

Article

Not peer-reviewed version

Flue Gas Oxygen Content Model Based on Bayesian Optimization Main-Compensation Ensemble Algorithm in Municipal Solid Waste Incineration Process

Weiwei Yang , [Jian Tang](#) ^{*} , [Hao Tian](#) , [Tianzheng Wang](#)

Posted Date: 25 October 2024

doi: 10.20944/preprints202410.1999.v1

Keywords: Municipal solid waste incineration (MSWI); Flue gas oxygen content; Main-compensation tree ensemble model; Ensemble TS fuzzy regression tree (EnTSFRT); Long short term memory network (LSTM); Bayesian optimization (BO)



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

Flue Gas Oxygen Content Model Based on Bayesian Optimization Main-Compensation Ensemble Algorithm in Municipal Solid Waste Incineration Process

Weiwei Yang ^{1,2}, Jian Tang ^{1,2,*}, Hao Tian ^{1,2} and Tianzheng Wang ^{1,2}

¹ School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China

² Beijing Laboratory of Smart Environmental Protection, Beijing 100124, China

* Correspondence: freeflytang@bjut.edu.cn or freeflytang@126.com

Abstract: Flue gas oxygen content is one of the important parameters in municipal solid waste incineration (MSWI) process. And its stable control is closely related to the incineration efficiency and pollutant emission. The construction of a high-precision and interpretable flue gas oxygen content model is the basis for achieving its optimal control. However, the existing methods have problems such as poor interpretability, low model accuracy, and complex manual hyperparameter adjustment. In view of the above problems, the article proposes a flue gas oxygen content model based on Bayesian optimization (BO) main-complement ensemble algorithm. Firstly, the ensemble TS fuzzy regression tree (EnTSFRT) is used to construct the main model. Then, the long short term memory network (LSTM) is used to construct the compensation model with the error of the EnTSFRT model as the true value. The weighted value of the main and compensation models is used as the output. Finally, based on the BO algorithm, the hyperparameters of the constructed main-complement ensemble model are optimized to achieve high performance model. Experimental results based on real MSWI process data show that the proposed method has good performance.

Keywords: Municipal solid waste incineration (MSWI); Flue gas oxygen content; Main-compensation ensemble model; Ensemble TS fuzzy regression tree (EnTSFRT); Long short term memory network (LSTM); Bayesian optimization (BO)

1. Introduction

The total amount of municipal solid waste (MSW) in developing countries such as China has grown rapidly with the advancement of urbanization [1], resulting in the phenomenon of “garbage siege” in many cities [2]. To promote the goal of “carbon peak carbon neutralization” [3], the treatment of MSW is urgent. At present, MSW incineration (MSWI) technology has been widely used in the world because of its advantages of harmlessness, reduction and resource utilization [4–6]. However, it also emits pollutants. The use of artificial intelligence (AI) technology for pollution reduction research is one of the current hot spots.

In the MSWI process, flue gas oxygen content is one of the key controlled variables that can characterize the incineration efficiency and pollutant emissions [7]. When flue gas oxygen content is too low, the combustion efficiency of MSW will be greatly reduced. And many toxic and harmful gases such as dioxin (DXN) and sulfur dioxide (SO₂) will be produced [8]. When flue gas oxygen content is too high, it will increase the emission of combustion NO_x gas [Error! Bookmark not defined.]. Therefore, it is necessary to accurately control flue gas oxygen content. How to construct an effective flue gas oxygen content model is the basis for its accurate control, which is of great significance for the smooth, efficient and environmentally friendly operation of MSWI process.

At present, the modeling of flue gas oxygen content for MSWI process mainly includes mechanism modeling and data-driven modeling [9–10]. Due to the complex mechanism reaction and nonlinear dynamic characteristics, it is difficult to establish an accurate mathematical model of flue gas oxygen content [11]. In the context of AI and information construction, a large amount of industrial data containing mechanism knowledge can be obtained, which makes data-driven modeling methods receive more and more attention [12]. Data-driven modeling is mainly divided into two categories: neural network (NN)-based algorithm [13–14] and tree-based algorithm [15]. The former is weaker than the latter in terms of interpretability. For complex industrial process modeling, there are many reports based on NN modeling. Ma et al. [16] proposed a modeling method of flue gas oxygen content in power plants based on stacked target enhanced autoencoder and long short-term memory (LSTM). Qiao et al. [Error! Bookmark not defined.] proposed an improved LSTM (ILSTM) modeling method. In the references [17–19], a Takagi-Sugeno (TS) fuzzy neural network (TSFNN) controlled object modeling method based on multiple-input multiple output (MIMO) was proposed. Since NN models are composed of many neurons, the connection between neurons is different. With the increase of NN depth and width, the number of parameters increases rapidly, resulting in large memory consumption and resource consumption. Compared with NN, tree models have better generalization and interpretability in terms of small sample modeling. In researches of tree-based modeling, Tang et al. [20] proposed a deep forest regression modeling method based on cross-layer full connection for soft measurement of key process parameters such as product quality and environmental pollution indicators in complex industrial processes. Xia et al. [21] proposed the furnace temperature modeling of MSWI process based on linear regression decision tree (LRDT). However, the MSWI process is complex, and the data has certain uncertainty. And small changes in the input data may lead to completely different tree structures, which makes the model difficult to repeat and explain in some cases, and will reduce the modeling accuracy. Therefore, Xia et al. [22] proposed an ensemble T-S fuzzy regression tree (EnTSFRT) algorithm, introduced fuzzy reasoning [23] to deal with uncertainty, and illustrated the effectiveness of the EnTSFRT model through experiments.

In order to further improve the performance of the model, Yang et al. [24] proposed a dynamic model of a hybrid structure of selective catalytic reduction (SCR) denitrification system. Luo et al. [25] proposed an integrated modeling method that combines the mechanism model with the data-driven model to realize the roller kiln temperature modeling. Dong et al. [26] proposed a hybrid model based on mechanism modeling and data-driven modeling for the construction of leaching system models. The research showed that the compensation model of secondary modeling of the model error can effectively extract the effective information in the error [27]. This main-compensation model mechanism can adopt the same model or different models. For example, gradient boosting decision tree (GBDT) used the same decision tree to construct a multi-level error compensation model [28], and reference [29] adopted NN compensated linear master model. Recently, long short term memory network (LSTM) has been used to model pollutant emission concentration such as NO_x and CO in MSWI process with satisfactory performance [30,31]. At present, the main-compensation ensemble controlled object model for flue gas oxygen content has not been reported.

In addition, hyperparameter adjustment based on manual approach is complex and time-consuming in the modeling process, and it is easy to fall into local optimum. To solve this problem, Ma et al. [32] proposed the particle swarm optimization algorithm to optimize the multi-task NN model. Zhou et al. [33] proposed Bayesian optimization (BO)-based online robust parameter selection for sequential support vector regression. Deng et al. [34] used the improved sparrow search algorithm to optimize the squeeze casting process parameters intelligently.

In summary, the article proposes the BO-EnTSFRT-LSTM algorithm. The algorithm takes primary air volume, secondary air volume, feeder speed average, drying grate speed average and ammonia injection amount as the manipulated variables (MVs). At the same time, considering that there are many hyperparameters in the main-compensation ensemble model, BO is proposed to optimize the hyperparameters. Firstly, the main model is constructed based on EnTSFRT. Secondly,

the error of the EnTSFRT model is taken as the true value to construct the LSTM-based model. Finally, BO is used to optimize the hyperparameters.

The innovation of the article is reflected in: (1) The main modeling based on tree algorithm is proposed to improve the interpretability of the controlled object model. (2) A method of hyperparameters selection of the main-complement ensemble model based on BO algorithm is proposed. It reduces the complexity and time-consuming of manual hyperparameter adjustment approach and enhances the model performance. (3) A main-complement ensemble model based on BO optimization for flue gas oxygen content is constructed for the first time. It provides support for related intelligent control research.

The rest of the organizational structure of this article is arranged as follows. Section II describes the MSWI process for flue gas oxygen content , the overall modeling strategy and the implementation of the algorithm. In section III, the experimental research is carried out, and the experimental results are analyzed to verify the feasibility and superiority of the proposed method. Section IV summarizes and proposes future research directions.

2. Materials and Methods

2.1. MSWI Process Description for Key Controlled Variables

This study is based on an MSWI plant of typical grate furnace in Beijing. Its process flow is shown in Figure 1.

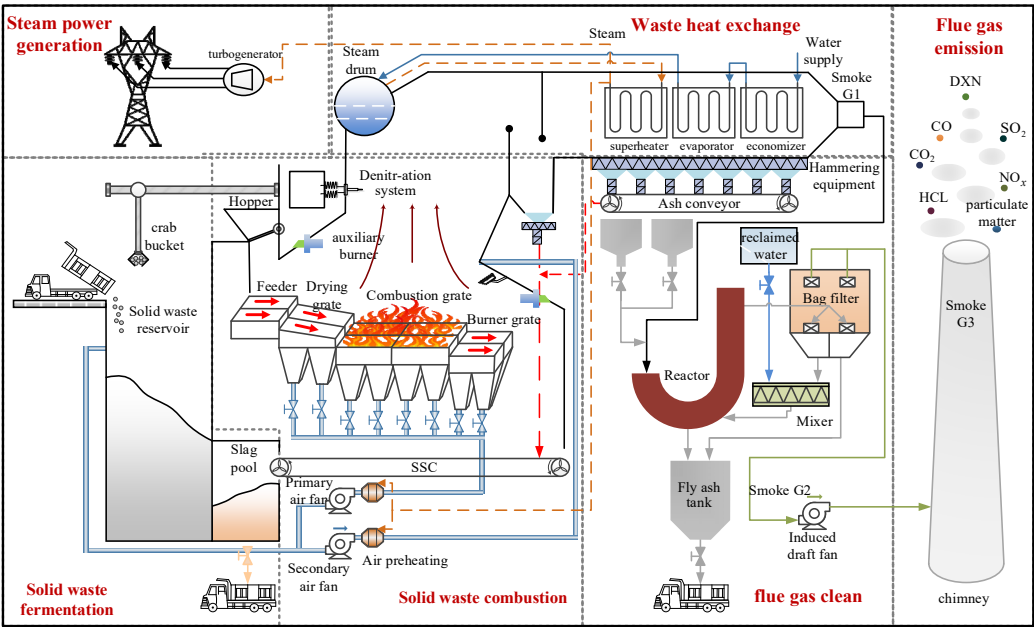


Figure 1. MSWI process flow of typical grate furnace.

Figure 1 shows that this MSWI process mainly includes six process stages: solid waste fermentation, solid waste combustion, waste heat exchange, flue gas cleaning, flue gas emission, and steam power generation [35–36]. The detailed description is as follows.

(1)Solid waste fermentation stage. The original MSW contains a large amount of water, which is not conducive to combustion. Therefore, it needs to be bio-fermented in a solid waste reservoir. This process can increase the calorific value of the solid portion by about 30% with about 5-7 days. Then, the fermented MSW transfers it to the hopper and pushes into the incinerator by the feeder and enters the solid waste combustion stage.

(2)Solid waste combustion stage. This stage is to convert MSW into high temperature under the coupling interaction of multi-phase such as solid-gas-liquid and multi-field such as heat flow force.

It can be divided into three phases: drying, combustion and burnout. The drying phase is realized based on the furnace's heat radiation phenomenon and the preheated primary air drying function. During combustion phase, MSW is ignited and produces flue gas. After about 1.5 hours of high-temperature combustion, the combustible components are fully burned and then enter the burnout phase. The non-combustible ash is pushed out of the furnace by the burnout grate.

(3) Waste heat exchange stage. Firstly, the high temperature flue gas is initially cooled by the water wall. Secondly, heat energy is transferred to the boiler by radiation and convection using equipment such as superheaters, evaporators and economizers. Then, the water in the boiler is converted into high-pressure superheated steam and enters the steam generation stage. Finally, the outlet flue gas temperature of the boiler is reduced to 200 °C. In this stage, the cooling rate needs to be strictly controlled to prevent the re-generation of pollutants.

(4) Steam power generation stage. Using the high temperature steam generated by the waste heat boiler. Converting mechanical energy into electrical energy to achieve self-sufficiency in plant-level electricity consumption and external supply of surplus electricity. At the same time, it also can realize resource utilization and obtain economic benefits.

(5) Flue gas cleaning stage. Firstly, the denitrification system is used to remove NO_x. Secondly, the acid gas is neutralized by semi-dry deacidification process. Then, activated carbon is used to adsorb DXN and heavy metals in flue gas. Finally, the particles, neutralizing reactants and activated carbon adsorbates in the flue gas are removed by the bag filter.

(6) Flue gas emission stage. The flue gas that meets the national emission standards is discharged into the atmosphere through the chimney through the induced draft fan.

The key controlled variables involved in the solid waste combustion stage are furnace temperature, flue gas oxygen content, steam flow and combustion line. This stage includes three processes: drying, combustion and burnout. In the combustion process, a strong oxidation reaction occurs, indicating that the combustible component and oxygen undergo a complete combustion reaction. Obviously, the oxygen content (air supply volume) and MSW feed rate (grate speed) in the combustion process are crucial to the combustion process. In the burnout process, under the action of high temperature and primary air, the coke will oxidize with oxygen. Therefore, maintaining the stability of flue gas oxygen content is of great significance for the stable operation of combustion process.

2.2. Flue Gas Oxygen Content Control Description

In the MSWI process, flue gas oxygen content is a characterization of the excess oxygen air coefficient. It can characterize the combustion state to a certain extent [Error! Bookmark not defined.]. When flue gas oxygen content is too low and the excess air ratio is too low, a large amount of toxic and harmful gases will be produced in addition to the increase of heat loss and the decrease of combustion efficiency [Error! Bookmark not defined.]. When flue gas oxygen content is too large and the excess air coefficient is too large, it shows that the amount of air is too large. The excess air can take away a lot of heat and dust, i.e., high flue gas oxygen content, also increase the emission of NO_x pollutants [Error! Bookmark not defined.].

Normally, the flue gas oxygen content at the outlet of the waste heat boiler is taken as one of the key controlled variables to reflect the MSWI process. Only when flue gas oxygen content is controlled within a certain range, it is possible to make the MSW and combustible flue gas in the furnace fully burned. Thus, the emission concentration of secondary pollutants can be ensured to meet the environmental standards. Therefore, it is of great significance to establish an accurate controlled object model of flue gas oxygen content to support the research of related intelligent control algorithms to realize the smooth, efficient and environmentally friendly operation of MSWI process [Error! Bookmark not defined.].

The MSWI process is complex, and there are many variables that affect the O₂ at the outlet of the waste heat boiler, and some of them are coupled. Therefore, first of all, the variables are preliminarily selected according to the expert experience [37]. The primary air volume, secondary air volume, feeder speed average, drying grate speed average and ammonia injection amount are selected as

MVs. In addition, because the mechanism model is difficult to construct, the construction of data-driven model is the current research hotspot. Without affecting the normal operation of the industrial site, collecting process data for modeling at the same time is the first problem to be solved in building the model.

2.3. Data Acquisition Devices Description and Experimental Data Analysis

Usually, in order to ensure the safe and reliable operation of the MSWI process, DCS systems with closed characteristics in industrial sites are strictly integrated with external devices for data collection and algorithm testing. To address this issue, we collected data using an edge verification platform of MSWI power plant equipped with a safety isolation acquisition device as shown in Figure 2.

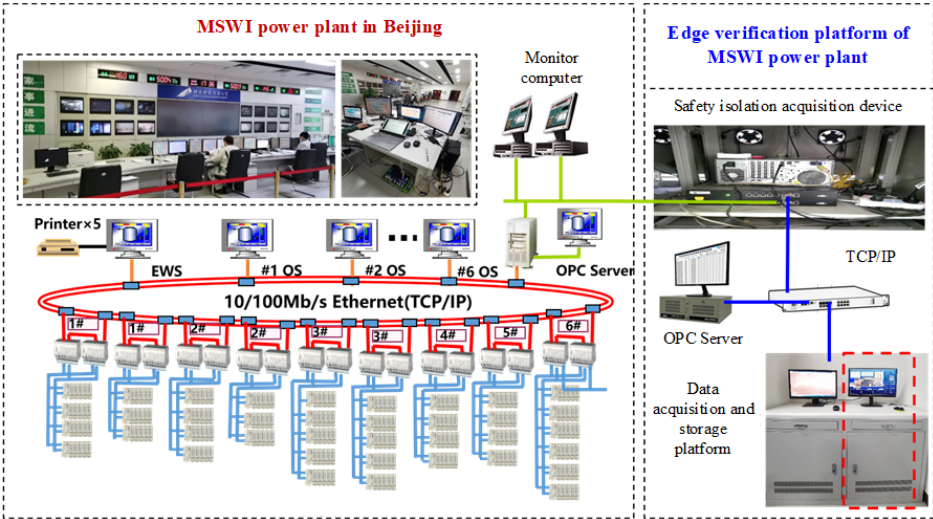


Figure 2. Structure of edge verification platform of MSWI power plant.

The device in Figure 1 facilitates one-way data transmission, mitigating interference issues during the data acquisition process. The industrial data for the experiment are obtained from an actual MSWI power plant in Beijing, China, with a collection frequency of 1 second. 16 hours of continuous process data from 8:00 to 24:00 on a certain day are used. The data are processed through moving averages to eliminate outliers, 857 samples were generated and used to build the FT controlled object model. Further, we divide these samples into training set and testing set in a ratio of 3:1.

There are five key MVs (primary air volume, secondary air volume, feeder speed average, drying grate speed average and ammonia injection amount) and one controlled variable (CV) (flue gas oxygen content). Pearson correlation coefficient (PCC) is used to evaluate the correlation between the key MVs and the CV, as shown in Figure 3.

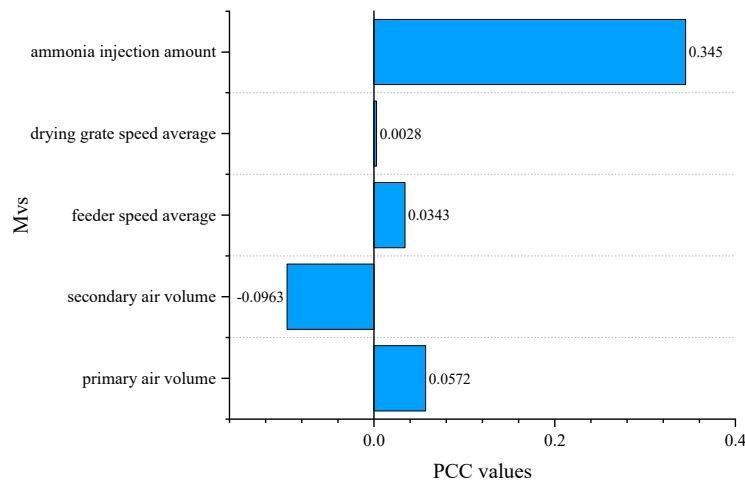


Figure 3. PCC values between MVs and key flue gas oxygen content.

Figure 3 shows that there is a strong linear relationship between ammonia injection amount and flux gas oxygen content for the modeling data used in this article. Obviously, the flux gas oxygen content of the actual MSWI process should have a stronger mapping relationship with other MVs. Since PCC can only measure the linear relationship between MVs and CVs, Figure 3 also indicates that there is a complex nonlinear or even uncertain mapping relationship between flux gas oxygen content and MVs. Therefore, it is necessary to adopt a modeling strategy that can characterize strong uncertainty and complex mapping relationships to construct a controlled object model for flux gas oxygen content.

2.4. Modeling Strategy

Based on the previous analysis, it can be concluded that constructing a flue gas oxygen content model for the MSWI process is the foundation for implementing stable control algorithms. In this article, an interpretable flue gas oxygen content based on BO-EnTSFRT-LSTM is proposed. In which, the main model is constructed by using EnTSFRT to model the uncertainty, the LSTM is used to construct the compensation model by using the main model error as the true value, and the hyperparameters of the constructed main-complement ensemble model are optimized based on the BO algorithm. The overall modeling strategy is shown in Figure 4.

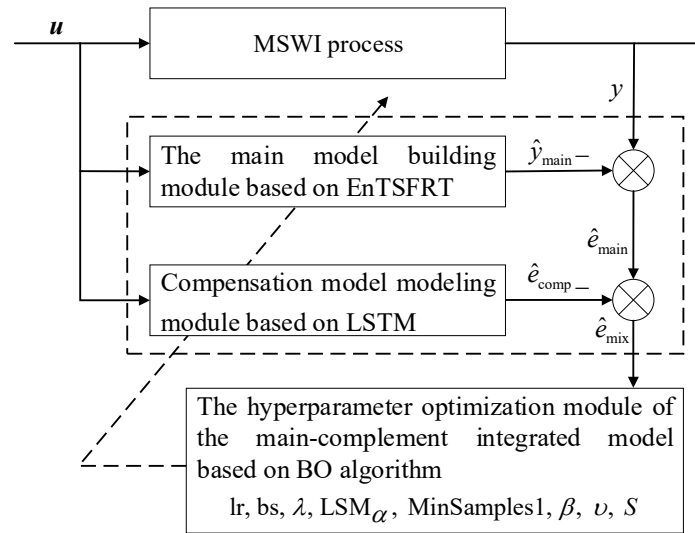


Figure 4. Modeling strategy.

In Figure 3, where u denotes MVs, y represents the true value of flue gas oxygen content, $\hat{y}_{main}$ represents the output value of EnTSFRT main model, $\hat{e}_{main}$ represents the error value of EnTSFRT main model, $\hat{e}_{comp}$ denotes the output value of LSTM compensation model, $\hat{e}_{mix}$ denotes the error value of LSTM compensation model, $\hat{y}$ represents the output value of BO-EnTSFRT-LSTM. The relationships among above the variables are as follow:

$$\hat{e}_{main} = y - \hat{y}_{main} \quad (1)$$

$$\hat{y} = \hat{y}_{main} + \hat{e}_{comp} \quad (2)$$

In addition, $lr, bs, \lambda, LSM_{\alpha}, MinSamples1, Tn, \beta, \nu$ and S in Figure 4 represent the hyperparameters of the main-complement ensemble model that the BO algorithm needs to optimize. They are learning rate, number of batches, the intermediate variable in the recursive calculation process, regularization coefficient, the minimum number of samples and the number of decision trees of the main model and the learning rate, regularization coefficient and maximum number of iterations of the compensation model, respectively.

The functions of different modules are as follows:

- 1) Main model construction module based on EnTSFRT: It is obtained by constructing multiple TSFRT sub-models with parallel fusion mode;
- 2) Compensation model construction module based on LSTM: It is used to construct the compensation model with the error true value of the EnTSFRT model;
- 3) Hyperparameter optimization module based on BO algorithm: The hyperparameters of the constructed main-complement ensemble model are optimized based on the BO algorithm.

2.5. Algorithm Implementation

2.5.1. Main Model Construction Module Based on EnTSFRT

The EnTSFRT model requires parallel fusion of multiple TSFRT sub-models. The TSFRT sub-model mainly includes a screening layer and a fuzzy inference layer. Among them, the screening layer is used for feature screening, and the fuzzy inference layer uses TS fuzzy inference.

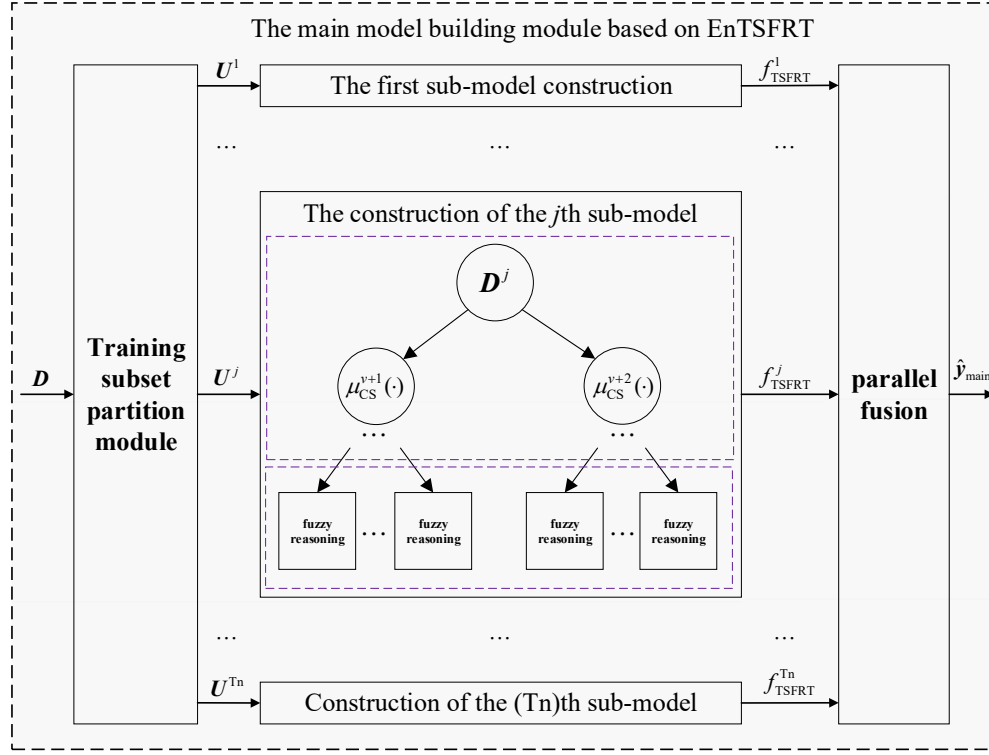


Figure 5. Struct of main model construction module based on EnTSFRT.

where $\mathbf{D} = \{\mathbf{u}_n, \mathbf{y}_n\}_{n=1}^N \in \mathbb{R}^{N \times M}$ denotes the input of EnTSFRT, $\mathbf{u}_n$ denotes MVs, n denotes the n th input sample, N represents the total number of data set samples, M represents dimension of the input feature, $\mathbf{U}^j$ represents the input variable matrix of the j th subset sample, $\mathbf{D}^j$ denotes the j th training subset obtained by random feature sampling of $\mathbf{D}$, $\mu_{CS}^{v+1}(\cdot)$ and $\mu_{CS}^{v+2}(\cdot)$ denote the membership degree on the non-leaf nodes of $v+1$ and $v+2$, f_{TSFRT} denotes the TSFRT sub-model.

2.5.1.1. Training Subsets Partition Submodule

The training subset $\mathbf{D}^j$ of M features is obtained by feature random sampling of $\mathbf{D}$. It can be expressed as:

$$\mathbf{D}^j = \{\mathbf{u}_n^j, \mathbf{y}_n^j\}_{n=1}^N = \{\mathbf{U}^j, \mathbf{y}^j\} \in \mathbb{R}^{N \times M} \quad (3)$$

where $\mathbf{U}^j = \{\mathbf{u}_n^j\}_{n=1}^N$ denotes the input variable matrix, $\mathbf{y}^j = \{\mathbf{y}_n^j\}_{n=1}^N$ represents the true value vector of flue gas oxygen content.

Correspondingly, from the perspective of input features, $\mathbf{u}^j$ with M input features can be expressed as:

$$\mathbf{u}^j = [u_1, u_2, \dots, u_M] \in \mathbb{R}^{1 \times M} \quad (4)$$

Therefore, a subset denoted as $\{\mathbf{D}^1, \dots, \mathbf{D}^j, \dots, \mathbf{D}^{Tn}\}$ can be obtained.

2.5.1.2. Sub-Model Construction Submodule

The construction of the j th TSFRT sub-model is described as an example.

The data set $\mathbf{D}^j$ is used as the input of the screening layer. And the best segmentation position is determined. Each recursive coordinate point (j, m) in $\mathbf{D}^j$ is obtained. And its mean square error (MSE) value is calculated to minimize MSE to determine the segmentation node. The estimation process of the loss function value is as follows:

$$\begin{aligned}\Phi &= \arg \min_{\phi} \left[f_{\text{MSE}}^j(\mathbf{D}_L^j) + f_{\text{MSE}}^j(\mathbf{D}_R^j) \right] \\ &= \arg \min_{\phi} \left[\left((\mathbf{y}_L^j - \mathcal{G}_{i_L}^j) \mu_{\text{cs}}^i(\mathbf{u}^j) \right)^2 + \left((\mathbf{y}_R^j - \mathcal{G}_{i_R}^j) \mu_{\text{cs}}^i(\mathbf{u}^j) \right)^2 \right]\end{aligned}\quad (5)$$

where Φ represents the loss value of data set $\mathbf{D}^j$, $f_{\text{MSE}}^j(\cdot)$ denotes the MSE value of the left subset $\mathbf{D}_L^j$ (or right subset $\mathbf{D}_R^j$), $\mathbf{y}_L^j$ denotes the true vector of the left subset $\mathbf{D}_L^j$, $\mathbf{y}_R^j$ denotes the true vector of the right subset $\mathbf{D}_R^j$, $\mu_{\text{cs}}^i(\mathbf{u}^j)$ denotes the clear membership function of the input $\mathbf{u}^j$ on the i th non-leaf node, $\mathcal{G}_{i_L}^j$ represents the average of the target values of the left subset $\mathbf{D}_L^j$, $\mathcal{G}_{i_R}^j$ represents the average of the target values of the right subset $\mathbf{D}_R^j$.

In this article, both $\mathcal{G}_{i_L}^j$ and $\mathcal{G}_{i_R}^j$ are represented as $\mathcal{G}_i^j$. Its calculation is as follows:

$$\mathcal{G}_i^j = \frac{1}{N_{\text{Subset}}^j} \sum_{n=1}^{N_{\text{Subset}}^j} y_n \quad (6)$$

where N_{Subset}^j denotes the number of $\mathbf{D}_L^j$ (or $\mathbf{D}_R^j$).

In Equation **Error! Reference source not found.**, $\mu_{\text{cs}}^i(\mathbf{u}^j)$ is calculated as:

$$\mu_{\text{cs}}^i(\mathbf{u}^j) = \begin{cases} 1 & \text{if } u_{j,m} \geq \phi_v \\ 0 & \text{if } u_{j,m} < \phi_v \end{cases}, v = 1, 2, \dots, \left(\frac{V}{2} - 1 \right) \quad (7)$$

where $\mu_{\text{cs}}^i(\mathbf{u}^j)$ denotes the clear membership function of the input $\mathbf{u}^j$, ϕ_v indicates that the segmentation threshold can be expressed as $\phi_v = u_{j,m}$. It is assumed that the TSFRT sub-model in this article consists of V nodes. So, the number of non-leaf nodes is $\frac{V}{2} - 1$, and the sum of membership sets is $\left\{ \mu_{\text{cs}}^v(\cdot) \right\}_{v=1}^{\frac{V}{2}-1}$.

$\mathbf{D}^j$ is divided into left and right subsets as follows:

$$\begin{cases} \mathbf{D}_L^j : \{ \mathbf{D}_L^j \in \mathbb{R}^{N_L \times M} \mid \mu_{\text{cs}}^1(\mathbf{u}) \equiv 1 \} \\ \mathbf{D}_R^j : \{ \mathbf{D}_R^j \in \mathbb{R}^{N_R \times M} \mid \mu_{\text{cs}}^1(\mathbf{u}) \equiv 0 \} \end{cases} \quad (8)$$

where $\mu_{\text{cs}}^1(\cdot)$ denotes the first membership degree of a clear set $C_{\text{cs}}^{N_{\text{leaf}}^j}$ obtained by using the minimum MSE. The first element of the clear set is defined by Equation **Error! Reference source not found.** The clear set is described as:

$$C_{cs}^{v_{leaf}^j} : \{\mu_{cs}^1(\mathbf{u})\} \quad (9)$$

Repeat the previous step until the minimum number of samples at the leaf node reaches the set threshold. The i th clear set can be expressed as:

$$C_{cs}^{v_{leaf}^j} : \left\{ \mu_{cs}^1(\mathbf{u}), \dots, \mu_{cs}^v(\mathbf{u}) \mid v \ll \left(\frac{V}{2} - 1 \right) \right\} \quad (10)$$

It can be further expressed as:

$$C_{cs}^{v_{leaf}^j} : \left\{ \phi_1, \dots, \phi_v \mid v \ll \left(\frac{V}{2} - 1 \right) \right\} \quad (11)$$

The training data $\mathbf{D}^{v_{leaf}^j}$ of the node is determined according to $C_{cs}^{v_{leaf}^j}$. And the T-S fuzzy inference is constructed. $\mathbf{D}^{v_{leaf}^j}$ denotes the following:

$$\mathbf{D}^{v_{leaf}^j} = \left\{ \mathbf{u}^j \subseteq C_{cs}^{v_{leaf}^j}, \mathbf{y}^j \right\}_{j=1}^{N_{v_{leaf}^j}^j} \in \mathbb{R}^{N_{v_{leaf}^j}^j \times v}, v \ll \left(\frac{V}{2} - 1 \right) \quad (12)$$

where $\mathbf{D}^{v_{leaf}^j}$ represents the training data of fuzzy reasoning, v_{leaf}^j denotes the v_{leaf}^j th leaf node, and $N_{v_{leaf}^j}^j$ denotes the number of samples in the v_{leaf}^j th leaf node.

In the fuzzy inference layer, K rules are defined to represent the local linear relationship between the input variable and the target. Define the k th rule as follows:

$$\begin{aligned} \text{Rule}^k : & \text{if } \phi_1 \text{ is } A_1^k \text{ and } \phi_l \text{ is } A_m^k, \\ & \text{then } y^k = g^k(u_1, \dots, u_m) \end{aligned} \quad (13)$$

where y^k represents the output of the consequent of the k th fuzzy rule, that is, the weight of the k th fuzzy rule. Further, we simplify Equation **Error! Reference source not found.** as follows:

$$\begin{aligned} \text{Rule}^k : & \text{if } u_1^{v_{leaf}^j} \text{ is } \mu_{A_1^k}^k(u_1^{v_{leaf}^j}) \dots \text{ and } u_v^{v_{leaf}^j} \text{ is } \mu_{A_m^k}^k(u_v^{v_{leaf}^j}), \\ & \text{then } y^k = g^k(u_1, \dots, u_m) \end{aligned} \quad (14)$$

where $u_1^{v_{leaf}^j}$ denotes a variable of $C_{cs}^{v_{leaf}^j}$, k denotes the membership degree of the k th fuzzy rule, $\mu_{A_1^k}^k(\cdot)$ denotes the membership function of A_1^k , and describes the membership of $u_1^{v_{leaf}^j}$ to A_1^k .

In the article, Gaussian function is used as membership function, as follows:

$$\mu_{A_1^k}^k(u_1^{v_{leaf}^j}) = \exp \left[-\frac{\left(u_1^{v_{leaf}^j} - \delta_{m,k} \right)^2}{\sigma_{m,k}^2} \right] \quad (15)$$

where $\delta_{m,k}$ denotes the center of the membership function, and $\sigma_{m,k}^2$ represents the width of membership function.

The activation intensity of the k th fuzzy rule of the fuzzy rule of m input variables is calculated as follows:

$$\alpha^k = \prod_{m=1}^M \mu_{A_m^k}^k \left(u_m^{v_{\text{leaf}}^j} \right) \quad (16)$$

where α^k denotes the product output of the k th fuzzy rule.

Further normalization operation is as follows:

$$\bar{\alpha}^k = \frac{\alpha^k}{\sum_{k=1}^K \alpha^k} \quad (17)$$

According to Equations **Error! Reference source not found.**–**Error! Reference source not found.**, y^k and $\bar{\alpha}^k$ are linearly combined and accumulated. The predicted output values $\hat{y}$ are obtained as follows:

$$\begin{aligned} \hat{y}_{\text{main}}^n &= \sum_{k=1}^K \bar{\alpha}^k g^k(u_n) \\ &= \sum_{k=1}^K \frac{\alpha^k}{\sum_{k=1}^K \alpha^k} (\omega_1^k u_1 + \omega_2^k u_2 + \cdots + \omega_m^k u_m) \end{aligned} \quad (18)$$

where $g(\cdot)$ represents the consequent output function, and ω denotes the consequent weight parameter.

Repeat the above steps until $\frac{V}{2}$ leaf nodes are obtained. By obtaining $\frac{V}{2}$ fuzzy inferences to construct a multi-input single-output model, the TSFRT simplified model of multi-input single-output can be obtained as follows:

$$\hat{\mathbf{y}}_{\text{main}}^j = f_{\text{TSFRT}}^j(\mathbf{U}, K, \theta_{\text{leaf}}, \boldsymbol{\omega}, \boldsymbol{\delta}, \boldsymbol{\ell}) \quad (19)$$

where f_{TSFRT}^j denotes the j th TSFRT sub-model, $\mathbf{U} = \{\mathbf{U}^1, \dots, \mathbf{U}^j, \dots, \mathbf{U}^{\text{Tn}}\}$, θ_{leaf} denotes the minimum number of samples, $\boldsymbol{\omega} = \{\omega_1, \omega_2, \dots, \omega_m\}$ denotes the consequent weight parameter matrix, $\boldsymbol{\delta}$ represents the center matrix of membership function, $\boldsymbol{\ell}$ represents the width matrix of membership function.

In addition, the article uses the least squares method to update the weight of the consequent. Equation **Error! Reference source not found.** can be merged as follows:

$$\hat{\mathbf{y}}_{\text{main}}^n = \omega_1^k \bar{\alpha}_1^k u_1 + \omega_2^k \bar{\alpha}_2^k u_2 + \cdots + \omega_m^k \bar{\alpha}_m^k u_m = \boldsymbol{\omega} \mathbf{u}^* \quad (20)$$

where $\mathbf{u}^* = [\bar{\alpha}_1^k u_1, \bar{\alpha}_2^k u_2, \dots, \bar{\alpha}_m^k u_m] \in \mathbb{R}^{1 \times m}$.

Given the input and output matrices, the T-S consequent weights can be estimated as follows:

$$\boldsymbol{\omega} = \left((\mathbf{U}^*)^T \mathbf{U}^* \right)^{-1} (\mathbf{U}^*)^T \mathbf{y} \quad (21)$$

where $\mathbf{U}^*$ is composed of $\mathbf{u}^*$.

The least square method is used for parameter learning. The recursive calculation is as follows:

$$\boldsymbol{\omega}_{l+1} = \boldsymbol{\omega}_l + H_{l+1} (\mathbf{u}_{l+1}^*)^T (y_{l+1} - \mathbf{u}_{l+1}^* \boldsymbol{\omega}_l) \quad (22)$$

$$H_{l+1} = H_l - \frac{H_l (\mathbf{u}_{l+1}^*)^T \mathbf{u}_{l+1}^* H_l}{1 + (\mathbf{u}_{l+1}^*)^T H_l \mathbf{u}_{l+1}^*} \quad (1) \quad (23)$$

where ω_0 is the weight of the initial time, which is a random vector; l denotes the l th recursive process; H_l is the intermediate variable in the recursive process; H_0 is defined by $H_0 = \gamma \mathbf{I}$, λ denotes a large positive value, and $\mathbf{I}$ represents the unit matrix.

2.5.1.3. Parallel Fusion Submodule

Firstly, according to the modeling process of TSFRT, total TSFRT models $\{f_{\text{TSFRT}}^j\}_{j=1}^{\text{Tn}}$ are constructed, which are defined as $\mathbf{F} \in \mathbb{R}^{N \times \text{Tn}}$.

Then, to estimate the weight with the minimum training error, the following optimal problem is used to calculate the pseudo-inverse:

$$\arg \min_{\mathbf{W}_{\text{LSM}}^{\text{Tn}}} : (\mathbf{F} \mathbf{W}_{\text{LSM}}^{\text{Tn}} - \mathbf{y})^2 + \text{LSM}_\alpha (\mathbf{W}_{\text{LSM}}^{\text{Tn}})^2 \quad (24)$$

where LSM_α denotes the regularization coefficient.

Using the ridge regression theory, that is, the Moore-Penrose inverse matrix, the weight matrix is calculated. The calculation results are as follows:

$$\mathbf{W}_{\text{LSM}}^{\text{Tn}} = \mathbf{F}^T (\text{LSM}_\alpha \mathbf{I} + \mathbf{F} \mathbf{F}^T)^{-1} \mathbf{y} \quad (25)$$

Finally, the output of EnTSFRT is as follows:

$$\hat{\mathbf{y}}_{\text{main}} = \mathbf{F} \mathbf{W}_{\text{LSM}}^{\text{Tn}} = \left[f_{\text{TSFRT}}^j (\mathbf{U}, K, \theta_{\text{leaf}}, \omega, \delta, \ell) \right]_{j=1}^{\text{Tn}} \mathbf{W}_{\text{LSM}}^{\text{Tn}} \quad (26)$$

2.5.1.4. Flow Chart of Main Model Modeling

The flow chart of the construction of the main model is as follows:

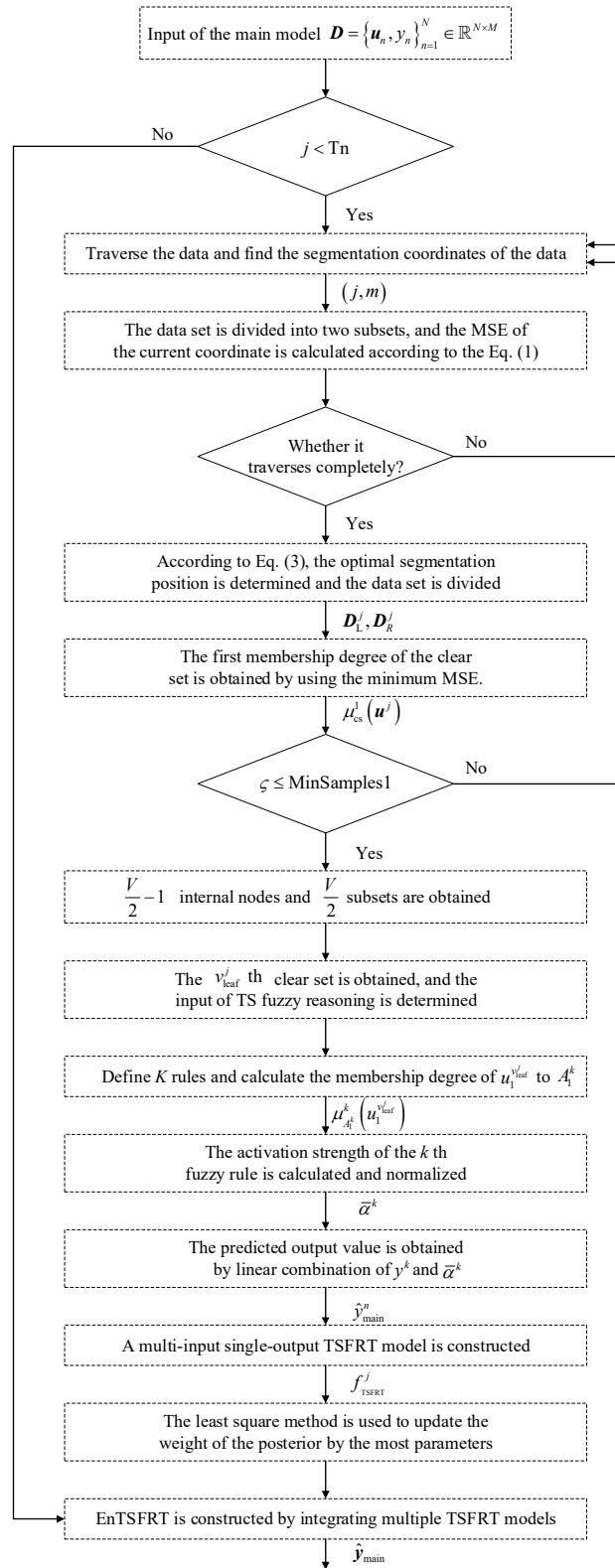


Figure 6. The flow chart of the construction of the main model.

2.5.2. Compensation Model Construction Module Based on LSTM

LSTM is a special recurrent neural network (RNN). The difference is that LSTM introduces three gates of forget gate, input gate and output gate and one cell state. The structure of LSTM is shown as follows :

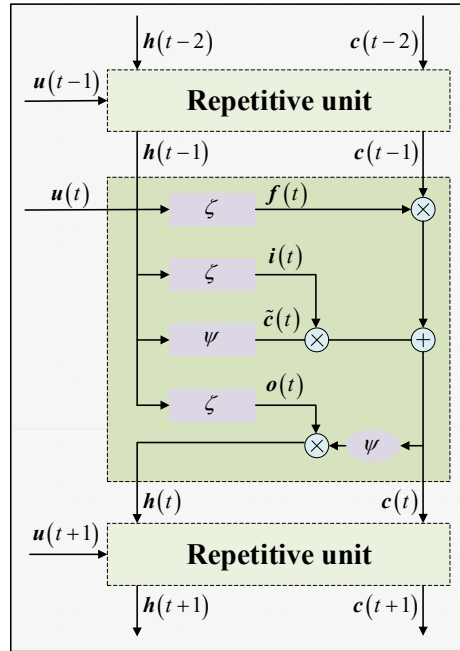


Figure 7. Structure of compensation model construction module based on LSTM.

where t denotes the current moment, $\mathbf{h}$ denotes the hidden state, $\mathbf{c}$ denotes the cell state, $\tilde{\mathbf{c}}$ denotes the candidate cell state, ζ represents the sigmoid function, ψ represents the tanh function, $\mathbf{f}$ denotes the value of the forget gate, $\mathbf{i}$ denotes the value of the input gate, $\mathbf{o}$ represents the value of the output gate, $\otimes$ represents element-by-element multiplication.

Its forward calculation and back propagation are as follows.

2.5.2.1. Forward Calculation Process

The output of forget gate, input gate, output gate, candidate cell state, cell state and hidden state can give the network memory ability of repetitive unit [Error! Bookmark not defined.]. The forward calculation process at time t are as follows.

Firstly, update the output of the forget gate as follows:

$$\mathbf{f}(t) = \zeta(\mathbf{W}_{uf}\mathbf{u}(t) + \mathbf{W}_{hf}\mathbf{h}(t-1) + \mathbf{b}_f) \quad (27)$$

where $\mathbf{W}$ represents the weight matrix, $\mathbf{b}$ represents the bias matrix.

Secondly, the output of the input gate is updated and new candidate cell state is generated, as follows:

$$\mathbf{i}(t) = \zeta(\mathbf{W}_{ui}\mathbf{u}(t) + \mathbf{W}_{hi}\mathbf{h}(t-1) + \mathbf{b}_i) \quad (28)$$

$$\tilde{\mathbf{c}}(t) = \psi(\mathbf{W}_{uc}\mathbf{u}(t) + \mathbf{W}_{hc}\mathbf{h}(t-1) + \mathbf{b}_{\tilde{c}}) \quad (29)$$

Further, the cell state is updated as follows:

$$\mathbf{c}(t) = \mathbf{c}(t-1) \odot \mathbf{f}(t) + \tilde{\mathbf{c}}(t) \odot \mathbf{i}(t) \quad (30)$$

Then, the output gate is updated and the cell state at the current moment is generated, as follows:

$$\mathbf{o}(t) = \zeta(\mathbf{W}_{uo}\mathbf{u}(t) + \mathbf{W}_{ho}\mathbf{h}(t-1) + \mathbf{b}_o) \quad (31)$$

$$\mathbf{h}(t) = \mathbf{o}(t) \odot \psi(\mathbf{c}(t)) \quad (32)$$

Finally, the output value $\hat{e}_{\text{comp}}(t)$ of LSTM is obtained as follows :

$$\hat{e}_{\text{comp}}(t) = \mathbf{W}_{out}\mathbf{h}(t) \quad (33)$$

where $\mathbf{W}_{out}$ is the output weight matrix.

2.5.2.2. Back Propagation Process

The loss function $J(t)$ in the training process of LSTM is defined as follows:

$$J(t) = \frac{1}{2} [\hat{e}_{\text{main}}(t) - \hat{e}_{\text{comp}}(t)]^2 \quad (34)$$

The weights and biases are updated based on the back propagation algorithm. Taking the update of $\mathbf{W}_{uf}$ as an example, as follows:

$$\mathbf{W}_{uf}(t+1) = \mathbf{W}_{uf}(t) - \beta \frac{\partial J(t)}{\partial \mathbf{W}_{uf}(t)} \quad (35)$$

Using the chain rule, $\frac{\partial J(t)}{\partial \mathbf{W}_{uf}}$ is calculated as follows:

$$\frac{\partial J(t)}{\partial \mathbf{W}_{uf}(t)} = \frac{\partial J(t)}{\partial \hat{e}_{\text{comp}}(t)} \frac{\partial \hat{e}_{\text{comp}}(t)}{\partial \mathbf{W}_{uf}(t)} = -[\hat{e}_{\text{main}}(t) - \hat{e}_{\text{comp}}(t)] \frac{\partial \hat{e}_{\text{comp}}(t)}{\partial \mathbf{W}_{uf}(t)} \quad (36)$$

The calculation of $\frac{\partial \hat{e}_{\text{comp}}(t)}{\partial \mathbf{W}_{uf}(t)}$ is as follows :

$$\begin{aligned} \frac{\partial \hat{e}_{\text{comp}}(t)}{\partial \mathbf{W}_{uf}(t)} &= \frac{\partial \hat{e}_{\text{comp}}(t)}{\partial \mathbf{h}(t)} \frac{\partial \mathbf{h}(t)}{\partial \mathbf{c}(t)} \frac{\partial \mathbf{c}(t)}{\partial \mathbf{f}(t)} \frac{\partial \mathbf{f}(t)}{\partial \mathbf{W}_{uf}(t)} \\ &= \mathbf{v}_{out} \mathbf{o}(t) \odot [\text{sech}^2 \mathbf{c}(t)] \mathbf{c}(t-1) \odot \mathbf{f}(t) [1 - \mathbf{f}(t)] \mathbf{u}(t) \end{aligned} \quad (37)$$

Further, $\mathbf{W}_{uf}(t+1)$ can be obtained.

The update process of the remaining weights and biases is the same as the update process of $\mathbf{W}_{uf}$.

2.5.2.3. Flow Chart of Compensation Model Modeling

The flow chart of the construction of the compensation model is as follows:

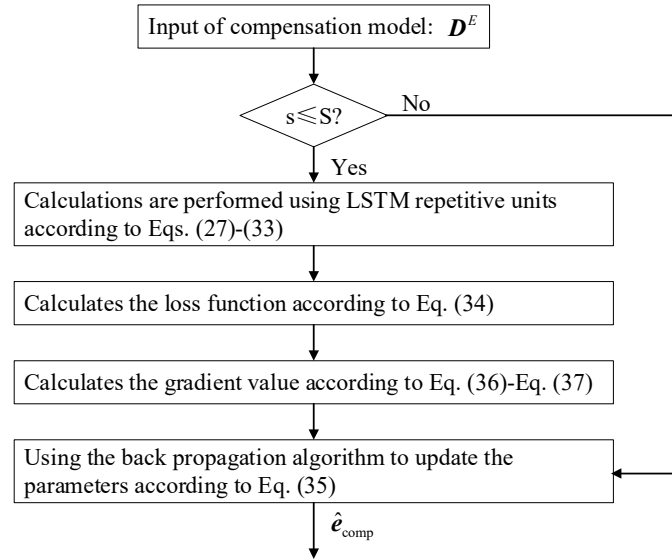


Figure 8. The flow chart of the construction of the compensation model.

where D^E is the input of the compensation model, denoted as $D^E = \{u(t), \hat{e}_{main}(t)\}_{t=1}^N$; and s represents the s th iteration number.

2.5.3. Hyperparameter Optimization Module Based on BO Algorithm

2.5.3.1. The Principle of BO Algorithm

BO is a kind of optimization method based on machine learning. Its purpose is to minimize the objective function. In the article, the eight hyperparameters in the main complement ensemble model are selected based on the BO algorithm to achieve the purpose of minimizing the objective function. The objective function selected in this article is the negative value of R-square (R^2) indicator.

The optimization process of BO algorithm is as follows.

Step 1: Set the boundary space

Set the value range of the hyperparameters $X = \{x_1; \dots; x_9\}$ of BO. It includes the upper boundary (UB) and the lower boundary (LB) of the sample space, which is set as follows:

$$\begin{cases} LB_1 < x_1 < UB_1 \\ \dots \\ LB_9 < x_9 < UB_9 \end{cases} \quad (38)$$

where $\{x_1; \dots; x_8\}$ denotes as follows:

$$\begin{cases} \mathbf{x}_1 \leftarrow \text{lr} \\ \mathbf{x}_2 \leftarrow \lambda \\ \mathbf{x}_3 \leftarrow \text{bs} \\ \mathbf{x}_4 \leftarrow \text{LSM}_\alpha \\ \mathbf{x}_5 \leftarrow \text{MinSamples1} \\ \mathbf{x}_6 \leftarrow \text{Tn} \\ \mathbf{x}_7 \leftarrow \beta \\ \mathbf{x}_8 \leftarrow \nu \\ \mathbf{x}_9 \leftarrow S \end{cases} \quad (39)$$

Step 2: Construct the initial data set

Randomly initialize P_{init} input samples $\mathbf{X} = \{\mathbf{x}_1; \dots; \mathbf{x}_{P_{\text{init}}}\}$. And observe the corresponding objective function value $\mathbf{\Gamma} = [\Gamma(\mathbf{x}_1), \dots, \Gamma(\mathbf{x}_{P_{\text{init}}})]^\top$ of the sample to form the initial data set.

Step 3: Construct / train the GPR model

The Gaussian process regression (GPR) model is constructed by using the data set. The GPR model uses the known data as a prior probability distribution that satisfies the multivariate normal distribution. It is described as follows:

$$\mathbf{\Gamma} \sim N(\xi_0, \Sigma_0) \quad (40)$$

where $\xi_0 = [\xi_0(\mathbf{x}_1), \dots, \xi_0(\mathbf{x}_p)]$ is the mean function, p represents the number of data set samples at the current time, and Σ_0 denotes the covariance matrix as follows:

$$\Sigma_0 = \begin{bmatrix} \Sigma_0(\mathbf{x}_1, \mathbf{x}_1) & \dots & \Sigma_0(\mathbf{x}_1, \mathbf{x}_p) \\ & \dots & \\ \Sigma_0(\mathbf{x}_p, \mathbf{x}_1) & \dots & \Sigma_0(\mathbf{x}_p, \mathbf{x}_p) \end{bmatrix} + \sigma^2 \mathbf{I} \in \mathbb{R}^{p \times p} \quad (41)$$

where σ^2 represents the variance of the measurement noise, $\Sigma_0(\mathbf{x}_a, \mathbf{x}_b)$ is the kernel function between the a th and b th sample points. It is calculated as follows:

$$\Sigma_0(\mathbf{x}_a, \mathbf{x}_b) = \sigma_\Gamma^2 \exp\left(-\frac{1}{2} \sum_{o=1}^d \frac{(x_{ao} - x_{bo})^2}{\eta_o^2}\right) \quad (42)$$

where x_{ao} represents the o th dimension feature of the a th sample, σ_Γ is the kernel amplitude of the kernel function, η_o denotes the length ratio of the o th dimension of the sample.

For a new sample point in the sample space, the posterior probability distribution of the sample point can be calculated by the prior probability distribution. According to the definition of GP, the joint Gaussian distribution $\mathbf{\Gamma}$ and the predicted value $\hat{\mathbf{\Gamma}}^*$ of the known data also obey the Gaussian distribution, which can be expressed as:

$$\begin{pmatrix} \mathbf{\Gamma} \\ \hat{\mathbf{\Gamma}}^* \end{pmatrix} \sim N\left(0, \begin{pmatrix} \Sigma_0 & \Sigma_0^* \\ (\Sigma_0^*)^\top & \Sigma_0^{**} \end{pmatrix}\right) \quad (43)$$

where $\Sigma_0^* = [\Sigma_0(\mathbf{x}_1, \mathbf{x}_1), \dots, \Sigma_0(\mathbf{x}_p, \mathbf{x}^*)]^T \in \mathbb{R}^{p \times 1}$ represents the covariance of training testing set, and $\Sigma_0^{**} = \Sigma_0(\mathbf{x}^*, \mathbf{x}^*) \in \mathbb{R}^{1 \times 1}$ denotes the covariance of the testing set.

Since the training set is known, the conditional distribution for Γ given $\hat{\Gamma}^*$ can be calculated based on Equation **Error! Reference source not found.** as follows:

$$\hat{\Gamma}^* | \Gamma \sim N(\xi_p, \sigma_p^2) \quad (44)$$

where ξ_p and σ_p^2 are as follows:

$$\begin{cases} \xi_p = (\Sigma_0^*)^T \Sigma_0^{-1} (\Gamma - \xi_0) + \xi_0(\mathbf{x}^*) \\ \sigma_p^2 = \Sigma_0^{**} - (\Sigma_0^*)^T \Sigma_0^{-1} \Sigma_0^* \end{cases} \quad (45)$$

where ξ_p is the posterior mean value, and σ_p^2 denotes the posterior variance.

Step 4: Estimate the parameters of the GPR model

We estimate the parameters in the mean function and covariance matrix. Maximum likelihood estimation (MLE) is used to estimate the above parameters, which improves the fitting and prediction performance of the model [38,39].

Step 5: Sampling

In order to find the optimal point $\mathbf{x}_{op}$ to minimize the objective function value $\Gamma(\mathbf{x}_{op})$, the expected improvement (EI) sampling function is used to sample the objective function.

Step 6: Update the data set

We calculate the objective function value $\Gamma(\mathbf{x}^*)$ corresponding to the new sample point $\mathbf{x}^*$.

Then, we update the data sets $\mathbf{X} = \{\mathbf{x}_1; \dots; \mathbf{x}_p; \mathbf{x}^*\}$ and $\Gamma = [\Gamma(\mathbf{x}_1), \dots, \Gamma(\mathbf{x}_p), \Gamma(\mathbf{x}^*)]^T$.

Repeat steps 3-6 until the preset maximum number of iterations is reached. And find the optimal sample point $\mathbf{x}_{op}$ that meet the requirements from the dataset.

2.5.3.2. The Flow Chart of BO Algorithm

The flow chart of BO algorithm is as follows:

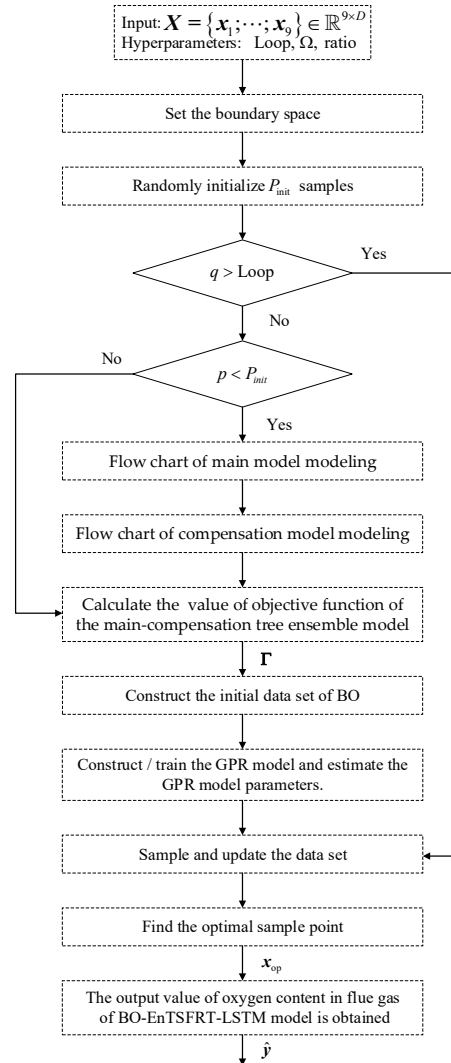


Figure 9. The flow chart of BO algorithm.

2.6. pseudo code

The pseudo code of the pseudocode algorithm used in this article is as follows:

where Loop represents the number of optimization iterations, Ω represents the sampling function, ratio represents the exploration ratio, ς is the current minimum number of samples.

Table 1 shows that the parameters of BO process are Loop, Ω , and ratio. Among them, BO establishes a probability model based on the past evaluation results of the objective function and finds the value that minimizes the objective function. The commonly used sampling functions include: expectation boosting, knowledge gradient, and entropy search and predictive entropy search. Its selection needs to consider how likely it is to obtain the optimal value at a certain point, and to evaluate whether the point can reduce the uncertainty of the Bayesian statistical model. The exploration ratio is a parameter that controls the balance between exploration and exploitation, and it needs to be adjusted according to specific optimization tasks and application scenarios.

Table 1. Pseudocode of BO-EnTSFRT-LSTM algorithm.

1	Input and hyperparameter settings: Data set $X = \{x_1; \dots; x_9\} \in \mathbb{R}^{9 \times D}$; Hyperparameters of BO algorithm: Loop, Ω , ratio; Hyperparameters of EnTSFRT: $lr, \lambda, bs, LSM_\alpha$, MinSamples1 and Tn; Hyperparameters of LSTM: β, ν and S Output: The output value $\hat{y}$ of flue gas oxygen content
2	Read the data, set the boundary space, and randomly initialize P_{init} samples;
3	If $q < \text{Loop}$
4	If $p < P_{init}$
5	Take $D = \{u_n, y_n\}_{n=1}^N \in \mathbb{R}^{N \times M}$ as the input of the EnTSFRT model;
6	If $j < \text{Tn}$
7	If $\varsigma \leq \text{MinSamples1}$
8	The data are traversed, and the optimal segmentation position is determined by minimizing MSE;
9	The data are divided into two subsets of D_L^j and D_R^j ;
10	Gets the first membership degree in a clear set $\mu_{cs}^1(u)$;
11	end
12	The internal nodes and $\frac{V}{2}$ subsets of $\frac{V}{2}-1$ TSFRT sub-models are obtained;
13	The clear set of v_{leaf}^j is obtained, and the input D_{leaf}^j of TS fuzzy inference is determined;
14	K rules are defined to represent the local linear relationship between the input variable and the target.
15	The membership degree $\mu_{A_i^k}^k(u_1^{v_{leaf}^j})$ of $u_1^{v_{leaf}^j}$ to A_i^k is calculated. The Gaussian function is selected as the membership function.
16	The activation intensity of the kth fuzzy rule is calculated, and the normalization operation is carried out to obtain $\bar{\alpha}^k$;
17	The predicted output value $\hat{y}_{main}^n$ is obtained by linear combination of y^k and $\bar{\alpha}^k$.
18	Construct a multi-input single-output TSFRT sub-model f_{TSFRT}^j ;
19	The most parameters of the least square method are used to update the weights of the consequents.
20	end
21	EnTSFRT is constructed by ensemble multiple TSFRT sub-models;
22	Calculate the error value of the EnTSFRT sub-model;
23	D^E is used as the input of LSTM model;
24	Initialize the parameters;
25	If $s \leq S$
26	Calculations are performed using LSTM repetitive units according to Equations Error! Reference source not found.-Error! Reference source not found. ;
27	Calculates the loss function according to Equation Error! Reference source not found. ;
28	Calculates the gradient value according to Equation Error! Reference source not found.-Equation Error! Reference source not found. ;
29	Using the back propagation algorithm to update the parameters according to Equation Error! Reference source not found. ;
30	end
31	end
32	Construct BO initial data set;
33	Construction / training of GPR model;
34	Sample and update the data set;
35	Find the optimal sample point;

33 end

34 The output value $\hat{y}$ of flue gas oxygen content is obtained.

3. Results and Discussion

3.1. Performance Metrics

The evaluation indexes used in the article are root mean squard error (RMSE), mean absolute error (MAE) and R-Square (R^2), as follows:

$$RMSE = \sqrt{\frac{\sum_{n=1}^N (y_n - \hat{y}_n)^2}{N}} \quad (46)$$

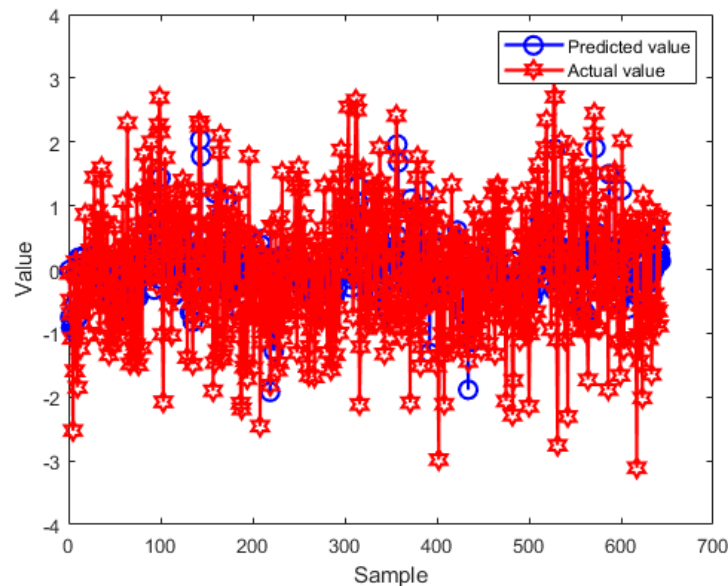
$$MAE = \frac{1}{N} \sum_{n=1}^N |\hat{y}_n - y_n| \quad (47)$$

$$R^2 = 1 - \frac{\sum_n (\hat{y}_n - y_n)^2}{\sum_n (\bar{y}_n - y_n)^2} \quad (48)$$

where RMSE represents the standard deviation of the sample for the difference between the predicted and true values, reflecting the degree of dispersion of the sample. MAE represents the average absolute error between the predicted and true values. And when doing nonlinear fitting, the smaller the RMSE, the better[40]. R^2 is used to quantify the fitting degree of the model to the data. Generally speaking, the closer the value of R^2 is to 1, the better the fitting degree of the model is.

3.2. Experimental Results

The parameters of BO are set based on Thumb rule. The detaied values are follows: Loop = 20 , Ω = EI function, and ratio = 0.5. The optimal hyperparameters are selected as lr = 0.1829 , λ = 23 , bs = 21 , LSM_α = 0.6714 , MinSamples1 = 42 , Tn = 23 , β = 0.0263 , ν = 0.1923 and S = 345 . The fitting curves of the training set and testing set of the main-compensation ensemble model are shown in Figures 10 and 11.



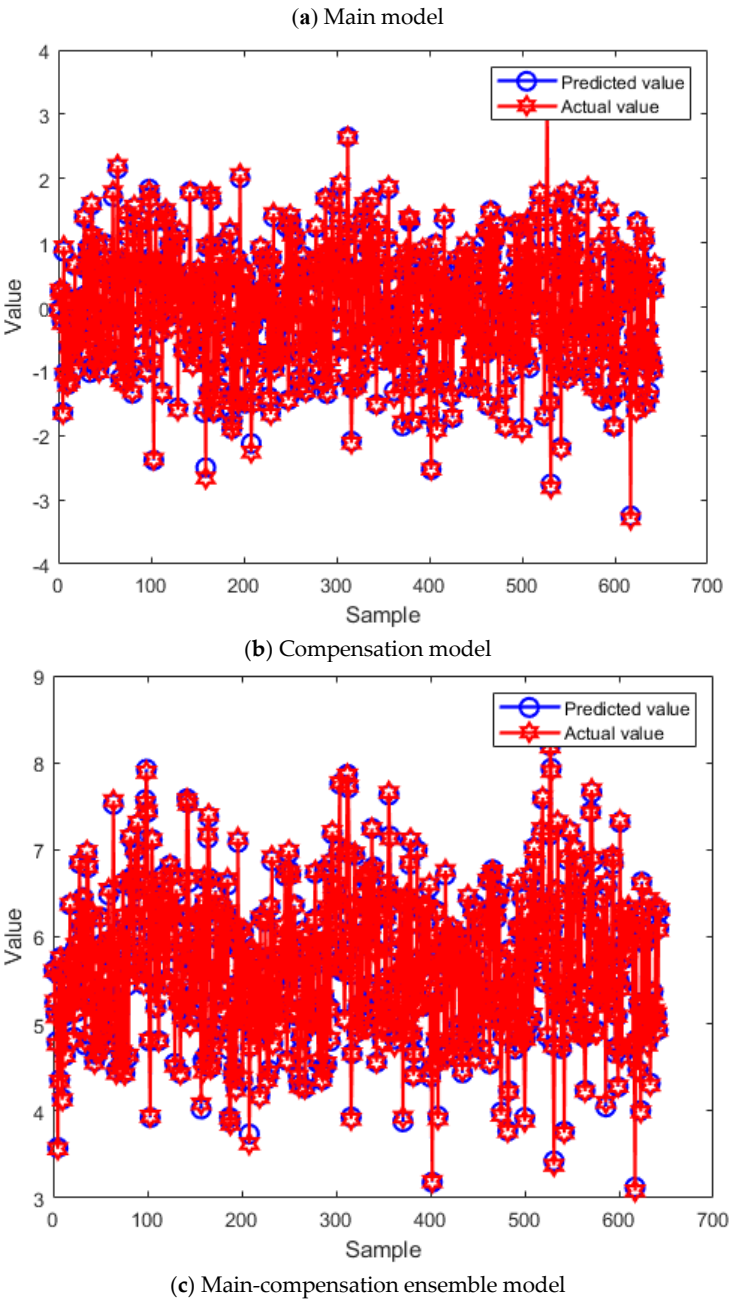
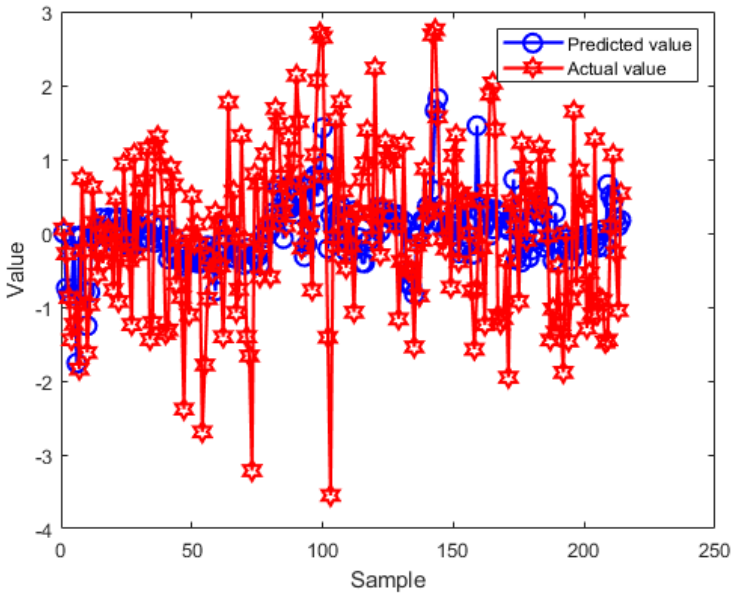
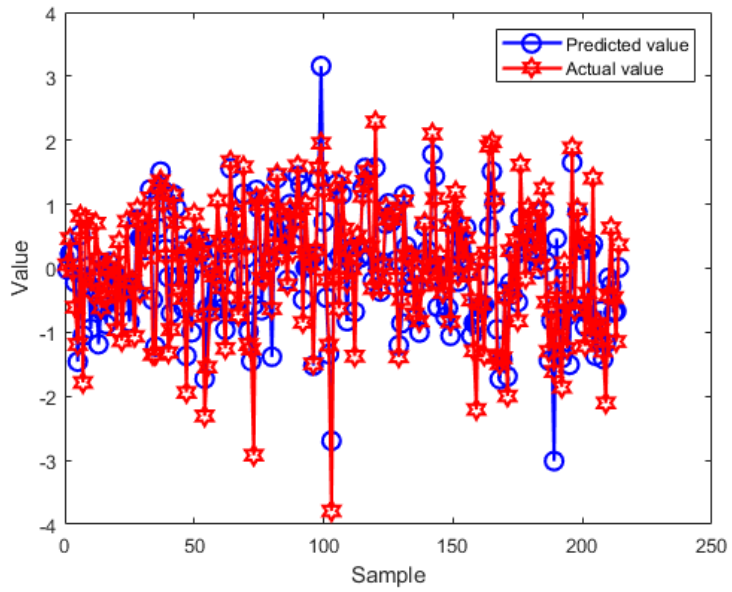


Figure 10. Fitting curves of training set.



(a) Main model



(b) Compensation model

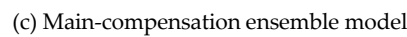


Figure 11. The fitting curves of the testing set.(进行了修改).

According to Figures 10-11, only using the EnTSFRT-based main model, the predicted value cannot fit the actual value effectively. The compensation model has a better fitting effect on the error value of the main model. At the same time, by comparing Figures 10(a)-11(a) with Figures 10(c)-11(c), it can be seen that the fitting effect of the proposed method on the training set and the testing set has a great improvement.

The weights of all trees constructed by the main model are shown in Figure 12:



The curve of the objective function in the BO optimization process is as follow:

Figure 13 shows that the value of the objective function is in the range of -0.5079 to -0.7988 in the BO optimization process.

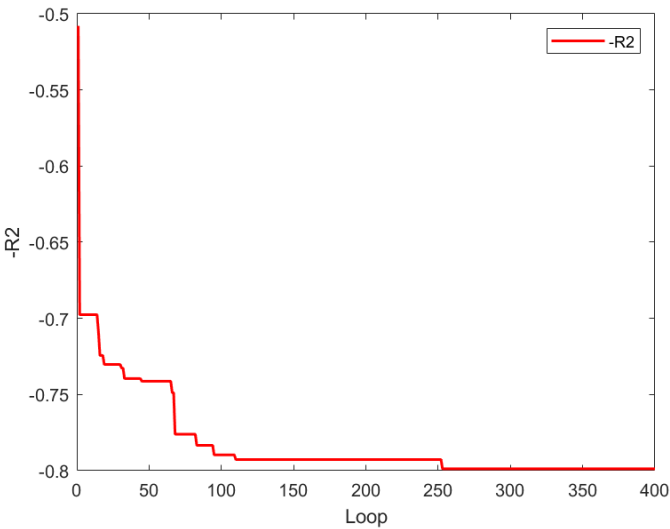
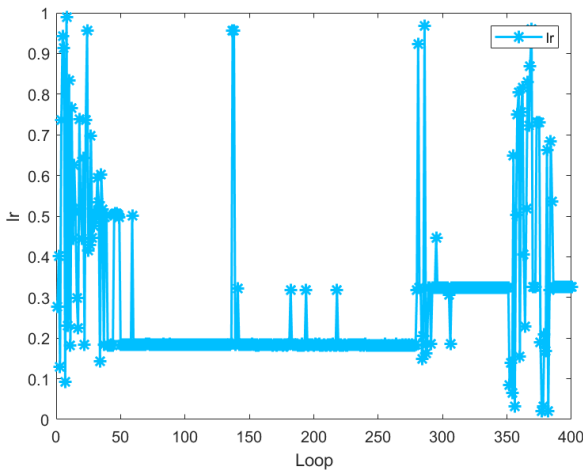


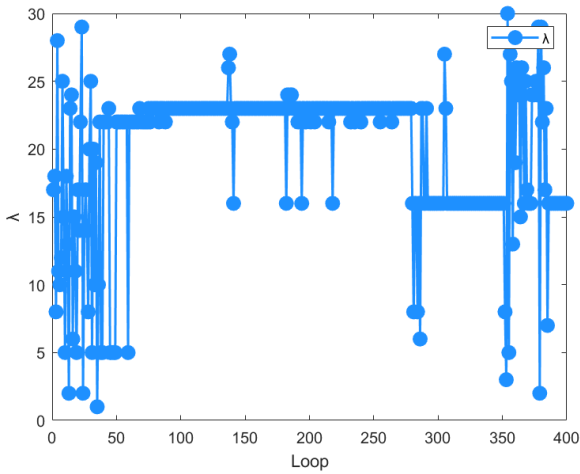
Figure 13. The curve of the objective function in the BO optimization process.

The variation curves of the 9 hyperparameters in the BO optimization process are as follows:

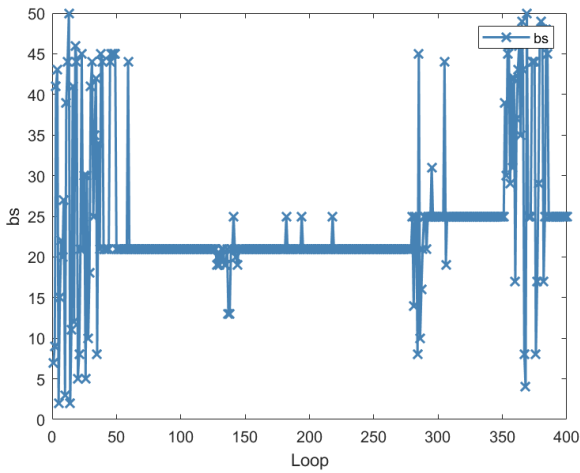
Figure 14 shows the change process of hyperparameters of the main-complement ensemble model in the BO optimization process. The specific detailed analysis is as follow:



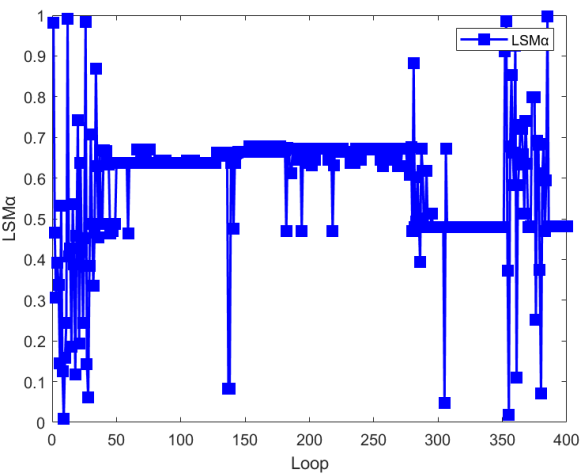
(a)



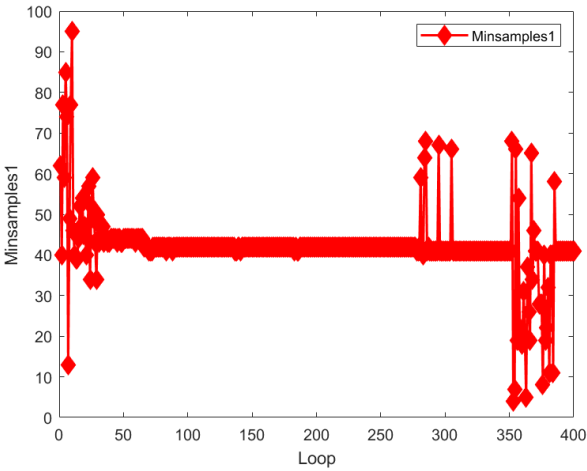
(b)



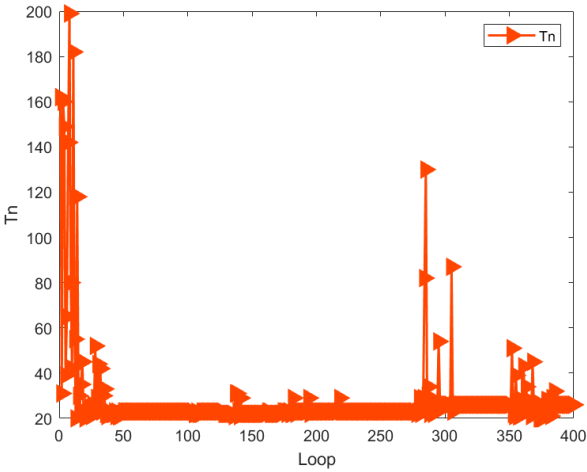
(c)



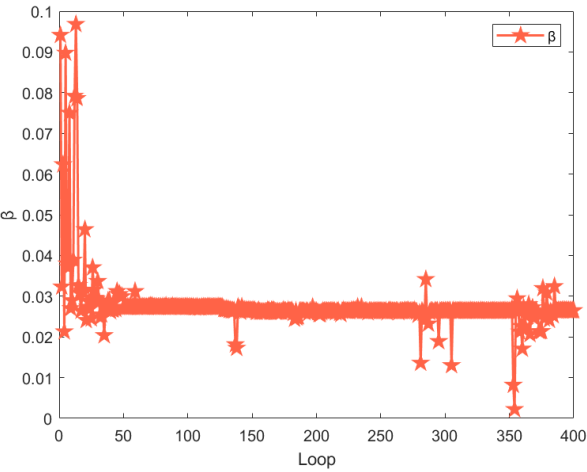
(d)



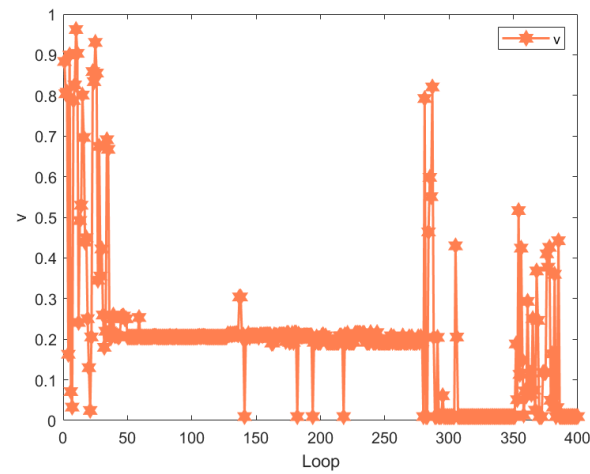
(e)



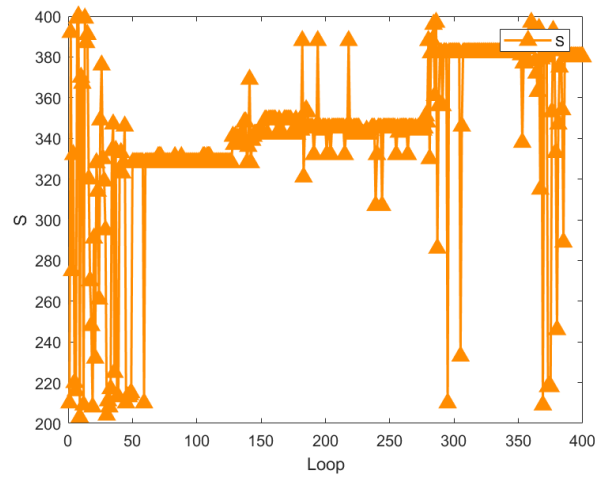
(f)



(g)



(h)



(i)

Figure 14. The variation curves of the nine hyperparameters in the BO optimization process.

1) Figure 14(a) shows the change curves of lr , which is the learning rate of the main model. When the range of $Loop$ is 0 to 50, the change of lr is dramatic. When the range of $Loop$ is 50 to 280, the value of lr is basically stable at 0.18. When the $Loop$ is greater than 280, it first stabilizes at about 0.3. Then changes sharply and finally tends to be stable at about 0.18.

2) Figure 14(b) shows the change curves of λ , which is the intermediate variable in the recursive calculation process of the main model. With the increase of $Loop$, its change trend is similar to that of lr . Its value is finally stabilized at about 16.

3) Figure 14(c) shows the change curves of bs , which is the the number of batches of the main model training. With the increase of $Loop$, its change trend is similar to that of lr . Its value is finally stabilized at about 25.

4) Figure 14(d) shows the change curves of LSM_a , which is the regularization coefficient of the master model. With the increase of $Loop$, its change trend is similar to that of lr . Its value is finally stabilized at about 0.48.

5) Figure 14(e) shows the change curves of $MinSamples1$, which is the minimum number of leaf node samples of the main model. With the increase of $Loop$, the value of $MinSamples1$ first changed dramatically. Then its value is basically stable at about 41.

6) Figure 14(f) shows the change curves of Tn , which is the number of decision trees of the main model. When $Loop$ is 0 to 20, Tn changes drastically. When $Loop$ is greater than 20, the value of Tn is stable at about 23.

7) Figure 14(g) shows the change curves of β , which is the learning rate of the compensation model. With the increase of $Loop$, the change trend of its value is similar to that of Tn . Its value eventually converges to about 0.026.

8) Figure 14(h) shows the change curves of ν , which is the regularization coefficient of the compensation model. When $Loop$ is in the range of 0 to 40, its value fluctuates greatly. Then, its value converges to about 0.2. When $Loop$ is greater than 280, the value is mostly around 0.01.

9) Figure 14(i) shows the change curves of S , which is the iteration times of the compensation model. When $Loop$ is in the range of 0 to 50, its value changes dramatically. When $Loop$ is greater than 50, its values converge to 328, 345 and 382 respectively.

3.3. Comparison and Discussion

To further illustrate the effectiveness, the proposed method is compared with tree-based method such as random forest (RF), LRDT, EnTSFRT-LRDT and BO-EnTSFRT. The hyperparameters of each model are set as follows. In RF model, the minimum number of samples is set to 50, and the number of features is 4. In LRDT model, the minimum number of samples is set to 20, and the regularization coefficient is 0.4. In EnTSFRT-LRDT model, $lr = 0.1$, $\lambda = 10$, $bs = 18$, $LSM_a = 0.3$, $MinSamples1 = 30$, $Tn = 160$, $MinSamples2 = 35$, and $\beta = 0.3$. In EnTSFRT-LSTM model, $lr = 0.1$, $\lambda = 15$, $bs = 18$, $LSM_a = 0.3$, $MinSamples1 = 55$, $Tn = 80$, $\beta = 0.05$, $\nu = 0.3$, $S = 320$. In BO-EnTSFRT model, $lr = 0.8554$, $\lambda = 18$, $bs = 41$, $LSM_a = 0.919$, $MinSamples1 = 45$, and $Tn = 200$. In BO-EnTSFRT-LRDT model, $lr = 0.024$, $\lambda = 6$, $bs = 48$, $LSM_a = 0.6123$, $MinSamples1 = 77$, $Tn = 184$, $MinSamples2 = 22$, and $\beta = 0.1346$, where $MinSamples2$ is the minimum sample size of LRDT model.

Based on the above models, the test is performed on the testing set, and repeated 20 times. One of the fitting curves and total statistical results are as follows.

It can be seen from Figure 15 that the fitting effect of the proposed method is the best based on different methods for the testing set. At the same time, it shows that the overall performance of the proposed method is the best one.

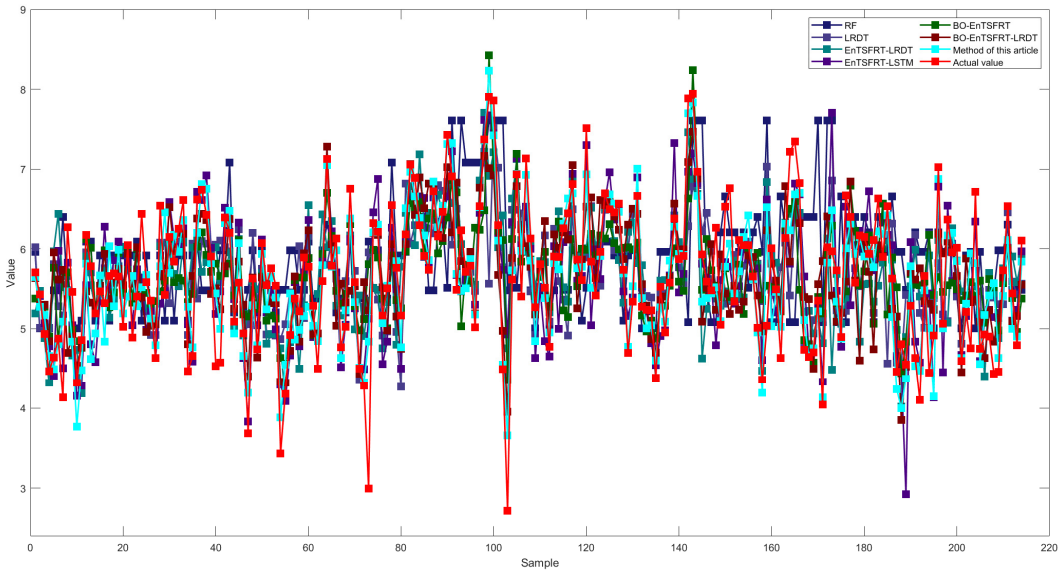


Figure 15. Fitting curves of different methods on the testing set.

The calculation results of performance evaluation indexes of different methods are as follows.

Table 2 show the statistical results of performance evaluation indexes of different methods in terms of training set and testing set with 20 repeated times. The proposed BO-EnTSFRT-LSTM model has the lowest average RMSE and MAE for the training set (2.6610E-02, 2.0684E-02) and the testing set (4.3991E-01, 3.1771E-01). At the same time, the proposed method has the highest R² on the training set (9.9896E-01) and the testing set (7.5649E-01±2.4804E-04). By comparing the method proposed in this article with EnTSFRT-LSTM and BO-ENTSFRT, the important role of BO algorithm and LSTM-based compensation model is illustrated respectively. In addition, the comparison between the proposed method and the BO-EnTSFRT-LRDT further illustrates the superiority of the LSTM algorithm. The proposed BO-EnTSFRT-LRDT method utilizes EnTSFRT to handle uncertainty in industrial modeling data, fits error data using LSTM, and avoids hyperparameters that are prone to overfitting using BO, resulting in optimal modeling performance. By comparing with other methods, the superiority of the proposed method is proved.

Table 2. Statistical results of performance evaluation indexes of different methods.

Dataset	Method	RMSE	MAE	R2
Training set	RF	8.6432E-01±1.1677E-31	6.4993E-01±1.2975E-32	-8.8061E-02±8.1092E-34
	LRDT	5.2664E-01±5.1899E-32	4.0872E-01±1.2975E-32	5.9604E-01±1.2975E-32
	EnTSFRT-LRDT	4.4932E-01±1.1640E-04	3.5266E-01±8.4882E-05	7.0088E-01±2.0645E-04
	EnTSFRT-LSTM	4.5666E-02±4.1522E-04	3.3945E-02±1.6713E-04	9.9639E-01±1.7202E-05
	BO-EnTSFRT	6.4563E-01±9.0752E-05	5.1347E-01±6.8136E-05	5.8308E-01±1.5481E-04
	BO-EnTSFRT-LRDT	4.3465E-01±2.9193E-32	3.4153E-01±1.2975E-32	7.2484E-01±5.1899E-32
Testing set	BO-EnTSFRT-LSTM	2.6610E-02±5.8122E-06	2.0684E-02±3.3649E-06	9.9896E-01±3.8281E-08
	RF	9.7815E-01±5.1899E-32	7.4468E-01±5.1899E-32	-2.0272E-01±1.2975E-32
	LRDT	7.6825E-01±0.0000E+00	5.9607E-01±5.1899E-32	2.5808E-01±0.0000E+00
	EnTSFRT-LRDT	7.3435E-01±3.5825E-04	5.7583E-01±1.6158E-04	2.4216E-01±1.4912E-03
	EnTSFRT-LSTM	5.7702E-01±1.0583E-03	4.2354E-01±6.3545E-04	5.8020E-01±2.1789E-03
	BO-EnTSFRT	8.5004E-01±5.7774E-04	6.7746E-01±2.1606E-04	3.1419E-01±1.4888E-03

BO-EnTSFRT-LRDT	6.4225E-01±1.2975E-32	4.9011E-01±5.1899E-32	4.8149E-01±2.9193E-32
BO-EnTSFRT-LSTM	4.3991E-01±2.0766E-04	3.1771E-01±1.1515E-04	7.5649E-01±2.4804E-04

Through the above comparative experiments and causal analysis, it indicates that the proposed method has the best performance compared to BO-EnTSFRT-LSTM-based modeling methods used in this study.

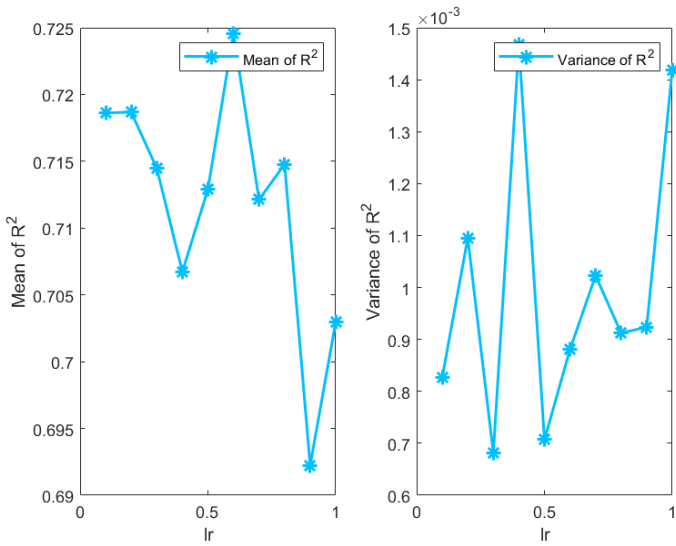
3.4. Hyperparameter Discussion

In the article, the single factor sensitivity analysis of the hyperparameters optimized by BO algorithm is carried out. Since the objective function of the BO algorithm is the negative value of the R^2 of the ensemble model, the hyperparameters are changed according to a certain step size in the range of hyperparameters and negative value of R^2 of the testing set are recorded for analysis. In the article, a total of 10 sets of experiments are set for each hyperparameter. The interval setting is as shown in Table 3:

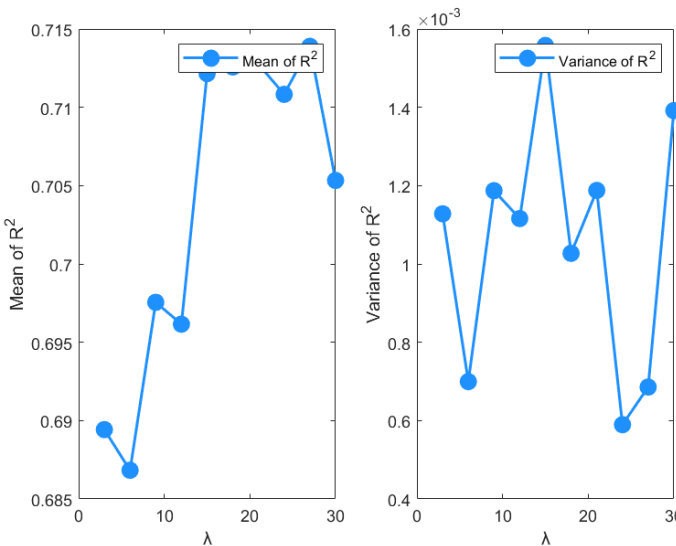
Table 3. Interval setting of hyperparameters of the main-complement ensemble model.

Model	Hyper Parameter	Range
Main Model	lr	[0,1]
	λ	[1, 30]
	bs	[1, 50]
	LSM_a	[0, 1]
	$Minsamplel$	[1, 100]
	Tn	[20, 200]
Compensation Model	β	[0, 0.1]
	ν	[0, 1)
	S	[200, 400]

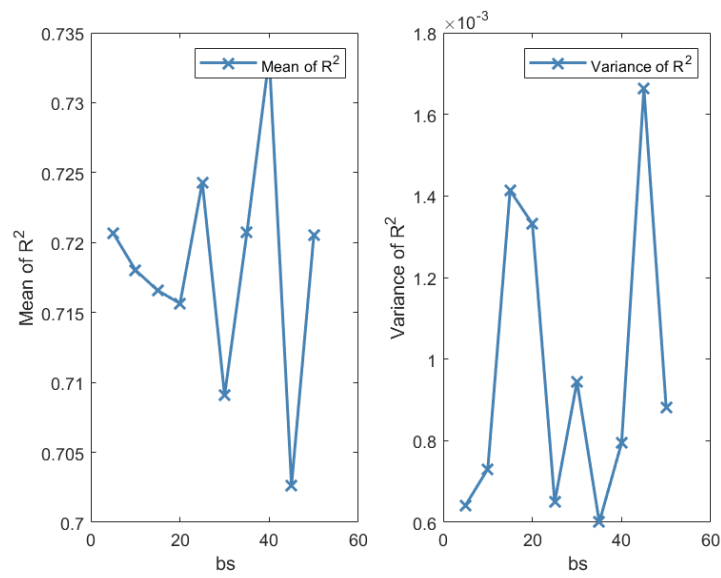
The selection of step size should consider the sensitivity of hyperparameters and the limitation of computing resources. In this article, the experiment is carried out in a fixed step size according to the setting interval of the hyperparameters based on author’s debug experience and PC hardware performance. For lr of the main model, the initial value is set to 0.1, and the step size is 0.1; the initial value of λ is 3 and the step size is 3; the initial value of bs is 5 and the step size is 5; the initial value of LSM_a is set to 0.1, and the step size is 0.1; the initial value of $MinSamplesl$ is 10 and the step size is 10; the initial value of Tn is 20, and the step size is 20. For the β of the compensation model, the initial value is 0.01 and the step size is 0.01; the initial value of ν is 0.05, and the step size is 0.1; the initial value of S is 220, and the step size is 400. In the experiment, the single factor analysis strategy is used. By performing 20 repeated experiments, the mean and variance change curves of RMSE on the testing set are shown in Figure 16.



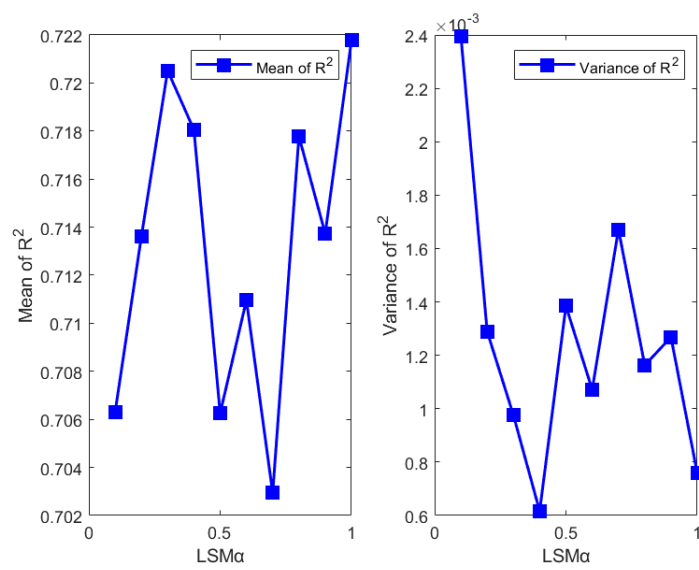
(a)



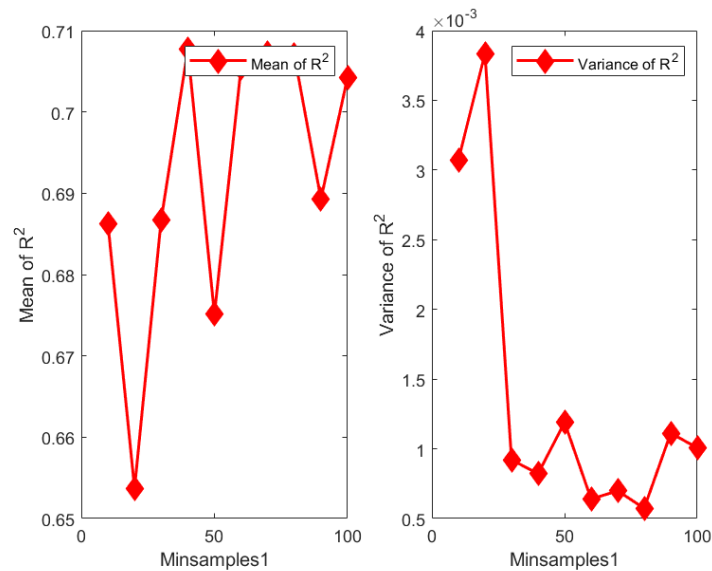
(b)



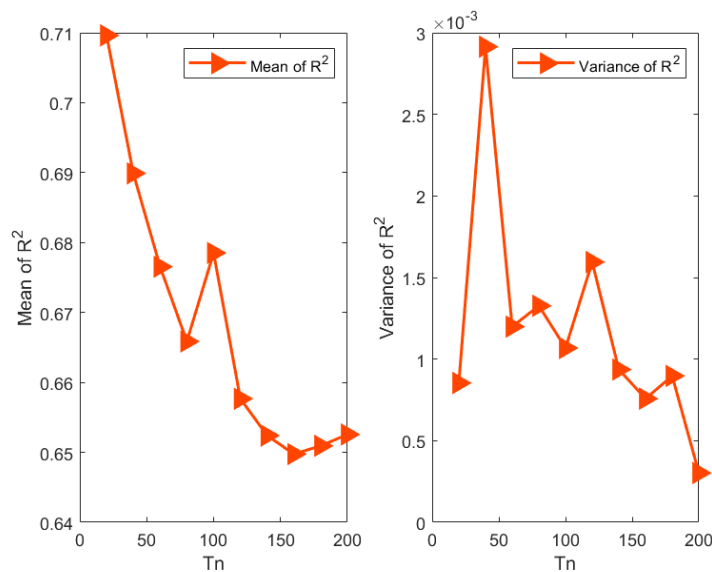
(c)



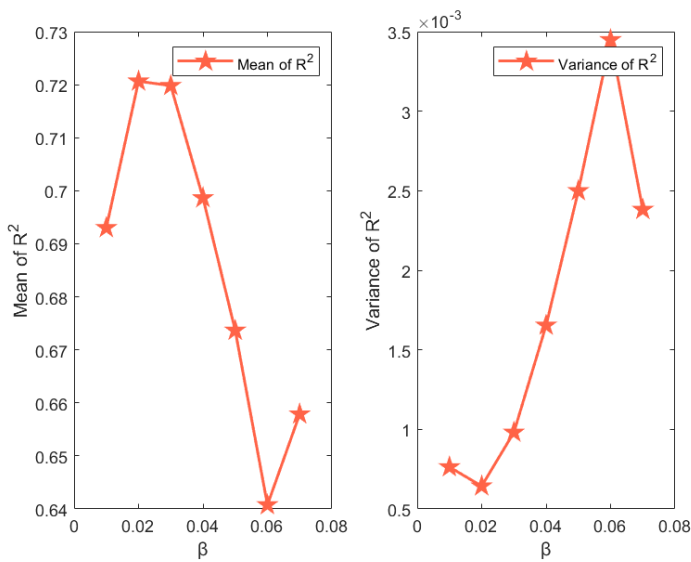
(d)



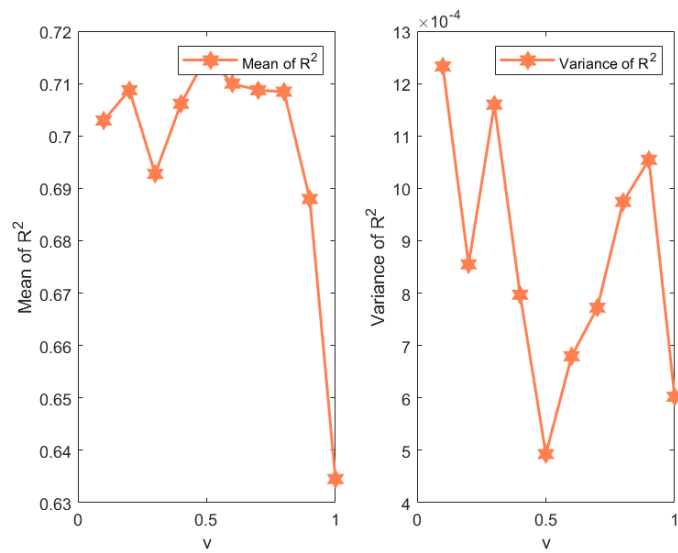
(e)



(f)



(g)



(h)

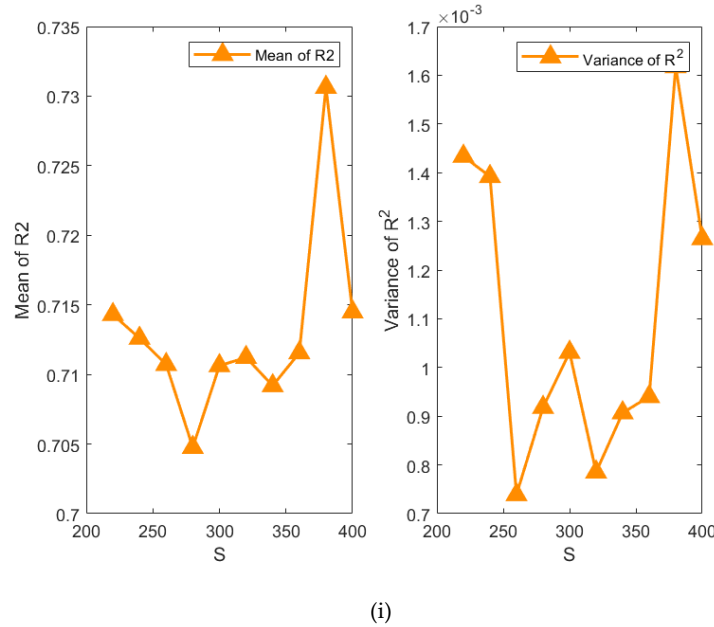


Figure 16. The curve of hyperparameter sensitivity analysis in terms of average value of 20 times.

Figure 16 shows a single factor-based hyperparameter analysis for mean and variance of R^2 in terms of average value of 20 times, and the results indicate that different hyperparameters of the constructed BO-EnTSFRT-LSTM model have different effects. The specific detailed analysis is as follow:

1) Figure 16(a) shows the effect of lr on the performance of the BO-EnTSFRT-LSTM model. When lr is less than 0.8, the mean value of R^2 fluctuates in a small range. When lr is greater than 0.8, R^2 decreases rapidly. It shows that it is easier to meet the requirements when lr is small. If the learning rate is too large, it may lead to skipping the optimal solution, which makes the model unable to achieve the best performance.

2) Figure 16(b) shows the effect of λ on the performance of the BO-EnTSFRT-LSTM model. When λ is in the range of 0 to 15, the mean value of R^2 generally shows a rapid growth. And when its value is in the range of 15 to 27, only a small fluctuation is produced. At the same time, the variance in this range also shows an overall downward trend. It shows that it is easier to meet the conditions when the value of λ is larger.

3) Figure 16(c) shows the effect of bs on the performance of the BO-EnTSFRT-LSTM model. With the increase of bs , the mean and variance of R^2 have relatively large fluctuations. It shows that the influence of bs on the performance of the model is more complicated and needs to be determined in combination with other hyperparameters.

4) Figure 16(d) shows the effect of LSM_a on the performance of the BO-EnTSFRT-LSTM model. With the increase of LSM_a , the mean value of R^2 increases rapidly. When LSM_a is in the range of 0.3 to 0.7, its value shows a decreasing trend as a whole. When LSM_a is 0.7, the mean value of R^2 is the smallest (0.7029). And then the mean value of R^2 increases rapidly. At the same time, the variance of R^2 also shows a downward trend.

5) Figure 16(e) shows the effect of $MinSamples1$ on the performance of the BO-EnTSFRT-LSTM model. When the value of $MinSamples1$ is greater than 30, the mean and variance of R^2 fluctuate in a small range. And when $MinSamples1=40$, the mean value of R^2 is the largest (0.7077) and the variance of R^2 is small (0.0008209). It shows that the larger $MinSamples1$ is easier to obtain higher performance model.

6) Figure 16(f) shows the effect of Tn on the performance of the BO-EnTSFRT-LSTM model. When Tn is in the range of 20 to 80 and 100 to 160, the mean value of R^2 decreases rapidly. And

when tn is 20, the mean value of R^2 is the largest (0.7096) and the variance of R^2 is small (0.0008537). It shows that it is easier to obtain a high performance model when Tn is small.

7) Figure 16(g) shows the effect of β on the performance of the BO-EnTSFRT-LSTM model. When the value of β is 0.02, the mean value of R^2 is largest (0.7206) and the variance value of R^2 is smallest (0.0006435). When the value of β is greater than 0.07, the learning rate is too large, resulting in gradient explosion. Therefore, in order to avoid the occurrence of gradient explosion, a smaller β should be selected.

8) Figure 16(h) shows the effect of ν on the performance of the BO-EnTSFRT-LSTM model. When ν is less than 0.8, the mean fluctuation range of R^2 is small. When ν is 0.5, the mean value of R^2 is the largest (0.7152) and the corresponding variance is the smallest (0.0004927). And when ν is greater than 0.8, the mean value of R^2 decreases rapidly.

9) Figure 16(i) shows the effect of S on the performance of the BO-EnTSFRT-LSTM model. When S is in the range of 220 to 340, the mean value of R^2 fluctuates in a small range. The mean value of R^2 increases rapidly when S is in the range of 340 to 380. And when S is 380, the mean value of R^2 is the largest (0.7306).

It should be noted that the sensitivity analysis conducted here is based on single factor hyperparameters. In fact, these numerous hyperparameters are interdependent, which is one of the reasons why BO is used for hyperparameter selection in this article.

3.5. Comprehensive Analysis

For the method proposed in this article, there are some limitations in each stage as follows.

(1) When the main model is constructed, it consists of multiple decision trees, and the training and prediction process consumes a lot of computing resources. Further research is needed on how to perform structural reduction on the constructed decision tree based main model. Moreover, how to further enhance the model's ability to represent uncertainties in the data also needs to be strengthened.

(2) When constructing the compensation model, the problem of gradient disappearance or gradient explosion may occur. At the same time, the structure of LSTM is complex and the training process is slow. Therefore, the structure of the model can be improved by researching its lightweight algorithms in the future.

(3) In the process of hyperparameters optimization, it is easy to fall into local optimum. For high-dimensional and non-convex functions with unknown smoothness and noise, it is often difficult for BO algorithm to fit and optimize them. How to improve the BO algorithm to enhance its ability and adaptability to the above problems is a problem that needs to be solved. In addition, further consideration is needed on how to perform BO optimization on non-gaussian distributed data.

4. Conclusion

The stable control of flue gas oxygen content during MSWI process is beneficial to improve incineration efficiency and reduce pollutant emissions. Aiming at the problem of lack of high-precision and interpretable flue gas oxygen content model, the article proposes a flue gas oxygen content model based on BO-optimized main-complement ensemble algorithm. The main contributions are as follows: (1) A hybrid model integrating parallel fusion and main-complement ensemble is proposed. (2) The strategy of using BO to optimize the hyperparameters of the main-complement ensemble model is proposed. (3) An interpretable and high-precision controlled object model for flue gas oxygen content is constructed for the first time. Experiments show that the performance of the proposed method is better than other methods, which shows the feasibility and effectiveness of the proposed method. In the subsequent research, the control algorithm will be carried out on this basis to ensure the efficient and stable operation of the MSWI process.

In addition, there are some limitations in this article. Due to the use of TSFRT and LSTM, this model consumes a lot of computer resources in its training stage. At the same time, how to improve the ability and adaptability of the BO algorithm to the above problems is a problem that needs to be

solved. In addition, further consideration is needed on how to perform BO optimization on non-gaussian distribution industrial data.

Appendix

Table A1. Abbreviations and their meanings.

Order	English abbreviation	English full name
1.	MSW	Municipal solid waste
2.	MSWI	MSW incineration
3.	AI	Artificial intelligence
4.	DXN	Dioxin
5.	SO ₂	Sulfur dioxide
6.	NN	Neural network
7.	LSTM	Long short-term memory
8.	ILSTM	Improved LSTM
9.	TS	Takagi-Sugeno
10.	TSFNN	TS fuzzy neural network
11.	MIMO	Multiple-input multiple output
12.	LRDT	Linear regression decision tree
13.	EnTSFRT	T-S fuzzy regression tree
14.	SCR	Selective catalytic reduction
15.	GBDT	Gradient boosting decision tree
16.	BO	Bayesian optimization
17.	MV	Manipulated variable
18.	CV	Controlled variable
19.	PCC	Pearson correlation coefficient
20.	MSE	Mean square error
21.	RNN	Recurrent neural network
22.	R ²	R-square
23.	UB	Upper boundary
24.	LB	Lower boundary

25.	GPR	Gaussian process regression
26.	MLE	Maximum likelihood estimation
27.	EI	Expected improvement
28.	RMSE	Root mean square error
29.	MAE	Mean absolute error

Table A2. symbols and their meanings.

Order	Symbol	Meaning
1.	u	Manipulated variables
2.	y	True value of flue gas oxygen content
3.	$\hat{y}_{\text{main}}$	Output value of flue gas oxygen content of EnTSFRT main model
4.	$\hat{e}_{\text{main}}$	Error value of EnTSFRT main model
5.	$\hat{e}_{\text{comp}}$	Output value of the LRDT compensation model
6.	$\hat{e}_{\text{mix}}$	Error value of LRDT compensation model
7.	$\hat{y}$	Output value of flue gas oxygen content of BO-EnTSFRT-LSTM model
8.	lr	Learning rate
9.	bs	Number of batches
10.	λ	The intermediate variable in the recursive calculation process
11.	LSM_{α}	Regularization coefficient of the main model
12.	MinSamples1	Minimum number of samples of the main model
13.	Tn	Number of decision trees of the main model
14.	β	Learning rate of the compensation model
15.	ν	Regularization coefficient of the compensation model
16.	S	Maximum number of iterations of the compensation model
17.	$\boldsymbol{D}$	Input of EnTSFRT
18.	n	Subscript of the n th input sample
19.	N	Total number of data set samples
20.	M	Dimension of input features

21.	U^j	Input variable matrix of the j th subset sample
22.	D^j	The j th training subset obtained by random feature sampling of D
23.	$\mu_{CS}^{v+1}(\cdot)$	Membership degree on the $(v+1)$ th non-leaf node
24.	$\mu_{CS}^{v+2}(\cdot)$	Membership degree on the $(v+2)$ th non-leaf node
25.	f_{TSFRT}	TSFRT sub-model
26.	Φ	Loss value of data set D^j
27.	$f_{MSE}^j(\cdot)$	MSE value of D_L^j (or D_R^j)
28.	D_L^j	Left subset
29.	D_R^j	Right subset
30.	y_L^j	True vector of D_L^j
31.	y_R^j	True vector of D_R^j
32.	$\mu_{cs}^v(u^j)$	Clear membership function of the input u^j on the v th non-leaf node
33.	$\mathcal{G}_{v_R}^j$	Average value of D_L^j target value
34.	$\mathcal{G}_{v_R}^j$	Average value of D_R^j target value
35.	N_{Subset}^j	Number of D_L^j (or D_R^j)
36.	$\mu_{cs}(u^j)$	Clear membership function of the input u^j
37.	ϕ_v	Segmentation threshold
38.	V	Number of nodes in the TSFRT model
39.	$\mu_{cs}^1(\cdot)$	The first membership of the clear set $C_{cs}^{leaf^j}$ is obtained by using the minimum MSE
40.	$D^{v_{leaf}^j}$	Training data set of fuzzy reasoning
41.	v_{leaf}^j	v_{leaf}^j th leaf node
42.	$N_{v_{leaf}^j}^j$	Number of samples in the v_{leaf}^j th leaf node

43.	K	The number of fuzzy rules
44.	k	The k th fuzzy rules
45.	y^k	Output of consequents of the k th fuzzy rules
46.	$u_1^{v_{leaf}}$	Variables in $C_{cs}^{v_{leaf}}$
47.	$\mu_{A_1^k}^k(\cdot)$	Membership function of A_1^k
48.	$\delta_{m,k}$	Center of membership function
49.	$\sigma_{m,k}^2$	Width of membership function
50.	α^k	Product output of k fuzzy rules
51.	$\mu_{A_m^k}^k$	Article k Membership of fuzzy rules
52.	$\bar{\alpha}^k$	Result of Normalization of Activation Intensity of Fuzzy Rule k
53.	$g(\cdot)$	Consequent output function
54.	ω	Weight parameters of consequents
55.	f_{TSFRT}^j	The j th TSFRT sub-model
56.	θ_{leaf}	Minimum sample amount
57.	$\boldsymbol{\omega}$	Consequent weight parameter matrix
58.	$\boldsymbol{\delta}$	Center matrix of membership function
59.	$\boldsymbol{\ell}$	Width matrix of membership function
60.	$\boldsymbol{u}^*$	Vector consisting of the product of t $\bar{\alpha}u$
61.	$\boldsymbol{U}^*$	Matrix composed of $\boldsymbol{u}^*$
62.	$\boldsymbol{\omega}_0$	Initial time consequent weight
63.	l	The l th recursive process
64.	H_l	Intermediate variables in the recursive process
65.	λ	A larger positive value
66.	$\boldsymbol{I}$	Identity matrix
67.	$\boldsymbol{F}$	EnTSFRT model

68.	LSM_{α}	Regularity coefficient of EnTSFRT
69.	W_{LSM}^{Tn}	Weight matrix of TSFRT in EnTSFRT model
70.	t	The current moment
71.	h	Hidden state
72.	c	Cell state
73.	$\tilde{c}$	Candidate cell state
74.	ζ	Sigmoid function
75.	ψ	The tanh function
76.	f	Value of the forget gate
77.	i	Value of the input gate
78.	o	Value of the output gate
79.	$\otimes$	Element-by-element multiplication
80.	W	Weight matrix
81.	b	Bias matrix
82.	W_{out}	Output weight matrix
83.	$J(t)$	The loss function in the training process of LSTM
84.	D^E	Input of the compensation model
85.	X	Input of BO
86.	P_{init}	BO algorithm initializes the number of inputs
87.	Γ	Objective function value corresponding to the observation sample
88.	ξ_{s_0}	Mean function
89.	p	Number of data set samples at the current time
90.	Σ_0	Covariance matrix
91.	σ^2	Measure the variance of noise
92.	$\Sigma_0(x_a, x_b)$	Kernel function between a th and b th sample points
93.	x_{ao}	The o th dimensional feature of the a th sample
94.	σ_{Γ}	Kernel amplitude of kernel function

95.	η_o	Length ratio of the o th dimension of the sample
96.	Σ_0^*	Training testing set covariance
97.	ξ_p	Posterior mean value
98.	σ_p^2	Posterior variance
99.	$\mathbf{x}_{op}$	Optimal point that minimizes the objective function value $\Gamma(\mathbf{x}_{op})$
100.	$\mathbf{x}^*$	New samples obtained by updating the data set
101.	D	The number of sample features
102.	$Loop$	Optimize the number of iterations
103.	Ω	EI sampling function
104.	$ratio$	Exploration ratio
105.	q	Number of iterations of the q th optimization
106.	ς	Current minimum number of samples
107.	$\hat{\mathbf{y}}_p$	Corresponding output value of the hybrid model with $\mathbf{x}_p$ as the input of BO.
108.	MinSamples2	Minimum sample size of LRDT model

References

1. 1] Jian, T., Heng, X., Wen, Y. & Jun-Fei, Q. (2023). Research status and prospect of intelligent optimization control of municipal solid waste incineration process. *Acta Automatica Sinica*, 49(10), 2019-2059.

2. 1] An, C., Jing-Rui, C., Jing, C., Chao, F. & Wei, W. (2019). Research on risks and countermeasures of “cities besieged by waste” in China-an empirical analysis based on DIIS. *Bulletin of Chinese Academy of Sciences*, 34(7), 797-806.

3. 1] *Efforts to Promote the Realization of Carbon Peak Carbon Neutral Target*. [Online]. Available: https://www.ndrc.gov.cn/wsdwhfz/202111/t20211111_1303691.html

4. 1] Ding, H., Tang, J., & Qiao, J. (2021). Control Methods of Municipal Solid Wastes Incineration Process: A Survey. In *2021 40th Chinese Control Conference (CCC)*(pp. 662-667). IEEE.

5. 1] Can-Lin, C., Jian, T., Heng, X., & Jun-Fei, Q. (2024). Early warning of dioxin emission in municipal solid waste incineration process based on fuzzy neural network confrontation generation. *Control Theory & Applications*, 1-9.

6. 1] Pan, X., Tang, J., & Xia, H. (2023). Flame combustion state identification based on CNN in municipal solid waste incineration process. In *2023 5th International Conference on Industrial Artificial Intelligence (IAI)*(pp. 1-4). IEEE.

7. 1] Qiao, J., Sun, J. & Meng, X. (2024). Event-triggered adaptive model predictive control of oxygen content for municipal solid waste incineration process. *IEEE Transactions on Automation Science and Engineering*, 21(1), 463-474.

8. 1] Jian, S., Xi, M. & Jun-Fei, Q. (2024). Data-driven predictive control of flue gas oxygen content in municipal solid waste incineration process. *Control Theory & Applications*, 41(3), 484-495.

9. 1] Kai-Cheng, H., Ai-Jun, Y & Jian, T. (2024). Multi-objective robust prediction model of furnace temperature and flue gas oxygen content in municipal solid waste incineration process. *Acta Automatica Sinica*, 50(5), 1001-1014.

10. 1] Huang, W., Ding, H. & Qiao, J. (2023). Adaptive multi-objective competitive swarm optimization algorithm based on kinematic analysis for municipal solid waste incineration. *Applied Soft Computing*, 149, 110925.
11. 1] Sun, J., Meng, X., & Qiao, J. (2024). Event-Based Data-Driven Adaptive Model Predictive Control for Nonlinear Dynamic Processes. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(4), 1982-1994.
12. 1] Li, L., Ding, S., Yang, Y., Peng, K. & Qiu, J. A fault detection approach for nonlinear systems based on data-driven realizations of fuzzy kernel representations. *IEEE Transactions on Fuzzy Systems*, 26(4), 1800-1812.
13. 1] Cho, S., Kim, Y., Kim, M., Cho, H. Moon, I. & Kim, J. (2022). Multi-objective optimization of an explosive waste incineration process considering nitrogen oxides emission and process cost by using artificial neural network surrogate models[J]. *Process Safety and Environmental Protection*, 162, 813-824.
14. 1] Sildir, H., Sarrafi, S. & Aydin E. (2022). Optimal artificial neural network architecture design for modeling an industrial ethylene oxide plant. *Computers & Chemical Engineering*, 163, 107850.
15. 1] Rahimieh, A., Mehriar, M., Zamir, S. & Nostrati. (2024). Fuzzy-decision tree modeling for H₂S production management in an industrial-scale anaerobic digestion process. *Biochemical Engineering Journal*, 208, 109380.
16. 1] Ma, Y., Zhou, Y., Peng, J., Chen, R., Dai, H. Ma, H. Hu, G & Xie, Y. (2024). Prediction of flue gas oxygen content of power plant with stacked target-enhanced autoencoder and attention-based LSTM. *Measurement*, 235, 115036.
17. 1] Hai-Xu, D., Jian, T, Heng, Xia. & Jun-Fei, Q. (2022). MIMO controlled object modeling of municipal solid waste incineration process based on TS-FNN. *Control Theory & Applications*, 39(8), 1529-1540.
18. 1] Lin, C., & Lee, C. (1991). Neural-network-based fuzzy logic control and decision system. *IEEE Transactions on Computers, Institute of Electrical and Electronics Engineers*, 40(12), 1320-1336.
19. 1] Wu, D., Peng, R. & Mendel, J. (2023). Type-1 and interval type-2 fuzzy systems. *IEEE Computational Intelligence Magazine*, 18(1), 81-83.
20. 1] Tang, J., Xia, H., Zhang, J., Qiao, J. & Yu, W. (2021) . Deep forest regression based on cross-layer full connection. *Neural Computing and Applications*, 33, 9307-9328.
21. 1] Xia, H., Tang, J., Wang, T., Tian, H., Cui, C. & Yu, W. (2023) . Interpretable controlled object model of furnace temperature for MSWI process based on a novel linear regression decision tree. In 2023 35th Chinese Control and Decision Conference (CCDC)(pp. 325-330). IEEE.
22. 1] Xia, H., Tang, J., Yu, W., et al. Takagi-Sugeno Fuzzy Regression Trees With Application to Complex Industrial Modeling[J]. *IEEE Transactions on Fuzzy Systems, Institute of Electrical and Electronics Engineers (IEEE)*, 2023, 31(7): 2210-2224.
23. 1] Meng T, Zhang W, Huang J, Cui, C. & Qiao, J. (2023) . Fuzzy reasoning based on truth-value progression: A control-theoretic design approach. *International Journal of Fuzzy Systems*, 25(4), 1559-1578.
24. 1] Yang, T., Ma, K., Lv, Y., Fang, F. & Chang, T. Hybrid dynamic model of SCR denitrification system for coal-fired power plant. (2019). In 2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)(pp. 106-111). IEEE.
25. 1] Luo, A. & Liu B. (2022). Temperature prediction of roller kiln based on mechanism and SSA-ELM data-driven integrated model. In 2022 China Automation Congress (CAC)(pp. 5598-5603). IEEE.
26. 1] Dong, S., Zhang, Y., Liu, J., Zhou, X. & Wang, X. (2022). Intelligent compensation prediction of leaching process based on GRU neural network. In 2022 China Automation Congress (CAC)(pp. 1663-1668). IEEE.
27. 1] Heng, Xia., Jian, Tang., Wen, Yu., & Jun-Fei, Qiao.(2024). Dioxin emission modeling based on simulation mechanism and improved regression decision tree. *Acta Automatica Sinica*, 50(8), 1001–1019.
28. 1] Friedman, J. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29, 1189-1232.
29. 1] Wei, Wang., Chang-Hui, Deng. & Li-Jie, Zhao. (2014). Research on the application of ellipsoid bound algorithm in hybrid modeling. *Acta Automatica Sinica*, 40(9), 1875-1881.
30. 1] Duan, H.,Meng, X., Tang, J., & Qiao, J. (2023) NOx emission prediction for MSWI process based on dynamic modular neural network, *Expert Systems with Applications*, 2023, 238, 122015.
31. 1] Zhang, R., Tang, J., Xia, H., Chen, J., Yu, W., & Qiao J. (2004). Heterogeneous ensemble prediction model of CO emission concentration in municipal solid waste incineration process using virtual data and real data hybrid-driven. *Journal of Cleaner Production*, 2024, 445, 141313.
32. 1] Ma, S., Chen, Z., Zhang, D.,Du, Y., Zhang, X. & Liu, Q. (2024). Interpretable multi-task neural network modeling and particle swarm optimization of process parameters in laser welding. *Knowledge-Based Systems*, 300, 112116.
33. 1] Zhou, X., Tan, J., Yu, J., Gu, X. & Jiang T. (2024). Online robust parameter design using sequential support vector regression based Bayesian optimization. *Journal of Mathematical Analysis and Applications*, 540(2),128649.

34. 1] Deng, J., Liu, G., Wang, L., Liu, W. & Wu, X. (2024). Intelligent optimization design of squeeze casting process parameters based on neural network and improved sparrow search algorithm. *Journal of Industrial Information Integration*, 39, 100600.
35. 1] Tian, H., Tang, J., Xia, H., Wang, T., Cui, C. & Pan, X. (2023). Furnace temperature control based on adaptive TS-FNN for municipal solid waste incineration process. In *2023 35th Chinese Control and Decision Conference (CCDC)*(pp. 360-365). IEEE.
36. 1] Ding, H., Tang, J. & Qiao, J. (2023). Dynamic modeling of multi-input and multi-output controlled object for municipal solid waste incineration process. *Applied Energy*, 339, 120982.
37. 1] Ding, H., Tang, J. & Qiao, J. (2022). MIMO modeling and multi-loop control based on neural network for municipal solid waste incineration. *Control Engineering Practice*, 127, 105280.
38. 1] Rasmussen C E , Williams C K I .Gaussian Processes for Machine Learning[M].MIT Press,2005.
39. 1] Rasmussen C E .Gaussian Processes in Machine Learning;[C]//Advanced Lectures on Machine Learning, ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures.2003.
40. 1] Wen-Ju, Y, Ying-Dong, Z., Hong-Hua, Y. (2024). Prediction of sediment discharge in Jinghe River based on optimized BP neural network. *Journal of China Hydrology*, 1-8.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.