Chinese Journal of Chemical Engineering 85 (2025) 238–250

Contents lists available at ScienceDirect

Chinese Journal of Chemical Engineering

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

Full Length Article

A soft sensing method for mechanical properties of hot-rolled strips based on improved co-training

Bowen Shi, Jianye Xue, Hao Ye *

Department of Automation, Tsinghua University, Beijing 100084, China

A R T I C L E I N F O

Article history:
Received 8 November 2024
Received in revised form
11 April 2025
Accepted 14 April 2025
Available online 13 May 2025

Keywords:
Mechanical properties
Co-training
Soft sensing method

A B S T R A C T

Accurately soft sensing of the mechanical properties of hot-rolled strips is essential to ensure product quality, optimize production, and reduce costs. However, it faces the difficulty caused by limited labeled samples, for which co-training based semi-supervised learning offers a potential solution. So in this paper, a novel soft sensing method for mechanical properties based on improved co-training (ICO) is proposed. Compared with the existing co-training framework, the proposed ICO introduces improvements from the aspects of multiple view partition, confidence estimation, and pseudo-label assignment. Specifically, (i) in the stage of multiple view partition, ICO integrates metallurgical mechanisms of hot rolling processes and statistical mutual information to achieve a balance between view sufficiency and independence, which improves model performance and interpretability; (ii) in the stage of confidence estimation, ICO evaluates the confidence of unlabeled samples at the cluster level rather than at the level of a single sample, which facilitates the exploration of sample distribution and the selection of representative samples; (iii) in the pseudo-label assignment stage, ICO adopts a safe pseudo-label algorithm (which is called SAFER by its author and originally used for each single sample) to assign pseudo-labels for cluster of samples with the highest confidence determined in the previous step stage, to take advantage of the merit of handling unlabeled samples at the cluster level mentioned above on one hand, and the merit of SAFER in enhancing the quality of pseudo-labels on the other hand. The proposed soft sensing method effectively predicts mechanical properties on the real hot rolling dataset, achieving approximately 5% improvement in R^2 compared to traditional supervised learning.

1. Introduction

In the steel industry, the hot rolling process is crucial for producing high-performance steel, which involves passing steel billets through high-temperature rolling machinery to modify their shape and properties. Typically conducted when the steel billets are in a high-temperature state, this process enhances material pliability, eliminates internal stresses, and improves physical properties [1].

The mechanical properties of hot-rolled strips, including yield strength (YS), tensile strength (TS), and elongation (EL), are important indicators to evaluate their ability to withstand various external forces [2]. However, real-time measurement of the mechanical properties proves elusive due to intricate production models, coupled variable relationships, and time-consuming inspection procedures [3]. Traditional mechanical properties testing procedures, typically conducted after production, involve severing samples several meters from the coil's end. This process, which includes sample preparation and physical testing, is time-consuming and reduces steel coil yields due to destructive cutting.

Soft sensing is an effective way to deal with the difficulty. It aims to predict these important yet challenging-to-measure mechanical properties by establishing models that link readily measurable process variables with the target variables and can be further used to optimize processes, guarantee product quality, and shorten production cycles [4,5]. Many researchers have developed soft sensing models based on physical metallurgy principles to predict the mechanical properties of hot-rolled strips [6]. However, the hot rolling process is inherently complex, featuring uncertainty, high nonlinearity, and strong coupling. So the mechanism-based soft sensing models may struggle to achieve accurate predictions of mechanical properties and fail to meet the requirements of actual demands [7].

Consequently, an increasing number of studies have embraced data-driven soft sensing models for hot-rolled strips, leveraging extensive hot-rolling historical data, including alloy composition, rolling process parameters, and subsequent heat treatments [8].

* Corresponding author.
E-mail address: haoye@tsinghua.edu.cn (H. Ye).

https://doi.org/10.1016/j.cjche.2025.04.010

Yang *et al.* [9] constructed a Bayesian neural network with twenty-two features, including chemical composition, geometric dimensions, and rolling parameters, to predict the tensile strength of hot-rolled products. Xu *et al.* [10] converted one-dimensional rolling process data into a two-dimensional format and applied a convolutional neural network to create a soft sensor for measuring mechanical properties. Mohanty *et al.* [11] developed a deep neural network for the prediction of elongation, yield strength, and ultimate tensile strength of cold-rolled coils.

However, since the mechanical properties inspection in actual production is only carried out on a small number of sampled steel coils, it's difficult to accumulate enough data with the labels of mechanical properties, which hinders the achievement of an accurate soft sensing model based on supervised learning methods. Semi-supervised learning offers a promising solution to address this issue, which is applicable to the situations where the amount of labeled samples is limited, while unlabeled samples (*i.e.* samples with the measurable process variables but without the mechanical properties) are easily accessible. Semi-supervised learning dexterously utilizes unlabeled samples to broaden the training set, thereby improving the performance of the soft sensing model [12,13].

Currently, many studies have implemented semi-supervised learning for industrial soft sensors, which can be mainly classified into four categories: active learning, graph, self-training, and co-training based methods [14]. In active learning based methods, the most informative unlabeled samples are manually labeled for training a good performance model [15,16], which benefits from expert knowledge, but is tedious and time-consuming [17]. Graph-based methods build a graph structure based on the relationships between labeled and unlabeled samples [18,19]. In this kind of approaches, samples are typically defined as the graph's nodes, and the similarity between samples is depicted as the graph's edges. Then the pseudo-labels are assigned to unlabeled samples by label propagation. Since graph-based methods consider local relationships between data points, they can be robust to outliers [20]. However, the requirement of a known data distribution for constructing the graph limits the application of these methods in dealing with data of complex distributions [21]. In addition, the construction of complex graph structures may lead to increased computational and storage burdens [22].

Compared to active learning based methods, both self-training and co-training based methods augment the labeled dataset through pseudo-labeling of unlabeled data. This reduces the need for human annotators to label additional data, resulting in lower annotation costs. In contrast to graph-based methods, self-training and co-training based methods do not rely on explicit assumptions about data distribution or graph structure, and therefore are more adaptable to various datasets [23]. In the realm of soft sensors for mechanical properties in hot rolling using semi-supervised learning, most research has focused on self-training based methods. Wu *et al.* [24] proposed a self-training soft sensing model supporting Bayesian optimization deep neural network to elevate mechanical properties by using both the labeled and unlabeled data. Dong *et al.* [25] proposed a just-in-time learning-based soft sensor for the mechanical properties of strip steel *via* multi-block weighted semi-supervised models. They prioritized selecting unlabeled samples that were most similar to labeled samples. And the models' parameters were updated by utilizing the expanded dataset including these unlabeled samples. One difference between self-training and co-training lies in that self-training only uses a single model, whereas co-training utilizes multiple models [26]. Soft sensing based on co-training involves training multiple models on the sufficient and redundant views [27]. Throughout the training phase, each base model extracts valuable information from unlabeled data and improves performance through mutual collaboration [28].

Suppose the dataset X for training can be divided into a labeled dataset $X_L = \{L, Y_L\}$ and an unlabeled dataset $X_U = \{U, Y_U\}$, where L and U are known feature matrices, Y_L is the known label matrix corresponding to L, and Y_U is the unknown outputs corresponding to U. Taking the case of two views as an example, Fig. 1 illustrates the general framework of co-training methods, which comprise the following four steps [26]: (i) View partition. The features of L and U are divided into View 1 and View 2. Then correspondingly, L is divided into L_1 and L_2, and similarly, U is also divided into U_1 and U_2; (ii) Confidence estimation. By feeding each unlabeled sample in U_1 to Base Model 1 (denoted by h_1 in Fig. 1), which is trained on $\{L_1, Y_L\}$, the label confidence of unlabeled samples in U_1 is evaluated one by one. Following the similar procedure, the label confidence of the unlabeled samples

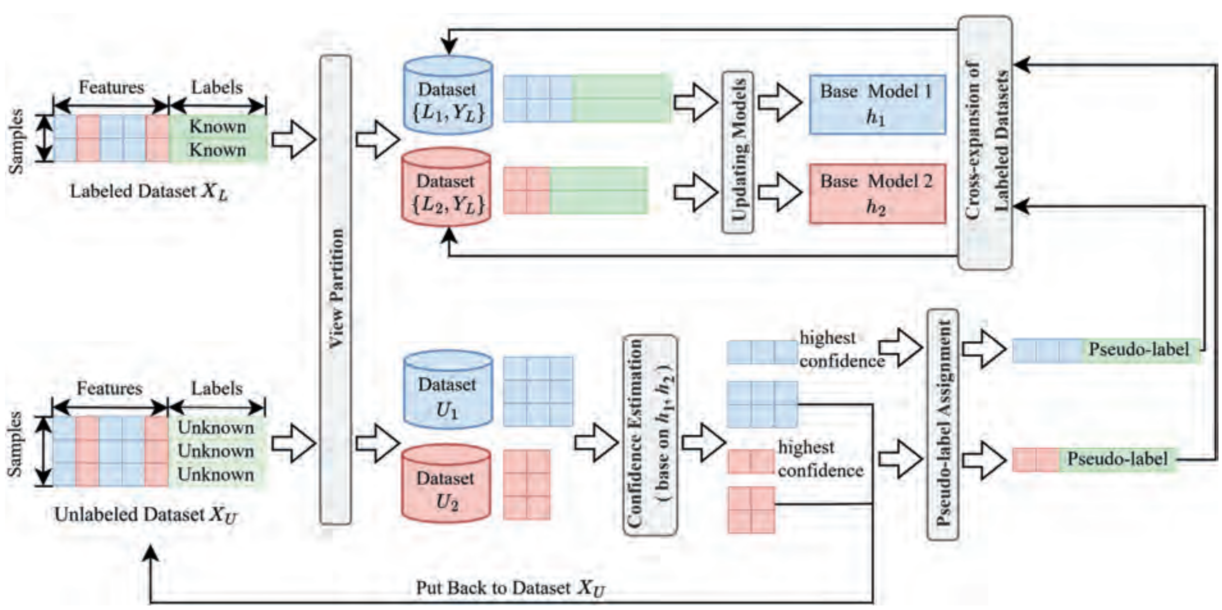

Fig. 1. Framework of co-training. To better distinguish features in View 1 and View 2, blue squares and red squares are respectively used to represent them in this figure.

in U_2 is also assessed by feeding them to Base Model 2 (denoted by h_2 in Fig. 1), which is trained on $\{L_2, Y_L\}$; (iii) Pseudo-label assignment. Pseudo-labels are assigned to the most confident unlabeled samples in U_1 and U_2, respectively; (iv) Cross-expansion of labeled datasets and updating models. The most confident unlabeled samples from U_1 and U_2, along with their corresponding pseudo-labels, are added to the labeled datasets $\{L_2, Y_L\}$ and $\{L_1, Y_L\}$, respectively, in a crossing way. Then with the expanded labeled datasets $\{L_1, Y_L\}$ and $\{L_2, Y_L\}$, h_1 and h_2 can be updated, respectively. This iterative process of confidence estimation, pseudo-label assignment, and cross-expansion of labeled datasets allows the co-training framework to leverage unlabeled data effectively and improve the performance of the models through mutual collaboration [26].

Since step (iv), *i.e.* cross-expansion of labeled datasets and updating models, remains consistent across co-training methods [26,29,30], and no modifications are made to this step in our study, we mainly review the primary strategies adopted by existing co-training methods in steps (i) to (iii) *i.e.* view partition, confidence estimation, and pseudo-label assignment. In the following, we will describe the current methods for each of these steps, and discuss their shortcomings and limitations.

Step (i) of co-training, view partition, as shown in Fig. 1, divides the datasets X_L and X_U into two views $\{L_1, Y_1, U_1\}$ and $\{L_2, Y_2, U_2\}$. Typically, when partitioning views, two assumptions, *i.e.* the sufficiency assumption and the independence assumption are expected to be satisfied. The sufficiency assumption refers to that the data partitioned into each individual view is sufficient to predict the true label, while the independence assumption states that the data in all views are conditionally independent [30,31]. Currently, there are three main strategies for view partitioning [26]: (i) random view partitioning [32,33], (ii) independence-based view partitioning [34,35], and (iii) sufficiency-based view partitioning [36]. Random view partitioning adopts the random division of original features, which is inherently blind without considering the two assumptions and is time-consuming [37]. Independence-based and sufficiency-based view partitioning methods focus on either the independence assumption or the sufficiency assumption, but cannot effectively balance them, because as indicated in [38], the two assumptions are inherently contradictory. Specifically, when views are independent, the correlation among features within each view is strong, making it challenging to ensure the sufficiency of views. Conversely, when the sufficiency of views is strong, the features in each view will be more comprehensive, and the independence between views will diminish. In addition, the view partitioning strategies mentioned above do not leverage domain-specific knowledge, resulting in less interpretable partitioning results [26]. Hence, during the step of view partitioning in co-training, it remains challenging to balance the independence and sufficiency of the views.

In step (ii) of co-training, confidence estimation is to evaluate the label confidence of the unlabeled samples in U_1 and U_2, as shown in Fig. 1. It is an essential step in co-training, because correct confidence evaluation on each unlabeled sample is the premise of selecting the most proper unlabeled sample and assigning a pseudo-label to it in step (iii) and step (iv). At present, the evaluation methods of label confidence can be summarized into three categories: (i) those based on distances in prediction results of different base models [39,40], (ii) those based on feature similarity between samples [41,42], and (iii) those based on model improvement [43]. Methods based on distances in prediction results of different base models take the idea of voting, *i.e.* higher label confidence is assigned to an unlabeled sample if its prediction labels given by different base models are closer. Evaluation based on feature similarity between samples is performed based on the hypothesis that samples with similar features are likely to have similar labels [44]. Thus, the greater the similarity between the unlabeled sample and the labeled samples in the neighborhood, the higher the label confidence of the unlabeled sample.

Different from the aforementioned two classes of label confidence assessment methods which are suitable for classification tasks but challenging to apply to regression tasks for soft sensing due to the continuous nature of label values, methods based on model improvement can be used to deal with regression problems. For convenience, let u_* denote an unlabeled sample in U_*, where $* = 1$ or 2 corresponds to View 1 and View 2, respectively. In the model improvement based confidence evaluation methods, for View 1 or View 2 (*i.e.* $* = 1$ or 2), the label confidence of u_* is assessed by comparing the fitting accuracy of h_* on a set of labeled samples in $\{L_*, Y_L\}$ (*e.g.* the set of k-nearest neighboring labeled samples of u_*), to the fitting accuracy of an updated model, which is obtained by updating h_* using the newly added sample $\{u_*, h_*(u_*)\}$. However, many methods based on model improvement evaluate confidence at the level of individual unlabeled samples, which can lead to the selection of non-representative samples, especially those located at the edges of the data distribution [4,45,46]. As a result, these methods may fail to ensure the representativeness of the chosen samples, leading to random model updates that can negatively affect the model's robustness. This issue is further exacerbated during the steps of pseudo-label assignment and cross-expansion of datasets. Such reliance on individual samples means that the model's performance can be significantly influenced by outliers or marginal samples, making the learning process less stable and more prone to fluctuations.

Step (iii) of co-training, pseudo-label assignment, involves assigning pseudo-labels to the most confident unlabeled samples in U_1 and U_2, as shown in Fig. 1. For convenience, let u_*^M denote the most confident unlabeled sample (*i.e.* the sample with the highest confidence) in U_*, where $* = 1$ or 2 corresponds to View 1 or View 2. It is intuitive that the output of h_*, with u_*^M as input, *i.e.* $h_*(u_*^M)$ can be regarded as a reasonable pseudo-label of u_*^M [47,48]. However, it is difficult to ensure the quality of pseudo-labels obtained based on this strategy, which may further lead to performance degradation [49]. To address this issue, Li *et al.* [49] proposed a safe pseudo-label technique, called SAFER in pseudo code, which integrates predictions from multiple models to reduce pseudo-label errors and give a safe prediction that is no worse than a direct supervised learner using only labeled data. Min *et al.* [29] further proposed a self-paced safe co-training for regression (SPOR), which uses the safe pseudo-label technique to enhance the quality of pseudo-labels, and uses a self-paced paradigm for unlabeled sample selection. It is worth mentioning that although SPOR allows multiple samples to be included in the labeled dataset in each round, it still evaluates the confidence of each individual sample in step (ii) (*i.e.* confidence estimation) and may lead to the problem of random model updates.

Co-training shows promise in soft sensing methods when labeled samples are limited, but it has not yet been explored in soft sensing for the mechanical properties of the hot rolling process. To address this gap and build on the motivations discussed above, we propose an improved co-training (ICO) method for soft sensing of mechanical properties of hot-rolled strips. Corresponding to classical co-training, ICO comprises four key steps: (i) multiple view partition, (ii) confidence estimation for unlabeled sample clusters, (iii) safe pseudo-label assignment, and (iv) cross-expansion of labeled datasets. It is notable that ICO mainly makes improvements in steps (i) to (iii) of classical co-training.

In step (i), we develop a strategy for partitioning views on the features of the hot rolling process based on the metallurgical mechanism of hot rolling and the statistical analysis of mutual information. This strategy considers both the sufficiency and independence of the views and achieves a balance between them through a trade-off. Detailed view partition strategies will be provided in Section 3.1.

In step (ii), ICO innovates the confidence evaluation of unlabeled samples at the sample cluster level, in contrast to the single-sample level in the classical co-training methods. As opposed to a single sample, a cluster of unlabeled samples is more likely to represent the data distribution of the whole dataset, and is more likely to contain representative samples [50–52]. Furthermore, a cluster of samples contains richer information, facilitating model updates in a more favorable direction. Detailed algorithms are provided in Section 3.2.

In step (iii), SAFER proposed by [49] is applied to the unlabeled sample cluster with the highest confidence obtained in step (ii). Unlike [49], where SAFER is applied at the single-sample level, SAFER in this paper is applied at the cluster level of unlabeled samples, which may benefit model updates by offering more representative samples.

In conclusion, the contributions of this paper are summarized below:

1) We propose an ICO framework, an improved framework based on co-training, specifically designed for soft sensing of mechanical properties in hot rolling processes. ICO maintains the universality of the co-training framework, allowing it to be adapted to various regressor models and providing the flexibility to choose the suitable regressors for different tasks and data distributions.
2) ICO incorporates domain-specific knowledge of hot rolling and statistical analysis to refine the view partition process. This refinement achieves a balance between view sufficiency and independence, which improves the soft sensing performance of ICO as a multi-view method.
3) In ICO, confidence evaluation is performed at the sample cluster level, where a confidence score is assigned to each cluster of unlabeled samples. Compared to methods that evaluate confidence at the single-sample level [24,25,53], ICO's cluster-level evaluation better captures the sample distribution and selects more representative samples. And the richer information and representativeness of unlabeled clusters provide more reliable guidance for model updates.
4) ICO employs SAFER, a safety pseudo-labeling mechanism, to calculate pseudo-labels at the sample cluster level. This improves the quality of pseudo-labels, ensures the model's robustness, and prevents model degradation caused by the use of unlabeled samples.
5) Experiments are conducted to show that the proposed ICO outperforms other supervised and semi-supervised methods in soft sensing the mechanical properties of hot-rolled strips with very limited labeled samples.

2. Preliminaries

2.1. The physical metallurgy mechanism in the hot rolling process

The process of hot rolling steel is a complex manufacturing process that involves several macroscopic stages: (i) heating, (ii) rough rolling, (iii) finishing rolling, (iv) cooling, and (v) coiling [53]. During the heating stage, steel plates are heated to a high temperature in a furnace, which improves the material's plasticity for subsequent plastic deformation [54]. In the rough rolling stage, the cross-section of the steel plate is reduced, and its length is increased, resulting in a thinner and wider plate. In the finishing rolling stage, the steel plate's cross-section is further reduced, and its thickness is altered while adjustments are made to the flatness and surface quality of the strip [55]. The cooling stage lowers the temperature of the strip to help control its microstructure, enhance its mechanical properties, and solidify and adjust the material's structure [56]. Finally, the hot-rolled steel strip is rolled into coils during the coiling stage for storage, transportation, and further processing [57].

Table 1 presents the microstructural changes and their related variables in each macroscopic stage [58]. During the heating stage, steel billets typically have an initial austenite structure. As they are heated, the size and shape of the austenite grains change, which ultimately affects the steel's final properties [59,60]. During the two rolling stages, the steel undergoes mechanical deformation and temperature variations, which may result in grain recrystallization. This process replaces old grains with new and smaller ones, relieving stress and enhancing the material's strength and toughness [61,62]. In the laminar cooling process following steel rolling, austenite phase transformations take place. The austenite transforms successively into ferrite, pearlite, and bainite, significantly altering the steel's hardness and strength [63]. During the post-transformation structure, further precipitation of micro-alloy carbonitrides occurs, resulting in precipitation hardening. The microstructural changes and the related variables of each macroscopic stage can be summarized in Table 1.

Leveraging the hot rolling mechanism as prior knowledge can assist the co-training method in partitioning views more rationally, offering an opportunity to achieve a balance between the sufficiency and independence of views.

2.2. Notations

For the sake of clarity, we formally define the notations used throughout this paper, even though some were introduced in Section 1.

2.2.1. Notation 1

Suppose that the dataset X comprising N samples, D features, and three mechanical properties (*i.e.* YS, TS, EL) can be categorized into a labeled dataset $X_L = \{L, Y_L\} \in \mathbb{R}^{N_l \times (D+3)}$ and an unlabeled dataset $X_U = \{U, Y_U\} \in \mathbb{R}^{N_u \times (D+3)}$, where N_l and N_u are the numbers of labeled and unlabeled samples, respectively. $L \in \mathbb{R}^{N_l \times D}$ and $U \in \mathbb{R}^{N_u \times D}$ denote the feature matrices. $Y_L \in \mathbb{R}^{N_l \times 3}$ and $Y_U \in \mathbb{R}^{N_u \times 3}$ denote the mechanical properties corresponding to L and U, respectively. It is worth noting that Y_L is known and regarded as the label matrix of L, while Y_U is unknown. In addition, let u denote the unlabeled samples belonging to U.

2.2.2. Notation 2

Let $F = \{f(d), d = 1, \cdots, D\}$ represent the feature set corresponding to the labeled dataset L along the feature dimension, where $f(d) \in \mathbb{R}^{N_l \times 1}$. Let $K \geq 2$ denote the number of views, then both L and U can be partitioned into K sets corresponding to the K views, *i.e.* $\{L_k\}_{k=1:K}$ and $\{U_k\}_{k=1:K}$, where $L_k \in \mathbb{R}^{N_l \times D_k}$ and $U_k \in \mathbb{R}^{N_u \times D_k}$ represent the labeled dataset and unlabeled dataset under the k_{th} view, respectively, and there is $\sum_{k=1}^{K} D_k = D$. In addition, we use $h_k, \ k = 1, \cdots, K$ to denote the base regressor in each view.

Table 1
Microstructural changes and relevant variables in various macroscopic stages of the hot rolling process.

Macroscopic stage	Microstructural changes	Relevant variables
Heating	Austenite crystallization	Heating temperature, heating time, billet chemical composition
Rough rolling	Austenite crystallization Austenite recrystallization	Roller diameter, rolling pressure, rolling temperature
Finishing rolling	Austenite recrystallization	Roller diameter, rolling pressure, rolling temperature
Cooling	Austenite phase transformation	Cooling rate, cooling temperature
Coiling	Precipitation	Coiling temperature

2.3. SAFER

Semi-supervised learning for regression tasks faces a challenge in terms of model performance degradation when using unlabeled data [64]. To tackle this issue, Li *et al.* [49] introduced a safe semi-supervised regression method called SAFER. This method ensures that after incorporating unlabeled data in the training, the model's performance is at least not worse than that trained through fully supervised learning with only labeled data [29].

In SAFER, to ensure that the generated pseudo-labels closely approximate the ground truth and robustly provide safe pseudo-labels even in the worst-case scenario (*i.e.* the quality of pseudo-labels is not worse than the predictions by a fully supervised model), an optimization problem is defined as [49],

$$\max_{p\in\mathbb{R}^{|u|}} \|p_0 - p^*\|^2 - \|p - p^*\|^2 \tag{1}$$

where u represents some unlabeled samples in U, as defined in Section 2.2, and $|u|$ denotes the number of samples in u; p_0 is the predictions of u with the fully supervised regressor h_0 trained only using X_L; p^* represents the true labels of u, which is actually unknown; p represents the pseudo-labels to be assigned to u by SAFER.

By estimating the unknown p^* through the linear combination of predictions from multiple models, the optimization objective Eq. (1) can be modified as Eq. (2).

$$\begin{gathered}\max_{p\in\mathbb{R}^{|u|}} \min_{\alpha} \left\| p_0 - \sum \alpha_k p_k \right\|^2 - \left\| p - \sum \alpha_k p_k \right\|^2 \\ \sum \alpha_k = 1 \\ 0 \le \alpha_k \le 1 \end{gathered} \tag{2}$$

where α_k is the weight of base regressor h_k; p_k is the predictions of u by h_k.

According to [49], the optimal solutions of Eq. (2) is given by

$$\begin{aligned}\tilde{\alpha} &= \operatorname{argmin} \left\| p_0 - \sum \alpha_k p_k \right\|^2 \\ \tilde{p} &= \sum \tilde{\alpha}_k p_k \end{aligned} \tag{3}$$

The pseudocode of the SAFER method is given in Algorithm 1 [65].

Algorithm 1: SAFER

Input: Base regressors $\{h_1, h_2, \cdots, h_K\}$, unlabeled samples u, and the baseline prediction p_0 for u, *i.e.* $p_0 = h_0(u)$
Output: Predictions $\tilde{p}$ for unlabeled samples u
Calculate the predictions for u by base regressors as $p_k = h_k(u),\ k = 1, \cdots, K$
Construct a kernel matrix $\boldsymbol{M} = [p_i^{\mathrm{T}} p_j]_{ij},\ i, j = 1, \cdots, K$
Derive a vector $\boldsymbol{v} = 2[p_1^{\mathrm{T}} p_0; \cdots; p_K^{\mathrm{T}} p_0]$
Obtain the optimal $\tilde{\alpha} = \operatorname{argmin} \alpha^{\mathrm{T}} M \alpha - v^{\mathrm{T}} \alpha$
Obtain the predictions for u as $\tilde{p} = \sum \tilde{\alpha}_k p_k$

3. Methods

Fig. 2 shows the framework of the proposed semi-supervised soft sensor for the mechanical properties of hot-rolled steel strips, ICO, which includes four steps: (i) multiple view partition, (ii) confidence estimation for unlabeled sample clusters, (iii) safe pseudo-label assignment, and (iv) cross-expansion of labeled datasets and updating models. In step (i), ICO innovates the view partitioning strategy by constructing the mutual information vector $\boldsymbol{M}$ and the stage vector $\boldsymbol{S}$. The strategy balances the sufficiency and independence of views to ensure the rationality of view partition. In step (ii), ICO performs confidence estimation for unlabeled samples at the sample cluster level. A cluster of samples contains richer information relative to a single sample, such as the local distribution of samples, making it more likely to update the model in a favorable direction. In addition, selecting a cluster of unlabeled samples at a time reduces the time complexity of the algorithm. In step (iii), SAFER is employed to assign pseudo-labels to the cluster of unlabeled samples with the highest confidence, ensuring the accuracy of the pseudo-labels. The step (iv) is to add the unlabeled samples with pseudo-labels obtained in one view to the labeled dataset of another view. In this way, the model in each view is mutually enhanced by incorporating unlabeled samples from other views.

The first two steps would be detailed in Section 3.1 and Section 3.2. The third step has been covered in Section 2.3. The fourth step closely resembles that in classical co-training, so it would not be further elaborated.

3.1. Multi-view partition

In this paper, the high-dimensional features collected from the hot rolling process include dimensional information, temperature information, and chemical composition information. The process of partitioning features into views is to allocate D features $F = \{f(d), d = 1, \cdots, D\}$ to K views $\{V(k), k = 1, \cdots, K\}$. We first propose three vectors, *i.e.* the importance score vector $I \in \mathbb{R}^D$, the mutual information vector $M \in \mathbb{R}^K$, and the stage vector $S \in \mathbb{R}^K$. Then based on them, we propose Algorithm 2 for view partition, to enhance the ability to predict mechanical performance and balance the sufficiency and independence of views.

Algorithm 2: Multi-view partition

Input: Mechanical performance indicator Y_L, set of features $F = \{f(d)\}_{d=1:D}$, and tradeoff parameter β
Output: Set of views $\{V(1), V(2), \cdots, V(K)\}$
Initialize $V(k), k = 1 : K$ as an empty list
for $d = 1 : D$ **do**
    $I(d) = MI(f(d), Y_L)$
end for
Sort I from high to low and reorder F in this order
for $d = 1 : D$ **do**
    Calculate the mutual information vector M according to Eq. (4)
    Calculate the stage vector S according to Eq. (5)
    $k_{\text{select}} = \operatorname{argmin}_k (M + \beta \cdot S)$
    Add $f(d)$ to $V(k_{\text{select}})$
end for

In the following, we will define the vectors $\boldsymbol{I}$, $\boldsymbol{M}$, and $\boldsymbol{S}$ and discuss their meanings.

(i) Importance score vector $\boldsymbol{I}$

The importance score vector $\boldsymbol{I} \in \mathbb{R}^D$ is defined as the vector composed of the mutual information between each feature and the mechanical performance indicators Y_L, *i.e.* $\boldsymbol{I} = [I(1), \cdots, I(d), \cdots, I(D)]^{\mathrm{T}}$, with $I(d) = \boldsymbol{MI}(f(d), Y_L)$, where $\boldsymbol{MI}(\cdot, \cdot)$ denotes the calculation of mutual information between two features [66]. The importance score $I(d)$ quantifies the contribution of feature $f(d)$ to predicting mechanical performance, where a higher score indicates greater informational content.

(ii) Mutual information vector $\boldsymbol{M}$

The mutual information vector $\boldsymbol{M}$ shows the independence between the feature currently being assigned and those already assigned in the K views. Each element in $\boldsymbol{M} = [M(1), \cdots, M(k), \cdots, M(K)]^{\mathrm{T}}$ is defined as

$$M(k) = \begin{cases} 0, & \text{if } V(k) \text{ is empty} \\ \sum\limits_{f(j) \in V(k)} \boldsymbol{MI}(f(i), f(j)), & \text{else} \end{cases} \tag{4}$$

where $f(i)$ is the feature currently to be allocated; $\boldsymbol{MI}(\cdot, \cdot)$ denotes the mutual information operator that measures the mutual information between features; $V(k)$ is the feature sets of the k_{th} view; $M(k)$ is the total mutual information for the current allocated feature $f(i)$ with respect to the k_{th} view.

(iii) Stage vector $\boldsymbol{S}$

Features belonging to the same stage of microstructural evolution contain similar information. To ensure the sufficiency of views, we formulate the stage vector $\boldsymbol{S} = [S(1), \cdots, S(k), \cdots, S(K)]^{\mathrm{T}}$ based on the metallurgical mechanism of hot rolling. Each element in $\boldsymbol{S}$ is defined as

$$S(k) = \begin{cases} 0, & \text{if } V(k) \text{ is empty} \\ \sum\limits_{f(j) \in V(k)} f(i) \otimes f(j), & \text{else} \end{cases} \tag{5}$$

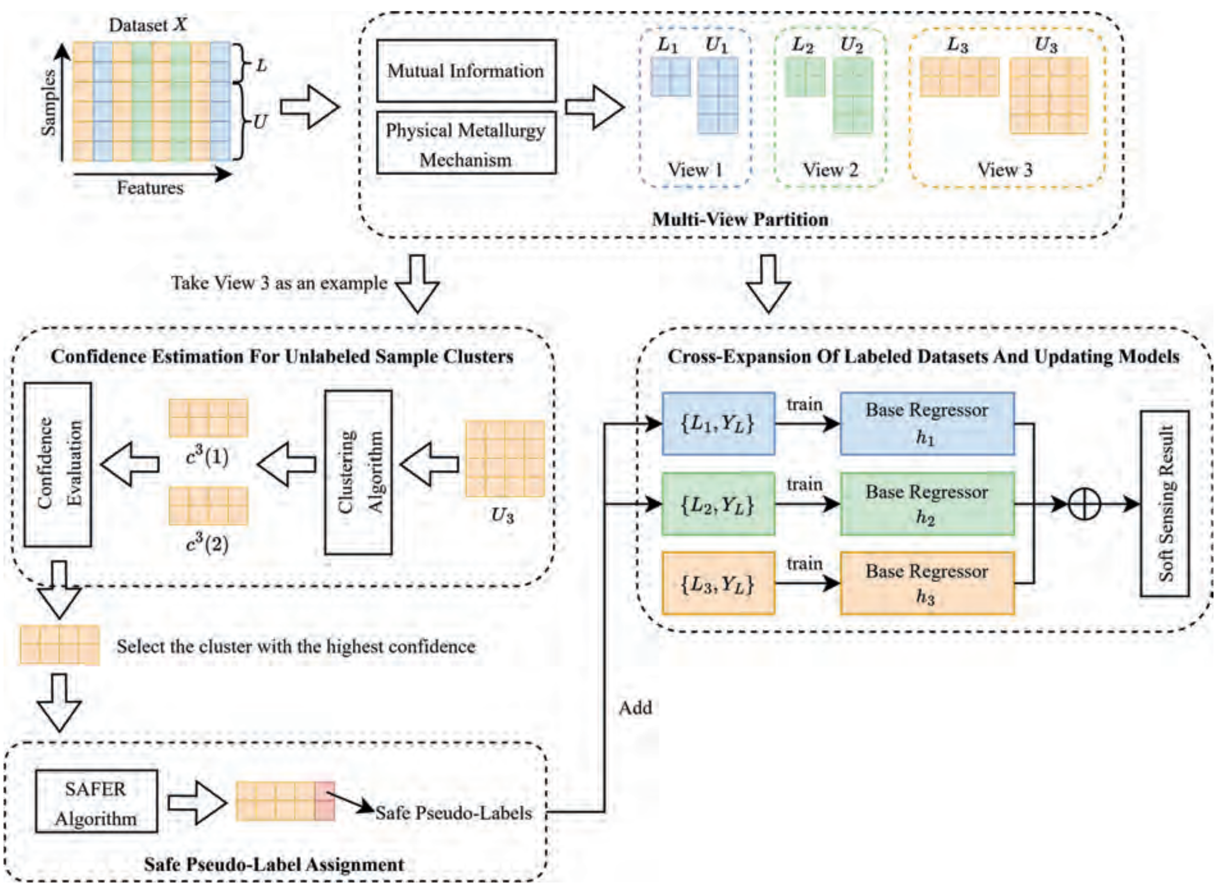

Fig. 2. Framework of ICO. (i) Multi-view partition: Based on mutual information and mechanism, all features are divided into several views, as exemplified by the number of views $K = 3$; (ii) Confidence estimation for unlabeled sample clusters: Illustrated by the 3_{rd} view as an example in this figure, unlabeled dataset U_3 is first clustered, and then confidence estimation is performed for each cluster, *i.e.* $c_3(p), p = 1, 2$. Finally the cluster with the highest confidence is selected; (iii) Safe pseudo-label assignment: Use the SAFER algorithm to assign pseudo-labels to the selected sample cluster; (iv) Cross-expansion of labeled datasets and updating models: Samples labeled with pseudo-labels are added to the labeled datasets of other views, corresponding to $\{L_1, Y_L\}$ and $\{L_2, Y_L\}$ in this figure.

where $f(i)$ is the feature currently to be allocated; the operation rule of the operator $\otimes$ is defined as

$$f(i) \otimes f(j) = \begin{cases} 1, & \text{if } f(i) \text{ and } f(j) \text{ in the same stage} \\ 0, & \text{else} \end{cases} \tag{6}$$

To balance view sufficiency and independence, we trade off $\boldsymbol{M}$, representing the independence between views, and $\boldsymbol{S}$, representing the sufficiency of views. Then the allocation result of the current feature $f(i)$ can be described as Eq. (7).

$$k_{\text{select}} = \underset{k}{\operatorname{argmin}}(\boldsymbol{M} + \beta \cdot \boldsymbol{S}) \tag{7}$$

where $\boldsymbol{M}$ is the mutual information vector which represents the independence between views; $\boldsymbol{S}$ is the stage vector which represents the sufficiency of views; β is the tradeoff between view independence and sufficiency; k_{select} is the view's index to which the current feature $f(i)$ should be assigned.

In addition, we could also affect the balance between sufficiency and independence of views by adjusting the number of views K. Generally, as the number of views K increases, the number of features within each view decreases, enhancing independence but diminishing sufficiency. Conversely, as K decreases, sufficiency improves but independence diminishes.

After the view partition, the original labeled set L is divided into $\{L_k\}_{k=1:K}$; the original unlabeled sample set U is divided into $\{U_k\}_{k=1:K}$ in the same way.

3.2. Confidence estimation for unlabeled sample clusters

Classical co-training typically evaluates the confidence of each unlabeled sample and chooses one unlabeled sample with the highest confidence in each round. However, to guarantee the representativeness of the selected unlabeled samples, ICO adopts a cluster-based strategy to create clusters of unlabeled samples and then evaluates the confidence (CFD) of these clusters as Eq. (8).

$$\text{CFD}(c_k(p)) = (Y_L - h_k(L_k))^2 - (Y_L - \widehat{h}_k(L_k))^2 \tag{8}$$

where $c_k(p)$ represents the p_{th} cluster of unlabeled samples under the k_{th} view; h_k represents the current regressor trained on X_L; $\widehat{h}_k$ represents the updated regressor after using the $c_k(i)$.

As a result, a cluster with the highest confidence among unlabeled samples can be directly selected during each iteration of training, which will be assigned pseudo-labels by SAFER in step (iii).

Soft sensing of mechanical properties is a regression task where the labels are continuous, so it's impossible to determine the number of label categories as in classification tasks [50–52]. Therefore, when using clustering methods to analyze the inherent distribution of data in regression tasks, it's generally not appropriate to use clustering methods that require a known number of categories, such as K-means [67]. Instead, it is more appropriate to use clustering methods that don't require the number of categories to be known in advance, such as density-based clustering [68]. In this paper, we utilize the affinity propagation (AP) algorithm [69] to create clusters of unlabeled samples. The AP algorithm clusters data points using message passing to iteratively update responsibility and availability values, resulting in high-quality cluster centers [70].

For a clearer description, the pseudocode for selecting unlabeled clusters under a given view is presented in Algorithm 3.

Algorithm 3: Confidence estimation for unlabeled sample clusters

Input: Labeled dataset L_k under the k_{th} view, unlabeled dataset U_k under the k_{th} view, mechanical performance indicator Y_L, regressor h_k under the k_{th} view
Output: Selected unlabeled cluster c_k^M
$AP(U_k) = \{c_k(1), c_k(2), \cdots, c_k(P)\}$//Cluster samples by AP algorithm
for $p = 1 : P$ **do**
  $\widehat{y}(p) = h_k(c_k(p))$
  Obtain $\widehat{h}_k$ by training on $(L_k, Y_k) \cup (c_k(p), \widehat{y}(p))$
  Calculate label confidence $\text{CFD}(c_k(p))$ according to Eq. (8)
end for
$c_k^M = \text{argmax}\,\text{CFD}(c_k(p))$

To conclude, performing confidence estimation and pseudo-label assignment at the sample cluster level offers two notable benefits over conventional methods of using a single unlabeled sample. Firstly, compared to a single sample, the sample cluster is more representative, *i.e.* the sample cluster can better reflect the distribution of the entire sample space, and a single sample is likely to come from the edge of the sample space. So using the clusters of unlabeled samples to update base models is a more robust model optimization process, which helps to address the problem of model degradation in semi-supervised learning. Secondly, it greatly reduces the time complexity of the algorithm.

3.3. ICO

The training process of ICO, as shown in Algorithm 4, begins with partitioning views to obtain sets of samples with different features. Next, the most confident cluster of unlabeled samples is selected sequentially under each view, and the pseudo-labels of the cluster samples are obtained using SAFER, presented in Algorithm 1. These clustered samples, along with their pseudo-labels, are integrated into the labeled sample set under other views. The regressors are then updated using the augmented set of labeled samples.

Algorithm 4: Training process of ICO

Input: Labeled dataset L, unlabeled dataset U, mechanical performance indicator Y_L, the number of views K, maximum iteration T
Output: Regressors $\{h_1, h_2, \cdots, h_k\}$
Obtain labeled datasets $\{L_k\}_{k=1:K}$ and unlabeled datasets $\{U_k\}_{k=1:K}$ after view partition by Algorithm 2
for $k = 1 : K$ **do**
  Initialize regressor h_k trained on L_k
end for
while $t < T$ **do**
  for $k = 1 : K$ **do**
    Obtain the most confident unlabeled cluster c_k^M by Algorithm 3
    Obtain pseudo-labels by SAFER algorithm $\widehat{y}^M = \text{SAFER}(c_k^M)$
    $U_k = U_k - c_k^M$//Remove the c_k^M set from the U_k set
    for $v = 1 : K$ **do**
      if $v \neq k$ **then**
        $L^v = L^v \cup (c_k^M, \widehat{y}^M)$
      end if
    end for
  end for
  for $k = 1 : K$ **do**
    Train h_k on L_k.
  end for
  $t = t + 1$
end while

During the testing process, a test unlabeled sample u^{test} is initially partitioned according to the consistent view partitioning method used during training, resulting in features $u_1^{\text{test}}, u_2^{\text{test}}, \cdots, u_K^{\text{test}}$ under each view. Utilizing the regressors $h_1, h_2, \cdots, h_k$ trained under each view, soft sensing of the current unlabeled sample is achieved, defined as Eq. (9).

$$\widehat{y} = \sum_k \gamma_k h_k(u_k^{\text{test}}) \tag{9}$$

where γ_k represents the weight of each regressor. In the absence of specific requirements, it can be set as $1/K$.

Similar to the process of the cross-expansion of labeled datasets and updating models in classical co-training, ICO effectively integrates information from different views, enhancing the model's performance. Moreover, the strategy of exchanging views to augment labeled samples efficiently alleviates the issue of error accumulation.

4. Experiments

4.1. Data description and experimental setup

The raw dataset in the experiments was obtained from a hot rolling production line of a steel factory, involving 34 types of steel. Based on the mechanism of hot rolling, 34 variables are selected for predicting target mechanical properties, *i.e.* YS, TS, EL. The raw dataset is divided into a labeled dataset X_L, an unlabeled dataset X_U, and a test dataset T. The numbers of samples in each dataset are $(N_l, N_u, N_t) = (537, 1556, 744)$, respectively.

In the first step of ICO, *i.e.* multi-view partition, we need to calculate two types of mutual information: the mutual information between features and the mutual information between features and three mechanical properties, as mentioned in Section 3.1. These are illustrated in Figs. 3 and 4, respectively.

To demonstrate the effectiveness of our soft sensor method, R-squared (R^2), root mean square error (RMSE), and mean absolute error (MAE) are used to evaluate the models' performance. The formulas are as follows:

$$R^2 = 1 - \frac{\sum_{n=1}^{N_t} (y(n) - \widehat{y}(n))^2}{\sum_{n=1}^{N_t} (y(n) - \overline{y}(n))^2} \tag{10}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N_t}(y(n) - \widehat{y}(n))^2} \tag{11}$$

$$\text{MAE} = \frac{1}{N_t}\sum_{n=1}^{N_t}|y(n) - \widehat{y}(n)| \tag{12}$$

where N_t is the number of the samples in test dataset T; $y(n)$ is the true label of the sample in T; and $\widehat{y}(n)$ is the predicted value.

The ICO framework proposed in Section 3.3 is general and permits the employment of any regressor model in both pseudo-labeling and soft sensing, and therefore offers the flexibility of using the same or different regressor models in the two stages. So in the experiments for comparison, we choose support vector regression (SVR), multilayer perceptron (MLP) and convolutional neural network (CNN) as the candidate regressors.

For the sake of clarity, the notation "ICO-*1-*2″ is used to represent the specific combination of models employed in pseudo-labeling and soft sensing respectively, where *1 and *2 can be SVR, MLP or CNN. For example, "ICO-MLP-SVR" refers to the ICO framework utilizing MLP for pseudo-labeling and SVR for soft sensing.

4.2. Selection of the number of views

The number of views (*i.e.* K) is a crucial factor that directly affects the performance of the proposed ICO framework. An appropriate choice of K can enhance the model's ability to capture the complex relationships within the data and fully leverage the advantages of the ICO framework. In this section, experiments investigating the impact of different K on model performance in

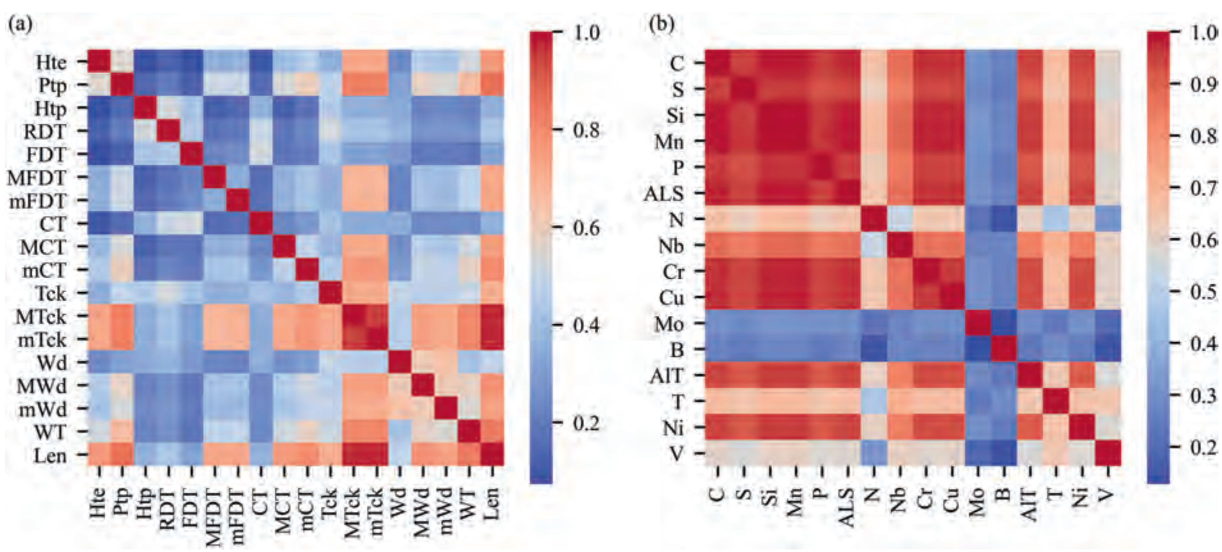

Fig. 3. MI between features: (a) MI between process parameter features; (b) MI between chemical components.

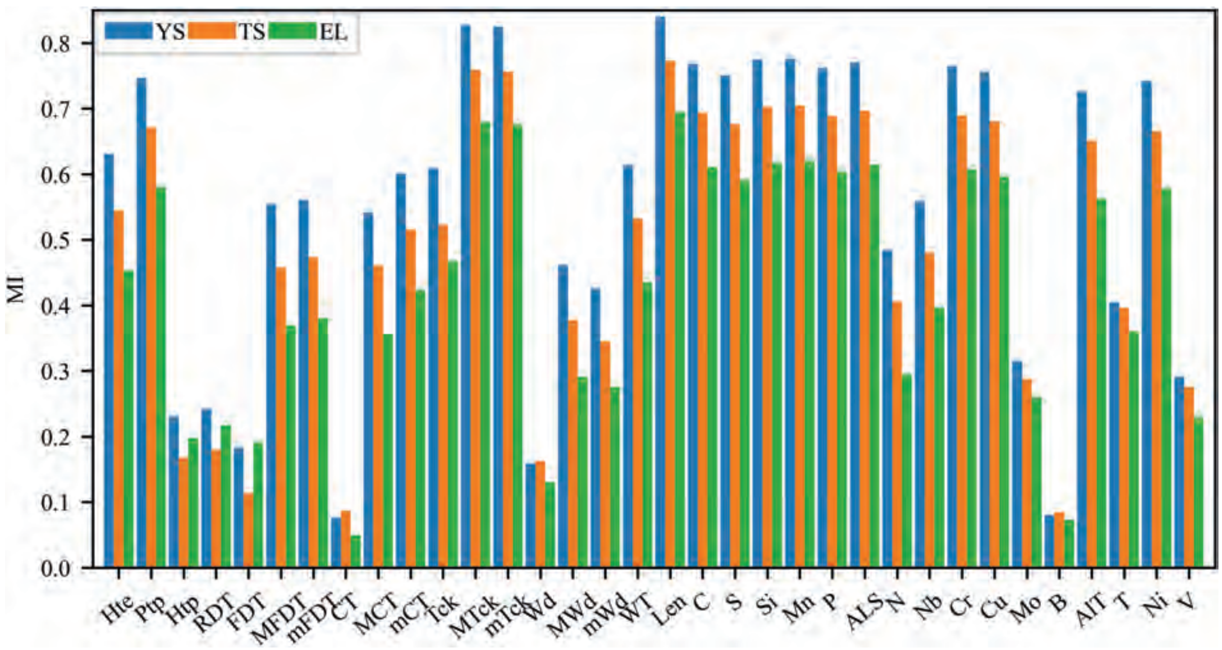

Fig. 4. MI between features and three mechanical properties.

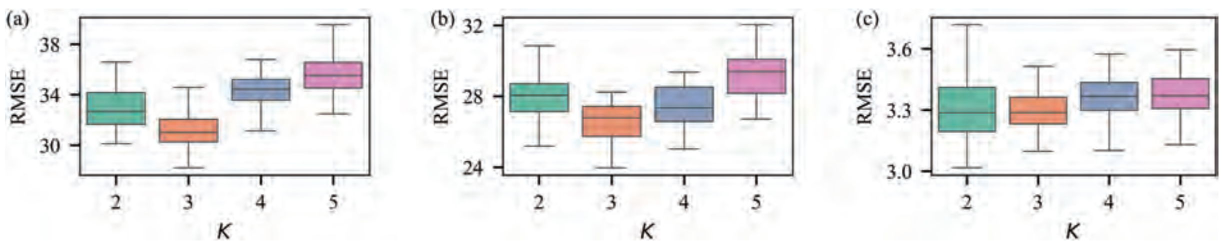

Fig. 5. The impact of the number of views: (a) YS; (b) TS; (c) EL.

predicting the mechanical properties of hot-rolled strips will be given, where the value of K varies from 2 to 6, and without loss of generality, SVR is used as the base model for each view.

Fig. 5 shows the boxplots of RMSE values for mechanical properties under different K. As K increases from 2 to 5, the number of variables in each view decreases, which weakens the sufficiency of each view but enhances the independence between views. As shown in Fig. 5, the model's performance initially improves as K increases from 2 to 3, but then deteriorates as K increases to 5, with the best performance achieved at $K = 3$. This indicates that $K = 3$ achieves a balance where each view contains enough features to allow the regressors to learn meaningful patterns on one hand, and a reasonable level of independence between views is maintained on the other hand. This balance facilitates effective interaction and learning between the views during the co-training process, ultimately leading to the best performance for predicting the mechanical properties of hot-rolled strips.

4.3. Soft sensing of mechanical properties

In this section, to show the effectiveness of ICO, ICO-SVR-SVR, ICO-MLP-MLP, and ICO-CNN-CNN are first tested. The soft sensing results are presented in Fig. 6, where the horizontal axis represents the true values of YS, TS, and EL, and the vertical axis represents the soft sensing predictions generated by ICO. The points closer to the diagonals indicate better performance, as they represent higher prediction accuracy.

To demonstrate the merit of ICO, a semi-supervised method that utilizes information from unlabeled samples to enhance soft measurement performance, we conduct comparative experiments with both supervised and semi-supervised soft-sensing methods. The compared methods are as follows:

1) SVR [71]: the supervised support vector regression method.
2) MLP [71]: the supervised multilayer perceptron designed with two hidden layers.
3) CNN [71]: the supervised convolutional neural network, designed with two convolutional layers, pooling layers, and a fully connected layer.
4) Self-SVR [72]: the semi-supervised SVR model based on the self-training paradigm.
5) SELM [73]: the semi-supervised extreme learning machine with graph Laplacian regularization.
6) Coreg [43]: the classical semi-supervised regression method using two different k-nearest neighbor regressors.
7) Co-SVR [72]: the semi-supervised support vector regression method based on the co-training paradigm.

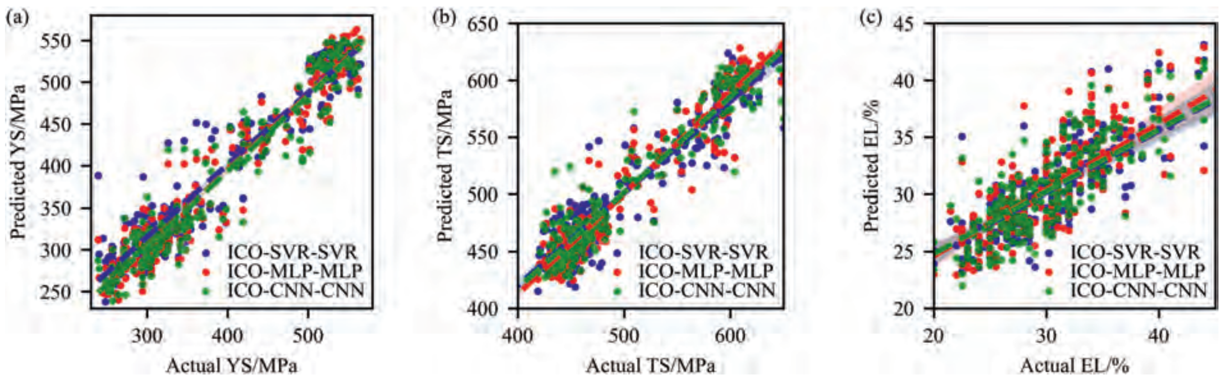

Fig. 6. The soft sensing performance of our methods: (a) YS; (b) TS; (c) EL.

Table 2
Performance comparison of different methods for soft sensing YS.

Models	R^2		RMSE		MAE	
	Mean	Std	Mean	Std	Mean	Std
SVR	0.8476	0.0179	41.8388	2.5258	27.6585	1.4695
MLP	0.8947	0.0149	35.0634	2.6945	23.6936	1.3212
CNN	0.8811	0.0177	37.0763	3.0609	24.0080	1.4387
Self-SVR	0.8629	0.0254	40.4690	3.3407	25.8975	2.2809
SELM	0.8807	0.0147	36.8915	2.0726	27.1365	1.4432
Coreg	0.8936	0.0226	33.3848	3.5807	24.4243	1.8118
Co-SVR	0.8879	0.0281	34.2620	3.9062	25.6682	2.7859
SSDBN	0.9206	0.0231	28.8364	3.2065	21.1700	2.0054
ICO-SVR-SVR	0.9071	0.0082	31.1598	1.4446	22.9035	0.9547
ICO-MLP-MLP	**0.9285**	0.0077	**27.3698**	1.4721	**20.2121**	0.8961
ICO-CNN-CNN	0.9238	0.0089	28.2473	1.4826	20.5268	1.0793

Table 4
Performance comparison of different methods for soft sensing EL.

Models	R^2		RMSE		MAE	
	Mean	Std	Mean	Std	Mean	Std
SVR	0.5014	0.0371	3.5304	0.1830	2.4659	0.1139
MLP	0.5270	0.0525	3.2993	0.2349	2.4222	0.1455
CNN	0.5451	0.0516	3.3681	0.1998	2.3988	0.1155
Self-SVR	0.5148	0.0520	3.5044	0.2296	2.4553	0.1232
SELM	0.5359	0.0395	3.4248	0.1709	2.3421	0.0913
Coreg	0.5415	0.0489	3.3742	0.2088	2.3785	0.1775
Co-SVR	0.5085	0.0473	3.4700	0.2268	2.6340	0.1023
SSDBN	0.5610	0.0414	**3.2014**	0.1801	**2.2914**	0.0874
ICO-SVR-SVR	0.5246	0.0226	3.2988	0.1002	2.3483	0.0411
ICO-MLP-MLP	0.5534	0.0261	3.3679	0.1930	2.3589	0.0987
ICO-CNN-CNN	**0.5733**	0.0317	3.2708	0.1802	2.4706	0.0843

8) SSDBN [74]: the semi-supervised deep belief network designed with the stacking of two fully connected layers.

Moreover, to ensure the robustness of the experimental results, each experiment has been repeated 50 times, and the results are presented in Table 2 including the mean and standard deviation of each performance metric (see Tables 3 and 4).

By comparing the original supervised regression models, *i.e.* SVR, MLP, and CNN, with their counterparts which cooperate with our proposed ICO, *i.e.*, ICO-SVR-SVR, ICO-MLP-MLP, and ICO-CNN-CNN, it can be seen that significant performance improvements are achieved in terms of R^2, RMSE, and MAE, highlighting the advantages of integrating the semi-supervised learning strategies within ICO.

Moreover, ICO-MLP-MLP generally outperforms other semi-supervised methods, such as Self-SVR, SELM, Co-SVR, and Coreg, which indicates that the strategies employed in ICO, including the view partition, confidence evaluation, and pseudo-label assignment strategies, are effective in improving the accuracy of soft sensing.

Table 3
Performance comparison of different methods for soft sensing TS.

Models	R^2		RMSE		MAE	
	Mean	Std	Mean	Std	Mean	Std
SVR	0.8227	0.0211	33.1689	1.9758	20.6948	1.3839
MLP	0.8914	0.0142	25.7423	1.7816	17.5330	0.9226
CNN	0.8744	0.0228	27.6314	2.6126	17.5952	1.2362
Self-SVR	0.8553	0.0338	29.9870	3.0394	18.6874	1.9719
SELM	0.8831	0.0208	26.0887	2.0052	18.6340	1.1853
Coreg	0.8420	0.0293	27.5257	1.7718	18.8239	1.0980
Co-SVR	0.8587	0.0335	29.2199	3.2612	18.7770	2.4004
SSDBN	0.9065	0.0331	23.3069	3.3201	16.7183	2.2913
ICO-SVR-SVR	0.8782	0.0109	26.6440	1.2614	18.4892	1.0333
ICO-MLP-MLP	**0.9128**	0.0110	**22.5125**	1.2411	**16.2721**	1.0619
ICO-CNN-CNN	0.8974	0.0116	24.7818	1.4409	16.8005	0.9277

It is worth noting that ICO-MLP-MLP and ICO-CNN-CNN outperform ICO-SVR-SVR, indicating that the choice of different regressor models impacts soft sensing performance. This will be analyzed in detail in the next section.

4.4. Model selection for pseudo-labeling and soft sensing in ICO

As mentioned in Section 4.1, the proposed ICO framework is general and permits the employment of any regressor model in both pseudo-labeling and soft sensing. Depending on the specific task requirements and dataset characteristics, ICO offers the flexibility of using the same or different regressor models in the two stages. In this section, using SVR, MLP and CNN as the candidate regressors, we investigate the influence of model selection on the soft sensing performance by exploring the combinations of SVR, MLP and CNN in pseudo-labeling and soft sensing, *i.e.* ICO-SVR-SVR, ICO-MLP-SVR, ICO-MLP-MLP, ICO-CNN-SVR, ICO-CNN-CNN.

Fig. 7 shows the R^2 values of the five aforementioned models. It can be seen that ICO-MLP-SVR achieves the best performance in predicting YS, ICO-MLP-MLP performs best for TS, while ICO-CNN-CNN and ICO-CNN-SVR achieve similarly good performance for EL. In comparison to ICO-SVR-SVR, these particular combinations lead to an improvement in R^2 of approximately 0.024 for YS, 0.035 for TS, and 0.049 for EL. These results demonstrate the ability of MLP and CNN to capture complex nonlinear patterns, making them effective in generating accurate pseudo-labels.

Moreover, the experimental results highlight the importance of selecting appropriate regressor models for pseudo-labeling and

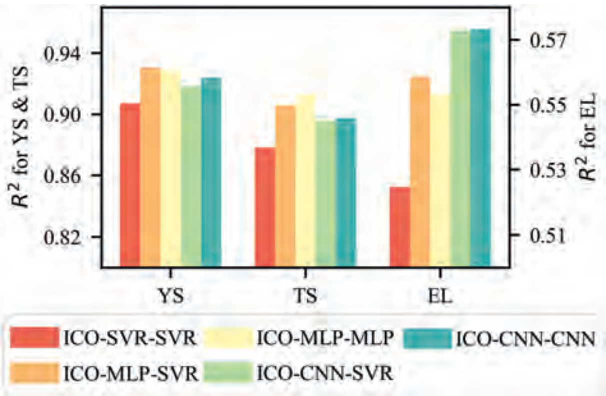

Fig. 7. The performance of different model combinations.

soft sensing based on the data distribution and the specific soft sensing task in practical applications. This adaptability is one of the notable strengths of the ICO framework, as it enables customization to diverse industrial and research scenarios, thereby enhancing the overall performance and reliability of soft sensing.

It is worth noting that the results of ICO-SVR-SVR, ICO-MLP-MLP, and ICO-CNN-CNN in Fig. 7 are consistent with those listed in Tables 2–4 of Section 4.3.

4.5. Sensitivity analysis of parameter β

In this section, taking ICO-SVR-SVR for soft sensing YS as an example, we present a sensitivity analysis of the parameter β in Eq. (7), which is crucial in the ICO for determining the tradeoff between sufficiency and independence in the view partitioning step.

Fig. 8 illustrates the performance of soft sensing YS obtained through a grid search, where β varies within the range from 0 to 50 with a grid size of 5. And for each value of β, we evaluate our model using a standard 10-fold cross-validation. As β increases, the model performance initially improves and then deteriorates. Notably, the experimental results reveal that the model performs well when β is in the range from 15 to 25. To further refine this range, a more detailed search was conducted within the narrower range from 10 to 25, with a grid size of 1. The results indicate that the model's performance remains relatively stable with minimal variation around $\beta = 18$.

4.6. Ablation study

To analyze the importance of several key ideas adopted in this paper, we further conduct the ablation studies. Specifically, the following two variants are analyzed:

(1) ICO-rvp: This variant replaces our view partition strategy, which takes into account both independence and sufficiency, with a random view partition strategy.

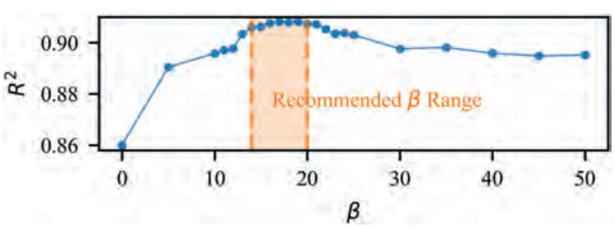

Fig. 8. Sensitivity analysis of parameter β.

Table 5
Ablation study results.

Properties	Metrics	ICO-rvp	ICO-ssl	ICO-SVR-SVR
YS	R^2	0.8618	0.8843	**0.9071**
	RMSE	40.2449	36.7182	**31.1598**
	MAE	26.3454	24.1535	**22.9085**
TS	R^2	0.8319	0.8626	**0.8782**
	RMSE	32.0377	30.7754	**26.6440**
	MAE	19.7381	18.8464	**18.4892**
EL	R^2	0.5071	0.5143	**0.5246**
	RMSE	3.5493	3.5372	**3.2988**
	MAE	2.4901	2.4564	**2.3483**

(2) ICO-ssl: This variant performs confidence evaluation and pseudo-label assignment at a single sample level, rather than the sample cluster level.

The results are summarized in Table 5. The comparison between ICO-SVR-SVR and ICO-rvp highlights the significance of appropriate view division in the co-training framework. The comparison between ICO and ICO-ssl shows the benefits of confidence assessment and pseudo-label assignment of sample clusters on model performance.

In addition, Fig. 9 shows the predictive performance of ICO-SVR-SVR and ICO-ssl on the test set as training iteration increases. As the number of training iterations increases, the prediction performance of ICO-SVR-SVR steadily improves. In the early iterations, the clusters with the highest confidence may contain more samples, which helps to update the base regressors and improve its performance on the test set quickly. In later iterations, primarily for fine-tuning the base regressors, it ultimately stabilizes at a higher predictive performance level. However, the performance curve of

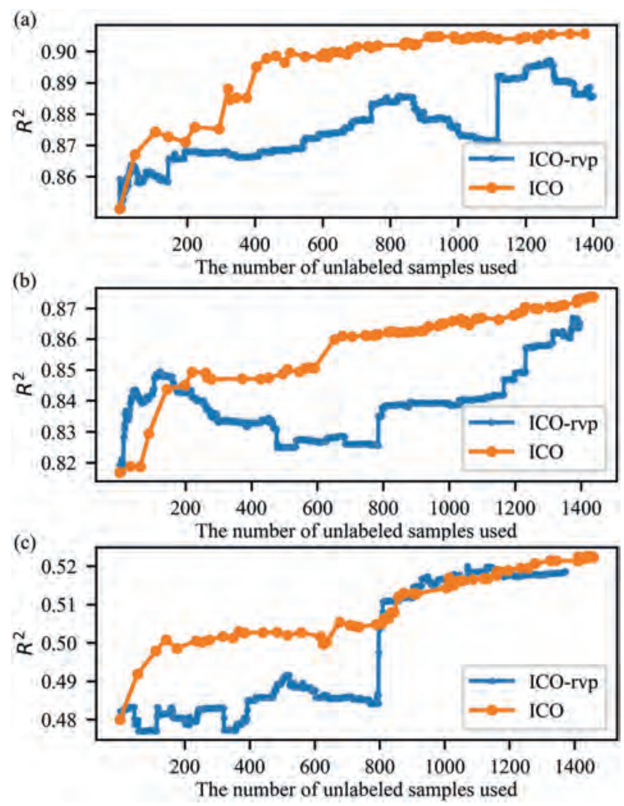

Fig. 9. The process of model improvement.

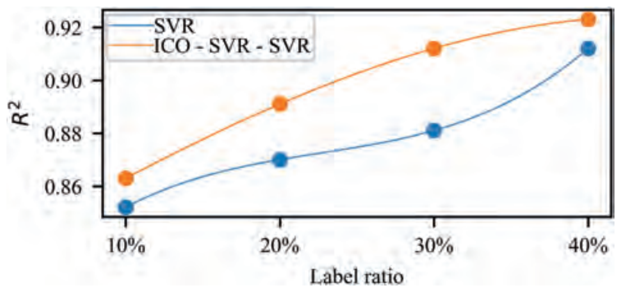

Fig. 10. Performance for soft sensing YS under different label ratios.

ICO-ssl is quite tortuous, indicating that the most confident sample selected in each iteration may not necessarily update the base regressors in the correct direction, potentially leading to the degradation of the soft sensing model.

4.7. Performance evaluation with different label ratios

To evaluate the adaptability of the proposed ICO framework under varying levels of supervision, experiments were conduct in this section using different label ratios, including 10%, 20%, 30%, and 40%. For each label ratio, we employ the same dataset and parameters for both ICO-SVR-SVR and SVR to ensure the fairness of the comparison. The experimental results are presented in Fig. 10.

When the label ratio is below 30%, there is a relatively high level of uncertainty due to the scarcity of labeled data. In this scenario, ICO-SVR-SVR demonstrates a clear advantage over SVR, indicating its ability to utilize the unlabeled samples effectively by the ICO framework. As the label ratio increases from 10% to 20% and then to 30%, both ICO-SVR-SVR and SVR show performance improvements. However, the improvement of SVR is not obvious, suggesting that relying solely on the labeled dataset is insufficient for accurate soft sensing. In contrast, ICO-SVR-SVR exhibits a significant enhancement in performance.

In conclusion, the experimental results demonstrate the effectiveness and adaptability of ICO-SVR-SVR. It not only exhibits superior performance in scenarios with limited labeled data but also shows a consistent improvement trend as the amount of labeled data increases.

5. Conclusions

In this paper, a novel co-training-based soft sensing method for mechanical properties (ICO) has been proposed, which can effectively predict the mechanical properties when labeled samples are limited. ICO mainly consists of four steps: multiple view partition, confidence estimation for unlabeled sample clusters, safe pseudo-label assignment, cross-expansion of labeled datasets, and updating models. We make improvements upon classical co-training methods in the first three steps. By integrating prior knowledge of the hot rolling process and statistical analysis, we achieve a more rational partition of views. Moreover, confidence estimation and safe pseudo-label assignment are conducted at the cluster level of unlabeled samples, ensuring more robust updates to the base model in the following step. The performance of ICO is validated on real hot rolling production line data. The experimental results demonstrate that ICO outperforms both the supervised method and the classical semi-supervised method.

While this paper focuses on the soft sensing of hot-rolled steel, the proposed ICO framework has broader applications. Future work may extend ICO to various industrial scenarios by adapting feature extraction and view partitioning to specific needs. Efforts will also focus on optimizing regressor selection with advanced algorithms. Additionally, we may integrate ICO with graph-based methods, such as the view-consistent heterogeneous network (VCHN) [75], which is also a multi-view method, similar to co-training. By combining ICO's SAFER algorithm with VCHN's view-consistency-driven pseudo-labeling, it is possible to enhance pseudo-label generation and collaborative training for better performance.

CRediT Authorship Contribution Statement

Bowen Shi: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. Jianye Xue: Writing – original draft, Supervision, Formal analysis, Data curation, Conceptualization. Hao Ye: Writing – review & editing, Validation, Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by National Key Research & Development Program of China (2021YFB3301200), and in part by the National Natural Science Foundation of China (61933015).

References

[1] M.A. Cavaliere, M.B. Goldschmit, E.N. Dvorkin, Finite element simulation of the steel plates hot rolling process, *Int. J. Numer. Methods Eng.* 52 (12) (2001) 1411–1430.
[2] H.Y. Ban, G. Shi, Y.J. Shi, Y.Q. Wang, Research progress on the mechanical property of high strength structural steels, *Adv. Mater. Res.* 250-253 (2011) 640–648.
[3] P. Šimeček, D. Hajduk, Prediction of mechanical properties of hot rolled steel products, *J. Achievements Mater. Manuf. Eng.* 20 (2007) 1–2.
[4] J. Wu, L.L. Tang, S.Q. Jin, X. Li, H. Liu, D. Li, Y.Q. Liu, Q.L. Wang, Modeling an adaptive hybrid soft sensor with co-training learning toward applications in wastewater treatment, *Ind. Eng. Chem. Res.* 62 (41) (2023) 16841–16853.
[5] H. Kay, S. Kay, M. Mowbray, A. Lane, C. Mendoza, P. Martin, D.D. Zhang, Constructing a symbolic regression-based interpretable soft sensor for industrial data analytics and product quality control, *Ind. Eng. Chem. Res.* 63 (9) (2024) 4083–4092.
[6] Y.K. Xie, Y.L. Deng, Y. Wang, X.B. Guo, Effect of asymmetric rolling and subsequent ageing on the microstructure, texture and mechanical properties of the Al-Cu-Li alloy, *J. Alloys Compd.* 836 (2020) 155445.
[7] W.W. Yan, R.C. Xu, K.D. Wang, T. Di, Z. Jiang, Soft sensor modeling method based on semisupervised deep learning and its application to wastewater treatment plant, *Ind. Eng. Chem. Res.* 59 (10) (2020) 4589–4601.
[8] F. Yan, L.Y. Kong, Y.R. Li, H.W. Zhang, C.J. Yang, L. Chai, A survey of data-driven soft sensing in ironmaking system: research status and opportunities, *ACS Omega* 9 (24) (2024) 25539–25554.
[9] Y.Y. Yang, M. Mahfouf, D.A. Linkens, Q. Zhang, Tensile strength prediction for hot rolled steels by Bayesian neural network model, *IFAC Proc.* 42 (23) (2009) 255–260.
[10] Z.W. Xu, X.M. Liu, K. Zhang, Mechanical properties prediction for hot rolled alloy steel using convolutional neural network, *IEEE Access* 7 (2019) 47068–47078.
[11] I. Mohanty, R. Banerjee, A. Santara, S. Kundu, P. Mitra, Prediction of properties over the length of the coil during thermo-mechanical processing using DNN, *Ironmak. Steelmak.* 48 (8) (2021) 953–961.
[12] G. Kostopoulos, S. Karlos, S. Kotsiantis, O. Ragos, Semi-supervised regression: a recent review, *J. Intell. Fuzzy Syst.* 35 (2)1483–1500.
[13] Q.Q. Sun, Z.Q. Ge, Deep learning for industrial KPI prediction: when ensemble learning meets semi-supervised data, *IEEE Trans. Ind. Inf.* 17 (1) (2021) 260–269.
[14] M.F. Abdel Hady, F. Schwenker, Semi-supervised learning. Handbook on Neural Information Processing, Springer Berlin Heidelberg, 2013, pp. 15–239.
[15] C.X. Jian, K.J. Yang, Y.H. Ao, Industrial fault diagnosis based on active learning and semi-supervised learning using small training set, *Eng. Appl. Artif. Intell.* 104 (2021) 104365.

[16] X.D. Shi, W.L. Xiong, Adaptive ensemble learning strategy for semi-supervised soft sensing, *J. Franklin Inst.* 357 (6) (2020) 3753–3770.
[17] Z.Q. Ge, Semi-supervised data modeling and analytics in the process industry: current research status and challenges, *IFAC J. Syst. Contr.* 16 (2021) 100150.
[18] P. Wang, Y.C. Yin, X.G. Deng, Y.C. Bo, W.M. Shao, Semi-supervised echo state network with temporal-spatial graph regularization for dynamic soft sensor modeling of industrial processes, *ISA Trans.* 130 (2022) 306–315.
[19] Y. Huang, C. Zhang, J. Yella, S. Petrov, X.Y. Qian, Y.F. Tang, X.Q. Zhu, S. Bom, GraSSNet: graph soft sensing neural networks. 2021 IEEE International Conference on Big Data (Big Data)., IEEE, Orlando, FL, USA, 2021, pp. 746–756.
[20] Z.X. Song, X.L. Yang, Z.L. Xu, I. King, Graph-based semi-supervised learning: a comprehensive review, *IEEE Transact. Neural Networks Learn. Syst.* 34 (11) (2023) 8174–8194.
[21] J.Y. Liang, J.B. Cui, J. Wang, W. Wei, Graph-based semi-supervised learning *via* improving the quality of the graph dynamically, *Mach. Learn.* 110 (6) (2021) 1345–1388.
[22] Y.W. Chong, Y. Ding, Q. Yan, S.M. Pan, Graph-based semi-supervised learning: a review, *Neurocomputing* 408 (2020) 216–230.
[23] K.M.O. Vale, A.C. Gorgônio, F. Da Luz E Gorgônio, A.M. De Paula Canuto, An efficient approach to select instances in self-training and co-training semi-supervised methods, *IEEE Access* 10 (2021) 7254–7276.
[24] S.W. Wu, J. Yang, G.M. Cao, Y.L. Qiu, G.G. Cheng, M.Y. Yao, J.X. Dong, Elevating prediction performance for mechanical properties of hot-rolled strips by using semi-supervised regression and deep learning, *IEEE Access* 8 (2020) 134124–134136.
[25] J. Dong, Y.Z. Tian, K.X. Peng, Just-in-time learning-based soft sensor for mechanical properties of strip steel *via* multi-block weighted semisupervised models, *IEEE Access* 8 (2020) 123869–123881.
[26] X. Ning, X.R. Wang, S.H. Xu, W.W. Cai, L.P. Zhang, L.N. Yu, W.F. Li, A review of research on co-training, *Concurr. Comput. Pract. Exp.* 35 (18) (2023) e6276.
[27] C. Gao, J. Zhou, D.Q. Miao, J.J. Wen, X.D. Yue, Three-way decision with co-training for partially labeled data, *Inf. Sci.* 544 (2021) 500–518.
[28] A. Rahate, R. Walambe, S. Ramanna, K. Kotecha, Multimodal co-learning: challenges, applications with datasets, recent advances and future directions, *Inf. Fusion* 81 (2022) 203–239.
[29] F. Min, Y. Li, L.Y. Liu, Self-paced safe co-training for regression. Advances in Knowledge Discovery and Data Mining, Springer International Publishing, 2022, pp. 1–82.
[30] R. Kihlman, M. Fasli, Improving the co-training algorithm to enhance semi-supervised learning results, in: 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan. IEEE, 2022, pp. 5962–5970..
[31] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in: Proceedings of the Eleventh Annual Conference on Computational Learning Theory. Madison Wisconsin USA. ACM, 1998.
[32] J. Wang, S.W. Luo, X.H. Zeng, A random subspace method for co-training, in: 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). Hong Kong, China, IEEE, 2008, pp. 195–200.
[33] Y. Yaslan, Z. Cataltepe, Co-training with relevant random subspaces, *Neurocomputing* 73 (10–12) (2010) 1652–1661.
[34] L.C. Zheng, Y. Cheng, H.X. Yang, N. Cao, J.R. He, Deep co-attention network for multi-view subspace learning, in: Proceedings of the Web Conference 2021, Ljubljana Slovenia, ACM, 2021.
[35] W.L. Weng, W.W. Zhou, J.Z. Chen, H. Peng, H.M. Cai, Enhancing multi-view clustering through common subspace integration by considering both global similarities and local structures, *Neurocomputing* 378 (2020) 375–386.
[36] X.C. Sheng, X.D. Yue, Novel co-training algorithm based on rough sets, *Appl. Res. Computers* 30 (12) (2013) 3546–3550. (in Chinese)
[37] Y.L. Gong, Q.W. Wu, SIVLC: improving the performance of co-training by sufficient-irrelevant views and label consistency, *Appl. Intell.* 53 (18) (2023) 20710–20729.
[38] J. Du, C.X. Ling, Z.H. Zhou, When does cotraining work in real data? *IEEE Trans. Knowl. Data Eng.* 23 (5) (2011) 788–799.
[39] Z.H. Zhou, M. Li, Tri-training: exploiting unlabeled data using three classifiers, *IEEE Trans. Knowl. Data Eng.* 17 (11) (2005) 1529–1541.
[40] A. Masmoudi, H. Bellaaj, K. Drira, M. Jmaiel, A co-training-based approach for the hierarchical multi-label classification of research papers, *Expert Syst.* 38 (4) (2021) e12613.
[41] H.M. Zhao, J.J. Zheng, W. Deng, Y.J. Song, Semi-supervised broad learning system based on manifold regularization and broad network, *IEEE Trans. Circuits Syst. I Regul. Pap.* 67 (3) (2020) 983–994.
[42] I. Nassar, S. Herath, E. Abbasnejad, W. Buntine, G. Haffari, All labels are not created equal: enhancing semi-supervision *via* label grouping and Co-training, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Nashville, TN, USA, 2021.
[43] Z.H. Zhou, M. Li, Semi-supervised regression with co-training, in: Proceedings of the 19th International Joint Conference on Artificial Intelligenc, Edinburgh, Scotland, UK, 2005.
[44] M.R. Amini, V. Feofanov, L. Pauletto, L. Hadjadj, E. Devijver, Y. Maximov, Self-Training: a Survey, *Neurocomputing* 616 (2025) 128904.
[45] D. Li, Y.Q. Liu, D.P. Huang, Development of semi-supervised multiple-output soft-sensors with co-training and tri-training MPLS and MRVM, *Chemometr. Intell. Lab. Syst.* 199 (2020) 103970.
[46] Q.X. Zhu, H.T. Zhang, Y. Tian, N. Zhang, Y. Xu, Y.L. He, Co-training based virtual sample generation for solving the small sample size problem in process industry, *ISA Trans.* 134 (2023) 290–301.
[47] K.L. Li, J. Zhang, H.Y. Xu, S.Z. Luo, H.X. Li, A semi-supervised extreme learning machine method based on co-training, *J. Comput. Inf. Syst.* 9 (1) (2013) 207–214.
[48] L. Bao, X.F. Yuan, Z.Q. Ge, Co-training partial least squares model for semi-supervised soft sensor development, *Chemometr. Intell. Lab. Syst.* 147 (2015) 75–85.
[49] Y.F. Li, D.M. Liang, Safe semi-supervised learning: a brief introduction, *Front. Comput. Sci.* 13 (4) (2019) 669–676.
[50] Y. Gong, J. Lu, Co-training method combined with semi-supervised clustering and weighted k-nearest neighbor, *Comput Eng. Appl.* 55 (2019) 114–118l.
[51] J. Lu, Y.L. Gong, A co-training method based on entropy and multi-criteria, *Appl. Intell.* 51 (6) (2021) 3212–3225.
[52] Y.L. Gong, Q.W. Wu, D.D. Cheng, A co-training method based on parameter-free and single-step unlabeled data selection strategy with natural neighbors, *Int. J. Mach. Learn. Cybern.* 14 (8) (2023) 2887–2902.
[53] G.Y. Deng, A.K. Tieu, L.H. Su, H.T. Zhu, M. Reid, Q. Zhu, C. Kong, Microstructural study and residual stress measurement of a hot rolling work roll material during isothermal oxidation, *Int. J. Adv. Manuf. Technol.* 102 (5) (2019) 2107–2118.
[54] X. Li, R.P. Yu, P.F. Wang, R. Kang, Z.Q. Shu, Z.S. Yue, Z.Y. Zhao, X. Wang, T.J. Lu, Plastic deformation and ductile fracture of L907A ship steel at increasing strain rate and temperature, *Int. J. Impact Eng.* 174 (2023) 104515.
[55] X.J. Sun, S.F. Yuan, Z.J. Xie, L.L. Dong, C.J. Shang, R.D.K. Misra, Microstructure-property relationship in a high strength-high toughness combination ultra-heavy gauge offshore plate steel: the significance of multiphase microstructure, *Mater. Sci. Eng. A* 689 (2017) 212–219.
[56] M.C. Zhao, K. Yang, Y.Y. Shan, The effects of thermo-mechanical control process on microstructures and mechanical properties of a commercial pipeline steel, *Mater. Sci. Eng. A* 335 (1–2) (2002) 14–20.
[57] S. Witek, A. Milenin, Numerical analysis of temperature and residual stresses in hot-rolled steel strip during cooling in coils, *Arch. Civ. Mech. Eng.* 18 (2) (2018) 659–668.
[58] X.Y. Jiang, Z.Q. Ge, Improving the performance of just-in-time learning-based soft sensor through data augmentation, *IEEE Trans. Ind. Electron.* 69 (12) (2022) 13716–13726.
[59] Z. Nasiri, S. Ghaemifar, M. Naghizadeh, H. Mirzadeh, Thermal mechanisms of grain refinement in steels: a review, *Met. Mater. Int.* 27 (7) (2021) 2078–2094.
[60] A. Kabanov, G. Korpala, R. Kawalla, U. Prahl, Effect of hot rolling and cooling conditions on the microstructure, MA constituent formation, and pipeline steels mechanical properties, *Steel Res. Int.* 90 (6) (2019) 1800336.
[61] A. Javaid, F. Czerwinski, Effect of hot rolling on microstructure and properties of the ZEK100 alloy, *J. Magnesium Alloys* 7 (1) (2019) 27–37.
[62] L.H. Zhou, H.Y. Bi, F.L. Sui, W. Du, X.S. Fang, Influence of finish rolling temperature on microstructure and properties of hot-rolled SUS436L stainless steel, *J. Mater. Eng. Perform.* 32 (18) (2023) 8441–8451.
[63] J. Teixeira, M. Moreno, S.Y.P. Allain, C. Oberbillig, G. Geandier, F. Bonnet, Intercritical annealing of cold-rolled ferrite-pearlite steel: microstructure evolutions and phase transformation kinetics, *Acta Mater.* 212 (2021) 116920.
[64] A. Oliver, A. Odena, C. Raffel, E.D. Cubuk, I.J. Goodfellow, Realistic evaluation of deep semi-supervised learning algorithms, in:Proceedings of the 32nd International Conference on Neural Information Processing System, Montreal, Canada , 2018.
[65] Y.F. Li, H.W. Zha, Z.H. Zhou, Learning safe prediction for semi-supervised regression, in: Proceedings of the AAAI Conference on Artificial Intelligence, US, San Francisco, 2017.
[66] A. Kraskov, H. Stögbauer, P. Grassberger, Estimating mutual information, *Phys. Rev. E* 69 (6) (2004) 066138.
[67] M. Ahmed, R. Seraj, S.M.S. Islam, The k-means algorithm: a comprehensive survey and performance evaluation, *Electronics* 9 (8) (2020) 1295.
[68] P. Bhattacharjee, P. Mitra, A survey of density based clustering algorithms, *Front. Comput. Sci.* 15 (2021) 151308.
[69] B.J. Frey, D. Dueck, Clustering by passing messages between data points, *Science* 315 (5814) (2007) 972–976.
[70] H.X. Xu, Research on Clustering Algorithms in Data mining. In: 2022 3rd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE). Xi'an, IEEE, China, 2022.
[71] J.M. Moreira de Lima, F.M. Ugulino de Araujo, Ensemble deep relevant learning framework for semi-supervised soft sensor modeling of industrial processes, *Neurocomputing* 462 (2021) 154–168.
[72] Z. Li, H.P. Jin, S.L. Dong, B. Qian, B. Yang, X.G. Chen, Semi-supervised ensemble support vector regression based soft sensor for key quality variable estimation of nonlinear industrial processes with limited labeled data, *Chem. Eng. Res. Des.* 179 (2022) 510–526.
[73] J.F. Liu, Y.Q. Chen, M.J. Liu, Z.T. Zhao, SELM: semi-supervised ELM with application in sparse calibrated location estimation, *Neurocomputing* 74 (16) (2011) 2566–2572.
[74] C. Shang, F. Yang, D.X. Huang, W.X. Lyu, Data-driven soft sensor development based on deep learning technique, *J. Process Control* 24 (3) (2014) 223–233.
[75] Z.L. Liao, X.L. Zhang, W. Su, K. Zhan, View-consistent heterogeneous network on graphs with few labeled nodes, *IEEE Trans. Cybern.* 53 (9) (2023) 5523–5532.