Measurement Science and Technology

PAPER

A novel random subspace method considering complementarity between unsupervised and supervised deep representation features for soft sensors

Meas. Sci. Technol. 33 (2022) 105119 (18pp) https://doi.org/10.1088/1361-6501/ac7b6b

A novel random subspace method considering complementarity between unsupervised and supervised deep representation features for soft sensors

Gang Wang[1,2,3], Hegong Zhu[1], Zhangjun Wu[1,2,3,*] and Min Yang[1,2,3]

[1] School of Management, Hefei University of Technology, Hefei, Anhui, People's Republic of China
[2] Key Laboratory of Process Optimization and Intelligent Decision-Making (Hefei University of Technology), Ministry of Education, Hefei, Anhui, People's Republic of China
[3] Ministry of Education Engineering Research Center for Intelligent Decision-Making & Information System Technologies, Hefei 230009, People's Republic of China

E-mail: wuzhangjun@hfut.edu.cn

Received 3 May 2022, revised 12 June 2022
Accepted for publication 22 June 2022
Published 26 July 2022

Abstract

Unsupervised and supervised deep learning extract effective and abstract features from different perspectives, which have been successfully applied in soft sensors. However, few studies have fused them and explored the complementary effect between the two kinds of features, which limits the utilization of comprehensive prediction information. To address the problem, a novel random subspace method with stacked auto-encoder (SAE) and bidirectional long short-term memory (Bi-LSTM), named RS-SBL, is proposed for soft sensors. Firstly, unsupervised and supervised deep representation features are extracted by SAE and Bi-LSTM, respectively. Secondly, to leverage the complementarity of the fusion features, an improved random subspcae (RS) method with a structure sparsity learning model is designed to discriminate the relative importance of different features and generate ensemble prediction results. Finally, the experiments on two real-world industrial nonlinear processes demonstrate that the proposed RS-SBL with the feature fusion strategy improves the prediction performance, and outperforms the other comparison soft sensor models.

Keywords: soft sensor, ensemble learning, sparse group lasso, fusion features, deep learning

(Some figures may appear in colour only in the online journal)

1. Introduction

In modern industrial processes, it is of significance to carry out real-time measurement for the key quality variables to ensure production safety and improve product quality [1, 2]. Nevertheless, most quality variables are difficult to measure online due to the limitations of the measurement environment, unacceptable expenses, and long measurement delay. Therefore, soft sensors have been designed to estimate the quality variables by building mathematical models between quality variables and process variables [3]. In this way, soft sensors can provide necessary information for process monitoring, optimization, and control, which have attracted growing attention in academia and for industrial applications [4].

In general, soft sensors can be divided into two categories, i.e. first-principle models (FPMs) and data-driven models (DDMs) [5, 6]. The FPMs are established through some

[*] Author to whom any correspondence should be addressed.

physical and chemical reaction mechanisms in the industrial processes; they require specific domain knowledge. However, it is difficult for FPMs to obtain domain knowledge and ensure system stability in complex industrial processes [7]. With the wide application of distributed control systems, a lot of industrial process data can be collected, stored, and analyzed. Hence, DDMs that can be established based on large historical data have been extensively used in modern industrial processes [8].

Generally, the data-driven soft sensors consist of two core parts: feature extraction and model construction. In the feature extraction part, principal component analysis (PCA) [9], partial least squares (PLS) [10], kernel PCA [11], and kernel PLS [12] have been applied for soft sensor modeling. However, it is difficult for the aforementioned methods that can be considered as shallow learning methods to extract effective representation features from the complex industrial processes [13, 14].

Recently, deep learning with a deeper network structure has attracted increasing attention from scholars due to its advantage of learning the high-level and nonlinear representation features in processing complex problems. Several deep learning methods have been introduced into soft sensors. In respect of unsupervised deep learning methods, these methods aim to reconstruct the original input as accurately as possible and extract meaningful information on the underlying structure and distribution from the industrial process data [15, 16]. Therefore, unsupervised deep representation features mainly focus on the hidden information of the original input data itself. For example, a deep belief network (DBN) [17] and stacked auto-encoder (SAE) [18] can be used to extract unsupervised deep representation features for soft sensor applications. Considering supervised deep learning methods, these methods have great potential in learning the quality-relevant features by capturing the potential relationship between the process variables and quality variables. For instance, the convolutional neural network (CNN) [7], recurrent neural network (RNN) [19], and its variant long short-term memory (LSTM) [20] have great potential in capturing quality-relevant features, which are popularly applied for soft sensor modeling [21, 22]. In summary, the unsupervised and supervised deep representation features exploit distinct prediction information in industrial process data and demonstrate effectiveness in soft sensor modeling. Therefore, if only one kind of feature is used, the extracted information from industrial process data may be one-sided to some degree. Nevertheless, few studies have fused them and considered the joint effect of the two kinds of features, which may complement each other. The fusion features may contain more comprehensive information, which should be further investigated for soft sensor modeling.

In the model construction part, machine learning methods, such as the artificial neural network (ANN) [23], support vector regression (SVR) [24], and extreme learning machine (ELM) [25], have been applied to soft sensors based on the above features. These single methods still face some problems in complex industrial processes, such as instability and low generalization ability [26]. To address these issues, ensemble learning methods have been proposed to obtain more robust prediction results by aggregating the base learners. Hence, ensemble learning methods, which are divided into instance partitioning methods and feature partitioning methods, can enhance the diversity and steadiness of soft sensor models [27, 28]. The instance partitioning methods, such as bagging and boosting, have been applied to soft sensors. Nevertheless, the fusion features, including unsupervised and supervised deep representation features, are usually high dimensional, leading to the poor performance of the instance partitioning methods. Under such circumstances, the feature partitioning methods, such as random subspace (RS), have great potential in dealing with the high-dimensional problem and are more suitable for soft sensors [29]. However, traditional RS fails to explore the joint effect of different features to retain more comprehensive information. Besides, RS randomly divides the whole feature space into a series of feature subsets, resulting in redundant and irrelevant features in the feature subsets. Therefore, an improved RS with a fusion feature strategy is needed to enhance the prediction performance of soft sensor models.

To address these issues, a novel RS method with SAE and bidirectional LSTM (Bi-LSTM), named RS-SBL, is proposed to predict the quality variables in this paper. In the feature extraction part, the deep representation features are extracted from unsupervised and supervised perspectives to retain more comprehensive information in the industrial process data. These two kinds of features are extracted by the SAE and Bi-LSTM respectively. In the model construction part, the weighted sparse group lasso (WSGL) is designed to reduce the negative impact of the redundant and irrelevant features and explore the complementary information of the two types of features. Specifically, the improved RS, which relies on the WSGL to calculate the weights of the different features, makes it possible to generate high-quality feature subsets. Finally, the feature subsets are used to train the base learners and ensemble prediction results are obtained by averaging the base learner results. Therefore, compared with the deep learning methods, the RS-SBL retains more comprehensive information and ensures the complementarity of different kinds of features. To validate the effectiveness of the proposed RS-SBL, we conducted an empirical study based on an industrial debutanizer column process (DCP) [30, 31] and an industrial sulfur recovery unit (SRU) [32, 33]. The experimental results show that the proposed RS-SBL has better performance than the comparison methods. Moreover, the complementarity and effectiveness of the fusion features extracted from unsupervised and supervised perspectives are verified.

The main contributions of this paper are as follows:

(a) An ensemble learning framework with a feature selection method is proposed to improve the performance and enhance the stability of soft sensor models. In the framework, the deep features extracted from unsupervised and supervised perspectives are investigated simultaneously to explore the complementary effect of these two kinds of features.

(b) A novel soft sensor model named RS-SBL is proposed in this paper. Specifically, the unsupervised and supervised deep representation features are extracted by the SAE and

Bi-LSTM, respectively. Then, the WSGL is designed to distinguish the relative importance of different features and reduce the redundant and irrelevant features. And the improved RS based on WSGL is used to generate high-quality feature subsets and produce the ensemble prediction results.

(c) To validate the effectiveness of the proposed RS-SBL, two real-world industrial nonlinear processes are utilized to conduct the empirical study, and the experimental results demonstrate that the proposed RS-SBL is superior to other soft sensor methods.

The rest of this paper is structured as follows. The related work of soft sensors is introduced in section 2. Section 3 presents the framework of the RS-SBL. The details of the experimental setup are described in section 4. Section 5 introduces the results and discussions of the experiment. Finally, the conclusions and future research are summarized in section 6.

2. Related work

In recent decades, a variety of DDMs have been proposed and applied in industrial processes. In general, there are two major steps in the DDMs: feature extraction and model construction. Next, the studies of the two steps are introduced in this section.

2.1. Feature extraction in soft sensors

In the feature extraction step, there are two main types of methods: shallow learning methods and deep learning methods, which have been widely employed to extract representation features in industrial processes.

The shallow learning methods, such as PCA and PLS, have been designed to capture representation features automatically for soft sensors. PCA has been identified to be an effective approach for learning the low-dimensional features and reducing the complexity of the soft sensor models. For instance, Zamprogna *et al* proposed a novel soft sensor model based on PCA to select the most suitable process variables, which could be considered as the soft sensor model input [34]. Yang *et al* used kernel PCA to reduce the dimension of the process variables by considering the nonlinear relationship in the industrial processes [35]. Compared with PCA, PLS can learn the representation features that are related to the quality variables by introducing the constraints of the target space. For example, Wang *et al* compared different variable selection methods based on PLS and designed a new metric to evaluate the performance of these methods [36]. Kim *et al* designed a soft sensor model based on locally weighted PLS, which considered the nonlinear relationship between process variables and quality variables [37]. However, the methods mentioned above are regarded as shallow learning methods and it is difficult for these methods to capture effective features from complex industrial processes.

In recent years, deep learning methods with a deeper network structure have been successfully used for soft sensors due to their advantages of learning the high-level and nonlinear features from complex industrial processes. To improve the performance of soft sensors, unsupervised and supervised deep learning methods have been investigated in the existing studies. Several unsupervised deep learning methods, such as DBN and SAE, aim to retain the significant and meaningful information of the input data itself and have been successfully applied for industrial processes. DBN can capture the abstract and effective representation features from the probabilistic view by stacking several restricted Boltzmann machines. For instance, Liu *et al* presented an ensemble deep kernel learning for soft sensors, which used DBN to learn the nonlinear and informative features from the input data [16]. Zhu and Zhang investigated the selection of DBN structure to estimate the polymer melt index [38]. SAE has advantages in retaining the important and meaningful representation features by aiming to reconstruct the input data. Moreover, the process of training SAE is more effective and efficient due to its simple structure and layer-wise pretraining [33]. For example, Yuan *et al* proposed a novel variable-wise weighted SAE, which extracted hierarchical output-related features by correlation analysis with the output variables [13]. Yan *et al* implemented a deep relevant representation learning approach based on SAE, which utilized mutual information analysis to eliminate irrelevant features [39].

Supervised deep learning methods, such as CNN and RNN, can learn the significant quality-related features. CNN can extract deep local features by using a sliding window to retain the dynamic information. For instance, Yuan *et al* presented a dynamic CNN strategy to consider the spatial and temporal relationship in the process variables [40]. Geng *et al* designed an improved gated CNN to capture the short-term relationship in the industrial process data [7]. To deal with the complex dynamics in the industrial processes, RNN and its variant LSTM have great potential in capturing the temporal dependence from continuous industrial processes. For example, Kataria and Singh proposed a RNN-based soft sensor to estimate the bottom product composition of the reactive distillation column [41]. However, due to the problems of gradient vanishing and gradient exploding, it is difficult for RNN to learn the long-term dependence. To address the issue, LSTM is proposed to learn the long-term relationship from the sequential data by introducing the memory cell and gate mechanism. Wang *et al* proposed a new soft sensor model for batch process data, which utilized the LSTM to extract the quality-relevant features from long-time data [42]. Zhu *et al* combined XGBoost and a bidirectional, converted gate LSTM to extract dynamic information hidden in the process data [43].

In summary, unsupervised and supervised deep learning methods, such as SAE and LSTM, can extract effective and abstract features from unsupervised and supervised perspectives. Nevertheless, the joint effect of these two types of features is often ignored. The fusion features may contain complementary information. Therefore, it is of great significance to consider these two kinds of features to comprehensively reflect the different aspects of industrial process data and obtain more accurate prediction results.

2.2. Model construction in soft sensors

In the model construction step, machine learning methods, such as ANN and SVR, have been widely used to predict quality variables based on the above representation features. ANN has great potential in extracting the nonlinear relationship between process variables and quality variables. For instance, Fortuna *et al* used multilayer perceptron neural networks to improve product quality monitoring in a refinery [44]. Gonzaga *et al* presented a soft sensor model based on ANN to extract nonlinear features for providing online estimates of the polymer viscosity [23]. SVR uses a nonlinear kernel transformation to map the input data into a feature space, which has been widely used in soft sensors. For example, Kaneko *et al* combined online SVR with an ensemble learning system to consider the nonlinear changes in the industrial processes [27]. Chitralekha and Shah demonstrated that the SVR is an efficient and effective technique for nonlinear processes in soft sensors [45]. However, these single methods still face some problems in complex industrial processes, such as instability and low generalization ability.

On the contrary, ensemble learning methods can enhance the stability and generalization ability by aggregating the base learner results, which have raised more and more attention for soft sensors. For instance, Wang *et al* developed a two-layer ensemble learning framework for soft sensors, in which the process variables were divided into subspace blocks and then used to construct base learners to increase the model diversity [46]. He *et al* designed a soft sensor method based on a novel robust bagging to obtain robust and stable prediction results, which integrated the improved ELM and PLS [47]. These instance partitioning methods perform poorly when facing high-dimensional features. Compared with the instance partitioning methods, the feature partitioning methods, such as RS, are more suitable for soft sensors due to their advantage in processing high-dimensional features. RS can divide the whole feature space into a series of feature subsets, which are used to train base learners and enhance the model diversity and stability. However, the traditional RS generates the feature subsets in a completely random manner, leading to redundant and irrelevant features in the feature subsets. Besides, the traditional RS ignores the joint effect of the multiple features. Therefore, it is worth designing an ensemble method with feature selection methods to obtain an optimal joint strategy for the multiple features and improve the performance of soft sensor models.

3. The proposed method for soft sensors

3.1. Framework of RS-SBL

In modern industrial processes, soft sensors are of significance to provide real-time information about the quality variables for process monitoring, control, and optimization [4]. To improve the performance of soft sensors, existing studies have proposed various soft sensor models based on either unsupervised or supervised deep representation features and introduced ensemble learning methods to enhance the model diversity and stability. However, the existing studies fail to consider the complementary effect between the unsupervised and supervised deep representation features. Moreover, the performance of the ensemble learning methods needs to be improved in current soft sensor studies. To address the above issues, a novel RS method with SAE and Bi-LSTM, RS-SBL, is proposed to integrate unsupervised and supervised deep representation features for soft sensors. The framework of the proposed RS-SBL is illustrated in figure 1, and can be summarized as follows:

(a) Data acquisition: various process variables, such as temperature, pressure, and flow rate in industrial process plants, are measured by several physical sensors. Correspondingly, the quality variables can be obtained by offline laboratory analysis.
(b) Feature extraction: the unsupervised deep representation features that focus on the information of data itself, are extracted by the SAE. Besides, the supervised deep representation features that are related to the quality variables, are extracted by the Bi-LSTM.
(c) Model construction: the improved RS relies on the WSGL to calculate the weights of the different features. Then, the features are divided into multiple feature subsets, which are trained by base learners. Finally, the ensemble prediction results are obtained by aggregating the results of base learners.

For a clear presentation, some formal definitions used in this section are introduced. Suppose the input data of the soft sensor models can be represented as $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_i, y_i), \ldots, (\boldsymbol{x}_N, y_N)\}$. $\boldsymbol{x}_i \in R^m$ is the ith sample with m process variables, $y_i \in R$ is the quality variable of $\boldsymbol{x}_i$, and N is the number of samples. At time t, the unsupervised and supervised deep representation features are expressed as $\mathbf{F}_{t,1}$ and $\mathbf{F}_{t,2}$ respectively.

3.2. Data acquisition

For process variables, some physical sensors are installed in ndustrial plants. Hence, the process variables, such as temperature, pressure, and flow rate are measured and stored with the progress of the reaction. Correspondingly, the quality variables are difficult to measure by physical sensors. They can be measured by offline laboratory analysis, such as gas chromatography [13], but with a long measurement delay. It is worth noting that the historical process variables and quality variables are related to the current quality variables since the industrial processes have intrinsic dynamic dependence. To consider the process dynamics, the sliding window is used to process the historical data in this paper. For the input sequence, the input length is set to T, then the input sequence $[\boldsymbol{x}_{t-T+1}, \boldsymbol{x}_{t-T+2}, \ldots, \boldsymbol{x}_t] \in R^{T \times m}$ is regarded as all input sequence at time t. Correspondingly, $y_{t+1} \in R$ is the quality variable of the input sequence. Hence, the process dynamics in the input sequence is retained by the sliding window.

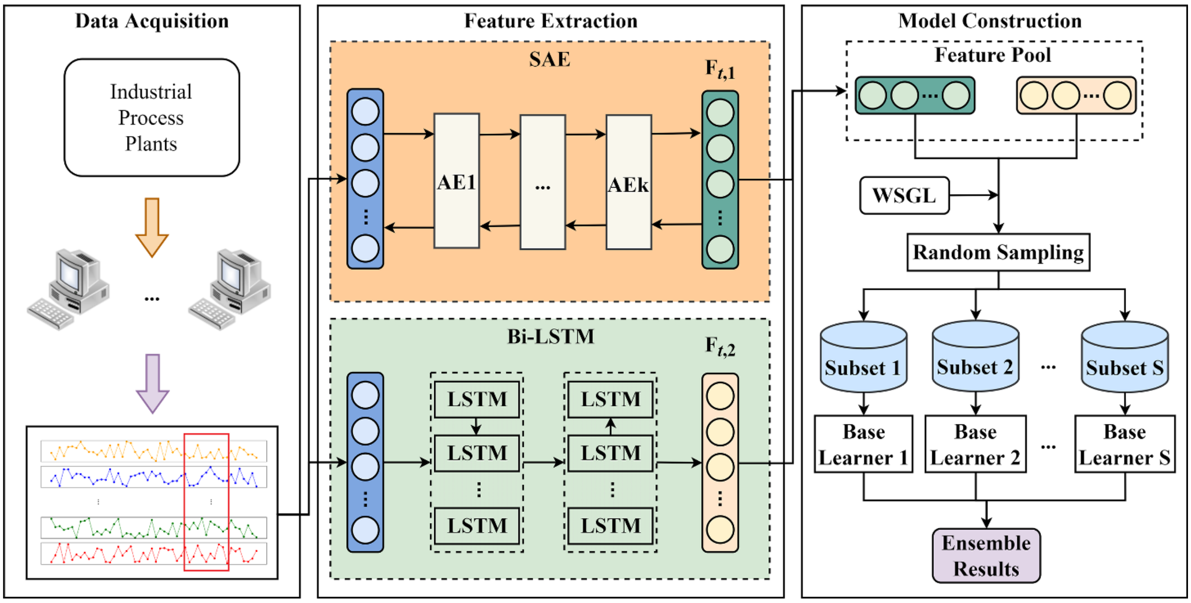


Figure 1. The framework of RS-SBL.

3.3. *Feature extraction for soft sensors*

After collecting the industrial data, the deep representation features are extracted from unsupervised and supervised perspectives. The unsupervised deep representation features are extracted by the SAE method, while the supervised deep representation features are extracted by the Bi-LSTM method.

3.3.1. Unsupervised deep representation features. Several unsupervised deep learning methods, such as DBN and SAE, which can learn the abstract features and reflect the potential information of the process variables, have been successfully applied in soft sensors. Compared with other unsupervised methods, SAE can capture the important and meaningful representation features by attempting to reproduce the input data. Moreover, the process of training SAE is more effective and efficient due to its simple structure and layer-wise pretraining. Therefore, SAE is employed to extract the unsupervised deep representation features for soft sensors in this paper. Specifically, the SAE is composed of multiple auto-encoders (AEs) layer by layer. The structure of SAE is shown in figure 2.

The AE consists of two parts: the encoder layer and the decoder layer. The encoder layer converts the input vector into the hidden feature vector, while the decoder layer restores the hidden feature vector to the input vector. The first step of the AE aims to map the input data $\boldsymbol{x}$ into the hidden feature vector $\boldsymbol{h}$:

$$h = f_{\mathrm{en}}(W_{\mathrm{en}}x + b_{\mathrm{en}}) \tag{1}$$

where f_{en} is the activation function of the encoder layer, the W_{en} and b_{en} are the parameters of the encoder layer. Correspondingly, the decoder layer restores the hidden feature vector to the input data:

$$\tilde{x} = f_{\mathrm{de}}(W_{\mathrm{de}}h + b_{\mathrm{de}}). \tag{2}$$

To obtain the parameter, set $\theta = \{W_{\mathrm{en}}, W_{\mathrm{de}}, b_{\mathrm{en}}, b_{\mathrm{de}}\}$. The reconstruction error between $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ is minimized by calculating the mean squared error as:

$$J(W_{\mathrm{en}}, W_{\mathrm{de}}, b_{\mathrm{en}}, b_{\mathrm{de}}) = \frac{1}{2N}\sum_{i=1}^{N}\|\tilde{\boldsymbol{x}}_i - \boldsymbol{x}_i\|^2. \tag{3}$$

Hence, the AE can learn the unsupervised representation features from the input data. To capture the high-level and abstract features, multiple AEs can be stacked to construct the SAE. As shown in figure 2, the original data $\boldsymbol{x}$ is fed into the encoder layer of the first AE (AE1) to train the parameter $\{W_1, b_1\}$ and obtain the feature vector $\boldsymbol{h}_1$ of AE1 by minimizing the reconstruction error between $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$. Then, the feature vector $\boldsymbol{h}_1$ is fed into the encoder layer of AE2 to train the parameter $\{W_2, b_2\}$ and obtain the feature vector $\boldsymbol{h}_2$ of AE2 by minimizing the reconstruction error between $\boldsymbol{h}_1$ and $\tilde{\boldsymbol{h}}_1$. In this way, the whole SAE is pre-trained layer by layer and the unsupervised deep representation features obtained can be denoted as $\mathbf{F}_{t,1} = \{\boldsymbol{f}_{t,1}^1, \boldsymbol{f}_{t,2}^1, \ldots, \boldsymbol{f}_{t,j}^1, \ldots, \boldsymbol{f}_{t,m_1}^1\}$, where m_1 is the number of unsupervised deep representation features.

3.3.2. Supervised deep representation features. The supervised deep learning methods, such as CNN and RNN, can learn the quality-relevant features by finding the potential relationship between process variables and quality variables. Because of the complex reaction processes, the delay of feedback control, and the retention of the raw materials and products in the device, there is complex temporal dependence in industrial processes. RNN has been widely employed in soft sensor applications due to its potential for learning complex temporal relationships. However, RNN only has short memory due to the problem of gradient vanishing. To address the problem, LSTM is a variant of the standard RNN, which can learn the long-term dependence by introducing cell state and controlling

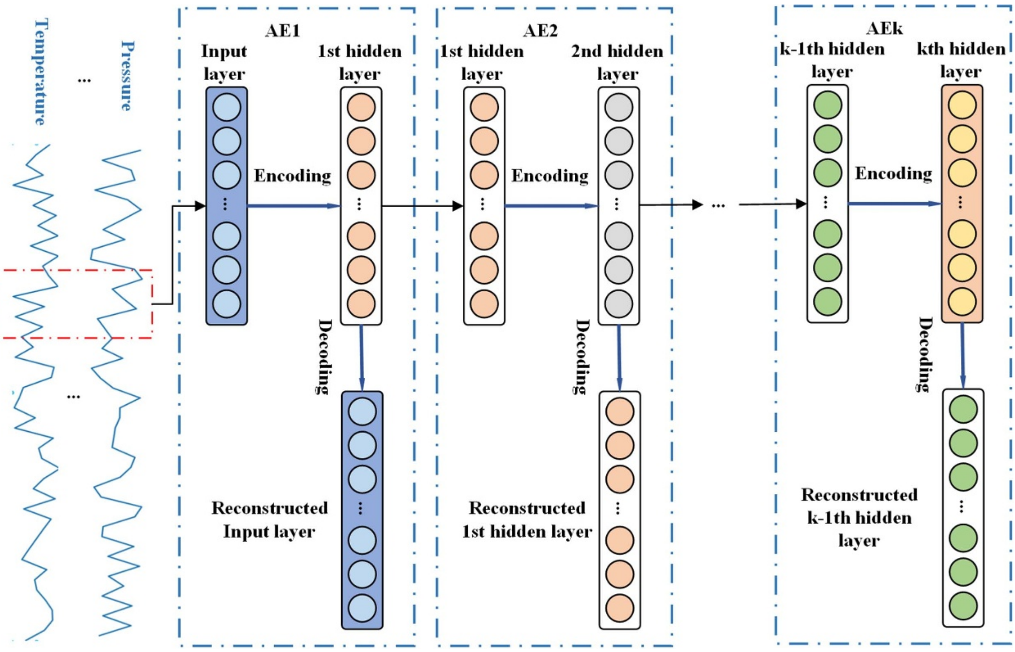


Figure 2. The structure of SAE.

information flow by using three gates [48]. Therefore, LSTM is utilized to learn the supervised and temporal representation features in this paper. The formula of LSTM is as follows:

$$f_t = \sigma\left(W_{fx}[x_t, h_{t-1}] + b_f\right) \tag{4}$$

$$i_t = \sigma\left(W_{ix}[x_t, h_{t-1}] + b_i\right) \tag{5}$$

$$o_t = \sigma\left(W_{ox}[x_t, h_{t-1}] + b_o\right) \tag{6}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tanh\left(W_{cx}[x_t, h_{t-1}] + b_c\right) \tag{7}$$

$$h_t = o_t \odot \tanh(C_t) \tag{8}$$

where σ is the sigmoid activation function, f_t, i_t, and o_t represent the forget gate, input gate and output gate, respectively; W_{fx}, W_{ix}, W_{ox}, and W_{cx} are the weight matrices of the different gates, and b_f, b_i, b_o, and b_c are bias terms. The previous cell state C_{t-1}, the previous hidden state h_{t-1}, and the current input vector x_t are three external inputs of the LSTM unit at the time step t.

However, the unidirectional LSTM is a network that spreads from the front to the back, so only historical information can be used. Bi-LSTM can obtain more temporal relationships from the original data. Therefore, Bi-LSTM is used to extract more effectively dynamic features for soft sensors in this paper. The structure of Bi-LSTM is shown in figure 3.

As shown in figure 3, Bi-LSTM has two unidirectional and opposite LSTM models. At the moment t, the process variables are fed into Bi-LSTM with opposite directions, then the hidden layer features $\vec{h}_t$ and $\overleftarrow{h}_{t-T+1}$ are calculated according to equations (4)–(8). The output of the last element is taken as the feature vector extracted by the Bi-LSTM, which can be denoted as $\boldsymbol{h} = [\vec{h}_t^1, \vec{h}_t^2, \ldots, \vec{h}_t^j]^{\mathrm{T}} \oplus [\overleftarrow{h}_{t-T+1}^1, \overleftarrow{h}_{t-T+1}^2, \ldots, \overleftarrow{h}_{t-T+1}^j]^{\mathrm{T}}$. Finally, the supervised representation features extracted from the process variables are defined as $\mathbf{F}_{t,2} = \{\boldsymbol{f}_{t,1}^2, \boldsymbol{f}_{t,2}^2, \ldots, \boldsymbol{f}_{t,j}^2, \ldots, \boldsymbol{f}_{t,m_2}^2\}$, where $\boldsymbol{f}_{t,j}^2$ is equal to the vector $\boldsymbol{h}$ and m_2 is the number of supervised deep representation features.

The training process of Bi-LSTM is as follows. Firstly, x_t, the previous hidden state h_{t-1}, and C_{t-1} are the input LSTM cell at the time t. Then h_t and C_t are calculated by equations (4)–(8), which go forward to the next time step. Finally, to train the Bi-LSTM, we adopted the mean square error as the loss function, which is calculated as follows:

$$J_{\mathrm{MSE}} = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \tilde{y}_i\right)^2 \tag{9}$$

where y_i and $\tilde{y}_i$ are the actual and predicted values respectively.

3.4. Model construction for soft sensors

After extracting the unsupervised and supervised deep representation features from the original data, the model construction based on the above features is the other important step for soft sensors. The ensemble learning methods, including instance partitioning methods and feature partitioning methods, have been introduced to enhance the diversity and stability of the soft sensor models. It is worth noting that the fusion features, including unsupervised and supervised deep representation features, are usually high dimensional. Under such circumstances, the feature partitioning methods, such as RS, are more suitable for soft sensors The RS can generate a series of feature subsets and train the base learners through feature subsets rather than the instance subsets.

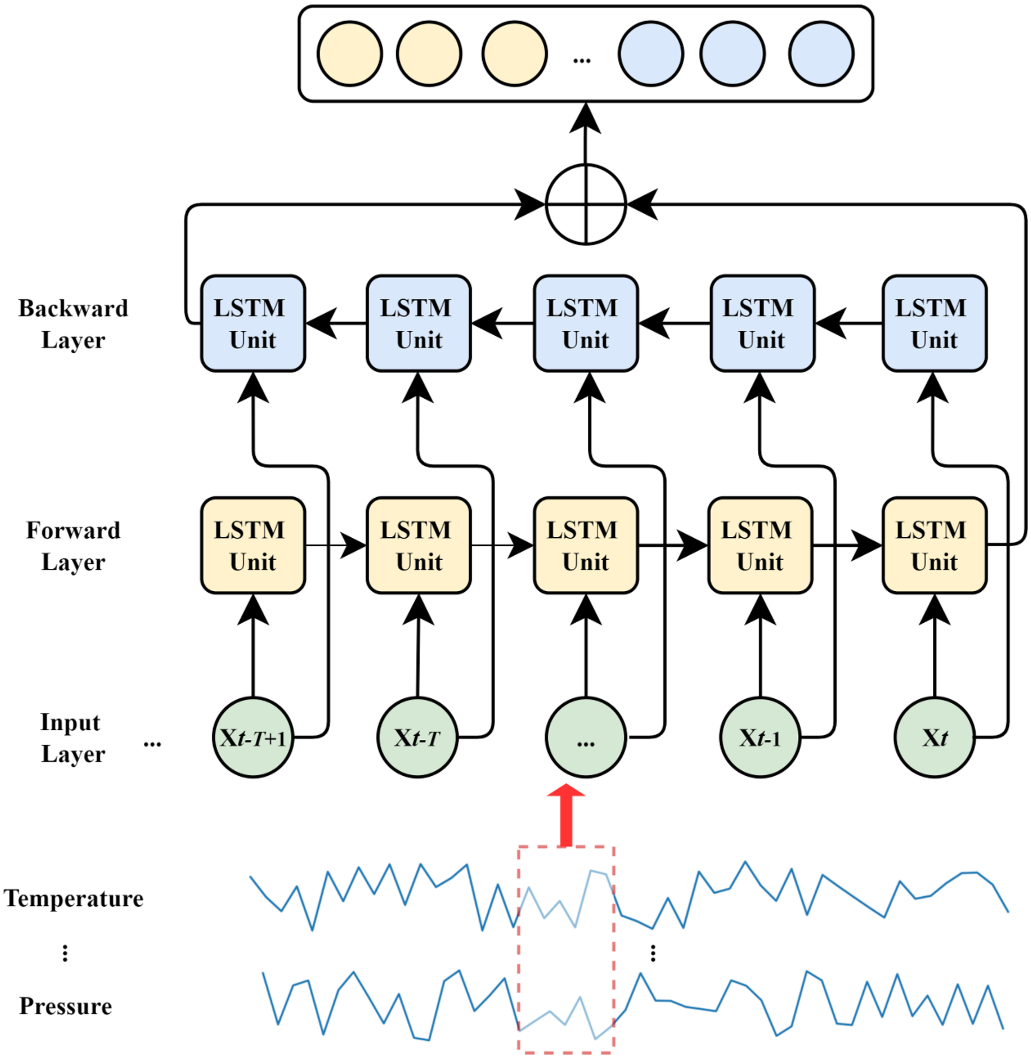


Figure 3. The structure of Bi-LSTM.

Nevertheless, the features are randomly selected by the traditional RS, leading to redundant and irrelevant features in the feature subsets. These redundant and irrelevant features have an adverse effect on the performance of the base learners. Hence, it is necessary to introduce feature selection methods to identify the important features and filter out the irrelevant features and improve the quality of feature subsets. Recently, structured sparsity learning has shown its superiority when facing high-dimensional data. However, the feature selection methods based on this structure only consider the individual features and ignore the group structure, which may contain significant information between different feature groups. These feature selection methods, such as group lasso (GL) and sparse GL (SGL), have been designed to address the issue. It is worth noting that the importance of the feature groups is different. Moreover, the single features in the same feature group have different contributions to predicting the quality variables. It is necessary to consider the distinguishment of the feature groups and the relative importance of the individual feature.

Therefore, to more effectively reduce the redundant and irrelevant features, the WSGL is designed to adaptively calculate the weights of different feature groups and the features in the same feature group. By the WSGL, the redundant and irrelevant features are effectively filtered out, which are given lower weights. Hence, the improved RS based on WSGL can improve the performance of soft sensor models. The improved RS mainly consists of two major steps: generation of the base learners and combination of the base learner results. The first step aims to generate the high-quality feature subsets, and the second step is to obtain the ensemble prediction results by combining the base learner results. The whole procedure of model construction is shown in figure 4.

3.4.1. Generation of the base learners. The generation of base learners is the first step in the model construction, which aims to generate a series of high-quality feature subsets, which are used to train the base learners. In this paper, the SAE and Bi-LSTM are employed to extract the unsupervised and supervised deep representation features, which can be divided into two feature groups. The data matrix input into the WSGL can be represented

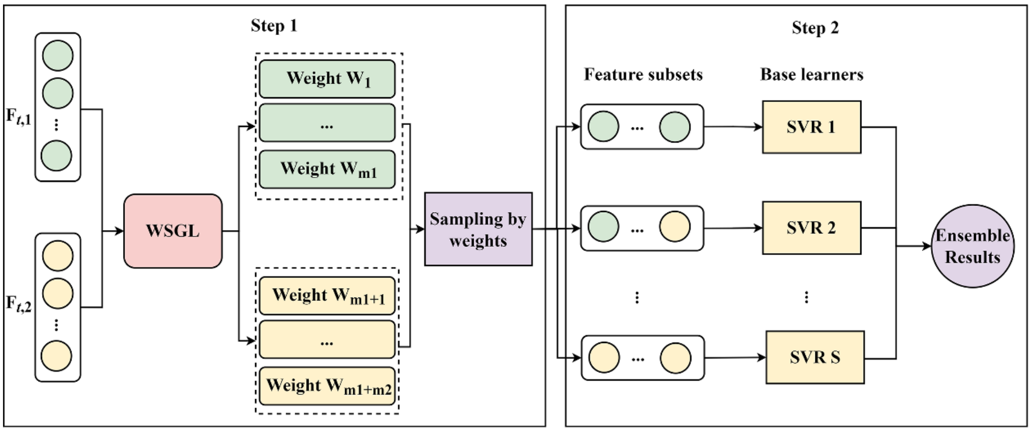


Figure 4. The model construction procedure.

as $\{(\boldsymbol{X}_1,y_1),\ (\boldsymbol{X}_2,y_2),\ \ldots,\ (\boldsymbol{X}_i,y_i),\ \ldots,\ (\boldsymbol{X}_N,y_N)\}$, where N is the number of samples, $\boldsymbol{X}_i=\{\boldsymbol{X}_i^1,\ \boldsymbol{X}_i^2\}$ is the input of the WSGL, $\boldsymbol{X}_i^1=[f_{i,1}^1,f_{i,2}^1,\ \ldots,f_{i,m_1}^1]$ and $\boldsymbol{X}_i^2=[f_{i,1}^2,f_{i,2}^2,\ \ldots,f_{i,m_2}^2]$ are the unsupervised and supervised deep representation features of the ith sample respectively, y_i is the quality variable of $\boldsymbol{X}_i$. Based on these, WSGL can be described as an optimization task as follows:

$$\begin{aligned}\min_{\beta,b}\frac{1}{2N}&\left\|\boldsymbol{y}-\sum_{c=1}^{2}v_c(\boldsymbol{\beta}_c\boldsymbol{X}^c+b_c)\right\|_2^2\\&+(1-\alpha)\lambda\sum_{c=1}^{2}\sqrt{m_c}\|\boldsymbol{\beta}_c\|_2+\alpha\lambda\|\boldsymbol{\beta}\|_1\\&\text{s.t.}\ \ \|v_c\|=1,v_c>0\end{aligned}\tag{10}$$

where the penalty parameter λ and the combination coefficient α determine the degree of contraction, $\boldsymbol{\beta}$ is the coefficient vector of the single feature and $\boldsymbol{\beta}_c$ is the coefficient vector of the cth feature group ($c=1$, 2), v_c is the weight of the cth feature group. Compared with the SGL, the WSGL considers the relative importance of different feature groups by introducing the parameters v_c, which can be learned adaptively in the process of training the WSGL. Specifically, v_c adaptively learns the weights of feature groups and determines the relative importance of each kind of feature group. Besides, the $(1-\alpha)\lambda\sum_{c=1}^{2}\sqrt{m_c}\|\boldsymbol{\beta}_c\|_2$ term makes the GL have group sparsity and further ensures that the features play predictive roles in collaboration. And the $\alpha\lambda\|\boldsymbol{\beta}\|_1$ places the sparsity upon single features. Therefore, the WSGL can effectively filter out irrelevant and redundant features by considering the weights of feature groups and ensuring the sparsity of the groups and within the groups at the same time. The WSGL is utilized to evaluate the weights of the different features by minimizing formula (10). The weights of the features of each feature group can be represented as $\boldsymbol{\beta}_c=[\beta_1^c,\ \beta_2^c,\ \ldots,\ \beta_{m_c}^c]^{\mathrm{T}}$.

Taking the derived weight vectors $\boldsymbol{\beta}_c$ as the sampling probabilities of features, we divide the features into several feature subsets. The subspace rate ratio refers to the proportion of the number of features in the subsets to the total number of features. Thus, under the joint effect of ratio and the sampling probabilities of features, the important features have higher weights and have a greater probability of being selected. In this way, the feature subsets of the improved RS can be represented as $\boldsymbol{D}_{\mathrm{sub}}^j=\{(\boldsymbol{f}_1',y_1),\ (\boldsymbol{f}_2',y_2),\ \ldots,\ (\boldsymbol{f}_N',y_N)\},1\leqslant j\leqslant S$. Then we can get a set of feature subsets, which are denoted as $\{\boldsymbol{D}_{\mathrm{sub}}^1,\ \boldsymbol{D}_{\mathrm{sub}}^2,\ \ldots,\ \boldsymbol{D}_{\mathrm{sub}}^j,\ \ldots,\ \boldsymbol{D}_{\mathrm{sub}}^S\}$. In this way, by the WSGL, the improved RS can select the significant features with a higher probability and enhance the diversity and stability of base learners.

3.4.2. Combination of the base learner results. In the step of base learner combination, to obtain the ensemble prediction results, the feature subsets are used to train the base learners. Due to the ability to deal with nonlinear problems, SVR has been successfully used for soft sensors. Therefore, the SVR is selected to be the base leaner in this paper. To address the nonlinear regression problems, the SVR formalism considers the following linear estimation function, which can be represented as $f(x)=(w^T\cdot\varphi(x))+b$, where $\varphi(x)$ denotes a functiontermed feature, w is the weight vector, and b is a constant. SVR can be obtained by solving the following optimization problem:

$$\min\frac{1}{2}w^Tw+C\sum_{i=1}^{N}(\xi_i^+ +\xi_i^-)\tag{11}$$

$$\text{s.t.}\begin{cases}y_i-w^T\phi(x)-b\leqslant\varepsilon+\xi_i,\\ w^T\phi(x)+b-y_i\leqslant\varepsilon+\xi_i^*,\\ \xi_i\geqslant 0,\xi_i^*\geqslant 0,C\geqslant 0.\end{cases}\tag{12}$$

where ε is the maximum deviation allowed during the training and C is the penalty coefficient. The ξ_i^+ and ξ_i^- are slack variables, which correspond to the size of the excess deviation for positive and negative deviations, respectively.

Finally, to reduce the prediction error, the prediction results of each base learner need to be combined reasonably. In

Table 1. Pseudo-code of the RS-SBL algorithm.

```
**RS-SBL** ($D$,$\alpha$,$\lambda$, ratio, $L$)

Input: Dataset $D = \{(\boldsymbol{x}_1, y_1),\ (\boldsymbol{x}_2, y_2),\ \ldots,\ (\boldsymbol{x}_i, y_i),\ \ldots,\ (\boldsymbol{x}_N, y_N)\}$;
  Combination coefficient $\alpha$;
  WSGL penalty parameter $\lambda$;
  Number of feature groups $K$;
  Random subspace rate ratio;
  Number of subsets: $S$;
  Base learners $L$;
Processing:
  $F_1 = \mathrm{SAE}(D)$ %Training the SAE to obtain unsupervised features.
  $F_2 = \mathrm{BiLSTM}(D)$ %Training the Bi-LSTM to obtain supervised features.
  Base learners $L \leftarrow \{\emptyset\}$;
  Number of feature subset num $\leftarrow$ floor($d \times$ ratio);
for $j \in \{\ 1, 2, \ldots, \mathrm{S}\}$ do:
  Feature subset $f_j \leftarrow \{\emptyset\}$;
  for $l \in \{\ 1, 2,\ \ldots, \mathrm{m}_c\},\ \ c = \{1,2\}$ do:
    $\beta_l^c = \mathrm{WSGL}(F_1, F_2, \alpha, \lambda)$
  end for
  Repeat until $|f_j| =$ num:
    Generate a random number rand between 0 and 1;
    sum $\leftarrow 0$;
    for $l \in \{\ 1, 2,\ \ldots, \mathrm{m}_c\},\ \ c = \{1,2\}$ do:
      sum $\leftarrow$ sum $+ \beta_l^c$;
    if (sum $+ \omega_c) \geqslant$ rand and the $l$th feature is not in $f_j$:
      Add the $l$th feature to $f_j$
    end if
  end for
 end of repeat
 $\tilde{y}^j = L(f_j)$; %Training the base learner with the sub dataset.
end for
Output: $\tilde{y}^{\mathrm{final}} = \frac{1}{S}\sum_{j=1}^{S}\tilde{y}^j$
```

this paper, we average the base learner results to obtain the ensemble prediction results:

$$\tilde{y}^{\mathrm{final}} = \frac{1}{S}\sum_{j=1}^{S}\tilde{y}^j \tag{13}$$

where $\tilde{y}^j$ is the prediction result of the jth base learner; $\tilde{y}^{\mathrm{final}}$ is the ensemble prediction result.

Table 1 shows the pseudo-code of the proposed RS-SBL.

4. Experiments

To validate the effectiveness of the proposed RS-SBL, the details of experiments are given in this section. Firstly, the experiment data sets are described in detail. Secondly, the evaluation metrics are introduced to evaluate the predictive performance. Finally, the experimental procedure is illustrated.

4.1. Experiment data set

4.1.1. DCP. In the industrial refining process, the debutanizer column is a significant processing unit. The flowchart of the debutanizer column is described clearly in literature [49]. The debutanizer column mainly consists of six reaction devices: heat exchanger, overhead condenser, bottom reboiler, head reflux pump, feed pump to the liquefied petroleum gas (LPG) splitter, and reflux accumulator. It is mainly used to separate the gasoline content (C5) from propane (C3) and butane (C4) to maximize C5 content and minimize C4 concentration in the debutanizer bottoms [49].

However, the concentration of C4 cannot be measured directly by the physical sensors, but can be measured by a gas chromatograph with a long measurement delay. To address the problem, soft sensors can be utilized to measure the concentration of C4. Seven process variables were used to build the soft sensor models. Table 2 gives a detailed description of these variables. There were a total of 2394 samples collected in this industrial process, of which 1400 samples were used to train the soft sensor model, and the remaining samples were utilized for model testing.

4.1.2. SRU process. In the industrial petroleum refinery, the SRU has a significant role. The SRU is designed to reduce the acid gas streams before they are released into the atmosphere. Moreover, sulfur can be effectively recovered in the SRU. Therefore, sulfur emissions can be effectively controlled. The

Table 2. Variable description for the debutanizer column.

Input variable	Variable description
u1	Top temperature
u2	Top pressure
u3	Reflux flow
u4	Flow to next process
u5	6th tray temperature
u6	Bottom temperature A
u7	Bottom temperature B

Table 3. Variable description for the SRU.

Input variable	Variables description
u1	Gas flow MEA GAS
u2	Air flow AIR MEA
u3	Secondary air flow AIR MEA 2
u4	Gas flow in SWS zone
u5	Air flow in SWS zone
y1	Concentration of SO_2 in the tail gas
y2	Concentration of H_2S in the tail gas

flowchart of the SRU is described clearly in literature [50]. There are four identical subunits of sulfur lines, which aim to transform two kinds of acid gases into sulfur. The two kinds of acid gases mainly includes monoethanolamine (MEA) gas and gas from sour water stripping (SWS). The MEA gas is rich in H_2S and the SWS gas is rich in NH_3 and H_2S. A more detailed description can be found in [50] for the SRU. The main chemical reactions in SRU are as follows:

$$2H_2S + 3O_2 \rightarrow 2SO_2 + 2H_2O \tag{14}$$

$$2H_2S + SO_2 \rightarrow 3S + 2H_2O. \tag{15}$$

To predict the quality variables including H_2S and SO_2, five process variables are selected to establish the soft sensor models. Table 3 gives a detailed description of these variables. In this study, 2000 samples were utilized to train the soft sensor models, and the other 1000 samples were used for model testing.

4.2. Evaluation metrics

In this paper, the performance of the soft sensor models is evaluated by the root mean squared error (RMSE) and r-square (R^2). RMSE can measure the size of the deviation between the actual output and the estimated output. When the RMSE is smaller, the prediction of the model is more accurate. In addition, R^2 can be used to measure the squared correlation between the actual output and the estimated output. The definitions of RMSE and R^2 are as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{(N_t - 1)} \sum_{i=1}^{N_t} (y_i - \hat{y}_i)^2} \tag{16}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N_t} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N_t} (y_i - \bar{y})^2} \tag{17}$$

where y_i and $\hat{y}_i$ are the labeled and predicted values of the ith testing sample, respectively. N_t is the number of testing samples. $\bar{y}$ is the mean of output values in the testing data set.

4.3. Experimental procedure

In the experiment, the proposed RS-SBL in this paper is compared with other soft sensor methods, including SAE, Bi-SLTM, SVR, bagging, Adaboost, random forest (RF), XGBoost, RS, and RS based on least absolute shrinkage and selection operator (RS-LASSO), which uses the LASSO to calculate the weights of the different features. To ensure the fairness of comparison, the base learners of ensemble learning methods such as bagging, Adaboost, RS, RS-LASSO, and the proposed RS-SBL are SVR, and a decision tree is the base learner of the RF. Table 4 shows the details of all the parameters used in the experiments.

To reduce the adverse influence of randomness, all the experiments were carried out repeatedly 20 times and the final results were calculated by averaging the results of the repeated experiments. All the experiments for this paper were processed by using Python 3.7.

5. Experimental results and discussion

5.1. Experimental results

The experimental results of the proposed RS-SBL and the mentioned methods are compared in tables 5 and 6 which show the mean and standard deviation (SD) of RMSE and R^2 values from the two data sets, respectively. The best results are marked in bold. It is worth noting that all results are based on the fusion of the unsupervised and supervised deep representation features except for SAE and Bi-LSTM, which are directly used to predict quality variables.

As shown in tables 5 and 6, firstly, the proposed RS-SBL achieves the best performance under the RMSE and R^2 metrics. Specifically, the RMSE values of the RS-SBL on the two data sets are 3.593% (DCP), 0.964% (SRU (SO_2)), and 1.124% (SRU (H_2S)). And the R^2 values of the RS-SBL on the two data sets are 95.20% (DCP), 86.32% (SRU (SO_2)), and 95.37% (SRU (H_2S)). Taking the DCP as an example, the RMSE value of RS-SBL decreases by 27.88%, 26.43%, 25.93%, 27.38%, 28.80%, 2.629%, 6.869%, 16.75%, and 9.199% in comparison with SAE, Bi-LSTM, SVR, Bagging, Adaboost, RF, XGBoost, RS, and RS-LASSO, respectively. Correspondingly, taking for example SRU (SO_2) , the RMSE value of RS-SBL decreases by 42.62%, 33.61%, 12.68%, 11.56%, 17.32%, 11.15%, 12.30%, 12.68%, and 4.837% in comparison with SAE, Bi-LSTM, SVR, Bagging, Adaboost, RF, XGBoost, RS, and RS-LASSO,

Table 4. Details of all the parameters used in the experiments.

Method	Parameters
SAE	Number of layers: 3; number of hidden units: 36-28-20; learning rate: 0.001; batch size: 32; epoch: 200.
Bi-LSTM	Number of layers: 2; number of hidden units: 20; learning rate: 0.001, batch size: 32, epoch: 200.
SVR	Kernel: RBF; gamma: auto; penalty parameter: 1.0.
Bagging	Number of ensembles: 10; Base learner: SVR.
Adaboost	Number of ensembles: 10; base learner: SVR.
RF	Number of ensembles: 10; base learner: DT; criterion: MSE; splitter: best.
XGBoost	Number of ensembles: 10; learning rate: 0.1; max_depth: 3.
RS	Subspace rate: {0.1, 0.3, 0.5, 0.7, 0.9}; number of ensembles: 10; base learner: SVR.
RS-LASSO	Penalty factor λ: {0.001, 0.01, 0.1, 1}; subspace rate: {0.1, 0.3, 0.5, 0.7, 0.9}; number of ensembles: 10; base learner: SVR.
RS-SBL	Combination coefficient α: {0.1, 0.3, 0.5, 0.7, 0.9}; penalty factor λ: {0.001, 0.01, 0.1, 1,10}; subspace rate: {0.1, 0.3, 0.5, 0.7, 0.9}; number of ensembles: 10; base learner: SVR.

Table 5. The RMSE (mean and SD) of different methods.

	DCP		SRU (SO_2)		SRU (H_2S)	
Methods	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)
SAE	4.982	0.354	1.680	0.213	2.171	0.253
Bi-LSTM	4.884	0.570	1.452	0.275	1.764	0.305
SVR	4.851	0.308	1.104	0.124	1.584	0.144
Bagging	4.948	0.229	1.090	0.086	1.577	0.081
Adaboost	5.046	0.268	1.166	0.063	2.575	0.986
RF	3.690	0.202	1.085	0.044	1.369	0.107
XGBoost	3.858	0.210	1.108	0.076	1.352	0.065
RS	4.316	0.233	1.104	0.083	1.401	0.101
RS-LASSO	3.957	0.213	1.013	0.064	1.212	0.082
RS-SBL	**3.593**	0.180	**0.964**	0.069	**1.124**	0.074

Table 6. The R^2 (mean and SD) of different methods.

	DCP		SRU (SO_2)		SRU (H_2S)	
Methods	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)
SAE	91.60	1.424	77.36	3.161	85.85	2.813
Bi-LSTM	93.05	1.476	80.02	3.384	88.62	3.157
SVR	93.16	0.603	81.28	2.692	90.29	2.733
Bagging	92.99	0.364	81.66	2.668	90.37	0.829
Adaboost	93.13	0.547	78.90	2.293	72.17	18.93
RF	95.16	0.620	81.38	1.680	94.19	0.769
XGBoost	93.93	0.419	83.51	2.625	94.12	0.792
RS	94.36	0.452	81.04	2.673	92.10	1.384
RS-LASSO	94.84	0.437	84.85	2.647	94.44	0.730
RS-SBL	**95.20**	0.424	**86.32**	2.602	**95.37**	0.702

respectively. Besides, the proposed RS-SBL based on the fusion features achieves a lower standard deviation, showing that the RS-SBL has better stability and generalization ability.

Secondly, the SVR based on fusion features achieves better performance in comparison with SAE and Bi-LSTM. Taking DCP as an example, the RMSE of SVR under the fusion features decreases by 2.629% and 0.676% in comparison with SAE and Bi-LSTM, respectively. Besides, the SVR based on the fusion features achieves a lower standard deviation than SAE and Bi-LSTM, showing its steadiness in comparison with deep learning methods.

Finally, compared with the SVR, most of the ensemble learning methods achieve better performance under the RMSE and R^2 metrics, showing that the ensemble scheme can improve the performance of the soft sensor models. Taking the SRU (SO_2) as an example, the RMSE and R^2 values of bagging are 1.090% and 81.66%, while the values of the SVR are 1.104% and 81.28%, respectively, indicating the advantage of the ensemble learning methods in soft sensing applications. Moreover, among ensemble learning methods, it is obvious that the RS-based methods have advantages over bagging and Adaboost, showing that the RS is more suitable for soft sensors when facing high-dimensional data.

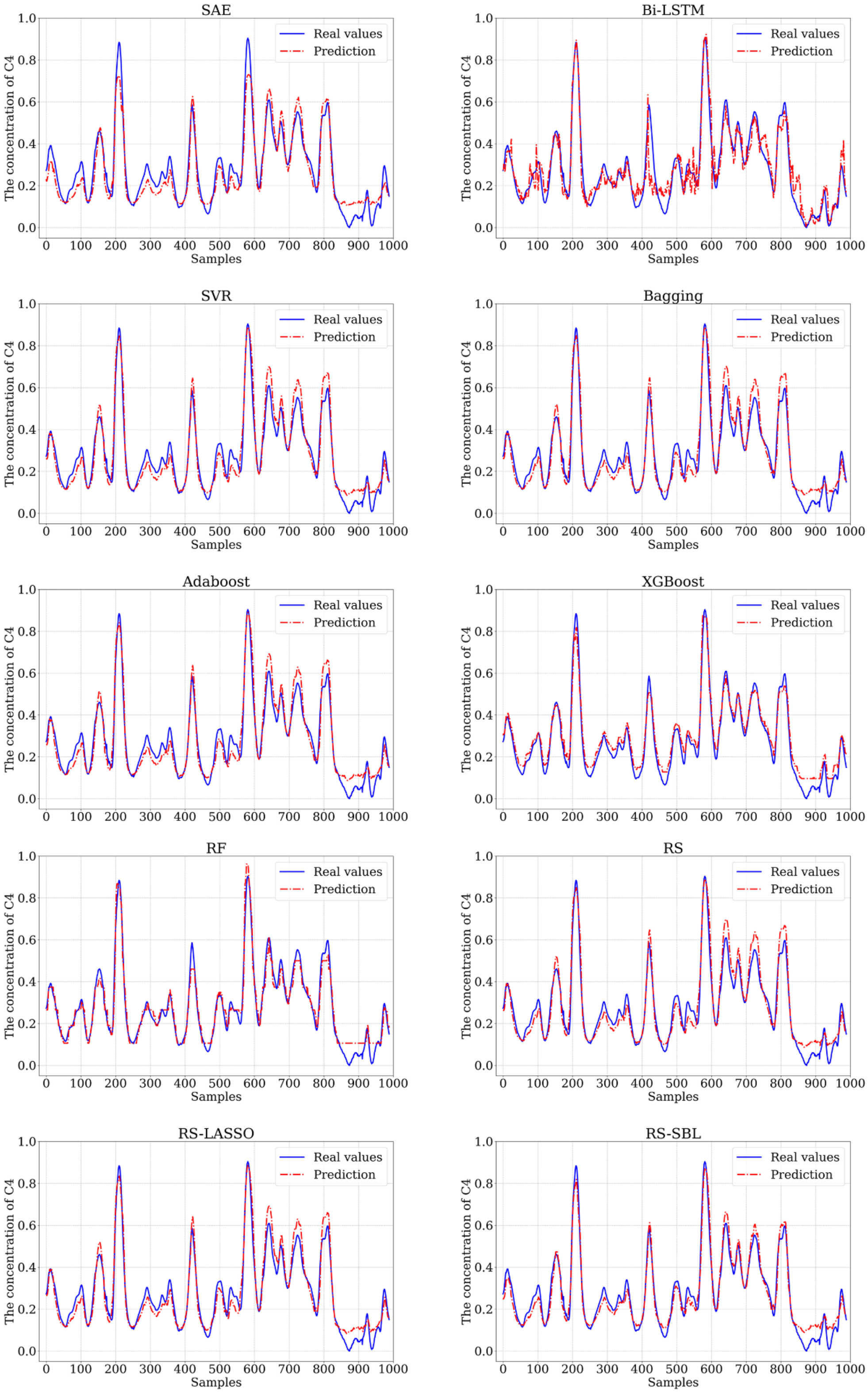


Figure 5. Detailed prediction results of the RS-SBL and other comparison methods.

5.2. Discussion

5.2.1. Evaluation of the different soft sensor models. To validate the effectiveness of the RS-SBL method, the prediction results of the proposed RS-SBL and other comparison methods are compared. Taking the DCP as an example, the detailed prediction results of the RS-SBL and other comparison methods when the fusion features are used as inputs are shown in figure 5.

As shown in figure 5, compared with other comparison methods, the predicted values of the RS-SBL best tracks the changes of real values in the testing data set. The prediction

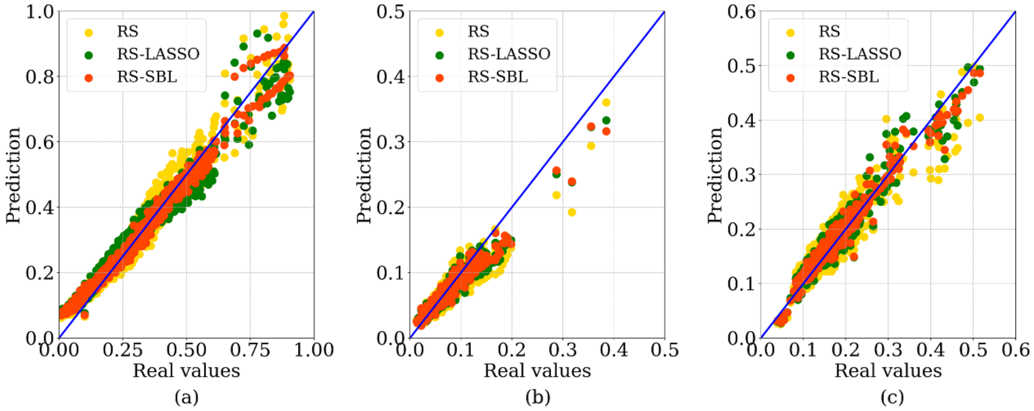


Figure 6. Scatterplots of predicted and real values of RS, RS-LASSO, and RS-SBL: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

errors mainly come from some outliers with extremely small or large quality variables. That is mainly because there are not sufficient samples to provide the information for training the soft sensor models [13]. Moreover, the performance of Adaboost is vulnerable to noise data, which may lead to the phenomenon of overfitting.

Furthermore, to validate the effectiveness of the proposed WSGL, the performance of the RS, RS-LASSO, and RS-SBL under the fusion features were compared. The detailed prediction results on the testing data set of the RS, RS-LASSO, and RS-SBL are shown in figure 6. It shows the scatterplots of the predicted and real values of the quality variables. Each point represents a test sample. The x-axis and y-axis are the real values and prediction of test samples, respectively.

As shown in figure 6, it is easy to find that the points of the RS-SBL are located more densely along the diagonal line than RS and RS-LASSO, showing that the WSGL can effectively select important features and generate high-quality feature subsets. The reason is that the unsupervised and supervised deep representation features have different contributions to predicting the quality variables, even if they are in the same feature group. The proposed WSGL method can effectively filter out the redundant and irrelevant features because it can adaptively and flexibly select the significant features by considering the importance of feature groups and individual features.

5.2.2. Evaluation of the fusion features. To validate the superiority of the fusion features, the performance of different features including unsupervised (F1), supervised deep representation features (F2), and the fusion features (F1 + F2) is compared. The RMSE and R^2 values from different methods using the above features as inputs are illustrated in figures 7 and 8.

As shown in figures 7 and 8, it can be observed that the performance of the fusion features is superior to the individual features. Taking the RS-SBL on DCP for example, the RMSE value of the fusion features decreases by 42.21% in comparison with F1, and 5.10% in comparison with F2. Correspondingly, the R^2 value of the fusion features increases by 5.62% in comparison with F1, and 1.22% in comparison with F2. This is mainly because the fusion features contain more comprehensive information, which is beneficial to predict the quality variables. Besides, as for the individual features, the supervised deep representation features show better prediction results than the unsupervised deep representation features. Taking the RS-SBL on the SRU (SO_2) for example, the RMSE of the F2 decreases by 19.48%, and the R^2 increases by 36.89% in comparison with F1. That is mainly because the supervised deep representation features extracted by the Bi-LSTM are relevant to the quality variables and contain the dynamic information in industrial processes, while the unsupervised deep representation features extracted by the SAE mainly focus on the data itself and may be irrelevant to quality variables.

Furthermore, the effect of the ensemble scheme and fusion features in RS-SBL is discussed. As shown in figures 7 and 8, taking the DCP as an example, the RMSE value of the RS-SBL decreases by 4.390% (F1), 21.97% (F2), and 27.47% (F1 + F2) in comparison with SVR, showing the effectiveness of the ensemble scheme. Besides, the RMSE value of the RS-SBL under fusion features decreases by 42.21% in comparison with F1, and 5.102% in comparison with F2, showing the superiority of the fusion features. Therefore, the improvement of the ensemble scheme is better than that of the fusion features when F2 is the individual feature. Correspondingly, the improvement of the fusion features is superior to the ensemble scheme when F1 is the individual feature.

Finally, to validate and evaluate the complementary effect of different kinds of features, the performance of RS, RS-LASSO, and RS-SBL is compared. The input of RS and RS-LASSO is the direct combination of two kinds of feature, ignoring the complementary information and not distinguishing different features. The proposed RS-SBL has superiority in comparison with RS and RS-LASSO, showing that the complementarity of different features can enhance the prediction performance. The correlation analysis of different features is shown in figure 9.

As can be seen in figure 9, the features have strong redundant components in the same feature group, showing that introducing the feature selection methods to reduce the redundant and irrelevant features is necessary. The different

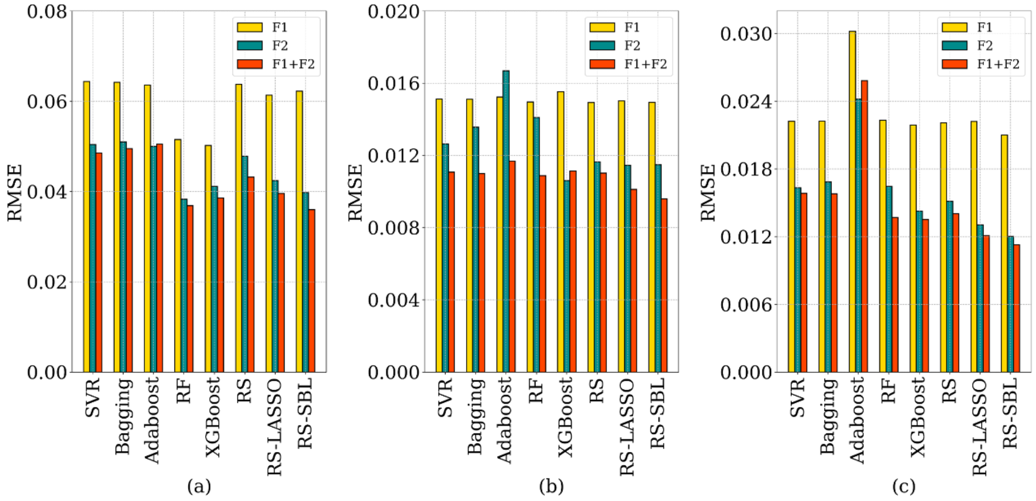


Figure 7. RMSE comparison of different features on two data sets: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

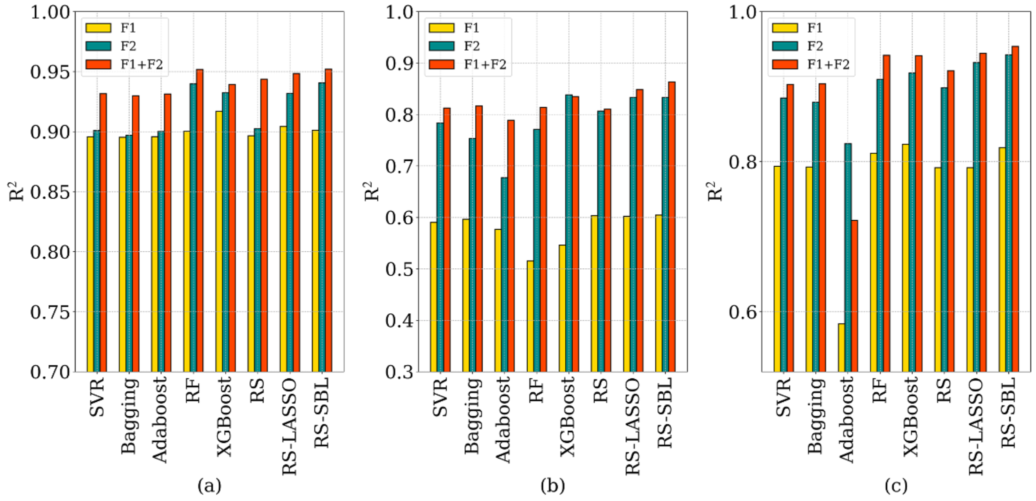


Figure 8. R^2 comparison of different features on two data sets: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

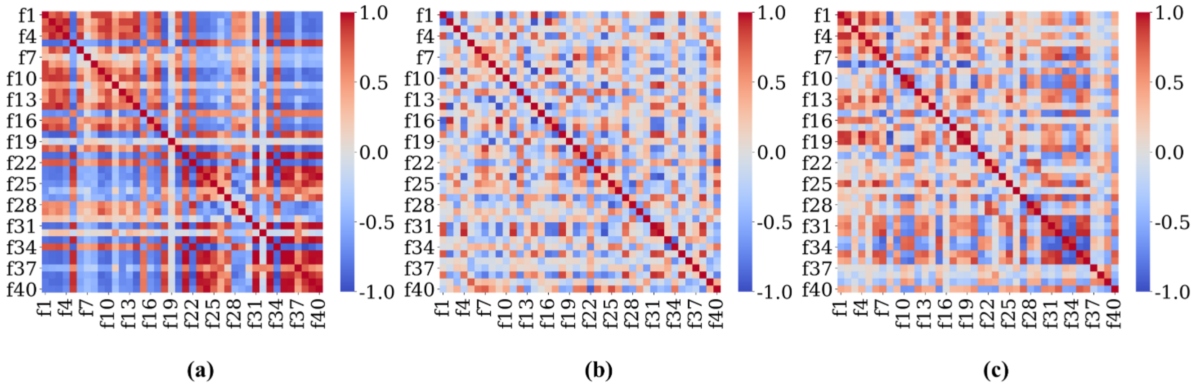


Figure 9. Feature correlation analysis on different data sets (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

feature groups have fewer inner correlations, indicating that the unsupervised and supervised deep representation features exploit distinct prediction information. And the weights of different features are shown in figure 10. The unsupervised deep representation features are represented as f1–f20, and the supervised deep representation features are represented as f21–f40. As shown in figure 10, the distinctions of different features are well identified, showing that the WSGL can evaluate the features and feature groups effectively. There are important features in both unsupervised and supervised feature

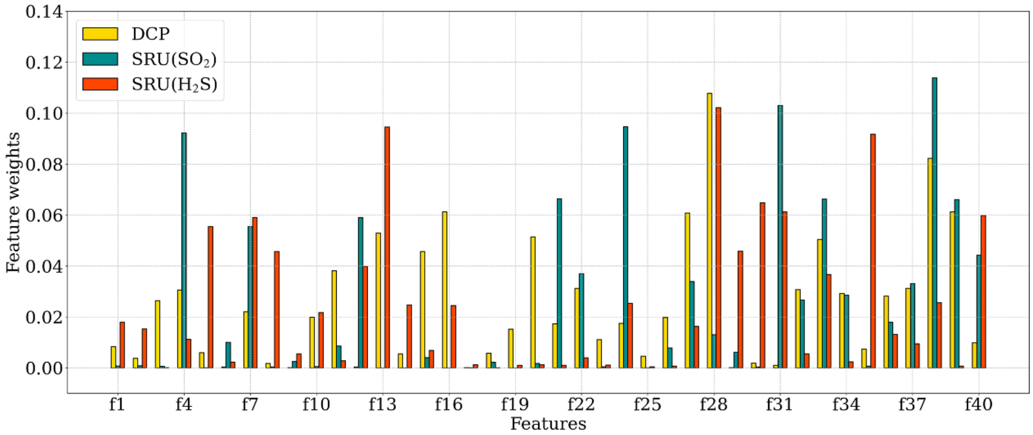


Figure 10. Feature weights on the two kinds of feature.

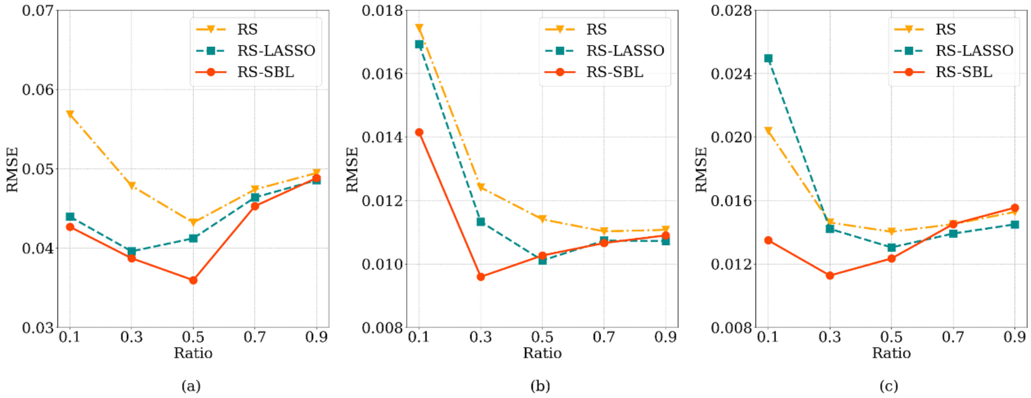


Figure 11. RMSE comparisons of RS, RS-LASSO, and RS-SBL on two data sets (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

groups, showing that these two kinds of feature can complement each other and the fusion features contain more comprehensive information.

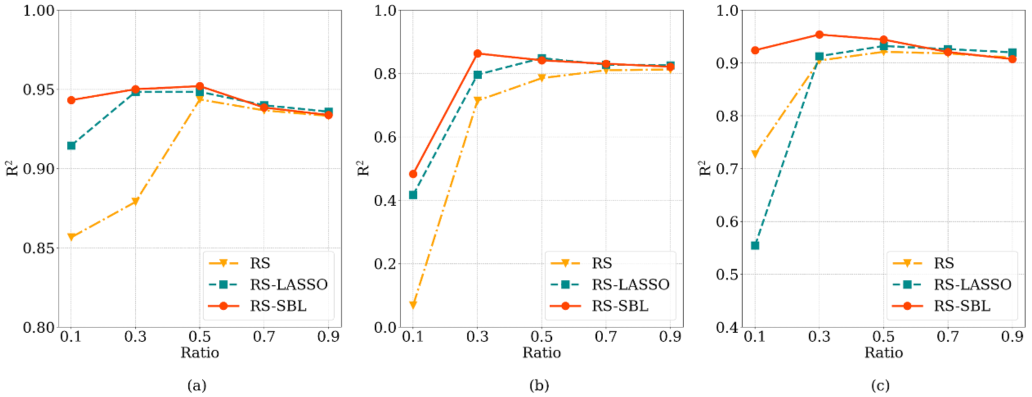
5.2.3. *Evaluation of the influence of the key parameters.*

The key parameters of the RS-SBL have a significant effect on the performance of predicting the quality variables. Thus, some key parameters, including subspace rate ratio, combination coefficient α, and penalty factor λ are discussed. Firstly, the influence of the subspace rate ratio is shown in figures 11 and 12.

As shown in figures 11 and 12, the proposed RS-SBL achieves the best performance when the ratio is set to 0.5 (DCP), 0.3 (SRU (SO_2)), and 0.3 (SRU (H_2S). As the ratio changes from 0.1 to 0.9, the RMSE values of the RS-SBL decrease and then increase gradually. Taking the DCP as an example, the RMSE values of the RS, RS-LASSO, and RS-SBL are 5.68%, 4.40%, and 4.27%, respectively, when the ratio is set to 0.1. Correspondingly, when the ratio is set to 0.9, the RMSE values of these three methods are 4.95%, 4.85%, and 4.88%, respectively. That is mainly because the original feature space contains some irrelevant and redundant features. Additionally, compared with RS and RS-LASSO, the proposed RS-SBL has the best performance and is least affected by the ratio. The reason for this phenomenon is that the important features are effectively selected with a higher probability by the WSGL.

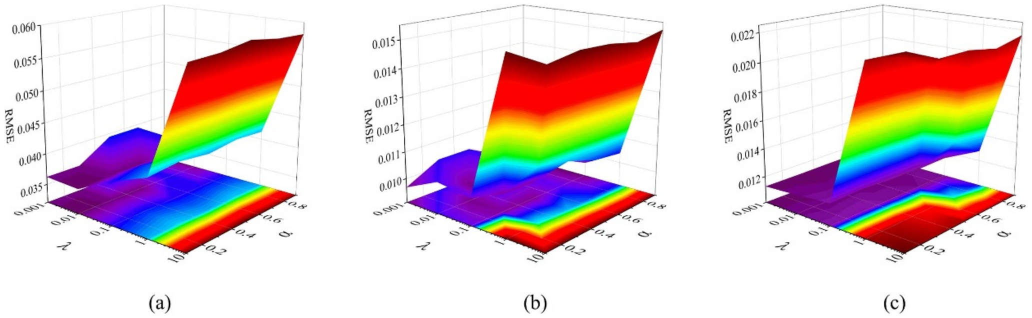
Furthermore, the effect of the combination coefficient α and penalty factor λ of the WSGL are discussed. Figures 13 and 14 illustrate the RMSE and R^2 values for different α and λ on the two types of data sets. The x-axis and y-axis represent the α and λ, respectively, and the z-axis is RMSE or R^2 value.

As shown in figures 13 and 14, the proposed RS-SBL has best performance when $\alpha = 0.1/\lambda = 0.01$ (DCP), $\alpha = 0.9/\lambda = 0.01$ (SRU (SO_2)), and $\alpha = 0.3/\lambda = 0.001$ (SRU (H_2S)). It is worth noting that the feature groups are more significant when α is closer to 0, showing that considering the weights of the different feature groups is necessary. Besides, when α is fixed, the performance of the RS-SBL is different when λ is varied from 0.001 to 10.0, showing that λ is a key parameter that can affect the performance of the RS-SBL. In conclusion, we can achieve accurate prediction results of the quality variables when some key parameters, such as ratio, α, and λ are selected properly.



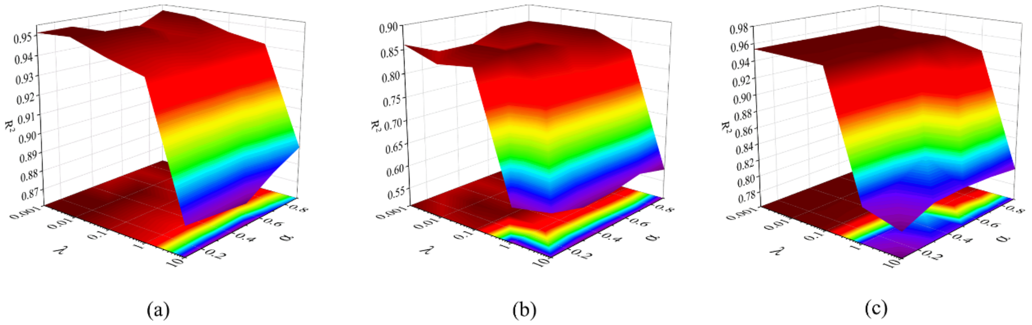
(a) (b) (c)

Figure 12. R^2 comparisons of RS, RS-LASSO, and RS-SBL on two data sets: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).



(a) (b) (c)

Figure 13. Sensitivity analysis of RMSE for the RS-SBL on two data sets: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).



(a) (b) (c)

Figure 14. Sensitivity analysis of R^2 for the RS-SBL on two data sets: (a) DCP; (b) SRU (SO_2); (c) SRU (H_2S).

6. Conclusions and future work

Soft sensors can provide information about key quality variables for process control and optimization, which is of significance in industrial processes. Existing studies have investigated unsupervised and supervised deep representation features for soft sensors but the complementary information between the two types of feature has been ignored and how to reduce irrelevant and redundant features remains to be explored. To address these issues, a novel RS method named RS-SBL is proposed to fuse unsupervised and supervised deep representation features for soft sensors. The core aspects of the proposed method are as follows. Firstly, the deep representation features are extracted from unsupervised and supervised methods. Secondly, RS-SBL is proposed to discriminate the relative importance of different features and generate accurate prediction results by taking advantage of the complementary information of the fusion features. To validate the superiority of the proposed RS-SBL, empirical studies on an industrial DCP and SRU are conducted. Experimental results show that the RS-SBL provides more accurate prediction results than the comparison methods.

Although the proposed method achieves superior performance, this method can be further explored. Firstly, the effectiveness of the RS-SBL needs to be further validated in more complex industrial processes. Secondly, consideration of the other structured sparsity learning methods is needed to select important features and enhance the performance of the base learners. Thirdly, introducing more effective aggregation strategies to consider the weights of the base learners is necessary. Finally, because the RS-SBL is computationally intensive, parallel computing techniques need to be utilized to address this problem in our future work.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (72071062), Anhui Provincial Key Research and Development Program (202104a05020038), and Fundamental Research Funds for the Central Universities (PA2021KCPY0032).

ORCID iDs

Gang Wang https://orcid.org/0000-0002-6395-9409
Zhangjun Wu https://orcid.org/0000-0003-2414-5768

References

[1] Shardt Y A W, Hao H and Ding S X 2015 A new soft-sensor-based process monitoring scheme incorporating infrequent KPI measurements *IEEE Trans. Ind. Electron.* **62** 3843–51

[2] Kano M and Nakagawa Y 2008 Data-based process monitoring, process control, and quality improvement: recent developments and applications in steel industry *Comput. Chem. Eng.* **32** 12–24

[3] de Morais G A P, Barbosa B H G, Ferreira D D and Paiva L S 2019 Soft sensors design in a petrochemical process using an evolutionary algorithm *Measurement* **148** 106920

[4] Curreri F, Graziani S and Xibilia M G 2020 Input selection methods for data-driven soft sensors design: application to an industrial process *Inf. Sci.* **537** 1–17

[5] Kadlec P, Gabrys B and Strandt S 2009 Data-driven soft sensors in the process industry *Comput. Chem. Eng.* **33** 795–814

[6] Cong Q and Yu W 2018 Integrated soft sensor with wavelet neural network and adaptive weighted fusion for water quality estimation in wastewater treatment process *Measurement* **124** 436–46

[7] Geng Z, Chen Z, Meng Q and Han Y 2022 Novel transformer based on gated convolutional neural network for dynamic soft sensor modeling of industrial processes *IEEE Trans. Ind. Inf.* **18** 1521–9

[8] Guo R, Liu H, Wang W, Xie G and Zhang Y 2021 A hybrid-driven soft sensor with complex process data based on DAE and mechanism-introduced GRU *2021 IEEE 10th Data Driven Control and Learning Systems Conf. (DDCLS)* pp 553–8

[9] Zhang J, Chen H, Chen S and Hong X 2019 An improved mixture of probabilistic PCA for nonlinear data-driven process monitoring *IEEE Trans. Cybern.* **49** 198–210

[10] Shang C, Huang X, Suykens J A K and Huang D 2015 Enhancing dynamic soft sensors based on DPLS: a temporal smoothness regularization approach *J. Process Control* **28** 17–26

[11] Ge Z, Yang C and Song Z 2009 Improved kernel PCA-based monitoring approach for nonlinear processes *Chem. Eng. Sci.* **64** 2245–55

[12] Si Y, Wang Y and Zhou D 2020 Key-performance-indicator-related process monitoring based on improved kernel partial least squares *IEEE Trans. Ind. Electron.* **68** 2626–36

[13] Yuan X, Huang B, Wang Y, Yang C and Gui W 2018 Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE *IEEE Trans. Ind. Inf.* **14** 3235–43

[14] Aguirre L A, Teixeira B O S, Barbosa B H G, Teixeira A F, Campos M C M M and Mendes E M A M 2017 Development of soft sensors for permanent downhole gauges in deepwater oil wells *Control Eng. Pract.* **65** 83–99

[15] Yuan X, Wang Y, Yang C and Gui W 2020 Stacked isomorphic autoencoder based soft analyzer and its application to sulfur recovery unit *Inf. Sci.* **534** 72–84

[16] Liu Y, Yang C, Gao Z and Yao Y 2018 Ensemble deep kernel learning with application to quality prediction in industrial polymerization processes *Chemom. Intell. Lab. Syst.* **174** 15–21

[17] Yuan X, Gu Y and Wang Y 2021 Supervised deep belief network for quality prediction in industrial processes *IEEE Trans. Instrum. Meas.* **70** 1–11

[18] Liu C, Wang K, Wang Y, Xie S and Yang C 2021 Deep nonlinear dynamic feature extraction for quality prediction based on spatiotemporal neighborhood preserving SAE *IEEE Trans. Instrum. Meas.* **70** 1–10

[19] Xie W, Wang J, Xing C, Guo S, Guo M and Zhu L 2021 Variational autoencoder bidirectional long and short-term memory neural network soft-sensor model based on batch training strategy *IEEE Trans. Ind. Inf.* **17** 5325–34

[20] Pan H, Su T, Huang X and Wang Z 2020 LSTM-based soft sensor design for oxygen content of flue gas in coal-fired power plant *Trans. Inst. Meas. Control* **43** 78–87

[21] Schmidhuber J 2015 Deep learning in neural networks: an overview *Neural Netw.* **61** 85–117

[22] Makantasis K, Karantzalos K, Doulamis A and Doulamis N 2015 Deep supervised learning for hyperspectral data classification through convolutional neural networks *2015 IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS)* pp 4959–62

[23] Gonzaga J C B, Meleiro L A C, Kiang C and Maciel Filho R 2009 ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process *Comput. Chem. Eng.* **33** 43–49

[24] Kaneko H and Funatsu K 2014 Application of online support vector regression for soft sensors *AlChE J.* **60** 600–12

[25] Huang G-B, Zhou H, Ding X and Zhang R 2011 Extreme learning machine for regression and multiclass classification *IEEE Trans. Syst. Man Cybern.* B **42** 513–29

[26] Wang Z-Y, Lu C and Zhou B 2018 Fault diagnosis for rotary machinery with selective ensemble neural networks *Mech. Syst. Signal Process.* **113** 112–30

[27] Kaneko H and Funatsu K 2014 Adaptive soft sensor based on online support vector regression and Bayesian ensemble learning for various states in chemical plants *Chemom. Intell. Lab. Syst.* **137** 57–66

[28] Shao W, Chen S and Harris C J 2018 Adaptive soft sensor development for multi-output industrial processes based on selective ensemble learning *IEEE Access* **6** 55628–42

[29] Ge Z and Song Z 2014 Ensemble independent component regression models and soft sensing application *Chemom. Intell. Lab. Syst.* **130** 115–22

[30] Fortuna L, Graziani S, Rizzo A and Xibilia M G 2007 *Soft Sensors for Monitoring and Control of Industrial Processes* (Berlin: Springer)

[31] Singh H, Pani A K and Mohanta H K 2019 Quality monitoring in petroleum refinery with regression neural network: improving prediction accuracy with appropriate design of training set *Measurement* **134** 698–709

[32] Wang K, Shang C, Yang F, Jiang Y and Huang D 2017 Automatic hyper-parameter tuning for soft sensor modeling based on dynamic deep neural network *2017 IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC)* pp 989–94

[33] Yuan X, Qi S and Wang Y 2020 Stacked enhanced auto-encoder for data-driven soft sensing of quality variable *IEEE Trans. Instrum. Meas.* **69** 7953–61

[34] Zamprogna E, Barolo M and Seborg D E 2005 Optimal selection of soft sensor inputs for batch distillation columns using principal component analysis *J. Process Control* **15** 39–52

[35] Yang H and Huang M 2010 A soft sensor based on kernel PCA and composite kernel support vector regression for a flotation circuit *2nd Int. Conf. on Advanced Computer Control* vol 5 pp 375–8

[36] Wang Z X, He Q P and Wang J 2015 Comparison of variable selection methods for PLS-based soft sensor modeling *J. Process Control* **26** 56–72

[37] Kim S, Kano M, Hasebe S, Takinami A and Seki T 2013 Long-term industrial applications of inferential control based on just-in-time soft-sensors: economical impact and challenges *Ind. Eng. Chem. Res.* **52** 12346–56

[38] Zhu C-H and Zhang J 2019 Developing soft sensors for polymer melt index in an industrial polymerization process using deep belief networks *Int. J. Autom. Comput.* **17** 44–54

[39] Yan X, Wang J and Jiang Q 2020 Deep relevant representation learning for soft sensing *Inf. Sci.* **514** 263–74

[40] Yuan X, Qi S, Wang Y and Xia H 2020 A dynamic CNN for nonlinear dynamic feature learning in soft sensor modeling of industrial process data *Control Eng. Pract.* **104** 104614

[41] Kataria G and Singh K 2018 Recurrent neural network based soft sensor for monitoring and controlling a reactive distillation column *Chem. Prod. Process Model.* **13** 20170044

[42] Wang K, Gopaluni R B, Chen J and Song Z 2020 Deep learning of complex batch process data and its application on quality prediction *IEEE Trans. Ind. Inf.* **16** 7233–42

[43] Zhu X, Hao K, Xie R and Huang B 2021 Soft sensor based on extreme gradient boosting and bidirectional converted gates long short-term memory self-attention network *Neurocomputing* **434** 126–36

[44] Fortuna L, Giannone P, Graziani S and Xibilia M G 2007 Virtual instruments based on stacked neural networks to improve product quality monitoring in a refinery *IEEE Trans. Instrum. Meas.* **56** 95–101

[45] Chitralekha S B and Shah S L 2010 Application of support vector regression for developing soft sensors for nonlinear processes *Can. J. Chem. Eng.* **88** 696–709

[46] Wang Y, Wu D and Yuan X 2019 A two-layer ensemble learning framework for data-driven soft sensor of the diesel attributes in an industrial hydrocracking process *J. Chemom.* **33** e3185

[47] He Y, Geng Z and Zhu Q 2016 Soft sensor development for the key variables of complex chemical processes using a novel robust bagging nonlinear model integrating improved extreme learning machine with partial least square *Chemom. Intell. Lab. Syst.* **151** 78–88

[48] Zhu X, Zhang P and Xie M 2021 A joint long short-term memory and adaboost regression approach with application to remaining useful life estimation *Measurement* **170** 108707

[49] Fortuna L, Graziani S and Xibilia M G 2005 Soft sensors for product quality monitoring in debutanizer distillation columns *Control Eng. Pract.* **13** 499–508

[50] Fortuna L, Rizzo A, Sinatra M and Xibilia M 2003 Soft analyzers for a sulfur recovery unit *Control Eng. Pract.* **11** 1491–500