





Continual Semisupervised Learning of Echo State Network for Quality Prediction of Multimode Processes

Chao Yang , Graduate Student Member, IEEE, Qiang Liu , Senior Member, IEEE, Yi Liu , Member, IEEE, and Yiu-Ming Cheung , Fellow, IEEE

Abstract—The successive switching nature of multimode processes, coupled with data scarcity, challenges traditional quality prediction models. Specifically, the difficulty of simultaneously collecting abundant labeled datasets from all modes forces the model to update its parameters as modes switch. This leads to the forgetting of historical mode knowledge and hinders the aggregation of knowledge, thereby degrading generalization across modes. To this end, we propose a novel continual semisupervised graph echo state network (CS²GESN). First, a semisupervised graph echo state network (S²GESN) is designed based on the graph smoothing assumption to extract dynamic information from unlabeled samples within each mode. The S²GESN model then evolves into a continual model, CS²GESN, employing an elastic weight consolidation strategy for parameter importance estimation derived from pseudoinverse parameter optimization, facilitating the accumulation of historically learned knowledge. This manner alleviates performance deterioration from data scarcity and information forgetting, and enables more flexible modeling of successive arriving operating modes. The superiority and feasibility of the proposed method are demonstrated through its application to the Tennessee Eastman process and the three-phase flow facility process.

Index Terms—Catastrophic forgetting, continual learning (CL), echo state network (ESN), multimode processes, quality prediction, semisupervised learning (SSL).

I. INTRODUCTION

IN THE competitive landscape of modern industries, multimode processes are indispensable, enabling the production of diverse and specialized products [1], [2]. Effective quality prediction is critical to reduce product defects and optimize operational efficiency [3], [4]. The collection of abundant process datasets has significantly fueled the development of data-driven models that predict product quality by measurable process variables. In particular, data-driven dynamic models [5], [6], [7], [8], [9] exhibit superior performance by concentrating on dynamic information within process datasets, usually implemented by various recurrent neural networks (RNNs), such as long short-term memory [6], gated recurrent unit [7], and echo state networks (ESN) [8], [9], [10], [11].

As a variant of RNNs, ESN has garnered significant interest for modeling dynamic relations, which stem from the concept of sparse reservoir computation. For example, Patané and Xibilia [10] employed ESN to monitor the tail gas concentrations of the sulfur recovery process. Yang et al. [11] introduced a deep memory ESN for predicting temperature in the blast furnace. In addition, to address the issue of collinearity, a distribution ESN integrated with autoencoder [8] is developed for quality prediction of dynamic processes. However, most ESN improvements are difficult to adapt to multimode processes. Specifically, frequent changes in operating modes or feed ingredient ratios can cause significant data distribution discrepancies between modes [12], [13], rendering existing ESN-based dynamic models inappropriate.

To tackle the complexity of modeling multimode processes, various strategies have been developed, such as mixture models [1], [14], [15], adaptive learning-based methods [16], [17], [18], and multiple model-based techniques [19]. However, these approaches require all mode datasets during training, which is often difficult in industrial scenarios. A defining characteristic of multimode processes is the sequential arrival of operating mode, rather than their simultaneous occurrence [20]. This makes it difficult to collect datasets from each mode simultaneously. When a new mode emerges, most approaches build a local prediction

Received 10 January 2025; revised 7 April 2025; accepted 11 May 2025. Date of publication 12 June 2025; date of current version 4 September 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62161160338, Grant U23A20328, Grant 61991401, and Grant U20A20189, in part by the National Natural Science Foundation of China (NSFC)/Research Grants Council (RGC) Joint Research Scheme (N_HKBU214/21), the General Research Fund of RGC under Grant 12202924, Grant 12202622, and Grant 12201323, in part by the RGC Senior Research Fellow Scheme under Grant SRFS2324-2S02, and in part by the China Scholarship Council under Grant 202306440060. Paper no. TII-25-0230. (Corresponding author: Qiang Liu.)

Chao Yang and Qiang Liu are with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, Liaoning 110819, China (e-mail: 2010303@stu.neu.edu.cn; liuq@mail.neu.edu.cn).

Yi Liu is with the Institute of Process Equipment and Control Engineering, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: yliuzju@zjut.edu.cn).

Yiu-Ming Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong 999077, China (e-mail: ymc@comp.hkbu.edu.hk).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TII.2025.3575101>, provided by the authors.

Digital Object Identifier 10.1109/TII.2025.3575101

model using only the new mode's datasets, but this may lead to information forgetting of historical modes, causing performance deterioration when historical modes reappear. An alternative is to retrain the model from scratch with all available datasets, including newly collected mode datasets and stored historical mode datasets. However, this solution imposes a significant burden on data storage and computing resources, and struggles to balance learning across multiple modes. Therefore, developing a quality prediction model that aligns with the nature of successive switching remains a challenge.

The label sparseness is another challenge in modeling multimode processes. In general, the sampling frequency of quality variables is significantly slower than that of process variables [14], [21], [22]. The frequent switching of modes further exacerbates the phenomenon of sparse labeled samples alongside abundant unlabeled samples within each mode [14], [21]. Under these circumstances, supervised learning models struggle to achieve satisfactory performance with limited labeled samples. To alleviate the effect of data scarcity, scholars have developed various methods, primarily including active learning (AL) [23], data augmentation (DA) [24], [25], [26], and semisupervised learning (SSL) [14], [15], [21], [22], [27], [28], [29]. AL and DA directly enrich labeled samples, but they cannot easily adapt to different scenarios. In contrast, SSL offers a viable alternative by mining valuable information from unlabeled samples, exhibiting better generalization and adaptability. For instance, Shao et al. [21] developed a semisupervised Gaussian mixture regression for quality prediction of multimode processes. Furthermore, Yao et al. [14] proposed a semisupervised mixture variational autoencoder regression model. However, these SSL-based approaches that require the precollection of datasets of each mode cannot deal with the successive switching nature of multimode processes. In other words, insufficient data acquired from the single mode make it difficult to aggregate across modes.

Recently, a promising learning paradigm known as continual learning (CL), inspired by the human learning process, has been developed and applied to sequential tasks [30], [31], [32], [33], [34], [35], [36], [37]. A major challenge in sequential tasks is the tendency of models to forget crucial information from previous tasks as they acquire new knowledge, leading to catastrophic forgetting [32], [33]. CL addresses this by facilitating the retention of previously learned information alongside new knowledge acquisition, effectively mitigating catastrophic forgetting and reducing the data storage demands for model retraining. In the literature, the development of CL is generally categorized into three types [36]: regularization-based CL [20], [37], [38], isolation-based CL [39], and memory replay-based CL [40]. Specifically, regularization-based CL introduces a quadratic term to preserve learned knowledge by estimating the importance of model parameters. Isolation-based CL aims to isolate and assign parameters. The basic idea of memory replay-based CL is to store past experiences and replay them whenever it is necessary. In general, regularization-based CL is easy to implement and integrate with the other techniques.

Given that the switching nature of multimode processes is successive and sequential [34], [35], [36], CL presents a viable

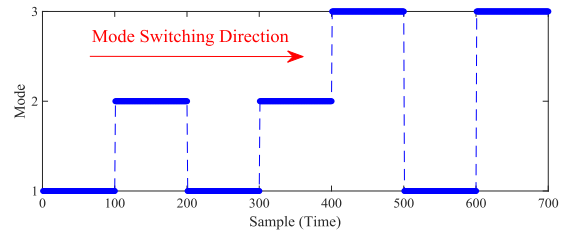


Fig. 1. Switching of different operating modes.

solution for constructing a single model capable of continuously learning new knowledge without forgetting the previous ones. Although CL has been explored for multimode processes [20], [38], quality prediction for multimode processes with sparse labels remains unsolved. It is worth noting that data scarcity may lead to overfitting in CL, preventing the effective consolidation of important knowledge. With this regard, this article presents a continual semisupervised dynamic quality prediction method, termed continual semisupervised graph ESN (CS²GESN). This method is specifically designed for multimode processes with successive arrival modes and sparse labels for each mode. First, leveraging the fast learning capabilities and dynamic description strengths of ESN, a semisupervised graph ESN (S²GESN) is designed to improve generalization performance within each operating mode characterized by sparse labels, which employs manifold regularization to extract valuable information from unlabeled datasets. Second, a regularization-based CL strategy known as elastic weight consolidation (EWC) [37] is introduced to derive a novel continual semisupervised learning (CSSL) algorithm, CS²GESN, which facilitates knowledge accumulation across all operating modes. Notably, EWC is primarily selected for the CL in terms of lower computational cost and the flexibility of integrating it with nondeep learning methods. Finally, the proposed CS²GESN enables flexible modeling by achieving effective combination between SSL and CL for multimode processes with sparse labels. The main contributions of this work are three-fold as follows.

- 1) The concept of CSSL is introduced for quality prediction in the presence of data scarcity, which is the first attempt to multimode processes with sparse labels.
- 2) A novel CSSL-based dynamic modeling method, CS²GESN, is developed to aggregate process dynamic information from each mode, facilitating knowledge accumulation across modes for quality prediction.
- 3) In the parameter consolidation phase, a Fisher information matrix (FIM) estimation formula based on pseudoinverse parameter optimization is designed to assess the significance of output weights \mathbf{W} concerning historical modes.

II. PROBLEM STATEMENT

Three distinct operating modes switch successively and repeatedly, as depicted in Fig. 1. This switching nature presents a challenge: a local model $p(\theta_{\text{old}}|\mathbf{S}_{\text{old}})$, trained on a historical old mode dataset $\{\mathbf{S}_{\text{old}}\}$, must constantly update its parameters $\theta_{\text{old}} \rightarrow \theta_{\text{new}}$ to adapt to new modes. However, this results in an

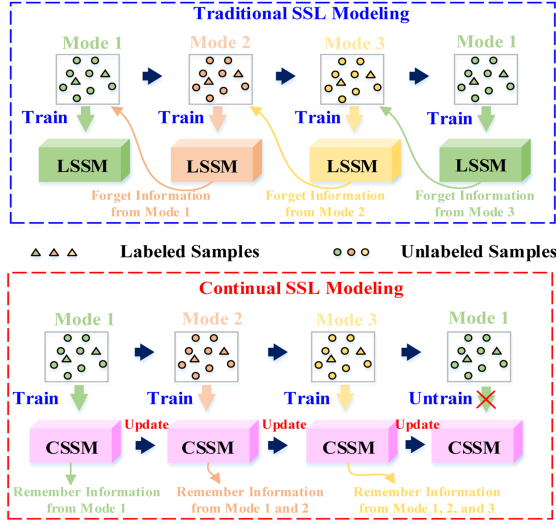


Fig. 2. Comparison diagram of traditional SSL modeling and the proposed continual SSL modeling for multimode processes.

intractable problem: when updating the model using new mode datasets $\{S_{\text{new}}\}$, the updated model $p(\theta_{\text{new}}|S_{\text{new}})$ may overwrite previously learned information from historical modes, causing catastrophic forgetting. As a result, the model's performance significantly deteriorates when a historical mode reappears. Alternatively, one might combine all historical mode datasets with the new mode dataset to update the model's parameters, $p(\theta_{\text{new}}|S_{\text{old}} \cup S_{\text{new}})$. However, this approach is not only resource-intensive in terms of data storage but also struggles with balancing learning across modes due to distribution discrepancies between them. Moreover, the frequent switching between modes complicates the collection of substantial labeled datasets for each mode, exacerbating model deterioration due to the scarcity of labeled samples. As shown in Fig. 2, traditional SSL modeling methods build a local SSL model (LSSM) that is effective only within each mode but fail to aggregate incomplete knowledge from sparse labeled samples across modes. This manner inevitably leads to catastrophic forgetting—where critical features from historical modes are overwritten by new mode datasets—resulting in model confined to specific operating mode. Furthermore, the scarcity of labeled samples allows a loss of modeling information, which in turn hampers the ability of CL to alleviate catastrophic forgetting. In summary, this work proposes a new idea for constructing a continual semisupervised model (CSSM), allowing mutual promotion between SSL and CL in a coupled manner. This strategy facilitates effective knowledge accumulation across modes, which is critical in enhancing comprehensive generalization for multimode processes with sparse labels.

III. PROPOSED CS²GESN ALGORITHM

A. Semisupervised Graph ESN

As a type of randomized RNN for dynamic modeling, ESN consists of three layers, i.e., input layer, reservoir pool layer, and output layer. It is well known that only output weights between

the reservoir pool layer and the output layer need to be trained. However, the scarcity of labeled samples prevents the traditional supervised ESN from accurately capturing dynamics within the abundant unlabeled samples. SSL provides a viable solution to make full use of unlabeled samples for dynamic modeling. In detail, the status of the reservoir can be updated directly using all labeled and unlabeled samples, while the calculation of output weights is performed using only labeled samples. Under the circumstances, the constructed model is regarded as a semisupervised ESN (S²ESN). Consider for the moment all collected input samples $\{X\} = \{X^l \cup X^u\} = [x(1), x(2), \dots, x(N)]^T \in \mathbb{R}^{N \times m}$, the corresponding reservoir state vectors is denoted as $V = [v(1), v(2), \dots, v(N)]^T \in \mathbb{R}^{N \times c}$, where $N = N_l + N_u$, m is the input size, and c is the reservoir size. The output of the labeled samples, denoted as $Y^l \in \mathbb{R}^{N_l}$, are first zero-padded to form another matrix $Y \in \mathbb{R}^N$. Essentially, Y preserves all rows from Y^l whenever $y(i)$ is recorded and populates remaining rows with zeros in instances where $y(i)$ is not recorded. To identify the locations where $y(i)$ is available, a measurement index matrix J is defined as

$$J = \begin{bmatrix} j_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \cdots & 0 \\ 0 & \cdots & j_i & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & j_N \end{bmatrix} \in \mathbb{R}^{N \times N}$$

where $j_i = 1$ when y_i is measured, and $j_i = 0$, otherwise.

Thus, the labeled state vectors $V^l \in \mathbb{R}^{N_l \times c}$ are organized from the state matrix V by the index position of labeled samples, which is described as

$$V^l = JV = [v(j_1^l), v(j_2^l), \dots, v(j_{N_l}^l)]^T \quad (1)$$

where $j_i^l (1 \leq i \leq N_l)$ denotes the index position of labeled sample.

The output weight W_{out} of S²ESN is calculated by labeled state vectors and corresponding output labels as

$$W_{\text{out}}^* = \left((V^l)^T V^l + \gamma I_c \right)^{-1} (V^l)^T Y^l \quad (2)$$

where γ is the regularized coefficient for alleviating overfitting.

Although S²ESN is superior to supervised ESN, it still ignores structure information within unlabeled samples. In general, we expect a smoothing hypothesis [27] that if two samples in the input space are close to each other, then they should have similar outputs, which enables label propagation between similar samples. Therefore, a semisupervised graph echo state network (S²GESN) is constructed by introducing manifold regularization based on the neighbor graph, and its optimization objective is formulated as

$$\begin{aligned} \mathcal{J}_{\text{S}^2\text{GESN}} = & \frac{1}{2} \|Y^l - V^l W_{\text{out}}\|^2 + \frac{\gamma}{2} \|W_{\text{out}}\|^2 \\ & + \frac{\lambda}{2} \sum_{i=1}^N \sum_{j=1}^N \omega(i, j) \|v(i) W_{\text{out}} - v(j) W_{\text{out}}\|^2 \end{aligned} \quad (3)$$

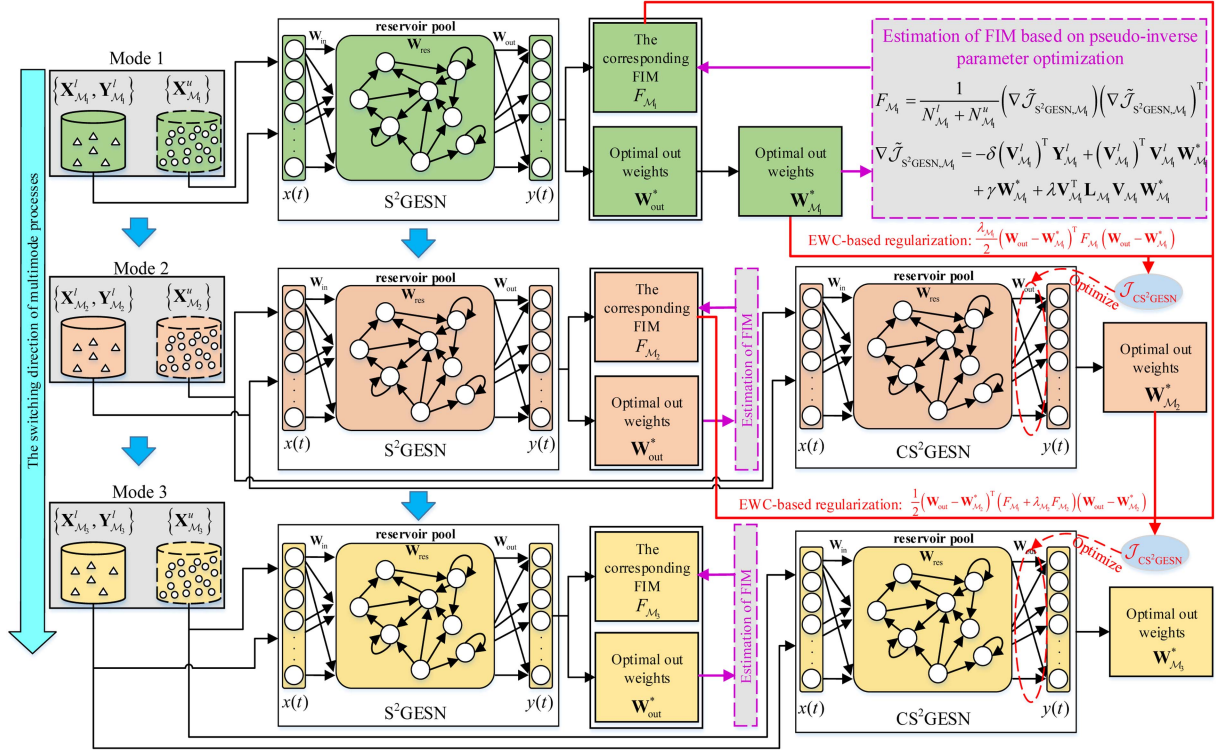


Fig. 3. Overall flowchart of the proposed CS²GESN.

where λ is a balance coefficient for SSL. $\omega(i, j)$, $i, j = 1, \dots, N$, is similarity weight coefficient between neighbors $x(i)$ and $x(j)$, calculated by the following function:

$$\omega(i, j) = \exp\left(-\frac{\|x(i) - x(j)\|^2}{2\sigma^2}\right). \quad (4)$$

Then, the third term of (3) can be converted into matrix form as

$$\begin{aligned} \hat{\Psi} &= \frac{\lambda}{2} \mathbf{W}_{\text{out}}^T \mathbf{V}^T \mathbf{L} \mathbf{V} \mathbf{W}_{\text{out}} \\ &= \frac{\lambda}{2} \mathbf{W}_{\text{out}}^T \mathbf{V}^T (\mathbf{D} - \mathbf{\Omega}) \mathbf{V} \mathbf{W}_{\text{out}} \end{aligned} \quad (5)$$

where \mathbf{L} is the Laplacian matrix, $\mathbf{\Omega} = [w(i, j)]$ represents the adjacency matrix, and \mathbf{D} is a diagonal matrix where each element, \mathbf{D}_{ii} , equals the sum of weights $\sum_{j=1}^{N_l+N_u} w(i, j)$.

Furthermore, (3) can be reformulated as

$$\begin{aligned} \mathcal{J}_{\text{S}^2\text{GESN}} &= \frac{1}{2} \|\mathbf{Y}^l - \mathbf{V}^l \mathbf{W}_{\text{out}}\|^2 + \frac{\gamma}{2} \|\mathbf{W}_{\text{out}}\|^2 \\ &\quad + \frac{\lambda}{2} \mathbf{W}_{\text{out}}^T \mathbf{V}^T \mathbf{L} \mathbf{V} \mathbf{W}_{\text{out}}. \end{aligned} \quad (6)$$

Taking derivatives of (6) concerning \mathbf{W}_{out} , and setting it to zero, we have the following equation:

$$\begin{aligned} \nabla \mathcal{J}_{\text{S}^2\text{GESN}} &= \frac{\partial \mathcal{J}_{\text{S}^2\text{GESN}}}{\partial \mathbf{W}_{\text{out}}} \\ &= -(\mathbf{V}^l)^T \mathbf{Y}^l + (\mathbf{V}^l)^T \mathbf{V}^l \mathbf{W}_{\text{out}} + \gamma \mathbf{W}_{\text{out}} \\ &\quad + \lambda \mathbf{V}^T \mathbf{L} \mathbf{V} \mathbf{W}_{\text{out}} = 0. \end{aligned} \quad (7)$$

In the end, (7) can be calculated as

$$\mathbf{W}_{\text{out}}^* = \left((\mathbf{V}^l)^T \mathbf{V}^l + \gamma \mathbf{I}_c + \lambda \mathbf{V}^T \mathbf{L} \mathbf{V} \right)^{-1} (\mathbf{V}^l)^T \mathbf{Y}^l. \quad (8)$$

Remark 1: Note that all labeled and unlabeled samples with noise provides valuable information for updating the reservoir status whenever there is significant lack of labeled samples. Meanwhile, we incorporate a manifold regularization term to mitigate the negative effects of noises.

B. Continual S²GESN Model

The successive switching nature of multimode processes poses a significant challenge for modeling. On one hand, incomplete knowledge acquired from a single mode through SSL is difficult to effectively aggregate, limiting its generalization across modes. On the other hand, in the case of data scarcity, traditional CL struggles to accumulate critical knowledge. To this end, we develop a novel continual semisupervised modeling method, i.e., CS²GESN, which achieves mutual promotion between SSL and CL in a coupled manner, facilitating knowledge accumulation across modes, and its overall flowchart is illustrated in Fig. 3. Consider mode \mathcal{M}_1 as representing the historical mode and mode \mathcal{M}_2 as representing the new mode, mode \mathcal{M}_1 collects datasets $\mathbf{S}_{\mathcal{M}_1}$ with $N_{\mathcal{M}_1}$ samples, including $N_{\mathcal{M}_1}^l$ labeled samples $\{\mathbf{X}_{\mathcal{M}_1}^l, \mathbf{Y}_{\mathcal{M}_1}^l\}$ and $N_{\mathcal{M}_1}^u$ unlabeled samples $\{\mathbf{X}_{\mathcal{M}_1}^u, \emptyset\}$, where $N_{\mathcal{M}_1}^u \gg N_{\mathcal{M}_1}^l$. Similarly, mode \mathcal{M}_2 collects datasets $\mathbf{S}_{\mathcal{M}_2}$ with $N_{\mathcal{M}_2}$ samples, including $N_{\mathcal{M}_2}^l$ labeled samples $\{\mathbf{X}_{\mathcal{M}_2}^l, \mathbf{Y}_{\mathcal{M}_2}^l\}$ and $N_{\mathcal{M}_2}^u$ unlabeled samples $\{\mathbf{X}_{\mathcal{M}_2}^u, \emptyset\}$, where $N_{\mathcal{M}_2}^u \gg N_{\mathcal{M}_2}^l$. After the training of mode \mathcal{M}_1 , the optimal parameters $\theta_{\mathcal{M}_1}^*$ of

S²GESN are obtained. We expect to be able to further learn the new mode \mathcal{M}_2 without degrading the performance of old mode \mathcal{M}_1 . Hence, by considering the importance of parameters $\theta_{\mathcal{M}_1}^*$ in the old mode \mathcal{M}_1 , the objective function of CS²GESN for training new mode \mathcal{M}_2 is defined as

$$\mathcal{J}_{\text{CS}^2\text{GESN}}(\theta) = \mathcal{J}_{\text{S}^2\text{GESN}}(\theta, \mathbf{S}_{\mathcal{M}_2}) + \mathcal{J}_{\text{loss}}(\theta, \theta_{\mathcal{M}_1}^*, \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1}) \quad (9)$$

where $\mathcal{J}_{\text{CS}^2\text{GESN}}(\theta)$ is the loss function of CS²GESN based on mode \mathcal{M}_2 datasets and mode \mathcal{M}_1 parameters. $\mathcal{J}_{\text{S}^2\text{GESN}}(\theta, \mathbf{S}_{\mathcal{M}_2})$ represents the fitting error term of S²GESN for \mathcal{M}_2 . $\mathcal{J}_{\text{loss}}(\theta, \theta_{\mathcal{M}_1}^*, \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1})$ denotes the information loss for the old mode \mathcal{M}_1 . Here, $\lambda_{\mathcal{M}_1}$ is penalty coefficient for CL and $\mathbf{F}_{\mathcal{M}_1}$ is the FIM for \mathcal{M}_1 .

The second term on the right-hand side of (9) can be reorganized as

$$\begin{aligned} \mathcal{J}_{\text{loss}}(\theta, \theta_{\mathcal{M}_1}^*, \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1}) &= \mathcal{J}_{\text{loss}}(\mathbf{W}_{\text{out}}, \mathbf{W}_{\mathcal{M}_1}^*, \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1}) \\ &= \frac{\lambda_{\mathcal{M}_1}}{2} (\mathbf{W}_{\text{out}} - \mathbf{W}_{\mathcal{M}_1}^*)^\top \mathbf{F}_{\mathcal{M}_1} \\ &\quad \cdot (\mathbf{W}_{\text{out}} - \mathbf{W}_{\mathcal{M}_1}^*). \end{aligned} \quad (10)$$

Substituting (6) and (10) into (9), we have

$$\begin{aligned} \mathcal{J}_{\text{CS}^2\text{GESN}}(\mathbf{W}_{\text{out}}) &= \frac{1}{2} \|\mathbf{Y}_{\mathcal{M}_2}^l - \mathbf{V}_{\mathcal{M}_2}^l \mathbf{W}_{\text{out}}\|^2 + \frac{\gamma}{2} \|\mathbf{W}_{\text{out}}\|^2 \\ &\quad + \frac{\lambda}{2} \mathbf{W}_{\text{out}}^\top \mathbf{V}_{\mathcal{M}_2}^\top \mathbf{L}_{\mathcal{M}_2} \mathbf{V}_{\mathcal{M}_2} \mathbf{W}_{\text{out}} \\ &\quad + \frac{\lambda_{\mathcal{M}_1}}{2} (\mathbf{W}_{\text{out}} - \mathbf{W}_{\mathcal{M}_1}^*)^\top \mathbf{F}_{\mathcal{M}_1} \\ &\quad \cdot (\mathbf{W}_{\text{out}} - \mathbf{W}_{\mathcal{M}_1}^*). \end{aligned} \quad (11)$$

It is clear that the objective function (11) can reduce to a S²GESN objective if $\lambda_{\mathcal{M}_1} = 0$, i.e., all information of old mode \mathcal{M}_1 is discarded.

Taking derivatives of (11) concerning \mathbf{W}_{out} , and setting it to zero, we have the following equation:

$$\begin{aligned} \nabla \mathcal{J}_{\text{CS}^2\text{GESN}} &= \frac{\partial \mathcal{J}_{\text{CS}^2\text{GESN}}}{\partial \mathbf{W}_{\text{out}}} \\ &= -(\mathbf{V}_{\mathcal{M}_2}^l)^\top \mathbf{Y}_{\mathcal{M}_2}^l + (\mathbf{V}_{\mathcal{M}_2}^l)^\top \mathbf{V}_{\mathcal{M}_2}^l \mathbf{W}_{\text{out}} \\ &\quad + \gamma \mathbf{W}_{\text{out}} + \lambda \mathbf{V}_{\mathcal{M}_2}^\top \mathbf{L}_{\mathcal{M}_2} \mathbf{V}_{\mathcal{M}_2} \mathbf{W}_{\text{out}} \\ &\quad + \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1} (\mathbf{W}_{\text{out}} - \mathbf{W}_{\mathcal{M}_1}^*) = 0. \end{aligned} \quad (12)$$

The output weight \mathbf{W}_{out} of CS²GESN is derived from (12)

$$\mathbf{W}_{\text{out}}^* = \mathbf{\Gamma}^{-1} \left((\mathbf{V}_{\mathcal{M}_2}^l)^\top \mathbf{Y}_{\mathcal{M}_2}^l + \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1} \mathbf{W}_{\mathcal{M}_1}^* \right) \quad (13)$$

where $\mathbf{\Gamma}$ is denoted as

$$\begin{aligned} \mathbf{\Gamma} &= (\mathbf{V}_{\mathcal{M}_2}^l)^\top \mathbf{V}_{\mathcal{M}_2}^l + \gamma \mathbf{I}_k \\ &\quad + \lambda \mathbf{V}_{\mathcal{M}_2}^\top \mathbf{L}_{\mathcal{M}_2} \mathbf{V}_{\mathcal{M}_2} + \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1}. \end{aligned} \quad (14)$$

Remark 2: To overcome data scarcity, on one hand, manifold regularization constraint in SSL is to decrease the upper bound of generalization error. The inclusion of unlabeled samples

helps maintain smoothing of the data manifold, thereby enabling the model generalization. On the other hand, an EWC-based regularization in CL is to consolidate learned knowledge from historical data, enabling knowledge accumulation across modes.

C. Estimation of FIM for CS²GESN Model

Before solving (11), output weight $\mathbf{W}_{\mathcal{M}_1}^*$ and FIM $\mathbf{F}_{\mathcal{M}_1}$ need to be firstly calculated. As the optimal parameters of the mode \mathcal{M}_1 , $\mathbf{W}_{\mathcal{M}_1}^*$ is easy to obtain by S²GESN. The calculation of $\mathbf{F}_{\mathcal{M}_1}$ about the optimal output weight $\mathbf{W}_{\mathcal{M}_1}^*$ mainly depends on the gradient of the S²GESN optimization objective

$$\mathbf{F}_{\mathcal{M}_1} = \frac{1}{N_{\mathcal{M}_1}^l + N_{\mathcal{M}_1}^u} (\nabla \mathcal{J}_{\text{S}^2\text{GESN}, \mathcal{M}_1}) (\nabla \mathcal{J}_{\text{S}^2\text{GESN}, \mathcal{M}_1})^\top. \quad (15)$$

Note that the calculation of (15) is close to zero, mainly because the optimization goal of S²GESN is expected to be minimized by making $\nabla \mathcal{J}_{\text{S}^2\text{GESN}}$ equal to zero. Therefore, employing $\mathbf{F}_{\mathcal{M}_1}$ in (15) approaching zero to estimate the parameters importance is inappropriate. To avoid this issue, a small perturbation coefficient δ is added to (7) to obtain a modified nonzero gradient

$$\begin{aligned} \nabla \tilde{\mathcal{J}}_{\text{S}^2\text{GESN}, \mathcal{M}_1} &= -\delta (\mathbf{V}_{\mathcal{M}_1}^l)^\top \mathbf{Y}_{\mathcal{M}_1}^l + (\mathbf{V}_{\mathcal{M}_1}^l)^\top \mathbf{V}_{\mathcal{M}_1}^l \mathbf{W}_{\mathcal{M}_1}^* \\ &\quad + \gamma \mathbf{W}_{\mathcal{M}_1}^* + \lambda \mathbf{V}_{\mathcal{M}_1}^\top \mathbf{L}_{\mathcal{M}_1} \mathbf{V}_{\mathcal{M}_1} \mathbf{W}_{\mathcal{M}_1}^* \end{aligned} \quad (16)$$

where $\nabla \tilde{\mathcal{J}}_{\text{S}^2\text{GESN}, \mathcal{M}_1}$ denotes the modified nonzero gradient for mode \mathcal{M}_1 , δ can be set to any value other than $\delta = 1$. The detailed explanation for (16) is provided in section Appendix I of the Supplementary Material.

Hence, (15) can be further formulated as

$$\mathbf{F}_{\mathcal{M}_1} = \frac{1}{N_{\mathcal{M}_1}^l + N_{\mathcal{M}_1}^u} (\nabla \tilde{\mathcal{J}}_{\text{S}^2\text{GESN}, \mathcal{M}_1}) (\nabla \tilde{\mathcal{J}}_{\text{S}^2\text{GESN}, \mathcal{M}_1})^\top. \quad (17)$$

Remark 3: When we get the optimal output weight, the FIM derived from these gradients of (15), with each diagonal element close to zero, is not an effective measure of output weights importance. According to [41], the gradients directly reflect the rate of change of the loss function. Therefore, an output label bias is introduced in the proposed S²GESN to obtain a nonzero gradient. The estimated FIM in (17) with nonzero gradients are sensitive to the historical operating modes, enabling the retention of historical data learned knowledge and alleviating catastrophic forgetting.

D. Recursive Form of CS²GESN Model

For more than three successive modes, the recursive form of CS²GESN is briefly derived in this section. When datasets $\mathbf{S}_{\mathcal{M}_3}$ from mode \mathcal{M}_3 arrives, and historical mode datasets $\mathbf{S}_{\mathcal{M}_1}$ and $\mathbf{S}_{\mathcal{M}_2}$ are not available, the Bayesian posterior is further derived as

$$\begin{aligned} \log p(\theta|\mathbf{S}) &= \log p(\theta|\mathbf{S}_{\mathcal{M}_1}, \mathbf{S}_{\mathcal{M}_2}, \mathbf{S}_{\mathcal{M}_3}) \\ &= \log p(\mathbf{S}_{\mathcal{M}_3}|\theta) + \log p(\theta|\mathbf{S}_{\mathcal{M}_1}, \mathbf{S}_{\mathcal{M}_2}) \\ &\quad + \text{constant} \end{aligned} \quad (18)$$

where \mathbf{S} contains datasets from three modes, i.e., $\mathbf{S}_{\mathcal{M}_1}$, $\mathbf{S}_{\mathcal{M}_2}$, and $\mathbf{S}_{\mathcal{M}_3}$. After learning historical mode \mathcal{M}_1 and \mathcal{M}_2 , the corresponding datasets $\mathbf{S}_{\mathcal{M}_1}$ and $\mathbf{S}_{\mathcal{M}_2}$ are discarded. Therefore, the recursive Laplace approximation is adopted to approximate (18).

Similar to (9), the objective function of CS²GESN for new mode \mathcal{M}_3 is described as

$$\mathcal{J}_{\text{CS}^2\text{GESN}}(\theta) = \mathcal{J}_{\text{S}^2\text{GESN}}(\theta, \mathbf{S}_{\mathcal{M}_3}) + \mathcal{J}_{\text{loss}}(\theta, \theta_{\mathcal{M}_2}^*, \Upsilon_{\mathcal{M}_2}). \quad (19)$$

With the successive arrival of new modes, the above derivation can be reformulated as a more general form. For the emerging k th ($3 \leq k \leq K$) mode to be learned, denoted as \mathcal{M}_k , the datasets are denoted as $\mathbf{S}_{\mathcal{M}_k}$, the modeling objective is formulated as follows:

$$\begin{aligned} \log p(\theta|\mathbf{S}) &= \log p(\mathbf{S}_{\mathcal{M}_k}|\theta) + \log p(\theta|\mathbf{S}_{\mathcal{M}_1}, \dots, \mathbf{S}_{\mathcal{M}_{k-1}}) \\ &+ \text{constant}. \end{aligned} \quad (20)$$

Using recursive Laplace approximation, (20) can be can be approximately formulated as

$$\begin{aligned} -\log p(\theta|\mathbf{S}) &\approx -\log p(\mathbf{S}_{\mathcal{M}_k}|\theta) + \frac{1}{2}(\theta - \theta_{\mathcal{M}_{k-1}}^*)^\top \\ &\quad \Upsilon_{\mathcal{M}_{k-1}}(\theta - \theta_{\mathcal{M}_{k-1}}^*) + \text{constant} \end{aligned} \quad (21)$$

where

$$\begin{aligned} \Upsilon_{\mathcal{M}_{k-1}} &= \Upsilon_{\mathcal{M}_{k-2}} + \lambda_{\mathcal{M}_{k-1}} \mathbf{F}_{\mathcal{M}_{k-1}}, \quad k \geq 3 \\ \Upsilon_{\mathcal{M}_1} &= \lambda_{\mathcal{M}_1} \mathbf{F}_{\mathcal{M}_1} \end{aligned} \quad (22)$$

where $\mathbf{F}_{\mathcal{M}_{k-1}}$ denotes the FIM for mode \mathcal{M}_{k-1} , which can be calculated in sequence by (17). $\lambda_{\mathcal{M}_{k-1}}$ is the penalty coefficient estimating the importance of mode \mathcal{M}_{k-1} . Accordingly, the recursive form of (19) is given as

$$\begin{aligned} \mathcal{J}_{\text{CS}^2\text{GESN}}(\mathbf{W}_{\mathcal{M}_k}) &= \mathcal{J}_{\text{S}^2\text{GESN}}(\mathbf{W}_{\mathcal{M}_k}, \mathbf{S}_{\mathcal{M}_k}) \\ &+ \mathcal{J}_{\text{loss}}(\mathbf{W}_{\mathcal{M}_k}, \mathbf{W}_{\mathcal{M}_{k-1}}^*, \Upsilon_{\mathcal{M}_{k-1}}) \end{aligned} \quad (23)$$

where $\mathbf{W}_{\mathcal{M}_{k-1}}^*$ is the output weight of CS²GESN in the previous mode \mathcal{M}_{k-1} . $\mathcal{J}_{\text{S}^2\text{GESN}}(\mathbf{W}_{\mathcal{M}_k}, \mathbf{S}_{\mathcal{M}_k})$ is the objective loss of S²GESN for mode \mathcal{M}_k . $\mathcal{J}_{\text{loss}}(\mathbf{W}_{\mathcal{M}_k}, \mathbf{W}_{\mathcal{M}_{k-1}}^*, \Upsilon_{\mathcal{M}_{k-1}})$ is an information loss approximating the loss sum of the historical $k-1$ modes. In the end, the implementation details of S²GESN and CS²GESN are summarized in Algorithm1 and Algorithm2 of the Supplementary Material.

E. Computation Complexity Analysis

Taking two available modes \mathcal{M}_1 and \mathcal{M}_2 as an example, the computational complexity of the proposed method includes three components: 1) The computational complexity for the optimization of S²GESN for the historical model \mathcal{M}_1 is $\mathcal{O}(N_{\mathcal{M}_1}mc + c^3 + N_{\mathcal{M}_1}^l c + (N_{\mathcal{M}_1})^2)$; 2) the computational complexity for the estimation of FIM denoted by $F_{\mathcal{M}_1}$ for the historical mode \mathcal{M}_1 is $\mathcal{O}(c^2)$; 3) the computational complexity for the optimization of CS²GESN model for the new mode \mathcal{M}_2 is $\mathcal{O}(N_{\mathcal{M}_2}mk + c^3 + c^2 + N_{\mathcal{M}_2}^l c + (N_{\mathcal{M}_2})^2)$. Meanwhile, the computational complexity for the optimization of mixture semisupervised graph ESN (MS²GESN)

TABLE I
MODE PARAMETERS OF TE PROCESS

Mode	G/H ratio	Production level (kg/h)
\mathcal{M}_1	50/50	14076
\mathcal{M}_2	10/90	14077
\mathcal{M}_3	50/50	Maximum

is $\mathcal{O}(\sum_{i=1}^2 N_{\mathcal{M}_i}mc + c^3 + \sum_{i=1}^2 N_{\mathcal{M}_i}^l c + (\sum_{i=1}^2 N_{\mathcal{M}_i})^2)$. It is observed that the proposed CS²GESN does not impose a significant increase on computational cost.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

To verify the effectiveness of the proposed CS²GESN, in this section, two typical multimode industrial cases are investigated, including the Tennessee Eastman (TE) process and the three-phase flow facility (TPFF) process. In our experiment, our proposed CS²GESN and some baselines are listed below.

ESN: ESN, which is a supervised model only suitable for single mode.

Continual ESN (CESN): CESN, which is a continual supervised model suitable for multiple successive modes.

S²ESN: S²ESN, which is a semisupervised model only suitable for single mode.

S²GESN: S²GESN, which is a semisupervised model that combines graph regularization and is only suitable for single mode.

MS²GESN: MS²ESN, which builds a model using datasets from all available modes.

CS²ESN: CS²ESN, which is a CSSM suitable for multiple successive modes.

CS²GESN (Proposed): CS²GESN, which is a CSSM that combines graph regularization and is suitable for multiple successive modes.

The experimental schemes for the proposed CS²GESN and other baselines are detailed in Table AI of the Supplementary Material. Three common metrics, root-mean-squared error (RMSE), mean absolute error (MAE), and R-square (R²), are adopted to evaluate the modeling performance.

A. TE Process Benchmark

The TE process in [42] is first utilized to evaluate the proposed CS²GESN. It includes a reactor, a stripper, a recycle compressor, a condenser, and a separator. In this process, there are 12 manipulated variables XMV (1–12) and 41 measured variables XMEAS (1–41). The measured variables are further classified into 22 continuous variables XMEAS (1–22) and 19 component variables XMEAS (23–41). To implement quality prediction, only 22 continuous variables XMEAS (1–22) are selected as input variables, and the component variable XMEAS 38 is chosen as the quality variable. Meanwhile, based on the mode parameters listed in Table I, three steady operating modes, denoted by $\{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3\}$, are simulated, with 1000 samples collected for each mode.

In the experimental phase, for each mode, the first 500 samples are considered the training datasets, while the remaining one

TABLE II
PREDICTION PERFORMANCE COMPARISON OF CS²GESN AND OTHER BASELINES FOR TE PROCESS

Scenario	Methods	Training sources (Data and Model)	Model label	Test sources	LR: 5%			LR: 10%			LR: 20%		
					RMSE	MAE	R ²	RMSE	MAE	R ²	RMSE	MAE	R ²
S1	S ² GESN	\mathcal{M}_1	A	\mathcal{M}_1	0.0224	0.0183	0.7975	0.0217	0.0176	0.8104	0.0216	0.0172	0.8132
S2	CS ² GESN	\mathcal{M}_2 and A	B	\mathcal{M}_2	0.0531	0.0391	0.7772	0.0438	0.0320	0.8481	0.0444	0.0345	0.8445
S3	CS ² GESN	—	B	\mathcal{M}_1	0.0250	0.0203	0.7493	0.0248	0.0197	0.7524	0.0246	0.0196	0.7560
S4	S ² GESN	\mathcal{M}_2	C	\mathcal{M}_2	0.0569	0.0448	0.7439	0.0490	0.0381	0.8098	0.0462	0.0370	0.8312
S5	S ² GESN	—	C	\mathcal{M}_1	0.0327	0.0264	0.5705	0.0289	0.0235	0.6646	0.0307	0.0255	0.6212
S6	CS ² GESN	\mathcal{M}_3 and B	D	\mathcal{M}_3	0.1297	0.0940	0.8095	0.1240	0.0954	0.8258	0.1403	0.1120	0.7769
S7	CS ² GESN	—	D	\mathcal{M}_2	0.0648	0.0503	0.6684	0.0561	0.0449	0.7512	0.0485	0.0387	0.8143
S8	CS ² GESN	—	D	\mathcal{M}_1	0.0305	0.0245	0.6262	0.0302	0.0246	0.6325	0.0287	0.0233	0.6682
S9	S ² GESN	\mathcal{M}_3	E	\mathcal{M}_3	0.1195	0.0852	0.8382	0.1055	0.0688	0.8738	0.0944	0.0699	0.8990
S10	S ² GESN	—	E	\mathcal{M}_2	0.0702	0.0553	0.6103	0.0284	0.0227	0.6755	0.0792	0.0671	0.5041
S11	S ² GESN	—	E	\mathcal{M}_1	0.0341	0.0286	0.5318	0.0869	0.0729	0.4027	0.0372	0.0308	0.4448
S12	MS ² GESN	$\mathcal{M}_1, \mathcal{M}_2$	F	\mathcal{M}_2	0.0637	0.0494	0.6786	0.0539	0.0411	0.7700	0.0479	0.0364	0.8185
S13	MS ² GESN	—	F	\mathcal{M}_1	0.0276	0.0222	0.6944	0.0243	0.0192	0.7630	0.0256	0.0206	0.7358
S14	MS ² GESN	$\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$	G	\mathcal{M}_3	0.1807	0.1464	0.6298	0.1560	0.1276	0.7240	0.1526	0.1212	0.7361
S15	MS ² GESN	—	G	\mathcal{M}_2	0.0627	0.0466	0.6895	0.0510	0.0384	0.7942	0.0541	0.0417	0.7682
S16	MS ² GESN	—	G	\mathcal{M}_1	0.0263	0.0212	0.7228	0.0282	0.0228	0.6799	0.0291	0.0229	0.6594
S17	S ² ESN	\mathcal{M}_1	H	\mathcal{M}_1	0.0234	0.0184	0.7809	0.0237	0.0184	0.7748	0.0217	0.0172	0.8109
S18	CS ² ESN	\mathcal{M}_2 and H	I	\mathcal{M}_2	0.0562	0.0421	0.7502	0.0492	0.0368	0.8086	0.0444	0.0345	0.8439
S19	CS ² ESN	—	I	\mathcal{M}_1	0.0291	0.0237	0.6591	0.0269	0.0214	0.7093	0.0249	0.0198	0.7508
S20	S ² ESN	\mathcal{M}_2	J	\mathcal{M}_2	0.0575	0.0433	0.7390	0.0545	0.0419	0.7654	0.0464	0.0372	0.8299
S21	S ² ESN	—	J	\mathcal{M}_1	0.0326	0.0265	0.5733	0.0302	0.0244	0.6344	0.0301	0.0250	0.6366
S22	CS ² ESN	\mathcal{M}_3 and I	K	\mathcal{M}_3	0.1347	0.0853	0.7945	0.1197	0.0858	0.8376	0.1416	0.1129	0.7728
S23	CS ² ESN	—	K	\mathcal{M}_2	0.0858	0.0713	0.4180	0.0738	0.0599	0.5688	0.0486	0.0388	0.8132
S24	CS ² ESN	—	K	\mathcal{M}_1	0.0394	0.0321	0.3774	0.0331	0.0261	0.5600	0.0290	0.0235	0.6632
S25	S ² ESN	\mathcal{M}_3	L	\mathcal{M}_3	0.1396	0.0898	0.7790	0.1153	0.0808	0.8494	0.0940	0.0693	0.8999
S26	S ² ESN	—	L	\mathcal{M}_2	0.0959	0.0798	0.2729	0.0786	0.0669	0.5110	0.0795	0.0675	0.4998
S27	S ² ESN	—	L	\mathcal{M}_1	0.0410	0.0349	0.3255	0.0353	0.0299	0.5006	0.0370	0.0307	0.4506
S28	ESN	\mathcal{M}_1	M	\mathcal{M}_1	0.0408	0.0329	0.3312	0.0286	0.0232	0.6720	0.0251	0.0206	0.7471
S29	CESN	\mathcal{M}_2 and M	N	\mathcal{M}_2	0.0734	0.0611	0.5738	0.0643	0.0487	0.6735	0.0472	0.0375	0.8238
S30	CESN	—	N	\mathcal{M}_1	0.0388	0.0303	0.3942	0.0356	0.0285	0.4919	0.0324	0.0261	0.5774
S31	ESN	\mathcal{M}_2	O	\mathcal{M}_2	0.1031	0.0780	0.1599	0.0815	0.0625	0.4747	0.0661	0.0509	0.6540
S32	ESN	—	O	\mathcal{M}_1	0.0454	0.0361	0.1713	0.0444	0.0371	0.2084	0.0351	0.0287	0.5044
S33	CESN	\mathcal{M}_3 & N	P	\mathcal{M}_3	0.2300	0.1839	0.4002	0.1930	0.1522	0.5779	0.1598	0.1314	0.7106
S34	CESN	—	P	\mathcal{M}_2	0.0884	0.0670	0.3818	0.0794	0.0611	0.5016	0.0558	0.0464	0.7534
S35	CESN	—	P	\mathcal{M}_1	0.0402	0.0319	0.3510	0.0391	0.0309	0.3867	0.0375	0.0318	0.4355
S36	ESN	\mathcal{M}_3	Q	\mathcal{M}_3	0.2751	0.2040	0.1422	0.1956	0.1193	0.5665	0.1748	0.1286	0.6537
S37	ESN	—	Q	\mathcal{M}_2	0.1254	0.0978	-0.2439	0.1003	0.0747	0.2037	0.0956	0.0772	0.2768
S38	ESN	—	Q	\mathcal{M}_1	0.0560	0.0465	-0.2613	0.0434	0.0357	0.2443	0.0438	0.0361	0.2278

* LR: The label ratio of training datasets.

constitute the test datasets. It is noted that only a small number of training datasets for each mode are annotated due to fast switching between modes. Consequently, three types of label ratios (LR), i.e., 5%, 10%, and 20%, are investigated to achieve modeling in scenarios with sparse labels. Through trial and error, the relevant parameters for all methods are set identically for a fair comparison: the input weight \mathbf{W}_{in} randomly initialized within the range $[-1, 1]$, the state weight \mathbf{W}_{res} randomly initialized within the range $[0, 1]$, the reservoir size $c = 500$, the leaking rate $\alpha = 0.95$, and the spectral radius $r = 1.8$. Subsequently, the regularized coefficient γ of all seven models is set to 0.05, while the balance parameter λ , supporting smooth assumptions, is set to 10^{-5} for S²GESN, MS²GESN, and CS²GESN. In addition, for CESN, CS²GESN, and CS²ESN, the penalty parameter for the FIM is set to 10^4 .

A total of 38 scenarios (S1–S38) are conducted for each LR in our work. The prediction performance comparison of CS²GESN and other methods are listed in Table II. Some key insights are summarized as follows.

- 1) By comparing the modeling scenario in a single mode of $\mathcal{M}_1, \mathcal{M}_2$, and \mathcal{M}_3 , that is, S1, S4, S9, S17, S20, S25, S28,

S31, and S36, it is evident that the ESN model performs the worst. Semisupervised models, including S²GESN and S²ESN, outperform ESN, with S²GESN showing the best performance. The primary reason is that the sparse label issue significantly degrades the modeling accuracy of the supervised model, and SSL effectively alleviates it.

- 2) When the new mode \mathcal{M}_2 arrives, S²GESN and S²ESN update their model parameters using the datasets from mode \mathcal{M}_2 . Observations from six scenarios (S1, S4, S5, S17, S20, and S21) indicate that they achieve satisfactory performance in the new mode \mathcal{M}_2 . However, they suffer from catastrophic forgetting in the historical mode \mathcal{M}_1 , leading to poor prediction performance. Furthermore, with the arrival of the new mode \mathcal{M}_3 , a similar phenomenon is concluded in ten scenarios (S1, S4, S9, S10, S11, S17, S20, S25, S26, and S27), where catastrophic forgetting of historical modes \mathcal{M}_1 and \mathcal{M}_2 occurs despite satisfactory performance in the new mode \mathcal{M}_3 .
- 3) In contrast to 2), observing four scenarios (S2, S3, S18, and S19) reveals that the proposed CS²GESN and CS²ESN effectively mitigate catastrophic forgetting by

TABLE III
COMPREHENSIVE PERFORMANCE COMPARISON OF CS²GESN, MS²GESN, CS²ESN, AND CESN FOR TE PROCESS

LR	RMSE / MAE / R ² / MAPE (%)											
	CS ² GESN				MS ² GESN				CS ² ESN			
5%	0.0855	0.0562	0.9329	5.7288	0.1115	0.0714	0.8860	6.8553	0.0949	0.0629	0.9173	6.4365
10%	0.0805	0.0550	0.9406	5.5801	0.0962	0.0629	0.9152	6.1456	0.0834	0.0573	0.9362	5.7387
20%	0.0873	0.0580	0.9301	5.6660	0.0950	0.0619	0.9173	6.0881	0.0880	0.0584	0.9289	5.7017

* The bold and underline value denotes the best performance, LR: The label ratio of training datasets.

retaining important knowledge from the historical mode \mathcal{M}_1 . Similarly, in six scenarios (S6, S7, S8, S22, S23, and S24), catastrophic forgetting of historical modes \mathcal{M}_1 and \mathcal{M}_2 is mitigated when the new mode \mathcal{M}_3 appears. Furthermore, CS²GESN outperforms CS²ESN due to the effective integration of CL and SSL advantages.

- 4) In scenarios S28–S32, although CESN can mitigate the catastrophic forgetting observed in ESN, it struggles to achieve satisfactory performance due to the sparse labels that hinder accurate knowledge capture.
- 5) From the five scenarios (S12–S16), it can be seen that although MS²GESN does not forget much about the historical modes, it is difficult to achieve satisfactory performance for all modes. Specifically, the MS²GESN model may be influenced by distribution differences between modes, hindering its ability to simultaneously describe the complicated data properties of multiple modes. In addition, training on all mode datasets when a new mode arrives is not an ideal solution, as it can lead to a data storage burden.

In summary, from Table II, when SSL and CL are separately applied to the S9–S11 scenarios and the S33–S35 scenarios, neither of them supports the unified modeling objective of aggregating the incomplete process dynamics of the modes, and thus it is difficult to obtain satisfactory performance. In the scenarios of S6–S8, the proposed CS²GESN model achieves an effective combination between SSL and CL, mitigating the negative effect of sparse labels and catastrophic forgetting across modes. Fig. A1 in the Supplementary Material displays the predicted values and absolute errors for CS²GESN, MS²GESN, CS²ESN, and CESN, using 5% labeled training samples. Furthermore, comprehensive prediction performance across all modes is presented in Table III, showing that CS²GESN is the best in all scenarios.

B. TPF Process

The TPF process [43], which aims to supply a controlled flow rate of water, oil, and air to a pressurized facility, is an industrial system from Cranfield University. It can operate in different modes by adjusting two manipulated variables, i.e., the air flow rate and the water flow rate. Fig. A2 of the Supplementary Material includes a gas–liquid separator, a three-phase separator (TPS), numerous coalescers, and storage tanks connected by pipelines of different specifications and geometries. In this process, 16 process variables are chosen as inputs and the TPS pressure is chosen as the quality variable. A detailed description

TABLE IV
SETTINGS FOR MODE PARAMETERS

Mode types	Water flow (Kg/s)	Air flow (m ³ /s)	Sample size
\mathcal{M}_1	2	0.0208	1000
\mathcal{M}_2	1	0.0347	700
\mathcal{M}_3	2	0.0417	800

of the chosen process variables can be found in the work of [43]. In this case, we only investigate three steady operating modes, denoted as $\{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3\}$, with the corresponding parameters listed in Table IV. There are 1000, 700, and 800 samples for \mathcal{M}_1 , \mathcal{M}_2 , and \mathcal{M}_3 , respectively.

In the experimental phase, the former 400 sample points in each mode serve as the training datasets, and the remaining samples constitute the test datasets. Given that the study in [43] provided high-quality datasets with different data distributions, the datasets are normalized first. Three types of LR, i.e., 5%, 10%, and 20%, are investigated to evaluate the proposed model under sparse label scenarios. Through trial and error, the relevant parameters for all methods are set identically for a fair comparison: the input weight \mathbf{W}_{in} randomly initialized within the range $[-1, 1]$, the state weight \mathbf{W}_{res} randomly initialized within the range $[0, 1]$, the size of reservoir $c = 1500$, the leaking rate $\alpha = 0.95$, and the spectral radius $r = 1.8$. Subsequently, the regularized coefficient γ of all seven models is set to 0.05, while the balance coefficient λ , supporting smooth assumptions, is set to 10^{-5} for S²GESN, MS²GESN, and CS²GESN. In addition, for CESN, CS²GESN, and CS²ESN, the penalty coefficient for the FIM is 10^6 .

The prediction performance comparison of CS²GESN and other baselines is listed in Table V. Several key insights are summarized below.

- 1) CS²GESN consistently outperforms other baselines, showing lower RMSE and MAE values and higher R² values. For instance, in scenario S2 of 5% LR, CS²GESN has an RMSE of 1.6063, an MAE of 1.3002, and an R² of 0.7503, outperforming both S²GESN in scenario S4 and MS²GESN in scenario S12.
- 2) Compared to four scenarios (S3, S5, S21, and S32), CS²GESN performs better than S²GESN, S²ESN, and ESN, effectively alleviating the catastrophic forgetting for historical mode. In addition, as observed in three scenarios (S3, S19, and S30), CS²GESN performs better than CS²ESN and CESN. This implies that accurate extraction of valuable knowledge information

TABLE V
PREDICTION PERFORMANCE COMPARISON OF CS²GESN AND OTHER BASELINES FOR TPF PROCESS

Scenario	Methods	Training sources (Data and Model)	Model label	Test sources	LR: 5%			LR: 10%			LR: 20%		
					RMSE ($\times 10^{-4}$)	MAE ($\times 10^{-4}$)	R ²	RMSE ($\times 10^{-4}$)	MAE ($\times 10^{-4}$)	R ²	RMSE ($\times 10^{-4}$)	MAE ($\times 10^{-4}$)	R ²
S1	S ² GESN	\mathcal{M}_1	A	\mathcal{M}_1	2.0951	1.7337	0.5587	2.0663	1.6991	0.5707	1.8390	1.5118	0.6600
S2	CS ² GESN	\mathcal{M}_2 & A	B	\mathcal{M}_2	1.6063	1.3002	0.7503	1.5525	1.2981	0.7668	1.4181	1.1787	0.8054
S3	CS ² GESN	—	B	\mathcal{M}_1	1.8579	1.5382	0.6530	1.7135	1.3490	0.7048	1.6930	1.3910	0.7119
S4	S ² GESN	\mathcal{M}_2	C	\mathcal{M}_2	1.8921	1.5573	0.6536	1.7809	1.4654	0.6931	1.5164	1.2391	0.7775
S5	S ² GESN	—	C	\mathcal{M}_1	2.2929	1.8689	0.4715	2.1494	1.7736	0.5355	2.0954	1.7662	0.5586
S6	CS ² GESN	\mathcal{M}_3 & B	D	\mathcal{M}_3	6.4252	5.1032	0.6006	5.5783	4.1910	0.6990	4.9750	3.8282	0.7606
S7	CS ² GESN	—	D	\mathcal{M}_2	1.5419	1.2487	0.7700	1.3446	1.1163	0.8251	1.5098	1.2178	0.7795
S8	CS ² GESN	—	D	\mathcal{M}_1	1.8265	1.5288	0.6646	1.7687	1.4716	0.6855	1.9900	1.6312	0.6019
S9	S ² GESN	\mathcal{M}_3	E	\mathcal{M}_3	4.4146	3.4730	0.8115	4.5120	3.4434	0.8030	3.9827	3.0875	0.8465
S10	S ² GESN	—	E	\mathcal{M}_2	2.0519	1.6729	0.5926	2.0645	1.7050	0.5876	2.1314	1.7453	0.5604
S11	S ² GESN	—	E	\mathcal{M}_1	2.8600	2.3582	0.1777	2.7477	2.2548	0.2409	2.8325	2.3018	0.1934
S12	MS ² GESN	$\mathcal{M}_1, \mathcal{M}_2$	F	\mathcal{M}_2	1.7026	1.4060	0.7195	1.4610	1.2302	0.7935	1.4816	1.2479	0.7876
S13	MS ² GESN	—	F	\mathcal{M}_1	1.8573	1.5624	0.6532	1.8418	1.5379	0.6589	1.5928	1.3295	0.7449
S14	MS ² GESN	$\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$	G	\mathcal{M}_3	6.6019	4.7094	0.5783	7.6255	5.4003	0.4375	7.1108	5.0207	0.5108
S15	MS ² GESN	—	G	\mathcal{M}_2	1.4682	1.2168	0.7914	1.4344	1.2096	0.8009	1.4192	1.1668	0.8051
S16	MS ² GESN	—	G	\mathcal{M}_1	1.7847	1.4865	0.6798	1.7336	1.4443	0.6979	1.4990	1.2318	0.7741
S17	S ² ESN	\mathcal{M}_1	H	\mathcal{M}_1	2.2528	1.8632	0.4898	2.0924	1.7209	0.5598	1.8369	1.5103	0.6608
S18	CS ² ESN	\mathcal{M}_2 & H	I	\mathcal{M}_2	1.7778	1.4285	0.6942	1.6906	1.4092	0.7235	1.4403	1.1961	0.7993
S19	CS ² ESN	—	I	\mathcal{M}_1	2.0081	1.6589	0.5946	1.9096	1.5628	0.6334	1.8122	1.4614	0.6698
S20	S ² ESN	\mathcal{M}_2	J	\mathcal{M}_2	1.9137	1.5674	0.6457	1.7357	1.4481	0.7085	1.5145	1.2395	0.7781
S21	S ² ESN	—	J	\mathcal{M}_1	2.4216	1.9705	0.4104	2.2976	1.9118	0.4693	2.0928	1.7648	0.5597
S22	CS ² ESN	\mathcal{M}_3 and I	K	\mathcal{M}_3	6.5030	5.2751	0.5909	6.1185	4.3824	0.6378	5.4016	4.0023	0.7177
S23	CS ² ESN	—	K	\mathcal{M}_2	1.5659	1.2576	0.7627	1.4578	1.2021	0.7944	1.6364	1.3287	0.7409
S24	CS ² ESN	—	K	\mathcal{M}_1	1.8380	1.5335	0.6604	1.7598	1.4594	0.6886	1.9981	1.6306	0.5986
S25	S ² ESN	\mathcal{M}_3	L	\mathcal{M}_3	4.5768	3.6174	0.7973	4.5325	3.4717	0.8013	3.9767	3.0826	0.8470
S26	S ² ESN	—	L	\mathcal{M}_2	2.0700	1.6879	0.5854	2.0702	1.6957	0.5853	2.1325	1.7450	0.5600
S27	S ² ESN	—	L	\mathcal{M}_1	2.9188	2.4069	0.1435	2.8113	2.3091	0.2054	2.8375	2.3056	0.1905
S28	ESN	\mathcal{M}_1	M	\mathcal{M}_1	2.7674	2.2394	0.2300	2.0097	1.7609	0.5567	1.8095	1.4943	0.6708
S29	CESN	\mathcal{M}_2 and M	N	\mathcal{M}_2	2.4011	1.8759	0.4422	1.8851	1.5427	0.6561	1.5989	1.3141	0.7526
S30	CESN	—	N	\mathcal{M}_1	2.7686	2.2453	0.2293	2.1553	1.8123	0.5330	2.0246	1.6445	0.5879
S31	ESN	\mathcal{M}_2	O	\mathcal{M}_2	2.5551	2.2378	0.3683	2.1046	1.7567	0.5714	1.8900	1.5463	0.6544
S32	ESN	—	O	\mathcal{M}_1	2.9775	2.4062	0.1087	2.7540	2.2745	0.2375	2.9218	2.4165	0.1417
S33	CESN	\mathcal{M}_3 & N	P	\mathcal{M}_3	9.1038	7.1538	0.1982	7.0592	5.1668	0.5179	6.7314	4.4988	0.5616
S34	CESN	—	P	\mathcal{M}_2	1.9493	1.5208	0.6324	1.7373	1.4141	0.7080	1.5817	1.3004	0.7579
S35	CESN	—	P	\mathcal{M}_1	2.2940	1.9139	0.4709	2.0799	1.7450	0.5651	2.0772	1.7345	0.5662
S36	ESN	\mathcal{M}_3	Q	\mathcal{M}_3	9.2435	7.6942	0.1734	7.2773	5.5926	0.4876	7.1393	4.8967	0.5069
S37	ESN	—	Q	\mathcal{M}_2	2.5280	1.9887	0.3816	2.3364	1.8892	0.4718	2.2667	1.8613	0.5028
S38	ESN	—	Q	\mathcal{M}_1	4.1800	3.5098	-0.7567	2.8715	2.3331	0.1710	2.8840	2.3536	0.1637

* LR: The label ratio of training datasets.

TABLE VI
COMPREHENSIVE PERFORMANCE COMPARISON OF CS²GESN, MS²GESN, CS²ESN, AND CESN FOR TPF PROCESS

LR	RMSE ($\times 10^{-4}$) / MAE ($\times 10^{-4}$) / R ² / MAPE (%)											
	CS ² GESN			MS ² GESN			CS ² ESN			CESN		
5%	3.8459	/ 2.5640	/ 0.6262	/ 0.2569	3.9215	/ 2.4159	/ 0.6114	/ 0.2424	3.8907	/ 2.6211	/ 0.6175	/ 0.2627
10%	3.3816	/ 2.2263	/ 0.7110	/ 0.2232	4.4445	/ 2.6074	/ 0.5008	/ 0.2612	3.6658	/ 2.2994	/ 0.6604	/ 0.2304
20%	3.1574	/ 2.2118	/ 0.7481	/ 0.2218	4.1304	/ 2.3826	/ 0.5689	/ 0.2389	3.3820	/ 2.2906	/ 0.7110	/ 0.2296
									4.0633	/ 2.4849	/ 0.5828	/ 0.2490

* The bold and underline value denotes the best performance, LR: The label ratio of training datasets.

- from unlabeled datasets can help mitigate catastrophic forgetting.
- 3) It is clear from 11 scenarios (S28–S38), that CESN and ESN exhibit poorer performance with higher RMSE and MAE values and lower R² values in most scenarios. For example, in S32 with 5% LR, ESN has an RMSE of 2.9775, an MAE of 2.4062, and an R² of 0.1087. Although CESN alleviates the catastrophic forgetting of ESN, their modeling performance is not ideal owing to the issue of sparse labels.

- 4) As observed in five scenarios (S12–S16), it is evident that MS²GESN struggles the most, exhibiting high errors and low overall performance. The primary reason is that MS²GESN cannot adequately describe the complex data properties of multiple modes. In addition, it is not suitable for practical situations due to a data storage burden.
- 5) More importantly, in three scenarios (S11, S27, and S38), S²GESN, S²ESN, and ESN do not perform very well against the historical mode \mathcal{M}_1 . The primary reason is the significant difference between mode \mathcal{M}_3 and

historical mode \mathcal{M}_1 . This further demonstrates that while preserving important knowledge from the historical mode \mathcal{M}_1 , the performance of CS²GESN in S6, CS²ESN in S22, and CESN in S33 is not ideal in the new mode \mathcal{M}_3 . However, CS²GESN still exhibits the best among them.

In the Supplementary Material, Fig. A3 presents the predicted values and corresponding absolute errors of the proposed CS²GESN compared to other approaches, including MS²GESN, CS²ESN, and CESN, with 5% labeled training samples. Subsequently, these four models are utilized to predict all modes under three LR simultaneously, and their comprehensive prediction performance are listed in Table VI. It is clear that the proposed CS²GESN exhibits superior performance for multimode processes with sparse labels.

V. CONCLUSION

In this article, we have developed the CS²GESN for online prediction of product quality in multimode processes, where each mode arrives successively with sparse labels. Unlike conventional modeling approaches, the proposed CS²GESN has CL capabilities that alleviate the issue of catastrophic forgetting when updating mode parameters for a new mode, aligning well with practical demands. Meanwhile, a SSL technique is combined to extract the valuable information of unlabeled samples for each single mode with sparse labels, which not only improves modeling performance within each single mode but also strengthens the retention of key parameters across all modes, further alleviating catastrophic forgetting. The comprehensive experimental results of the TE and TPFF processes have verified the effectiveness of the proposed CS²GESN model, achieving a more flexible solution for multimode processes.

In the future, we will make improvements in the following two aspects: 1) EWC-based CL methods still have a learning bottleneck despite their low computational complexity, and it is crucial to investigate more advanced CL strategies; 2) we will explore effective combination of SSL with CL to enable mutual promotion in noisy environments.

REFERENCES

- [1] X. Zhang, C. Song, J. Zhao, and D. Xia, "Gaussian mixture continuously adaptive regression for multimode processes soft sensing under time-varying virtual drift," *J. Process Control*, vol. 124, pp. 1–13, 2023.
- [2] L. Yao and Z. Ge, "Distributed parallel deep learning of hierarchical extreme learning machine for multimode quality prediction with big process data," *Eng. Appl. Artif. Intell.*, vol. 81, pp. 450–465, 2019.
- [3] F. A. Souza, R. Araújo, and J. Mendes, "Review of soft sensor methods for regression applications," *Chemometrics Intell. Lab. Syst.*, vol. 152, pp. 69–79, 2016.
- [4] Q. Sun and Z. Ge, "A survey on deep learning for data-driven soft sensors," *IEEE Trans. Ind. Informat.*, vol. 17, no. 9, pp. 5853–5866, Sep. 2021.
- [5] S. J. Qin, Y. Dong, Q. Zhu, J. Wang, and Q. Liu, "Bridging systems theory and data science: A unifying review of dynamic latent variable analytics and process monitoring," *Annu. Rev. Control*, vol. 50, pp. 29–48, 2020.
- [6] X. Yuan, L. Li, and Y. Wang, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3168–3176, May 2020.
- [7] Y. Tian, Y. Xu, Q. Zhu, and Y. He, "Novel stacked input-enhanced supervised autoencoder integrated with gated recurrent unit for soft sensing," *IEEE Trans. Instrum. Meas.*, vol. 71, 2022, Art. no. 2515009.
- [8] Y. He, L. Chen, Y. Xu, Q. Zhu, and S. Lu, "A new distributed echo state network integrated with an auto-encoder for dynamic soft sensing," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023, Art. no. 2500308.
- [9] Y. Bo, P. Wang, X. Zhang, and B. Liu, "Modeling data-driven sensor with a novel deep echo state network," *Chemometrics Intell. Lab. Syst.*, vol. 206, 2020, Art. no. 104062.
- [10] L. Patané and M. G. Xibilia, "Echo-state networks for soft sensor design in an SRU process," *Inf. Sci.*, vol. 566, pp. 195–214, 2021.
- [11] Y. Yang, X. Zhao, and X. Liu, "A novel echo state network and its application in temperature prediction of exhaust gas from hot blast stove," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9465–9476, Dec. 2020.
- [12] C. Yang, Q. Liu, Y. Liu, and Y. M. Cheung, "Transfer dynamic latent variable modeling for quality prediction of multimode processes," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 5, pp. 6061–6074, May 2024.
- [13] B. Shen, L. Yao, Z. Yang, and Z. Ge, "Mode information separated β -VAE regression for multimode industrial process soft sensing," *IEEE Sensors J.*, vol. 23, no. 9, pp. 10231–10240, May 2023.
- [14] L. Yao, B. Shen, L. Cui, J. Zheng, and Z. Ge, "Semi-supervised deep dynamic probabilistic latent variable model for multimode process soft sensor application," *IEEE Trans. Ind. Informat.*, vol. 19, no. 4, pp. 6056–6068, Apr. 2023.
- [15] W. Shao, Z. Ge, Z. Song, and J. Wang, "Semisupervised robust modeling of multimode industrial processes for quality variable prediction based on student's t mixture model," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 2965–2976, May 2020.
- [16] F. Guo, B. Wei, and B. Huang, "A just-in-time modeling approach for multimode soft sensor based on Gaussian mixture variational autoencoder," *Comput. Chem. Eng.*, vol. 146, 2021, Art. no. 107230.
- [17] H. Kaneko and K. Funatsu, "Moving window and just-in-time soft sensor model based on time differences considering a small number of measurements," *Ind. Eng. Chem. Res.*, vol. 54, no. 2, pp. 700–704, 2015.
- [18] B. Song, S. Tan, and H. Shi, "Key principal components with recursive local outlier factor for multimode chemical process monitoring," *J. Process Control*, vol. 47, pp. 136–149, 2016.
- [19] Y. Liu, T. Chen, and J. Chen, "Auto-switch Gaussian process regression-based probabilistic soft sensors for industrial multigrade processes with transitions," *Ind. Eng. Chem. Res.*, vol. 54, no. 18, pp. 5037–5047, 2015.
- [20] J. Zhang, D. Zhou, and M. Chen, "Monitoring multimode processes: A modified PCA algorithm with continual learning ability," *J. Process Control*, vol. 103, pp. 76–86, 2021.
- [21] W. Shao, Z. Ge, and Z. Song, "Soft-sensor development for processes with multiple operating modes based on semisupervised Gaussian mixture regression," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 5, pp. 2169–2181, Sep. 2019.
- [22] Y. S. Lee and J. Chen, "A robust semi-supervised learning scheme for development of within-batch quality prediction soft-sensors," *Eng. Appl. Artif. Intell.*, vol. 133, 2024, Art. no. 107920.
- [23] C. J. Lowrance and A. P. Lauf, "An active and incremental learning framework for the online prediction of link quality in robot networks," *Eng. Appl. Artif. Intell.*, vol. 77, pp. 197–211, 2019.
- [24] Y. Dai, C. Yang, K. Liu, A. Liu, and Y. Liu, "TimeDDPM: Time series augmentation strategy for industrial soft sensing," *IEEE Sensors J.*, vol. 24, no. 2, pp. 2145–2153, Jan. 2023.
- [25] X. Jiang and Z. Ge, "Improving the performance of just-in-time learning-based soft sensor through data augmentation," *IEEE Trans. Ind. Electron.*, vol. 69, no. 12, pp. 13716–13726, Dec. 2022.
- [26] X. Li, Q. Zhu, and Y. He, "Data mode-related generative adversarial network for industrial soft sensor application," *IEEE Trans. Ind. Informat.*, vol. 20, no. 3, pp. 4198–4205, Mar. 2023.
- [27] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, 2020.
- [28] C. Liu, S. Li, H. Chen, X. Xiu, and C. Peng, "Semi-supervised joint adaptation transfer network with conditional adversarial learning for rotary machine fault diagnosis," *Intell. Robot.*, vol. 3, no. 2, pp. 131–143, 2023.
- [29] W. Dai, X. Li, and K. Cheng, "Semi-supervised deep regression with uncertainty consistency and variational model ensembling via bayesian neural networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 6 2023, pp. 7304–7313.
- [30] S. Dohare, J. F. Hernandez-Garcia, Q. Lan, P. Rahman, A. R. Mahmood, and R. S. Sutton, "Loss of plasticity in deep continual learning," *Nature*, vol. 632, no. 8026, pp. 768–774, 2024.
- [31] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Netw.*, vol. 113, pp. 54–71, 2019.

- [32] G. M. Van de Ven, H. T. Siegelmann, and A. S. Tolias, "Brain-inspired replay for continual learning with artificial neural networks," *Nature Commun.*, vol. 11, no. 1, Dec. 2020, Art. no. 4069.
- [33] N. Y. Masse, G. D. Grant, and D. J. Freedman, "Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization," *Proc. Nat. Acad. Sci.*, vol. 115, no. 44, pp. 10467–10475, 2018.
- [34] R. Hadsell, D. Rao, A. A. Rusu, and R. Pascanu, "Embracing change: Continual learning in deep neural networks," *Trends Cogn. Sci.*, vol. 24, no. 12, pp. 1028–1040, 2020.
- [35] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci.*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [36] M. De Lange et al., "A continual learning survey: Defying forgetting in classification tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3366–3385, Jul. 2021.
- [37] F. Huszár, "On quadratic penalties in elastic weight consolidation," 2017, *arXiv:1712.03847*.
- [38] J. Zhang, D. Zhou, M. Chen, and X. Hong, "Continual learning for multimode dynamic process monitoring with applications to an ultra-supercritical thermal power plant," *IEEE Trans. Automat. Sci. Eng.*, vol. 20, no. 1, pp. 137–150, Jan. 2022.
- [39] P. Zhang et al., "Continual learning on dynamic graphs via parameter isolation," in *Proc. 46th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2023, pp. 601–611.
- [40] Z. Mai, R. Li, H. Kim, and S. Sanner, "Supervised contrastive replay: Revisiting the nearest class mean classifier in online class-incremental continual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3589–3599.
- [41] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.
- [42] N. L. Ricker and J. Lee, "Nonlinear model predictive control of the tennessee eastman challenge process," *Comput. Chem. Eng.*, vol. 19, no. 9, pp. 961–981, 1995.
- [43] C. Ruiz-Cárcel, Y. Cao, D. Mba, L. Lao, and R. Samuel, "Statistical process monitoring of a multiphase flow facility," *Control Eng. Pract.*, vol. 42, pp. 74–88, 2015.



Chao Yang (Graduate Student Member, IEEE) received the M.Eng. degree in power engineering and engineering thermophysics from the Zhejiang University of Technology, Hangzhou, China, in 2019. He is currently working toward the Ph.D. degree in control science and engineering with the State Key Laboratory of Synthetical Automation for Process Industry, Northeastern University, Shenyang, China.

Since March 2024, he has been visiting the University of Surrey, Guildford, U.K., sponsored by the China Scholarship Council to exchange. His current research interests include deep learning, transfer learning, process monitoring, fault diagnosis, and process data analytics.



Qiang Liu (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in control theory and engineering from Northeastern University, Shenyang, China in the years of 2003, 2006, and 2012, respectively.

He is currently a Full Professor with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University. He has authored or coauthored more than 70 peer-reviewed papers. His research interests include big data analytics, machine learning, statistical process monitoring and fault diagnosis of complex industrial processes.

Prof. Liu was the recipient of the Second Prize of Liaoning Provincial National Natural Science Award, the excellent doctor degree dissertation award from the Liaoning Province of China, and the Outstanding Young Scholar of Liaoning Revitalization Talents Program, China. His two papers were selected as one of the F5000-Top academic papers in Chinese top-quality SCI tech Journals in the years of 2019 and 2022, respectively.



Yi Liu (Member, IEEE) received the Ph.D. degree in control theory and engineering from Zhejiang University, Hangzhou, China, in 2009.

From 2011 to 2020, he was an Associate Professor with the Institute of Process Equipment and Control Engineering, Zhejiang University of Technology. From February 2012 to June 2013, he was a Postdoctoral Researcher with the Department of Chemical Engineering, Chung-Yuan Christian University, Taoyuan City, Taiwan. Since December 2020, he has been a

Full Professor with Zhejiang University of Technology. He has authored or coauthored more than 50 research papers at IEEE Transactions and international journals. His research interests include data intelligence with applications to modeling, control, and optimization of industrial processes.



Yiu-Ming Cheung (Fellow, IEEE) received the Ph.D. degree in the field of artificial intelligence from the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, in 2000.

He is currently a Chair Professor with the Department of Computer Science, Hong Kong Baptist University. His research interests include machine learning and visual computing, as well as their applications in data science, pattern recognition, and optimization.

Dr. Cheung is a Member of European Academy of Sciences and Arts, and a Fellow of AAAS, IET, and BCS. He is the Editor-in-Chief (2023–) of IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE. Also, he is an Associate Editor for several prestigious journals, including IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (2014–2020), *Pattern Recognition*, to name a few.