect relationship with the output variable, in the meantime has a weaker relationship with other variables. High-rank variable is more suitable for being an input variable. Based on ICI rank, we choose the input variable set.

3.1.3. Sequence adjustment

We suppose the time for an object moving from inlet to outlet is τs, the resolution of the record device is 1s. The observations of the inlet and outlet are two discrete-time variables $\boldsymbol{I}$ and $\boldsymbol{O}$.

$$\begin{aligned} \boldsymbol{I} &= [I_1, I_2, \ldots, I_n] \\ \boldsymbol{O} &= [O_1, O_2, \ldots, O_n] \end{aligned}. \tag{19}$$

The observation of the same object at inlet and outlet is I_t and $O_{t+\tau}$, respectively. In other words, the delay between these two variables is τs. To reduce the delays, we need to adjust the sequences. We displace the sequences to be aligned and get $\boldsymbol{I}_{new}$ and $\boldsymbol{O}_{new}$ as

$$\begin{aligned} \boldsymbol{I}_{new} &= [I_1, I_2, \ldots, I_{n-\tau}] \\ \boldsymbol{O}_{new} &= [O_{1+\tau}, O_{2+\tau}, \ldots, O_n] \end{aligned}. \tag{20}$$

For now, the variable processing procedure is completed.

3.2. FFOLSSVM

Another part of the soft sensor is the prediction model. We propose an FFOLSSVM (forgetting factor online least square support vector machine) model to consider the change of the performance caused by the variation of new input data.

3.2.1. LSSVM based iteration approach

LSSVM uses training samples to predict new samples. The accuracy depends on the weights which are trained by the optimizing algorithms. However, the training stage is time-consuming. Therefore we design an approach to change the weights according to the last prediction error instead of retraining based on new training samples. The approach obtains prediction value while updating weights.

First, we modify (10), so the model can function using the iteration approach.

$$\begin{aligned} y(x) &= \sum_{k=1}^{n} \alpha_k K(x, x_k) + b \\ &= \sum_{k=1}^{n} \beta_k K(x, x_k) = \boldsymbol{K}\boldsymbol{\beta} \end{aligned}, \tag{21}$$

where $\boldsymbol{\beta} = \boldsymbol{K}^{+}(\boldsymbol{K}\alpha+b)$, $\boldsymbol{K}^{+}$ is generalized inverse of $\boldsymbol{K}$. $\boldsymbol{\beta}$ could be several values, but it would not influence the following equations.

In this study, Gaussian kernel function is selected as kernel function.

$$K(x, x_k) = \exp\left(-\|x - x_k\|^2 / \sigma^2\right). \tag{22}$$

At initial stage,

$$\boldsymbol{\beta}_0 = \boldsymbol{K}_0^{-1} y_0 = (\boldsymbol{K}_0^{\mathrm{T}}\boldsymbol{K}_0)^{-1}\boldsymbol{K}_0^{\mathrm{T}} y_0, \tag{23}$$

where $\boldsymbol{K}_0$ is calculated by initial training samples. To keep $\boldsymbol{K}_0^{\mathrm{T}}\boldsymbol{K}_0$ being non-singular, regularization parameter is introduced. $\boldsymbol{K}_0^{\mathrm{T}}\boldsymbol{K}_0$ is transformed as

$$\boldsymbol{K}_0^{\mathrm{T}}\boldsymbol{K}_0 + \frac{1}{C} \quad s.t. \min_{\beta_0} \left\{ \left\| \boldsymbol{K}_0\boldsymbol{\beta}_0 - y_0 \right\| + \frac{\left\|\boldsymbol{\beta}_0\right\|^2}{C} \right\}. \tag{24}$$

For $\boldsymbol{Z}_0 = \boldsymbol{K}_0^{\mathrm{T}}\boldsymbol{K}_0 + \frac{1}{C}$

Moreover, we have

$$\boldsymbol{\beta}_0 = {\boldsymbol{Z}_0}^{-1}\boldsymbol{K}_0^{\mathrm{T}} y_0. \tag{25}$$

When a new sample is introduced into the model, $\boldsymbol{K}_1$ is constructed based on a new sample.

$$\boldsymbol{\beta}_1 = {\boldsymbol{Z}_1}^{-1}\begin{bmatrix} \boldsymbol{K}_0 \\ \boldsymbol{K}_1 \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} y_0 \\ y_1 \end{bmatrix}, \tag{26}$$

and,

$$\boldsymbol{Z}_1 = \tfrac{\boldsymbol{I}}{C} + \begin{bmatrix} \boldsymbol{K}_0 \\ \boldsymbol{K}_1 \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} \boldsymbol{K}_0 \\ \boldsymbol{K}_1 \end{bmatrix} = \boldsymbol{Z}_0 + \boldsymbol{K}_1^{\mathrm{T}}\boldsymbol{K}_1 \ . \tag{27}$$

Transform $\begin{bmatrix} \boldsymbol{K}_0 \\ \boldsymbol{K}_1 \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} y_0 \\ y_1 \end{bmatrix}$ as

$$\begin{aligned} \begin{bmatrix} \boldsymbol{K}_0 \\ \boldsymbol{K}_1 \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} &= \boldsymbol{K}_0^{\mathrm{T}} y_0 + \boldsymbol{K}_1^{\mathrm{T}} y_1 = \boldsymbol{Z}_0\boldsymbol{Z}_0^{-1}\boldsymbol{K}_0^{\mathrm{T}} y_0 + \boldsymbol{K}_1^{\mathrm{T}} y_1 \\ &= \boldsymbol{Z}_0\boldsymbol{\beta}_0 + \boldsymbol{K}_1^{\mathrm{T}} y_1 = (\boldsymbol{Z}_1 - \boldsymbol{K}_1^{\mathrm{T}}\boldsymbol{K}_1)\boldsymbol{\beta}_0 + \boldsymbol{K}_1^{\mathrm{T}} y_1 \\ &= \boldsymbol{Z}_1\boldsymbol{\beta}_0 - \boldsymbol{K}_1^{\mathrm{T}}\boldsymbol{K}_1\boldsymbol{\beta}_0 + \boldsymbol{K}_1^{\mathrm{T}} y_1 \end{aligned}, \tag{28}$$

and,

$$\boldsymbol{\beta}_1 = \boldsymbol{\beta}_0 + {\boldsymbol{Z}_1}^{-1}\boldsymbol{K}_1^{\mathrm{T}}(y_1 - \boldsymbol{K}_1\boldsymbol{\beta}_0). \tag{29}$$

Therefore, we have

$$\begin{aligned} \boldsymbol{Z}_{k+1}^{-1} &= (\boldsymbol{Z}_k + \boldsymbol{K}_{k+1}^{\mathrm{T}}\boldsymbol{K}_{k+1})^{-1} \\ \boldsymbol{\beta}_{k+1} &= \boldsymbol{\beta}_k + \boldsymbol{Z}_{k+1}^{-1}\boldsymbol{K}_{k+1}^{\mathrm{T}}(y_{k+1} - \boldsymbol{K}_{k+1}\boldsymbol{\beta}_k) \end{aligned} \ . \tag{30}$$

Because $\boldsymbol{Z}_{k+1}^{-1}$ is difficult to calculate, we simplify it and get

$$\boldsymbol{Z}_{k+1}^{-1} = (\boldsymbol{Z}_k + \boldsymbol{K}_{k+1}^{\mathrm{T}}\boldsymbol{K}_{k+1})^{-1} = \boldsymbol{Z}_k^{-1} - \frac{\boldsymbol{Z}_k^{-1}\boldsymbol{K}_{k+1}^{\mathrm{T}}\boldsymbol{K}_{k+1}\boldsymbol{Z}_k^{-1}}{I + \boldsymbol{K}_{k+1}\boldsymbol{Z}_k^{-1}\boldsymbol{K}_{k+1}^{\mathrm{T}}}. \tag{31}$$

We set $\boldsymbol{P}_k = \boldsymbol{Z}_k^{-1}$, and get

$$\boldsymbol{P}_{k+1} = \boldsymbol{P}_k - \frac{\boldsymbol{P}_k\boldsymbol{K}_{k+1}^{\mathrm{T}}\boldsymbol{K}_{k+1}\boldsymbol{P}_k}{I + \boldsymbol{K}_{k+1}\boldsymbol{P}_k\boldsymbol{K}_{k+1}^{\mathrm{T}}}, \tag{32}$$

and,

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + \boldsymbol{P}_{k+1}\boldsymbol{K}_{k+1}^{\mathrm{T}}(y_{k+1} - \boldsymbol{K}_{k+1}\boldsymbol{\beta}_k). \tag{33}$$

For the time-varying system, the current operation condition can be changed dramatically. Consequently, the output value drastically changes. The weights of the prediction model should be adjusted. Therefore, the forgetting factor of weights is added in the algorithm. The forgetting factor is updated based on the error of the prediction, which is particularly sensitive to new data [29].

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + \frac{\boldsymbol{P}_{k+1}\boldsymbol{K}_{k+1}^{\mathrm{T}}}{1+\xi_{k+1}}\hat{e}_{k+1}, \tag{34}$$

where

$$\hat{e}_{k+1} = y_{k+1} - \boldsymbol{K}_{k+1}\boldsymbol{\beta}_k, \tag{35}$$

and,

$$\xi_{k+1} = \boldsymbol{K}_{k+1}\boldsymbol{P}_k\boldsymbol{K}_{k+1}^{\mathrm{T}}, \tag{36}$$

where ξ_{k+1} is the error of new sample based on $\boldsymbol{\beta}_k$. ξ_{k+1} is the update flag, if $\xi_{k+1}= 0$, then $\boldsymbol{P}_{k+1} = \boldsymbol{P}_k$. If $\xi_{k+1}> 0$, then (32) is transformed as

$$\boldsymbol{P}_{k+1} = \boldsymbol{P}_k - \frac{\boldsymbol{P}_k\boldsymbol{K}_{k+1}^{\mathrm{T}}\boldsymbol{K}_{k+1}\boldsymbol{P}_k}{\varepsilon_{k+1}^{-1} + \xi_{k+1}}, \tag{37}$$

where

$$\varepsilon_{k+1} = \lambda_k - \frac{1 - \lambda_k}{\xi_{k+1}}. \tag{38}$$

λ_k is the forgetting factor, $0 < \lambda_k \leq 1$. The expression of the λ_{k+1} is shown as

$$\begin{aligned} \lambda_{k+1} = \Bigg\{ 1 + (1+\rho) \Bigg[& \ln(1+\xi_{k+1}) \\ & + \left(\frac{(\nu_{k+1}+1)\,\eta_{k+1}}{1+\xi_{k+1}+\eta_{k+1}} - 1 \right) \frac{\xi_{k+1}}{1+\xi_{k+1}} \Bigg] \Bigg\}^{-1} \end{aligned}, \tag{39}$$

where

$$\begin{aligned} \eta_{k+1} &= \tfrac{\hat{e}_{k+1}^2}{\zeta_{k+1}} \\ \zeta_{k+1} &= \lambda_k(\zeta_k + \tfrac{\hat{e}_{k+1}^2}{1+\xi_{k+1}}) \\ \nu_{k+1} &= \lambda_k(\nu_k + 1) \end{aligned} \ , \tag{40}$$

where ρ is a fixed number, initial ζ and ν are between 0 and 1.

Therefore, the LSSVM based iteration approach is shown as

$$\hat{y_{k+1}} = \boldsymbol{\beta}_{k+1}\boldsymbol{K}_{k+1}, \tag{41}$$

where $\hat{y_{k+1}}$ is the prediction value.

The aAlgorithm 2[3] illustrates the LSSVM based iteration approach.

[3] In this paper, the parameters of Algorithm 2 are set as $\lambda_0 = 0.1$, $\zeta_0 = 1, \nu_0 = 0.1, \rho = 25$.

Algorithm 2: LSSVM based Iteration Approach

Input: λ_0,ζ_0,ν_0, ρ,C,σ, initial data $\boldsymbol{D}_{initial}$, following data $\boldsymbol{D}_{following}$.
Output: Prediction value $\boldsymbol{y}$.
Initialize: Train the traditional LSSVM model based on $\boldsymbol{D}_{initial}$, and get $\boldsymbol{\beta}$ in (21);
Calculate $\boldsymbol{K}_0$, $\boldsymbol{\beta}_0$, $\boldsymbol{P}_0$ based on $\boldsymbol{D}_{initial}$;
$k \leftarrow 0$;
while $k <$ *length* ($\boldsymbol{D}_{following}$) **do**
  Calculate $\boldsymbol{K}_{k+1}$ $\boldsymbol{\beta}_{k+1}$, ξ_{k+1}, ε_{k+1} based on (k+1)th $\boldsymbol{D}_{following}$;
  if $\xi_{k+1}= 0$ **then**
    $\boldsymbol{P}_{k+1} \leftarrow \boldsymbol{P}_k$
  else
    $\boldsymbol{P}_{k+1} \leftarrow \boldsymbol{P}_k - \frac{\boldsymbol{P}_k \boldsymbol{K}_{k+1}^{\mathrm{T}} \boldsymbol{K}_{k+1} \boldsymbol{P}_k}{\varepsilon_{k+1}^{-1} + \xi_{k+1}}$
  end
  Calculate η_{k+1}, ζ_{k+1} and ν_{k+1};
  k $\leftarrow$ k+1;
  $\boldsymbol{y}_{k+1} \leftarrow \boldsymbol{K}_{k+1}\boldsymbol{\beta}_{k+1}$;
  return $\boldsymbol{y}_{k+1}$
end

3.2.2. Update strategy

The LSSVM based iteration approach uses the initial training samples for the whole progress. When the initial data no longer covers the current condition, we will not get the excellent accuracy of the prediction model. Thus, we propose an update strategy to renew the input samples of the kernel function, so it is one kind of the kernel update.

This study draws on the idea of on-line updating data in [17] and proposes an on-line update prediction model based on prediction error, which discards some old samples according to prediction error, adds the same number of new samples to update training samples, and keep the dimension of the kernel matrix. This procedure satisfies both calculating speed and prediction accuracy.

Define the prediction error as the deviation between the prediction value and the actual one.

$$e_i = y_i - \tilde{y}_i, \tag{42}$$

where y_i, $\tilde{y}_i$ and e_i is the actual value, the prediction value and the prediction error at the time i, respectively.

The average prediction error of the model is defined as

$$e_{ave} = \sum_{i=1}^{m} \frac{e_i}{m}, \tag{43}$$

where m is the number of the training samples.

A sliding window is established, and the average prediction error is used to determine whether the model should be updated. The flowchart of the update strategy is shown in Fig. 1.

n sets of training samples whose errors are higher than the threshold are inserted in the sliding window. n sets of training samples with the smallest Lagrange value (9) are deleted at the same time. Then the traditional LSSVM is retained; after that, the iterative approach is activated until it no longer satisfies the prediction accuracy. It can avoid deleting significant samples during retraining.

Compared with the study in [17], the approach updating is triggered by the average error. The error of the individual sample guides the adding/deleting of the samples, which is more aimed at the accuracy requirement. The strategy does not need frequent updates because of the forgotten factor and iterative algorithm. The number of training samples of the model is always the same in this study, and the size of the matrix of the model is not changed, which helps the stable operation of the computer, and reduces the computational cost.

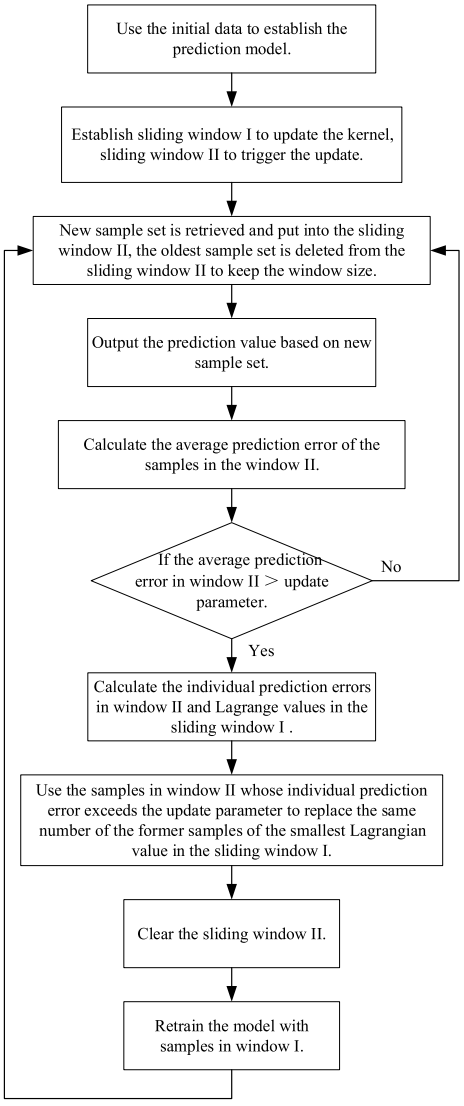

Fig. 1. Flowchart of the approach update strategy.

4. Nox prediction method based on real data

In this section, the NOx concentration of SCR reactor inlet of a 600 MW coal-fired unit in Inner Mongolia, China is used as a forecasting object. The real data with the resolution of 1s is retrieved from DCS for 201 600 s to conduct prediction experiments. 33 000 s of data is used to illustrate the result.

4.1. Brief description of the investigated power plant

The investigated power plant is a typical coal-fired boiler, which has a large furnace with a 19.08 m × 19.08 m cross-section and a 65.1 m height. Six medium-speed pulverizers are used to supply pulverized coal, which is then mixed with the primary air and blown into the furnace through the burner nozzles. An imaginary horizontal circle with a 7.69 m diameter is then formed in the center. Six layers of primary air (A, B, C, D, E, and F) and eight layers of input air (AA, AB, BC, CC, DD, DE, EF, and FF) are distributed alternately in a vertical direction. Six layers of circumferential air (A–F) are arranged to surround the burner nozzles to supply sufficient air and guarantee the combustion stability. Four layers of overfire air (OFA) are installed over the upper nozzles to replenish the air in the later combustion phase

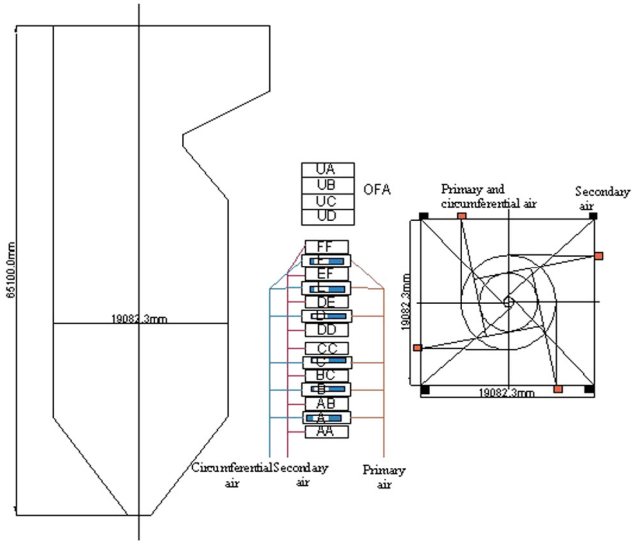

Fig. 2. The schematic diagram of the furnace and burner layout.

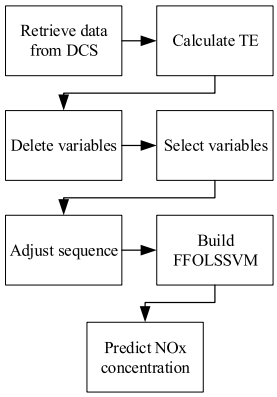

Fig. 3. The flowchart of the soft sensor method for NO*x* emission.

for better combustion efficiency. The schematic diagram of the furnace and burner layout is shown in Fig. 2.

4.2. Nox concentration prediction

The flowchart of the soft sensor method for NO*x* emission is shown in Fig. 3.

4.2.1. Retrieve data from DCS

The real data retrieved from DCS is included as follows.

- 78 variables from the boiler;
- 8 variables from the turbine;
- 21 variables from the reactor and flue.

There are 106 candidate variables in total, and the output variable: the NO*x* concentration of the SCR reactor inlet. Because the data has already been smoothed by DCS, it can be used directly for the soft sensor.

The data cover a broad range of the operation condition, and the variation of a part of variables is shown in Table 1.

4.2.2. Calculate TE

All the variables, including output variable, are paired in groups of two, and their TEs are calculated to obtain the results of time delay and cause–effect strength.

Table 1
Part of process variables and their ranges of the investigated boiler in the operation condition.

Variables	Unit	Variation range	Standard deviation
Variables from boiler			
Boiler load	MW	[333.8 605.9]	80.0
Total air rate	m^3/h	[1078.9 1862.7]	205.8
Circumferential air A	%	[8.2 46.5]	12.5
Over fire air A	%	[0.0 100.0]	28.0
Total coal rate	t/h	[151.4 328.4]	39.7
Variables from the turbine			
Pressure of main steam	Kpa	[16.1 139.0]	23.6
Temperature of main steam	°C	[526.6 548.1]	2.2
Variables from the reactor and flue			
NO*x* concentration of SCR inlet	mg/m^3	[201.5 431.8]	27.5
The proportion of O_2 in flue	%	[1.6 4.1]	0.6

Table 2
A part of input variables with high ICI.

NO.	Variables	ICI
1	The current of mill A	5.00
2	The deflection secondary air at level EF	4.98
3	The differential pressure at A side of the hearth	4.92
4	The current of mill C	4.65
5	The current of mill D	4.577
6	The deflection secondary air at level CD	4.552
7	The despun secondary air at level FF	4.519
8	The differential pressure at B side of the hearth	4.112
9	The despun secondary air at level OFA	4.110
10	The deflection secondary air at level DE	3.797

First, according to (13), discrete data should be binarized. The process of NO*x* concentration and boiler load is shown as an illustration. Fig. 4 shows the binarized process, the data is normalized by mapping into [0,1], and the parameter δ is set to 0.1.

The TE (15) between NO*x* concentration and boiler load is shown in Fig. 5. The parameters k and l are set to 15 and 5 respectively. $t + l - 1$ is from 1 to 600 s.

Then we obtain the cause–effect relation strength and time delay based on Algorithm 1.

4.2.3. Delete variables

Some variables from the 106 candidate variables are deleted. The deletion principle of the variables is as follows:

1. The variables are irrelevant (no cause–effect strength, C=0 in Algorithm 1) and redundant (same variable of others, flag=1 in Algorithm 1).
2. The time delay ranges from the output variable to the candidate one.

Principle 1 guarantees that the remain candidates all have a relationship with the output variable, and keep the number of the candidates as low as possible. Principle 2 guarantees that the remain candidates are the possible cause of the output variable. The principles help reduce the computational cost for the next step.

After deleting, there are 53 variables left for further selection.

4.2.4. Select variables

Among the remaining 53 variables, many variables are strongly correlated, such as coal-mill current and coal flow. Because multiple variables influence coal flow, and the coal mill current can directly reflect the coal flow without being influenced, so the coal mill current is more representative than others. HITS is used to obtain ICI, which is shown in Fig. 6.

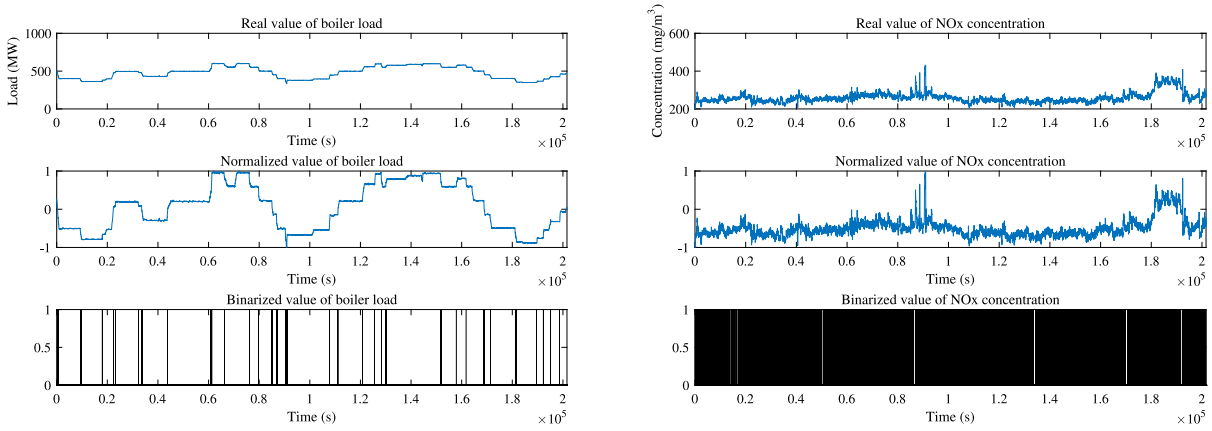

Fig. 4. The process of NO*x* concentration and boiler load.

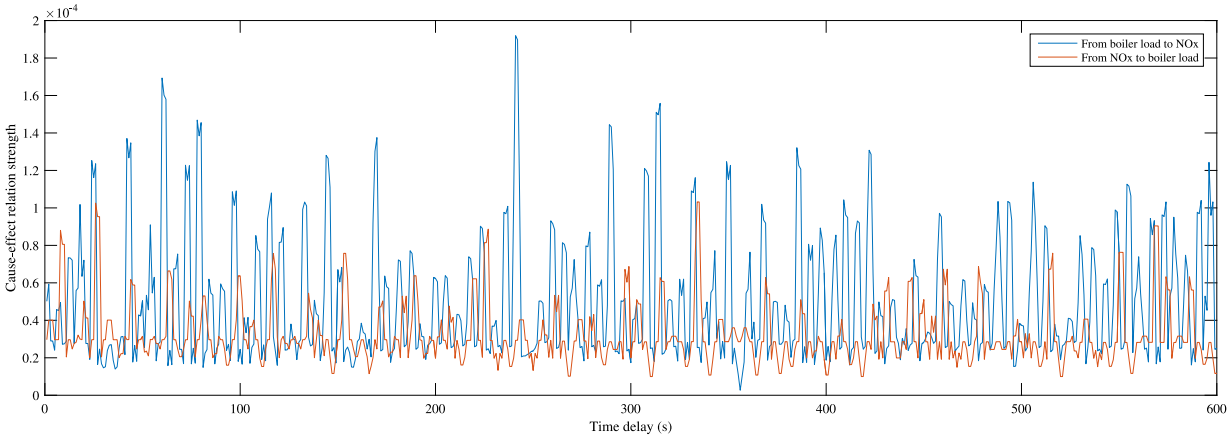

Fig. 5. TE calculation between boiler load and NO*x*.

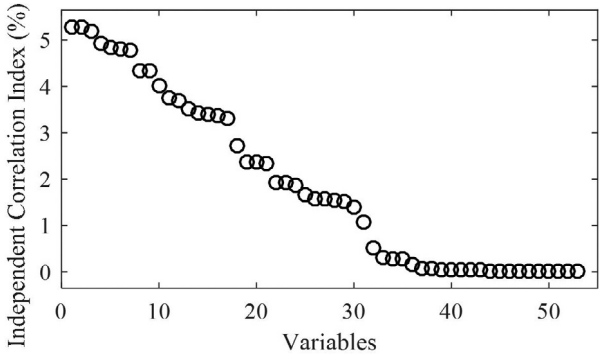

Fig. 6. ICI of input variables.

Table 3
TE results of selected variables.

$TE_{column \to row}$ ($\times 10^{-2}$)	1	2	3	4	5	6	7	8	9	10	NO*x*
1	0	0	0	0	0	0	0	0	0	0	4.838
2	0	0	0	0	0	0	0	0	0	0	5.474
3	0	0	0	0	0	0	0	0	0	0	4.478
4	0	0	0	0	0	0	0	0	0	0	4.426
5	0	0	0	0	0	0	0	0	0	0	4.327
6	0	0	0	0	0	0	0	0	0	0	3.765
7	0	0	0	0	0	0	0	0	0	0	4.655
8	0	0	0	0	0	0	0	0	0	0	5.082
9	0	0	0	0	0	0	0	0	0	0	4.979
10	0	0	0	0	0	0	0	0	0	0	4.155

In this paper, the first ten variables are selected as input variables to predict NO*x* through the prediction model.

TE results of the ten selected variables are shown in Table 2. The numbers in the first column and row are the same as that in Table 3 to present different input variables.

As shown in Table 3, the cause–effect strengths between input variables are all 0, which indicate that there are no cause–effect relationships between input variables. There still exist cause–effect relationships between input variables and output variable.

The time delay between variables are shown in Table 4.

According to the study in [30], the coal volume makes the most contribution to the NO*x* concentration. The coal is delivered into the furnace by the mill, and the current of the mill directly reflects the coal volume. The different mills are installed at different places of the furnace. Moreover, in the practical application, not all the mill keeps operating. Thus, only three mills selected as input variables. In [31], secondary airs are found out to be the great contributors to the NO*x* by combustion mechanism analysis and experiments. The contributions of secondary airs at different levels are varied. Thus, it is reasonable for the method to select secondary air as input variables. The differential pressures of the two sides of the hearth reflect the pressure in the furnace. It was pointed out that the surrounding pressure rise speed and NO*x* emissions exhibit similar tendencies [32]. Thus, the differential pressures are the cause of NO*x*.

Table 4
The time delay between input variables and output variable.

NO.	Variables	Time delay to output variable (s)
1	The current of mill A	341
2	The deflection secondary air at level EF	299
3	The differential pressure at A side of the hearth	325
4	The current of mill C	337
5	The current of mill D	330
6	The deflection secondary air at level CD	305
7	The despun secondary air at level FF	308
8	The differential pressure at B side of the hearth	267
9	The despun secondary air at level OFA	323
10	The deflection secondary air at level DE	333

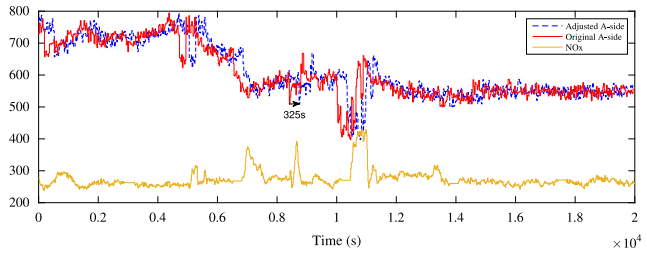

Fig. 7. The sequence adjustment of A-side.

4.2.5. Adjust sequence

The selected input variables and output variables are adjusted based on the calculated time delays in Table 4. As it is illustrated in Table 4, the time delay between differential pressure at A side of the hearth (A-side for convenience) and NO*x* concentration is 325 s, which means the current operation of A-side cause the respond of NO*x* in 325 s later. Thus, we displace the sequence of A-side for a 325s-delay. Fig. 7 shows the process of adjustment for A-side.

4.2.6. Build prediction model

The prediction model is built based on real data. The structure of the model is shown in Fig. 8.

Based on the proposed method, we select some input variables, namely the current of mill, the deflection secondary air at level EF, the differential pressure at A side of the hearth, the current of mill C, the current of mill D, the deflection secondary air at level CD, the despun secondary air at level FF, the differential pressure at B side of the hearth, the despun secondary air at level OFA and the deflection secondary air at level DE. The prediction model calculates the NO*x* concentration based on those secondary variables.

5. Results of NO*x* prediction

In this section, two experiments are carried out to testify

1. Viability and effectiveness of the proposed variable processing
2. Viability and effectiveness of the proposed soft sensor method

5.1. Experiment on variable processing

In this part, traditional LSSVM is built based on real data to verify the proposed variable processing method with the structure in Fig. 8. Then the LSSVM model is constructed based on continuous 4000s with 10 s resolution data. 75% of the data is training samples and 25% of the data is testing samples. LSSVM

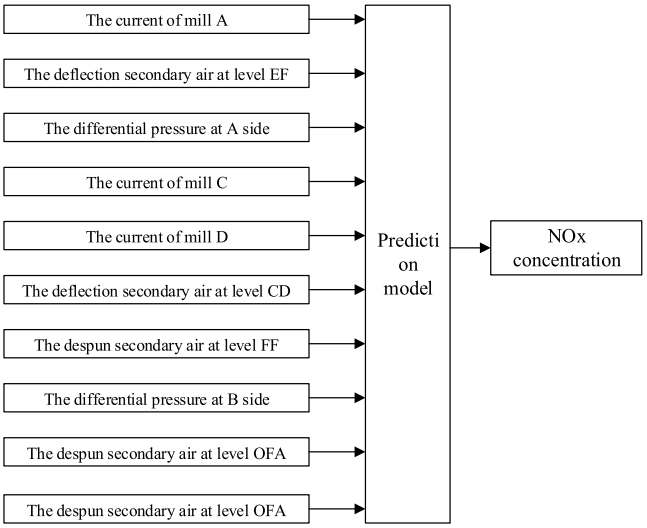

Fig. 8. Structure diagram of NO*x* emission model.

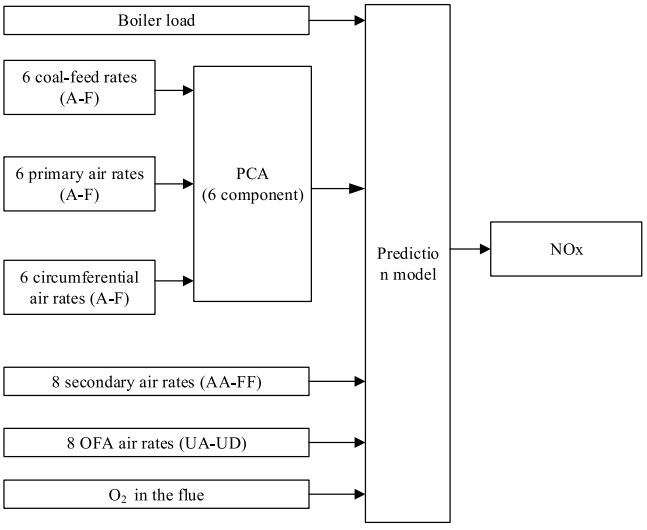

Fig. 9. Structure diagram of NO*x* emission model in [17].

uses RBF as a kernel function, and 5-fold cross-validation grid search optimizes parameters.

Root mean square error (RMSE) and mean relative error (MRE) are used as evaluation indicators.

The unit in this paper is similar to that in [17]. Therefore, as a comparison, using the input variable selection method in [17], a total of 20 input variables are selected referring to the combustion mechanism. The structure diagram is shown in Fig. 9.

Six different periods which represent six different conditions of the power plant are used to test the LSSVM and verify the proposed method. Fig. 10 shows the predictions and MREs based on different methods. The statistical results are shown in Table 5.

It is evident in Fig. 10 and Table 5 that the proposed method has the minimum RMSE and MRE compared with two other methods. The maximum MRE of the proposed method among the six different datasets is 8.17% while that of the former method and the method without sequence adjustment are 14.79% and 41.35%, respectively. The proposed method considerably increases the prediction accuracy and improves performance.

Subsequent data of that in Fig. 10(a) with the length of 2000 s is used to test the LSSVM. The model is built only based on the

(a) Predictions of Period 1 (b) Predictions of Period 2 (c) Predictions of Period 3

(d) MREs of Period 1 (e) MREs of Period 2 (f) MREs of Period 3

(g) Predictions of Period 4 (h) Predictions of Period 5 (i) Predictions of Period 6

(j) MREs of Period 4 (k) MREs of Period 5 (l) MREs of Period 6

Fig. 10. Predictions and MREs of different periods.

former method and the proposed method, to illustrate that it is necessary to update traditional LSSVM.

5.2. Experiment on prediction method

In this part, the entire prediction method is used to predict long-term NO*x* concentration. A total of 33 000 s with the 10 s resolution of real data is used to train and test it. To simulate the actual situation, we introduced the raw data from DCS directly into the model one by one. The data is not cleaned before being utilized to illustrate the proposed method and can be directly used in practical application. The actual value of NO*x* is delayed 50 s to be delivered into the model to calibrate the FFOLSSVM, to simulate the actual delay between measure equipment and DCS. The initial model of FFOLSSVM uses the first 300 sets of samples for training, and the parameters of the LSSVM DCS has already smoothed the data by 5-fold grid search cross-validation. RMSE and MRE are used to evaluate the model.

As a comparison, the same data is tested by the prediction method proposed in [17]. The model is referred to as LSSVMUP in the following.

The LSSVMUP model is built on the parameters of $\delta_1 = 0.8$ and $\delta_2 = 0.2$, and selected randomly because there are no introductions in the previous study. FFOLSSVM is constructed with

Table 5
Statistical results of testing based on different period of data.

Evaluation index	Period 1		Period 2		Period 3	
	RMSE (mg/m^3)	MRE (%)	RMSE (mg/m^3)	MRE (%)	RMSE (mg/m^3)	MRE (%)
Proposed method	4.4588	1.5213	1.4411	0.4802	3.4434	1.0174
Former method	7.0188	4.6906	2.6917	0.8904	7.6362	2.432
Unadjusted sequence	8.6168	16.1294	10.6342	3.4969	10.1039	3.4638
Evaluation index	Period 4		Period 5		Period 6	
	RMSE (mg/m^3)	MRE (%)	RMSE (mg/m^3)	MRE (%)	RMSE (mg/m^3)	MRE (%)
Proposed method	7.0177	1.4806	4.0143	1.2657	1.3237	0.4044
Former method	16.8756	2.2167	8.8755	3.1947	1.8345	0.5688
Unadjusted sequence	54.0213	2.7762	12.1678	4.2644	2.0173	0.7105

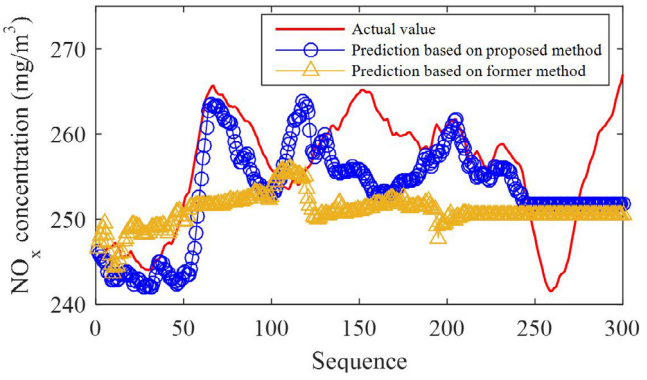

Fig. 11. Prediction results based on different model structure based on more data.

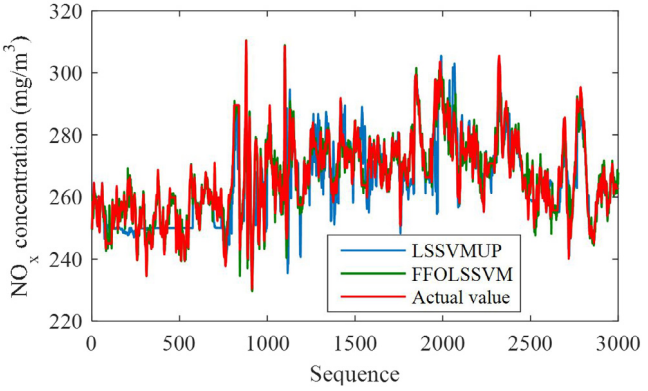

Fig. 12. Long-term prediction based on different methods.

$\varepsilon = 0.02$ to achieve high accuracy. The forecast results are shown in Fig. 12.

To show the parameters' influence on different models, LSSVMUP and FFOLSSVM are also constructed with different sets of parameters. The statistical results are shown in Table 6.

The relative errors are shown in Figs. 13 and 14.

6. Discussion

6.1. Discussion on variable processing

As shown in Fig. 10, predictions based on the proposed method can well predict the actual value, but the predictions based on the former method and that without adjusted sequence cannot follow the fluctuation of the actual value. As it is expected, the predictions based on the proposed method are more accurate

Table 6
Statistical results of long-term prediction based on different methods.

	LSSVMUP		FFOLSSVM	
Parameters	$\delta_1 = 0.8$ $\delta_2 = 0.2$	$\delta_1 = 0.9$ $\delta_2 = 0.3$	$\varepsilon = 0.02$	$\varepsilon = 0.06$
RMSE (mg/m^3)	8.23	6.51	3.15	5.67
MRE (%)	1.93	1.51	0.77	1.34
Calculation time (s)	1684	4351	1093	891
Update frequency	0.43	0.98	0.33	0.27
Calculating time per update (s)	1.31	1.48	1.1	1.09
Calculating time per sample (s)	0.56	1.45	0.36	0.30

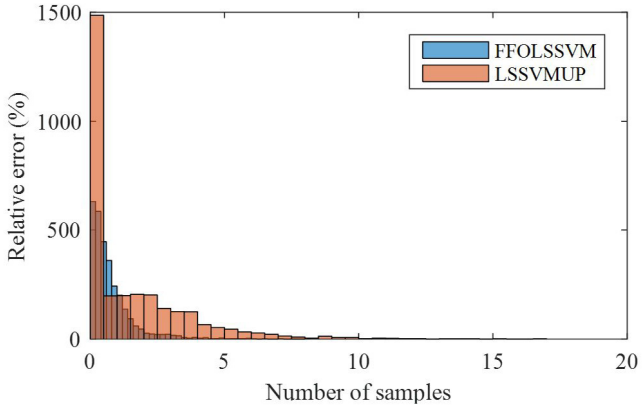

Fig. 13. Relative error distribution of two methods.

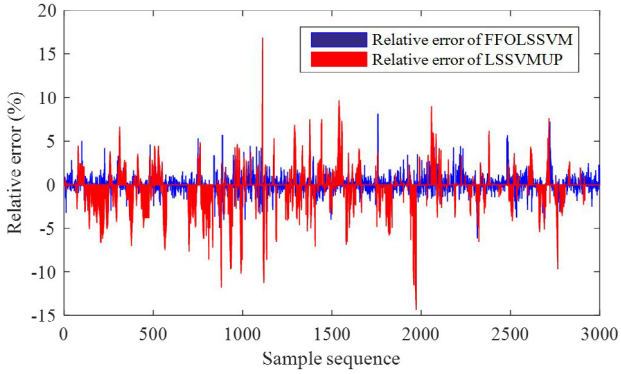

Fig. 14. Relative errors of two methods.

than that based on the previous method [17] in both periods. As for the aspect of sequence, adjustment of the sequence has a significant influence on improving prediction performance. Thus, Adjusting the sequence is very important for prediction and cannot be ignored. The time delays are detected through a statistical analysis of data without mechanism knowledge of the system; it can be migrated to other subjects.

The traditional LSSVM can only predict the first 100 samples based on the real data from DCS. As we mentioned before, the operation condition is continuously changing; old training samples cannot cover the feature of the new condition. In Fig. 11, 100 sequences stand for 1000 s of real-time, which is a long time for the condition to maintain. Therefore, to achieve long-term prediction, adaptive design is required.

6.2. Discussion on prediction method

In term of the number of parameters in the model, FFOLSSVM only has one parameter to choose, which is much more convenient. The parameter in FFOLSSVM directly relates to the prediction accuracy. As it is shown in Table 6, with the increase of the parameter, the accuracy, and computational time decrease. On the contrary, there are two parameters in LSSVMUP, and the unsuitable combination can cause failure in both accuracy and speed. In the field application, the data samples are too many for time-consuming optimization of parameters. The operation condition is changing; the initial parameters cannot always be the optimized value, which can cause the bad performance of the model. It is worth mentioning that when the model is running, the LSSVMUP with $\delta_1 = 0.8$ and $\delta_2 = 0.2$ adds the dimension of the model mapping matrix by adding samples. The final mapping matrix is 476 dimensions while the initial matrix is 300 dimensions. The final mapping matrix of LSSVMUP with $\delta_1 = 0.9$ and $\delta_2 = 0.3$ is 763 dimensions. A larger size of the matrix requires more time to be calculated. That is the reason for the LSSVMUP with different parameters have a difference in calculating time per update. Only when the computation time per sample is less than the sampling time of the practical applications, the method can be used. The sampling time of the investigated system is 2 s. Therefore, the proposed method can meet the requirement of computational cost and speed.

Regarding model prediction accuracy, 99.8% of the FFOLSSVM's prediction results are within 5% of the relative error, and the maximum relative error is only 8.57%, which fully meets the actual industry operation requirements. Compared with that, only 80% of the LSSVMUP is within 5%, and the maximum error is 16.76%.

6.3. Discussion on practical application

For practical application, a design must satisfy the requirements and avoid making it more difficult to implement at the same time.

NO*x* concentration measurement should be accurate, real-time, and stable. The soft sensor method is used because the hardware equipment cannot satisfy those requirements. Therefore, the design of soft sensor methods must consider the practical requirements.

(a) Accuracy
Accuracy reflects the generalization of a soft sensor method based on data-driven machine learning. Thus, the design of the method must come from a data-oriented view, and avoid mechanism analysis, for the mechanism analysis can hardly fully consider the operation all the time. The experiments in this paper also confirm it.

(b) Real-time property
The soft sensor can only be used when the computation time of each sample is less than sampling time.

(c) Stability
The stability of the soft sensor for a high-real-time-demand system relies on the computation time. It is important not to decrease the computation speed of each sample over time, and keep the time no more than half of the sampling time. Over-long computing time causes the system to halt.

Recent years have witnessed thriving researches on advanced machine learning technology. Most of them have excellent experimental accuracy. However, for practical application, they are still unable to be used because of its complex structure. Models having simple structures are more practical to construct and convenient for operators in the industry to maintain. That is why we use an LSSVM-based method.

However, this study only proposes a soft sensor method for measurement, which provides the value of NO*x* concentration for denitration system. The efficient control of denitration system still needs control logic design. At least, the numerical analysis based on the investigated power plant shows that the proposed method can save about 30% of the denitration materials per year when achieving the same denitration performance.

7. Conclusion

In this study, adaptive LSSVM based iterative prediction method for NO*x* concentration prediction in the coal-fired power plant is proposed. Firstly, we calculate the cause–effect relationship by modified TE, select an input variable set by HITS algorithm, and adjust the sequence of the variables. The cause–effect relationship depicted by TE is proposed as the criteria, and the HITS algorithm is migrated into the process of input variable selection. Moreover, it is the first time to identify time delay between variables and introduce the factor of time-delay to dissect and amend the practical prediction method by TE. Secondly, we propose an adaptive, LSSVM based iteration approach with the update procedure as a prediction model and use the selected variables to predict the output variable. We design a new iterative approach based on LSSVM, which only searches weights in the initial stage. The update procedure of the prediction model uses only one parameter, which directly guides the prediction accuracy. The prediction accuracy increases with the decrease in the value of the parameter. Finally, Numerical experiments based on real data of a 600 MW boiler illustrate the effectiveness and feasibility of the proposed method for the practical applications. Lastly, it is worth mentioning that the proposed method can also be migrated to solve other prediction problems because it does not rely on mechanism analysis.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106070.

Acknowledgments

This project was funded by the National Key R&D Program of China (NO. 2016YFB060071).

We thank professor Jianping He and Yushan Li for advising on revision.

References

[1] N.E. Administration, Action plan 2014-2020 for transforming the coal fired power industry and upgrading energy conservation and emission reduction, 2014, http://zfxxgk.nea.gov.cn/auto84/201409/t20140919_1840.htm.

[2] O.V. Ogidiama, T. Shamim, Performance analysis of industrial selective catalytic reduction (SCR) systems, Energy Procedia 61 (2014) 2154–2157.

[3] T. Korpela, P. Kumpulainen, Y. Majanne, A. Häyrinen, P. Lautala, Indirect NOx emission monitoring in natural gas fired boilers, Control Eng. Pract. 65 (2017) 11–25.

[4] L. Liu, A. Wiliem, S. Chen, B.C. Lovell, Automatic image attribute selection for zero-shot learning of object categories, in: 2014 22nd International Conference on Pattern Recognition, IEEE, 2014, pp. 2619–2624.

[5] L. Liu, F. Nie, A. Wiliem, Z. Li, T. Zhang, B.C. Lovell, Multi-modal joint clustering with application for unsupervised attribute discovery, IEEE Trans. Image Process. 27 (9) (2018) 4345–4356.

[6] J. Smrekar, M. Assadi, M. Fast, I. Kuštrin, S. De, Development of artificial neural network model for a coal-fired boiler using real plant data, Energy 34 (2) (2009) 144–152.

[7] C. Li, C. Liu, X. Yu, K. Deng, T. Huang, L. Liu, Integrating demand response and renewable energy in wholesale market, in: IJCAI, 2018, pp. 382–388.
[8] D.L. Pla, R. Kamyar, N. Hashemian, H. Mehdizadeh, M. Moshgbar, Moisture soft sensor for batch fluid bed dryers: A practical approach, Powder Technol. 326 (2018) 69–77.
[9] S. Li, P. Wang, L. Goel, A novel wavelet-based ensemble method for short-term load forecasting with hybrid neural networks and feature selection, IEEE Trans. Power Syst. 31 (3) (2016) 1788–1798.
[10] J. Liu, T. Qin, T. Yang, Y. Lü, Variable selection method based on partial mutual information and its application in power plant SCR system modeling, Proc. CSEE 36 (2016) 2438–2443, http://dx.doi.org/10.13334/j.0258-8013.pcsee.2016.09.015.
[11] D. Jie, L. Xie, X. Fu, X. Rao, Y. Ying, Variable selection for partial least squares analysis of soluble solids content in watermelon using near-infrared diffuse transmission technique, J. Food Eng. 118 (4) (2013) 387–392.
[12] L. Breiman, Classification and regression trees, Routledge, 2017.
[13] H. Taghavifar, H. Taghavifar, A. Mardani, A. Mohebbi, S. Khalilarya, S. Jafarmadar, Appraisal of artificial neural networks to the emission analysis and prediction of co2, soot, and NOx of n-heptane fueled engine, J. Cleaner Prod. 112 (2016) 1729–1739.
[14] B. Liu, F. Yan, J. Hu, R.F. Turkson, F. Lin, Modeling and multi-objective optimization of NOx conversion efficiency and nh3 slip for a diesel engine, Sustainability 8 (5) (2016) 478.
[15] J.A. Suykens, J. De Brabanter, L. Lukas, J. Vandewalle, Weighted least squares support vector machines: robustness and sparse approximation, Neurocomputing 48 (1–4) (2002) 85–105.
[16] Y. Lv, J. Liu, T. Yang, D. Zeng, A novel least squares support vector machine ensemble model for nox emission prediction of a coal-fired boiler, Energy 55 (2013) 319–329.
[17] Y. Lv, T. Yang, J. Liu, An adaptive least squares support vector machine model with a novel update for NOx emission prediction, Chemometr. Intell. Lab. Syst. 145 (2015) 103–113.
[18] Y. Lv, F. Hong, T. Yang, F. Fang, J. Liu, A dynamic model for the bed temperature prediction of circulating fluidized bed boilers based on least squares support vector machine with real operational data, Energy 124 (2017) 284–294.
[19] Y. Gu, W. Zhao, Z. Wu, Online adaptive least squares support vector machine and its application in utility boiler combustion optimization systems, J. Process Control 21 (7) (2011) 1040–1048.
[20] H. Zhou, J.P. Zhao, L.G. Zheng, C.L. Wang, K.F. Cen, Modeling NOx emissions from coal-fired utility boilers using support vector regression with ant colony optimization, Eng. Appl. Artif. Intell. 25 (1) (2012) 147–158.
[21] P. Tan, J. Xia, C. Zhang, Q. Fang, G. Chen, Modeling and optimization of NOX emission in a coal-fired power plant using advanced machine learning methods, Energy Procedia 61 (2014) 377–380.
[22] T. Schreiber, Measuring information transfer, Phys. Rev. Lett. 85 (2) (2000) 461.
[23] M. Paluš, V. Komárek, Z. Hrnčíř, K. Štěrbová, Synchronization as adjustment of information rates: detection from bivariate time series, Phys. Rev. E 63 (4) (2001) 046211.
[24] Z. Wu, P. Shang, Nonlinear transformation on the transfer entropy of financial time series, Physica A 482 (2017) 392–400.
[25] W. Hu, J. Wang, T. Chen, S.L. Shah, Cause-effect analysis of industrial alarm variables using transfer entropies, Control Eng. Pract. 64 (2017) 205–214.
[26] D. Gibson, J. Kleinberg, P. Raghavan, Inferring web communities from link topology, in: Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia: Links, Objects, Time and Space—Structure in Hypermedia Systems: Links, Objects, Time and Space—Structure in Hypermedia Systems, ACM, 1998, pp. 225–234.
[27] K. Kanathey, R. Thakur, S. Jaloree, Ranking of web pages using aggregation of page rank and hits algorithm, Int. J. Adv. Stud. Comput. Sci. Eng. 7 (2) (2018) 17–22.
[28] P. Jönsson, L. Eklundh, TIMESAT—A program for analyzing time-series of satellite sensor data, Comput. Geosci. 30 (8) (2004) 833–845.
[29] J. Mendes, R. Araújo, F. Souza, Adaptive fuzzy identification and predictive control for industrial processes, Expert Syst. Appl. 40 (17) (2013) 6964–6975.
[30] M. Dios, J. Souto, J. Casares, Experimental development of CO2, SO2 and NOx emission factors for mixed lignite and subbituminous coal-fired power plant, Energy 53 (2013) 40–51.
[31] D. Wang, E. Duan, Y. Guo, B. Sun, T. Bai, Numerical simulation of the effect of over-fire air on NOx formation in furnace, in: Materials for Renewable Energy and Environment, ICMREE, 2013 International Conference on, vol. 3, IEEE, 2014, pp. 780–783.
[32] N. Horibe, T. Ishiyama, Relations among nox, pressure rise rate, HC and CO in LTC operation of a diesel engine, Tech. rep., SAE Technical Paper, 2009.