





STAGED: A Spatial-Temporal Aware Graph Encoder–Decoder for Fault Diagnosis in Industrial Processes

Shizhong Li , *Student Member, IEEE*, Wenchao Meng , *Senior Member, IEEE*,
Shibo He , *Senior Member, IEEE*, Jichao Bi , *Member, IEEE*,
and Guanglun Liu , *Student Member, IEEE*

I. INTRODUCTION

Abstract—Data-driven fault diagnosis for critical industrial processes has exhibited promising potential with massive operating data from the supervisory control and data acquisition system. However, automatically extracting the complicated interactions between measurements and subtly integrating them with temporal evolutions have not been fully considered. Besides, with the increasing complexity of industrial processes, accurately locating fault roots is of tremendous significance. In this article, we propose an unsupervised spatial-temporal aware graph encoder–decoder (STAGED) model for industrial fault diagnosis. First, the high-dimensional measurements are constructed as a weighted graph to depict the complicated interactions. Then, the graph convolutional network, long short-term memory network and attention mechanism are applied to learn a comprehensive representation for multiserries. To enforce the model to better capture the temporal evolution, the dual decoder that performs reconstruction and prediction tasks simultaneously is adopted with a well-designed comprehensive loss function. By learning the spatial-temporal evolutions of datasets, faults can be diagnosed and located at a fine-grained level based on reconstruction deviations. To verify the performance of STAGED, experiments on the Cranfield three-phase flow facility and secure water treatment datasets are implemented and the results indicate that it can provide insight into fault evolution and accurately diagnose faults.

Index Terms—Complex industrial processes, encoder–decoder, fault diagnosis, graph convolutional networks, time series, unsupervised learning.

WITH the rapid development of digitalization and network communication, a mushrooming number of critical infrastructures in complex industrial processes are interconnected as cyber-physical systems (CPSs) to allocate resources and dispatch tasks systematically and efficiently [1]. Usually, CPSs are equipped with the supervisory control and data acquisition (SCADA) systems, which could gather monitoring signals from multiple critical components. Since the high-dimensional measurements from different sensors are coupled in a nonlinear and complex manner, manual inspection is becoming increasingly time-consuming or even infeasible [2]. Therefore, intelligent fault diagnosis has received wide attention from both industry and academia.

SCADA signals are mostly high-dimensional time series, which lack semantic information and accumulate rapidly over time. Besides, the inherent scarcity of labeled historical fault data and complex coupling among sensors make fault diagnosis even more challenging [3]. Generally, fault diagnosis methods can be classified into model-based and data-driven approaches [4]. As the model-based methods are established on precise mechanism models that are unavailable for most complex industrial processes in the real-world, their applications are greatly limited [2]. To take advantages of the enormous SCADA data and integrate with information platforms, the data-driven methods have become the mainstream. Typically, the data-driven machine learning approaches attempt to learn the normal patterns or construct statistical models to depict the data distribution, and then the faults can be detected through deviations from normal conditions [5]. For example, principal components analysis (PCA) [6] can learn a model to encode the normal data to low-dimensional representations, and the reconstruction error can be deemed as a fault score. Kernel density estimation [7] is widely used for nonparametric density estimation, which would label the data with low probability as a fault point. However, these traditional machine learning methods heavily rely on domain knowledge and feature engineering [8], which hinders their applications on large-scale industrial scenarios.

Inspired by the success of deep learning in image recognition and natural language process, an increasing number of industrial fault diagnosis tasks have begun to embrace the paradigm of

Manuscript received 1 March 2023; accepted 17 May 2023. Date of publication 29 May 2023; date of current version 19 January 2024. This work was supported in part by the National Key R&D Program of China under Grant 2022ZD0118702 and in part by the National Natural Science Foundation of China under Grant U1909207 and Grant U21B2029. Paper no. TII-23-0699. (*Corresponding author: Wenchao Meng.*)

Shizhong Li, Wenchao Meng, Shibo He, and Guanglun Liu are with the State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, and the Key Laboratory of CS&AUS of Zhejiang Province, Zhejiang University, Hangzhou 310027, China (e-mail: lysz@zju.edu.cn; wmengzju@zju.edu.cn; s18he@zju.edu.cn; silwaywliu@gmail.com).

Jichao Bi is with the Zhejiang Institute of Industry and Information Technology, Hangzhou 310000, China (e-mail: jonny.bijichao@zju.edu.cn).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TII.2023.3281083>.

Digital Object Identifier 10.1109/TII.2023.3281083

TABLE I
COMPARISON OF FAULT DETECTION METHODS BASED ON DEEP LEARNING

Reference	Unsupervised	Interaction graph	Temporal modeling	Prediction	Reconstruction	Fault localization
MSCRED[3]	✓	✗	✓	✗	✓	✓
IAGNN[4]	✗	✓	✗	✗	✗	✗
CCAE[9]	✓	✗	✗	✗	✓	✗
GCN-SA[10]	✗	✓	✗	✗	✗	✗
GID[11]	✗	✗	✗	✗	✗	✗
TopoMAD[12]	✓	✓	✓	✓	✗	✗
MTAD-GAT[13]	✓	✓	✓	✓	✓	✗
GTA[14]	✓	✓	✓	✓	✗	✗
STAGED	✓	✓	✓	✓	✓	✓

deep learning, which can cope with the complexity of industrial processes and take full use of the SCADA data [2]. In [9], a novel conditional convolutional auto-encoder (AE) that incorporated time stamps of data as input condition in reconstruction was devised to monitor wind turbine blade breakages. Zhang et al. [3] proposed a multiscale convolutional recurrent encoder-decoder (MSCRED) to diagnose anomalies in multivariate time series, which combined the long short-term memory (LSTM) network with the convolution operation to reconstruct the signature matrix.

However, typical deep learning models take the Euclidean data with grid structure as input and could hardly model the complicated coupling among sensors in complex industrial processes, which makes the feature extraction inefficient and hinders the fault diagnosis performance [4]. Considering that the graph is the dedicated model for analyzing relations between entities, the graph neural network (GNN) that has shown great success in learning non-Euclidean data [15] has been naturally introduced into industrial fault diagnosis. For example, Chen et al. [4] proposed an interaction-aware GNN to generate representative embeddings for the downstream tasks, which has been proved to be accurate for fault classification of complex industrial processes. In [10], graph learning was combined with structural analysis, which leveraged the advantages of prior knowledge and deep learning to improve the accuracy of fault diagnosis for power electronic equipments. In [11], a general framework based on GNN was proposed, which learned the graph structure by multiheaded-like similarity and network reconstruction technology to obtain representative embeddings for fault classification. Besides, plenty of works have further integrated such interactions with temporal patterns to achieve efficient spatial-temporal relationship perception. He et al. [12] utilized their prior knowledge to construct graphs for system metrics and replaced the gates in LSTM with graph convolution operations to capture the spatial-temporal dynamics. In [13], feature-oriented and time-oriented graph attention layers are devised to depict the multivariate correlations and temporal dependencies, which fuse more information from different aspects. In [14], a directed graph structure learning policy can automatically discover the hidden associations and the transformer-based architecture is adopted to capture long-distance context information.

Although lots of achievements have been made in GNN-based industrial fault diagnosis, there still exist some limitations. First, most of the existing research works have formulated an industrial fault diagnosis as a supervised graph (or node) classification problem, which may hinder their applications in practical industrial systems without fault labels. In fact, fault data is extremely rare in the real world and fault cases are hardly to be completely enumerated due to the complexity of modern industrial processes. Moreover, the high-dimensional time series from SCADA system are naturally coupled both spatially and temporally. Compared with traditional methods, GNN has made great achievements in spatial relationship modeling. However, the temporal evolution of time series has not been fully considered. Lastly, root cause localization is of tremendous significance for industrial fault diagnosis, because it can greatly reduce the workload of operators and improve maintenance efficiency [3]. But this has rarely been achieved by most researchers because it requires that the learned representations could delicately capture the couplings.

In consideration of the above-mentioned limitations, an unsupervised spatial-temporal aware graph encoder-decoder (STAGED) model is proposed in this article. Generally, STAGED is in the form of encoder-decoder, and it could learn the patterns from normal data and be trained in an unsupervised manner. First, a self-adaptive adjacent matrix could be learned for sensors, and in this way, the graph convolutional network (GCN) could be applied to further model the spatial coupling. In addition, the temporal evolution is also considered in STAGED, where node embeddings are updated and aggregated along the time axis. By learning the normal patterns of training data, STAGED can diagnose faults based on the reconstruction errors. The proposed model is tested on two real-world industrial processes, i.e., Cranfield three-phase flow facility and secure water treatment system. The results show that our model can improve the $F1$ -score by at least 3.55% compared with the classic machine learning algorithms and the state-of-the-art deep learning methods. Besides, case studies also verify the fault localization ability of our model. The comparison between our model and existing deep learning models is shown in Table I. The main contributions can be summarized as follows.

- 1) This article proposes an unsupervised framework for industrial fault diagnosis, which could capture the complex

spatial-temporal interactions among sensors with neural networks and tackle the lack of labeled fault samples.

- 2) The proposed STAGED model could dynamically formulate high-dimensional time series as adaptive graphs without manually setting weight threshold. Then, spatial-temporal information is subtly integrated by STAGED to form comprehensive graph embeddings, which makes the reconstruction-based fault diagnosis accurate.
- 3) The proposed framework could achieve node-wise fault diagnosis based on the efficient representations, which means it could provide auxiliary judgment information for fault localization. This is extremely important for industrial systems with large amount of measurements in real-world application.

The rest of this article is organized as follows. Section II introduces the basis of GCN and formulates the fault diagnosis problem. Section III elaborates the STAGED model. The experimental results are given in Section IV. Finally, Section V concludes this article.

II. PRELIMINARIES

A. Graph Convolutional Network

A graph is often denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where \mathcal{V} is the set of vertices and \mathcal{E} is the set of edges. The adjacent matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ can be defined as follows:

$$\mathbf{A}_{i,j} = \begin{cases} 1, & \text{if } \langle v_i, v_j \rangle \in \mathcal{E} \text{ and } i \neq j \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Obviously, for the undirected graph, \mathbf{A} is symmetric, and thus the graph can be represented by the Laplacian matrix as

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \quad (2)$$

where \mathbf{D} is the diagonal degree matrix and $D_{i,i} = \sum_j (\mathbf{A}_{i,j})$. Usually, the Laplacian matrix can be normalized as

$$\mathbf{L}^{sym} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}}. \quad (3)$$

Thus, the spectral GCN [16] can be defined in the Fourier domain with a filter $g_\theta = \text{diag}(\theta)$ parameterized by $\theta \in \mathbb{R}^N$ as follows:

$$g_\theta \star x = \mathbf{U} g_\theta(\Lambda) \mathbf{U}^T x \quad (4)$$

where \mathbf{U} and Λ are the eigenvectors matrix and the corresponding diagonal eigenvalue matrix of the normalized graph Laplacian with $\mathbf{U} \Lambda \mathbf{U}^T = \mathbf{L}^{sym}$.

To relieve the potential risk of intense computation and nonspatially localized filtering, Defferrard et al. [17] used the truncated expansion of $g_\theta(\Lambda)$ in terms of Chebyshev polynomials $T_k(\xi)$ up to K th order, and thus, got the ChebNet as follows:

$$g_\theta \star x \approx \sum_{k=0}^K \theta_k T_k(\tilde{\mathbf{L}}) x \quad (5)$$

where $\tilde{\mathbf{L}} = \frac{2}{\lambda_{\max}} \mathbf{L} - \mathbf{I}_N$, λ_{\max} is the largest eigenvalues of \mathbf{L} and $\theta \in \mathbb{R}^K$ is the coefficient. The Chebyshev polynomials are defined in recursive form as $T_k(\xi) = 2\xi T_{k-1}(\xi) - T_{k-2}(\xi)$, with $T_0(\xi) = 1$ and $T_1(\xi) = \xi$.

Kipf and Welling [18] further limited $K = 1$ to alleviate the problem of overfitting and approximates $\lambda_{\max} \approx 2$. To ensure numerical stability, a renormalization trick: $\mathbf{I}_N + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \rightarrow \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}}$ with $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ and $\tilde{D}_{i,i} = \sum_j \tilde{A}_{i,j}$ is introduced. Therefore, GCN for a signal $\mathbf{X} \in \mathbb{R}^{N \times C}$ with C channels is generalized as

$$\mathbf{Z} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \Theta \quad (6)$$

where $\Theta \in \mathbb{R}^{C \times F}$ is a matrix of filter parameters and $\mathbf{Z} \in \mathbb{R}^{N \times F}$ is the convoluted result. As a simplified localized spectral method, GCN can also be deemed as a spatial method because it only gathers information from the one-hop neighbors for updating.

B. Problem Statement

In industrial processes, SCADA systems could collect readings from multiple sensors deployed on critical infrastructures. The raw measurements from N sensors with the length of T can be denoted as $\mathbf{S} = [\mathbf{s}^{(0)}, \dots, \mathbf{s}^{(T)}] \in \mathbb{R}^{N \times T}$, where $\mathbf{s}^{(\tau)} \in \mathbb{R}^N$ is the measurements of N sensors at time tick τ . A sliding window of length w with stride of s could slide along the time axis to divide the original series into subsequences, and the t th segment can be denoted as $\omega_t = [\mathbf{s}^{(T)}, \dots, \mathbf{s}^{(T+w-1)}] \in \mathbb{R}^{N \times w}$. The width and stride of the sliding windows can be determined according to the stationarity of the time series and the accuracy requirements.

Notice that sensors in industrial processes interact with each other in complex ways. In order to depict such interactions, a graph denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is defined for each sliding window, where \mathcal{V} corresponds to N sensors and \mathcal{E} depicts the interactions. The raw measurement of each node in the corresponding sliding window can be regarded as its attributes, which means for the t th window, the node attribute $\mathbf{X}_t = \omega_t$. An edge $e = \langle v_i, v_j \rangle$ denotes the coupling score of node i and node j .

When the t th window comes, a sparse weight matrix \mathbf{W}_t to capture the complex coupling can be constructed by

$$\mathbf{W}_t = \mathcal{F}(\mathbf{X}_t) \quad (7)$$

where $\mathcal{F}(\cdot)$ is the structure learning function and will be introduced in Section III-A. Then, an efficient representation \mathbf{H}_t could be learned based on the past l sliding windows as

$$\mathbf{H}_t = \mathcal{H}((\mathbf{X}_t, \mathbf{W}_t), \dots, (\mathbf{X}_{t-l+1}, \mathbf{W}_{t-l+1})) \quad (8)$$

where $\mathcal{H}(\cdot)$ is the representation learning function elaborated in Sections III-B and III-C. Using \mathbf{H}_t , the measurements in the t th window can be reconstructed as $\tilde{\mathbf{X}}_t$. Thus, by analyzing the reconstruction deviations, fault diagnosis can be achieved by the anomaly score $\mathbf{A}(t)$ as follows:

$$\mathbf{A}(t) = \mathcal{O}(\mathcal{R}(\mathbf{H}_t, \mathbf{X}_t)) = \mathcal{O}(\tilde{\mathbf{X}}_t, \mathbf{X}_t) \quad (9)$$

and the details of reconstruction function $\mathcal{R}(\cdot)$ and diagnosis function $\mathcal{O}(\cdot)$ will be described in Sections III-D and III-E.

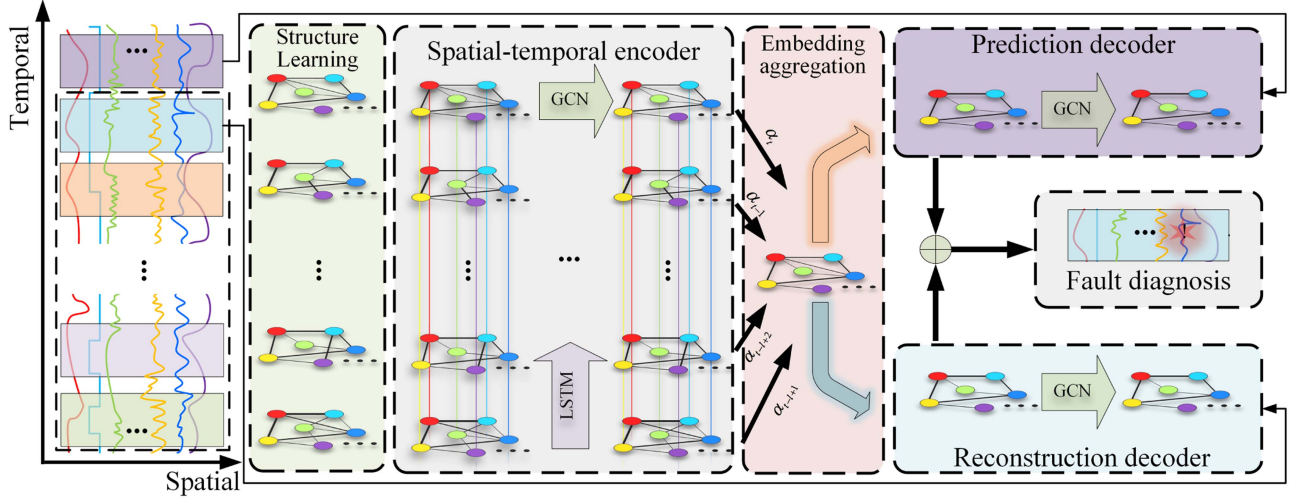


Fig. 1. Framework of proposed fault diagnosis scheme.

III. METHODOLOGY

This section proposes the STAGED model for unsupervised multiseries fault diagnosis. The overall framework of STAGED is illustrated in Fig. 1, which can be divided into graph structure learning layer, spatial-temporal encoder, embedding aggregation and dual decoder. The details of each module would be introduced.

A. Graph Structure Learning Layer

The multisensors in an industrial process are physically connected and intricately coupled. An attention-based graph structure learning layer is introduced to learn a weighted directed graph for each sliding window. For the measurements $\mathbf{X}_t \in \mathbb{R}^{N \times w}$ in the t th window, a fully connected (FC) layer is used to embed the readings into $\mathbf{V} \in \mathbb{R}^{N \times d}$, and thus, the attention-based edges with weights could be learned as

$$\begin{cases} \mathbf{V} = \sigma(\mathbf{X}_t \mathbf{W}_s^T + \mathbf{b}_s) \\ a_{i,j} = \mathbf{v}_i^T \mathbf{I} \mathbf{v}_j \end{cases} \quad (10)$$

where $\sigma(\cdot)$ is the activation function (i.e., $\text{Sigmoid}(\cdot)$), $\mathbf{I} \in \mathbb{R}^{d \times d}$ is the interaction attention matrix, and \mathbf{v}_i and \mathbf{v}_j are the corresponding node embeddings. The learned $a_{i,j}$ is the element at i th row and j th column of adjacent relation matrix \mathbf{A}_t , and denotes the relation score between node i and node j . In order to alleviate the computation complexity in the downstream fault diagnosis and highlight on the critical interactions, the sparsemax function is introduced to normalize the adjacent relation score for each sensor [19]. For the i th row $\mathbf{a}_i \in \mathbb{R}^N$ in \mathbf{A}_t , we first sort it in descending order as $a_{i,1} \geq \dots \geq a_{i,N}$. Then an integer index k for \mathbf{a}_i can be defined as follows:

$$k(\mathbf{a}_i) = \max \left\{ k \in [1, N] \mid 1 + k a_{i,k} > \sum_{j \leq k} a_{i,j} \right\}. \quad (11)$$

Based on $k(\mathbf{a}_i)$, an unique function $\tau(\mathbf{a}_i)$ can be expressed as

$$\tau(\mathbf{a}_i) = \frac{(\sum_{j \leq k(\mathbf{a}_i)} a_{i,j}) - 1}{k(\mathbf{a}_i)}. \quad (12)$$

Thus, the normalized weight can be calculated as $w_{i,j} := \max(a_{i,j} - \tau(\mathbf{a}_i), 0)$, where $w_{i,j}$ is the item in the i th row and j th column of the sparse normalized matrix \mathbf{W}_t .

Remark 1: The sparsemax function outputs sparse probabilities of a multinomial distribution, thus, filtering out noise from the mass of the distribution and the normalized relation score in \mathbf{W}_t is deemed as the edge weight in the following STAGEDs. The structure learning layer does not need to set threshold for similarity to obtain the adjacent matrix, which makes the model can be trained in an end-to-end approach and improves the consistency. For the graph structure learning layer, its computational and storage complexity are $O(Nw^2d + N^2d^2)$ and $O(dw + d^2)$, respectively.

B. Spatial-Temporal Encoder

The spatial-temporal encoder consists K spatial-temporal encoding layers and is shown in Fig. 2. The k th encoding layer takes the node embedding in the last layer (i.e., $\mathbf{H}^{(k-1)}$) as input, and uses GCN _{k} to gather the spatial information, which means that the node embedding of node i in the t th sliding window could be updated by

$$\tilde{\mathbf{h}}_{t,i}^{(k)} = \sigma \left(\sum_{j \in \mathcal{N}(i) \cup \{i\}} \frac{w_{i,j}}{\sqrt{\mathcal{D}(i)} \cdot \sqrt{\mathcal{D}(j)}} \cdot (\mathbf{W}_k^T \cdot \mathbf{h}_{t,j}^{(k-1)}) + \mathbf{b}_k \right) \quad (13)$$

where $\mathcal{N}(i)$ is the neighbors of node i , $\mathcal{D}(i)$ is the sum of the edge weights related to given node, $w_{i,j}$ is the learned edge weight, σ is the activation function (i.e., $\text{ReLU}(\cdot)$), \mathbf{b}_k is the bias and $\tilde{\mathbf{h}}_{t,i}^{(k)}$ is the updated hidden state. It should be noted that the first spatial-temporal encoding layer takes the raw measurements in sliding windows as input.

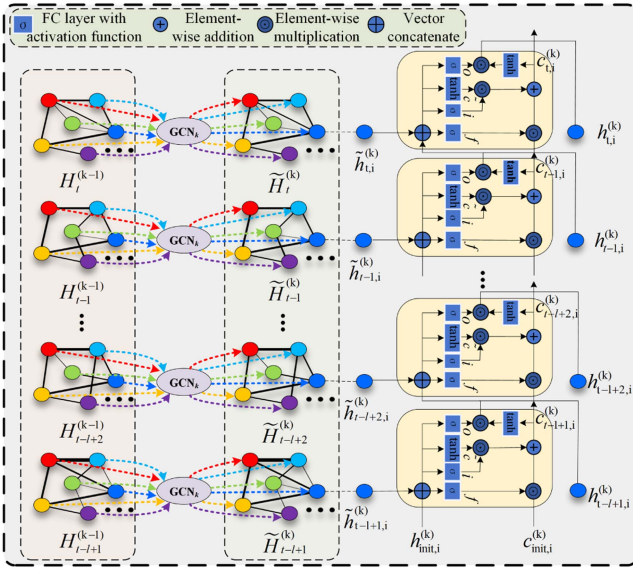


Fig. 2. Spatial-temporal encoding layer.

Then, the hidden states of each node would be sent to the $LSTM_k$ network as input, and thus, $\tilde{H}_{t-1}^{(k)}$ could be further updated. In Fig. 2, the update of node i with a blue color is depicted as an example and all the nodes would be updated in the same way. For the t th temporal window, the hidden state of node i is a sequence, which could contain the dynamic characteristics. Compared with vanilla recurrent neural network, LSTM introduces an internal memory state C to capture the long-range dependencies in sequences. LSTM update $h_{t,i}^{(k)}$ by

$$\begin{cases} c_{t,i}^{(k)} = f_{t,i}^{(k)} \odot c_{t-1,i}^{(k)} + i_{t,i}^{(k)} \odot \tilde{c}_{t,i}^{(k)} \\ h_{t,i}^{(k)} = o_{t,i}^{(k)} \odot c_{t,i}^{(k)} \end{cases} \quad (14)$$

where \odot is the element-wise production (i.e., Hadamard product), $c_{t-1,i}^{(k)}$ is the memory state in the last window, and $\tilde{c}_{t,i}^{(k)}$ is the candidate state calculated by

$$\tilde{c}_{t,i}^{(k)} = \tanh \left(W_c^{(k)} \tilde{h}_{t,i}^{(k)} + U_c^{(k)} h_{t-1,i}^{(k)} + b_c^{(k)} \right). \quad (15)$$

The three gates $f_{t,i}^{(k)}$, $i_{t,i}^{(k)}$, and $o_{t,i}^{(k)} \in [0, 1]^D$ are forget gate, input gate, and output gate, respectively. They are used to control the paths of information transmission, and can be derived by

$$\begin{cases} i_{t,i}^{(k)} = \sigma(W_i^{(k)} \tilde{h}_{t,i}^{(k)} + U_i^{(k)} h_{t-1,i}^{(k)} + b_i^{(k)}) \\ f_{t,i}^{(k)} = \sigma(W_f^{(k)} \tilde{h}_{t,i}^{(k)} + U_f^{(k)} h_{t-1,i}^{(k)} + b_f^{(k)}) \\ o_{t,i}^{(k)} = \sigma(W_o^{(k)} \tilde{h}_{t,i}^{(k)} + U_o^{(k)} h_{t-1,i}^{(k)} + b_o^{(k)}) \end{cases} \quad (16)$$

where σ is the activation function (i.e., logistic function).

Since spatial-temporal encoder is stacked by the encoding layers shown in Fig. 2, we only analyze the complexity for one layer. If the input and output dimension of node representations are F and F' , the computational and storage complexity can be expressed as $O(FF'|\mathcal{V}| + F'^2)$ and $O(FF' + F'^2)$, respectively, where $|\mathcal{V}|$ is the edge number.

In [12], [20], and [21], the spatial-temporal interactions are also extracted by LSTM and GCN to achieve efficient time series prediction or anomaly detection. However, in [12] and [21], the authors replaced the FC layers in LSTM with GCN, which means the internal states of LSTM are updated with the structural information of the last time step and GCN is a subcomponent of the modified LSTM. In [20], LSTM is used to further encode the latent representations learned by GCN, and it takes the whole representations as input and enforces the output to follow given distributions to construct a variational auto-encoder. Our spatial-temporal encoder uses LSTM to update node-wise representations learned by GCN, which could make the spatial-temporal learning much more efficient.

C. Embedding Aggregation

Considering that different historical windows are not equally correlated to the reconstruction and prediction tasks, an attention-based embedding aggregation strategy is adopted to gather the information from embeddings of the l temporal sliding windows after the spatial-temporal encoder. The goal is to select the steps that are relevant to current measurements to obtain a refined feature maps of shape $H_t \in R^{N \times E}$, which can be given by

$$\begin{cases} H_t = \sum_{i \in [t-l+1, t]} \alpha_i H_i^{(K)} \\ \alpha_i = \frac{\exp \left\{ Flat(H_t^{(K)})^T Flat(H_i^{(K)}) / \chi \right\}}{\sum_{i \in [t-l+1, t]} \exp \left\{ Flat(H_t^{(K)})^T Flat(H_i^{(K)}) / \chi \right\}} \end{cases} \quad (17)$$

where $Flat(\cdot)$ means flatten operation for tensors and χ is a scaling factor ($\chi=4$). That is, the embeddings of the l sliding windows are adaptively aggregated by the attention-based mechanism to form H_t , which summarizes the spatial-temporal information.

D. Prediction and Reconstruction Decoders

A dual decoder architecture which uses the aggregated embedding H_t to reconstruct X_t and predict X_{t+1} in the next sliding window at the same time is devised here to obtain more representative embeddings. Both the reconstruction and prediction decoders are multilayer GCNs with a FC layer at last. For the reconstruction decoder, the reconstruction error can be expressed as the Euclidean loss as follows:

$$L_{rec} = \frac{1}{N} \sum_{i=1}^N \|x_{t,i} - \tilde{x}_{t,i}\|_2^2 \quad (18)$$

where N is the number of sensors, $\tilde{x}_{t,i}$ is the reconstructed output for node i in the t th sliding window, and $x_{t,i}$ is the corresponding raw measurements.

The prediction decoder shares the same architecture with the reconstruction decoder, while it aims to predict the measurements in the next sliding window. Considering that the prediction result would gradually be unreliable with time going on, the prediction loss is formulated in a weight-decreasing form as

follows:

$$L_{pred} = \frac{1}{N} \sum_{i=1}^N \sum_{\gamma=1}^w \frac{(w-\gamma)}{\gamma^2} \|x_{t+1,i}^\gamma - \tilde{x}_{t+1,i}^\gamma\|_2^2 \quad (19)$$

where $x_{t+1,i}^\gamma$ is the measurement at γ th time tick in the $t+1$ sliding window, and $\tilde{x}_{t+1,i}^\gamma$ is the corresponding prediction. Finally, the optimization object of STAGED can be expressed as

$$\min_{\omega} \alpha L_{rec} + (1 - \alpha) L_{pred} + \lambda \|\omega\|_2^2 \quad (20)$$

where ω is the model parameter, α is the adjustable loss weight, and λ is the regularization parameter to control the complexity of the model.

Remark 2: The prediction task could guide the model to better capture the temporal characteristics and enforce the encoder to extract temporal dependency better. It should be noted that the prediction decoder is only used in training. When applying this model for fault diagnosis, only the reconstruction error will be considered because the measurements of the next sliding window are unavailable in on-line conditions.

E. Fault Diagnosis

To achieve fine-grained fault diagnosis, the fault score (i.e., the reconstruction error) for each sensor is calculated. Thus, the root causes of the faults can be localized, which is of great significance in industrial maintenance.

The initial fault score of the i th sensor in the t th sliding window is

$$e_{t,i} = \|x_{t,i} - \tilde{x}_{t,i}\|_2^2 \quad (21)$$

which is the mean square reconstruction error of its measurements. To eliminate the impact of noise and highlight on the deviation of faults, the fault score of sensor i smaller than $\beta \mathcal{E}_i$ will be set as zero, where β is an adjustable coefficient determined by experience and \mathcal{E}_i is the maximum validation error of sensor i . Thus, the final fault score can be obtained as

$$a_i(t) = \begin{cases} \frac{e_{t,i}}{\tilde{\sigma}_i}, & \text{if } e_{t,i} > \beta \mathcal{E}_i \\ 0, & \text{otherwise} \end{cases} \quad (22)$$

where $\tilde{\sigma}_i$ is the variance of e_i in all sliding windows. If $a_i(t) > 0$, sensor i could be labeled as fault in the t th window. Due to the in-depth learning of spatial-temporal dynamics in STAGED, the sensors related to fault roots will get higher fault scores.

Remark 3: Different from [4], the STAGED model is unsupervised, which means it could be trained without labeled data. For large-scale industrial processes, it's expensive or even impossible to obtain labeled data for some fault cases. Besides, due to the effective representations based on spatial-temporal modeling, STAGED can achieve node-wise diagnosis to guide the maintenance.

IV. EXPERIMENTAL STUDY

In this section, we conduct experiments to compare with the state-of-the-art algorithms, and discuss the impacts of the

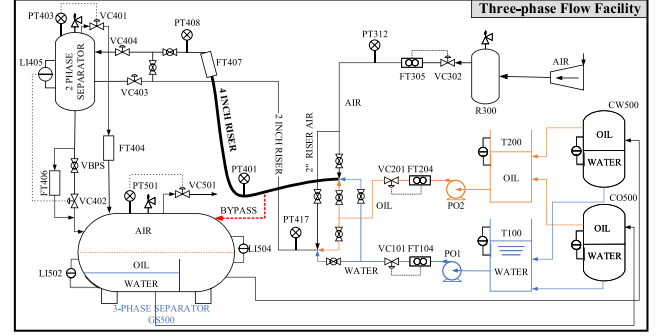


Fig. 3. TPF processes overview [22].

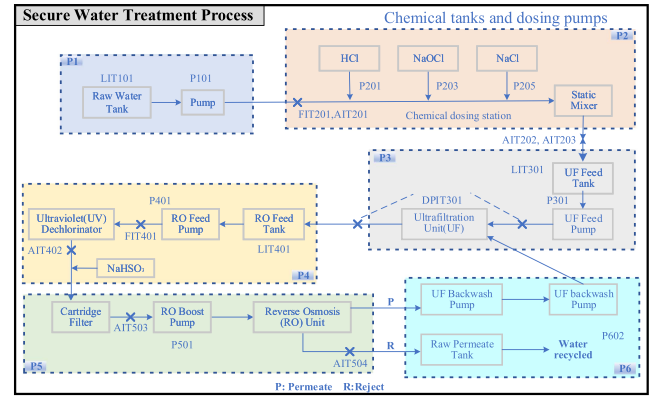


Fig. 4. SWaT testbed processes overview [23].

devised submodules. The parameter selection and fault diagnosis ability of STAGED would also be studied by cases.

A. Data Description

1) *Three-Phase Flow Facility Data:* The three-phase flow facility (TPFF) in Cranfield University could provide a controlled and measured flow speed of water, oil, and air to a pressurized system, which is shown in Fig. 3. The details of this dataset can be found in [22]. In SCADA system, 24 sensors measured critical process variables like pressure, flow rate, temperature, and density at different points, so it could guide the operators to get insight of the operating status in detail. To obtain the representative normal condition data, 20 different combinations of flow rates were tested. Besides, 6 fault conditions like air line blockage, slugging conditions, and top separator input blockage were tested to simulate typical malfunctions that could be experienced in real conditions.

2) *Secure Water Treatment Data:* The secure water treatment (SWaT) dataset is collected from a scaled down water treatment plant, and is provided by Singapore's Public Utility Board [23]. The SWaT testbed was run nonstop for 11 days, and several cyber and physical attacks were implemented in the last four days. A total of 51 sensors were deployed to collect signals like actuator status, flow rate, or pressure, which is shown in Fig. 4. The attacks were designed in a systematic way, and a corresponding attack list could help us to label the dataset. The

raw SWaT dataset is down-sampled to every 10 s by taking the median values and three sensors with fault measurements are removed.

B. Experimental Setup

1) **Baseline**: To verify the performance of the proposed STAGED framework, we compare it with some typical methods, including the classic fault diagnosis methods and the graph-based methods.

- 1) **PCA(1933)**: PCA [8] is a classical statistics technique to decompose the data matrix into vectors called principal components, and anomaly can be detected by reconstruction errors.
- 2) **LOF(2000)**: Local outlier factor [24] calculates the density of each point and its neighbors in the local region to judge the degree of abnormality, and has been a classic anomaly detection algorithm.
- 3) **IForest(2008)**: The isolation forests [25] use trees to separate data points, and the abnormal data is relatively dissimilar with the most data. So IForest could distinguish anomaly based on the separation complexity.
- 4) **AE(2006)**: Vanilla AE [26] uses multilayer perceptrons as encoder and decoder, and it does not specially take the spatial-temporal couplings into account. It diagnoses faults by reconstruction errors.
- 5) **MSCRED(2019)**: MSCRED [3] models the interactions between sensors by the signature matrices, and captures the temporal patterns by the convolutional LSTM.
- 6) **MTAD-GAT(2020)**: Multivariate time-series anomaly detection graph attention network [13] uses graph to model the interactions between sensors and detect faults by prediction deviations.
- 7) **GDN(2021)**: Graph deviation network [27] uses graph to model the interactions between sensors and detect faults by prediction deviations.
- 8) **STGAT-MAD(2022)**: Spatial-temporal graph attention network for multivariate time series anomaly detection [28] uses multiscale stacked modules to extract representations and detect faults by reconstruction errors.
- 9) **MemStream(2022)**: MemStream [29] is a streaming anomaly detection framework and uses a memory module to learn the dynamically changing trend in data without the need for labels.

2) **Evaluation Metrics**: The performance of different algorithms are measured by precision ($Prec$), recall (Rec) and $F1$ -scores. When the algorithms determine that a sliding window is fault/normal, the sliding window will be marked as positive/negative (We are interested in fault conditions). Given the ground truth, these labels can be further classified into True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). Thus, the metrics can be calculated by $F1 = \frac{2 \times Prec \times Rec}{Prec + Rec}$, where $Prec = \frac{TP}{TP + FP}$ and $Rec = \frac{TP}{TP + FN}$. $F1$ score is a comprehensive evaluation index for unbalanced classification problems, which is suitable for fault diagnosis. Following the strategy in [30], a whole fault segment would be treated

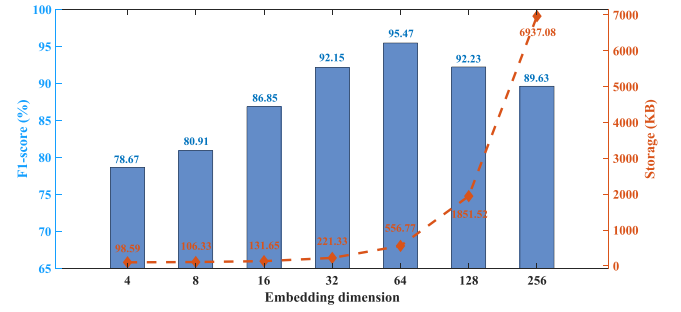


Fig. 5. Experiments on embedding dimensions.

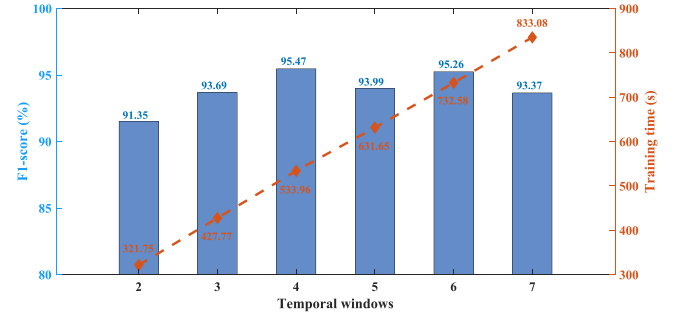


Fig. 6. Experiments on temporal windows.

as correctly detected if any sliding window within the segment is labeled as fault correctly.

3) **Implementation Details**: The proposed algorithm and its variants are implemented in PyTorch on a server with a GeForce RTX 3090 GPU and 64 Intel(R) Xeon(R) Gold 6226R CPU at 2.90 GHz. For both datasets, the length of sliding windows and sliding strides are all set as 32. The temporal length is 4, which means each sliding window would gather information from the past three sliding windows to capture the temporal evolution. The encoder layer, decoder layer, and dimension of hidden layer are set as 3, 3, and 64, respectively.

C. Parameter Selection

The embedding dimension and number of temporal windows are of great significance in the spatial-temporal modeling of STAGED. We conducted comparison experiments on SWaT here to investigate the parameter sensitivity and guide model selection. The encoder and decoder layers are all set to 3 in this section and all the models are trained for 50 epochs. The results are shown in Figs. 5 and 6.

Since the embedding dimension has great impacts on model size, the $F1$ -score and model storage cost are reported in Fig. 5. It can be found that with the increase of embedding dimension, the model size grows exponentially. Thus, it would be memory-costing if we increase the embedding dimension blindly. Also, the highest $F1$ -score is achieved with the embedding dimension of 64 and the model performance first raises and then descends with the increase of embedding dimension. These two observations inspire us to finally set the embedding dimension to 64.

TABLE II
STATISTICS OF THE TWO DATASETS

Datasets	#Features	#Train	#Test	Sampling frequency
TPFF	24	33397	23560	1 Hz
SWaT	48	47573	44992	0.1 Hz

TABLE III
FAULT DIAGNOSIS RESULT

Method	TPFF			SWaT		
	<i>Prec</i>	<i>Rec</i>	<i>F1</i>	<i>Prec</i>	<i>Rec</i>	<i>F1</i>
PCA	68.16	67.71	67.93	56.84	83.13	67.51
LOF	69.31	95.89	80.16	28.36	83.77	42.37
IForest	91.39	65.41	76.25	55.24	77.33	64.44
AE	82.21	91.78	86.73	62.27	84	71.38
MSCRED	87.08	95.19	90.96	95.28	65.41	77.56
MTAD-GAT	80.97	97.66	88.54	94.50	78.07	85.50
GDN	90.86	89.38	90.11	83.9	82.49	75.61
STGAT-MAD	81.04	96.04	87.91	92.22	88.53	90.34
MemStream	88.55	92.10	90.29	94.46	72.03	81.73
STAGED	91.72	97.48	94.51	97.28	93.72	95.47

TABLE IV
ABLATION STUDY RESULT

Method	TPFF			SWaT		
	<i>Prec</i>	<i>Rec</i>	<i>F1</i>	<i>Prec</i>	<i>Rec</i>	<i>F1</i>
STAGED	91.72	97.48	94.51	97.28	93.72	95.47
w/o LSTM	88.46	94.03	91.16	92.67	72.77	81.52
w/o Attention	89.97	90.25	90.11	93.64	84.82	89.01
w/o Prediction	90.12	94.65	92.33	95.69	93.19	94.43

The number of temporal windows would not impact the model size since LSTM networks can handle input of any length. From Fig. 6, it can be found that the training time increases linearly with the number of temporal windows, and the model performs better when the number of temporal windows is 4 or 6. The training results indicate that it would take more time for a larger model to converge. Besides, adding sliding windows will increase the inference delay. So it is recommended to set the number of temporal windows to 4 to achieve a balance between accuracy and efficiency.

D. Performance Evaluation

The results of different algorithms are reported on Table III. Obviously, the proposed method outperforms the baselines and achieves the beset *F1*-score on both datasets. MTAD-GAT, STGAT-MAD, and STAGED perform relatively better on the larger SWaT dataset with complex spatial-temporal coupling and all the deep learning based model performs well on the TPFF dataset with small scale. Moreover, the classic methods for general anomaly detection (i.e., PCA, LOF, and IForest) perform relatively worse since they are not specifically designed for time series, and the raw features from sliding windows are not representative enough without deep nonlinear transformations. MSCRED and GDN address the feature correlations more explicitly than AE with signature matrix or graph, so they are more suitable for multivariate time series anomaly detection and achieve performance improvement. MemStream is for streaming data anomaly detection and can be deployed directly without training. Although it does not design specific structure

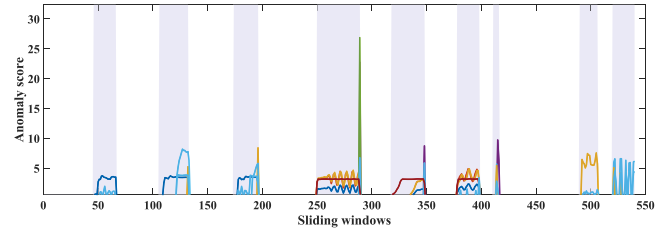


Fig. 7. Fault diagnosis performance on TPFF.

for temporal patterns, it introduces a memory module the storage the historical information for anomaly detection. STAGED captures the spatial-temporal interactions with GCN, attention mechanism and LSTM and is trained with both prediction and reconstruction tasks, so it could achieve better representation learning. For both datasets, STAGED can achieve *F1*-scores above 94%, which is at least 3.55% higher than the baselines.

E. Ablation Study

To further investigate the necessity of the devised modules, three model variants that remove the attention mechanism of embedding aggregation, LSTM for spatial-temporal encoder, and prediction decoder in training, respectively, are also tested.

- 1) Removing the LSTM module in encoder degrades the performance seriously, especially in SWaT. Since the measurements are naturally time series, and most physical components are governed by dynamic rules, the temporal evolution should be highlighted in the model structure design. This results verify the importance of spatial-temporal modeling in industrial fault diagnosis.
- 2) Without attention mechanism, the learned embeddings are aggregated by averaging the embeddings of all the past sliding windows. Thus, the performance degrades obviously because the embedding of the past windows should have different impacts on reconstructing the last window. The attention mechanism can model the temporal evolution in a overall perspective, and the *F1*-score drops 4.4% and 6.46% on TPFF and SWaT without it.
- 3) In the training process, the prediction decoder can guide the encoder optimization and enforce it to capture the system dynamics. Although it is not used in on-line diagnosis, using it in training still slightly improves the model performance. This implies that learning efficient representations is the basis for deep learning algorithms.

These findings indicate that the three mechanisms introduced in STAGED all contribute to its performance, and also validate the importance of spatial-temporal modeling in industrial fault diagnosis.

F. Fault Diagnosis

In Figs. 7 and 8, the fault detection results in the first 550 windows are depicted for case studies, in which the purple shaded regions represent the faulty periods, and the lines in different colors are fault scores for different sensors. It should be

TABLE V
FAULT LOCALIZATION CASE STUDIES

Datasets	Windows	Normal condition	Fault description	Diagnosis
TPFF	106-134	Manual valve at the input of air line was fully open	Manual valve of air line gradually closed, air line blockage	PT-312, FT-407, VC-501
	250-289	Top separator input valve VC404 was fully open	Valve VC404 closed, top separator input blockage	PT-312, PT-401, PT-408, PT-403, VC-404, PT-501
	414-416	Bypass after multi-phase flow mixing point was fully close	Direct bypass was opened to simulate leakage	PT-408, FT-407, PT-403
SWaT	33,34	Value of DPIT was < 40 kPa	Set value of DPIT > 40 kPa	DPIT-301, FIT-601, MV-301, P-602, MV-303
	360	Value of AIT-504 was < 15 $\mu\text{S}/\text{cm}$	Set AIT-504 to 255 $\mu\text{S}/\text{cm}$	AIT-504
	536	P-203 was on; P-205 was on	Turned off P-203 and P-205	P-204, P-206

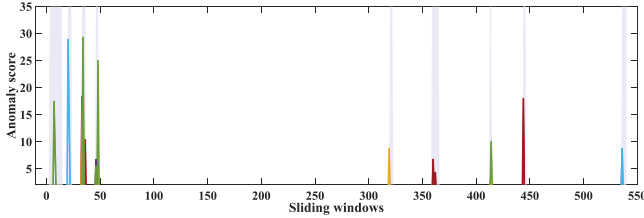


Fig. 8. Fault diagnosis performance on SWaT.

noted that most sensors are with fault scores of 0 because scores smaller than certain multiple of the validation errors have been eliminated. The results indicate that the proposed methods can accurately detect errors in industrial system.

In order to further analyze the performance of fault location, six cases from the test datasets are listed in Table V. The fault descriptions from log recordings are also listed, and we can verify whether the fault scores from STAGED can reflect the actual fault conditions and bring enlightenment to diagnosis.

For TPFF, the selected cases are analyzed as follows.

- 1) Windows 106–134 recorded an air line blockage fault, in which a manual valve of air line was gradually closed. As the fault started, the fault score of air delivery pressure in PT-312 gradually increased to 3.15. After 15 sliding windows (i.e., 480 s), because the air was blocked, the density in FT-407 grew and the air output of the 3-phase separator decreased, which were all indicated in the fault scores of the corresponding sensors.
- 2) Windows 250–289 witnessed the input blockage of 2 phase separator, and VC-404 was gradually closed. Soon after the fault, the pressures at PT-408, PT-403, and PT-312 (They were closely related to VC-404) decreased, and STAGED gave them high fault scores immediately. After 38 windows (i.e., 1216 s) the anomaly at downstream points PT-403 and PT-501 were also detected.
- 3) Windows 414–416 recorded the leakage at the bottom of the riser, where the bypass line directly isolated the input. Flow rate and density of top riser suddenly decreased, and STAGED noticed such phenomena. Also the pressure on top of the riser (i.e., PT-408 and PT-403) were affected.

For SWaT, three cases are selected, and analyzed as follows.

- 1) In windows 33 and 34, the valve of differential pressure indicating transmitter (DPIT) in P3 was set incorrectly, and STAGED recorded the deviation of DPIT-301. Besides, the components directly connected to DPIT (i.e., MV-301,

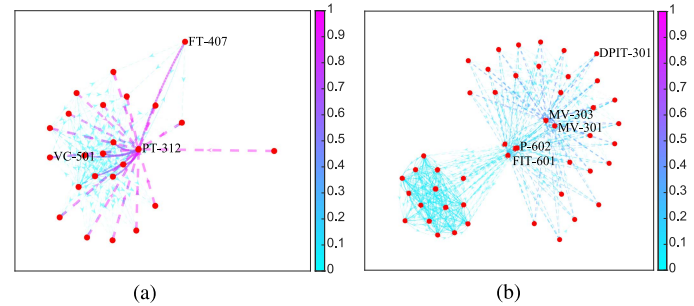


Fig. 9. Adjacent matrix of window 106 for TPFF and window 34 for SWaT. (a) TPFF. (b) SWaT.

P-602, and FIT-601) were influenced, which could also be observed with high fault scores.

- 2) In Windows 360, the permeate conductivity analyzer which measured the NaCl level in reverse osmosis process was maliciously set to an error value, STAGED accurately captured this short fault and gave alarms on AIT-504.
- 3) In window 536, the NaCl and HCl dosing pumps were mistakenly turned OFF, so the chemical dosing station would be seriously interfered. STAGED accurately found the changes in this two pumps and gave high fault scores to P-202 and P-204 (equal to P-201 and P-203 as they measured the same signals).

The above cases prove that STAGED can track the evolution of errors in industrial processes timely and accurately, thus, giving hints to operators, which could greatly improve fault localization speed and maintenance efficiency. Also, the fault information reflected by fault scores provides the interpretability of the proposed model, and greatly improves the practicality and reliability.

In order to further understand the sensor interaction modeling in STAGED, the adjacent matrices under the first listed fault conditions for both datasets are presented in Fig. 9. The red dots represent different sensors and the edges are the learned attention weights. The width and colors are related to the edge weights. For window 106 of TPFF, the fault is caused by the air blockage closed to PT-312, and many sensors pay great attentions to this change. Thus, the edge weights to PT-312 are mostly large. For window 34 of SWaT, the fault in sensor DPIT-301 interferes the ultra-filtration unit, and thus, the motorized valve MV-301 and MV-303 work inappropriately, which leads to larger scale failure in the system. Obviously, STAGED notices such changes and many sensors pay greater

attention to fault roots. The learned adjacent matrix under fault conditions also gives hints to fault diagnosis and improves the interpretability by indicating how sensors are related to each other.

V. CONCLUSION

This article proposed the STAGED model for industrial fault diagnosis based on high-dimensional time series, which fully considered the temporal evolution of time series and the complex interactions among sensors, and used LSTM network and GNN to simultaneously model the spatial-temporal characteristics. Considering the unsupervised nature of industrial fault diagnosis, the STAGED model took the reconstruction errors as the fault scores, which can be more practicable. Moreover, in order to force the model to better capture the spatial-temporal coupling, the attention mechanism and dual decoder structure are also introduced. A comprehensive weighted reconstruction and prediction loss function was devised to capture the temporal evolution better. The experiments on two typical industrial datasets verified the effectiveness of the designed modules and proved that STAGED could accurately localize the fault roots. Besides, further experiments on embedding dimensions and the number of temporal windows provide guidance for the parameter selection in practical deployment.

In future work, the multimodal data with different forms or sampling frequencies can be fused to improve performance, and fault diagnosis methods based on spatial-temporal modeling can be deployed in federated learning architecture to tackle with the dilemma of data islands in industry.

REFERENCES

- [1] D. Sinha and R. Roy, "Reviewing cyber-physical system as a part of smart factory in industry 4.0," *IEEE Eng. Manage. Rev.*, vol. 48, no. 2, pp. 103–117, Secondquarter 2020.
- [2] Y. Chi, Y. Dong, Z. J. Wang, F. R. Yu, and V. C. M. Leung, "Knowledge-based fault diagnosis in industrial Internet of Things: A survey," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 12886–12900, Aug. 2022.
- [3] C. Zhang et al., "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 1409–1416.
- [4] D. Chen, R. Liu, Q. Hu, and S. X. Ding, "Interaction-aware graph neural networks for fault diagnosis of complex industrial processes," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Dec. 17, 2021, doi: [10.1109/TNNLS.2021.3132376](https://doi.org/10.1109/TNNLS.2021.3132376).
- [5] L. Ruff et al., "A unifying review of deep and shallow anomaly detection," *Proc. IEEE*, vol. 109, no. 5, pp. 756–795, May 2021.
- [6] Q. Jiang, X. Yan, and B. Huang, "Performance-driven distributed PCA process monitoring based on fault-relevant variable selection and Bayesian inference," *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 377–386, Jan. 2016.
- [7] A. Giantomassi, F. Ferracuti, S. Iarlori, G. Ippoliti, and S. Longhi, "Electric motor fault detection and diagnosis by kernel density estimation and Kullback–Leibler divergence based on stator current measurements," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1770–1780, Mar. 2015.
- [8] C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning*, vol. 4, no. 4. Berlin, Germany: Springer, 2006.
- [9] L. Yang and Z. Zhang, "A conditional convolutional autoencoder-based method for monitoring wind turbine blade breakages," *IEEE Trans. Ind. Inf.*, vol. 17, no. 9, pp. 6390–6398, Sep. 2021.
- [10] Z. Chen, J. Xu, T. Peng, and C. Yang, "Graph convolutional network-based method for fault diagnosis using a hybrid of measurement and prior knowledge," *IEEE Trans. Cybern.*, vol. 52, no. 9, pp. 9157–9169, Sep. 2022.
- [11] Y. Zhang, C. Yang, K. Huang, and Y. Li, "Intrusion detection of industrial internet-of-things based on reconstructed graph neural networks," *IEEE Trans. Netw. Sci. Eng.*, early access, Jun. 21, 2022, doi: [10.1109/TNSE.2022.3184975](https://doi.org/10.1109/TNSE.2022.3184975).
- [12] Z. He et al., "A spatiotemporal deep learning approach for unsupervised anomaly detection in cloud systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 4, pp. 1705–1719, Apr. 2023.
- [13] H. Zhao et al., "Multivariate time-series anomaly detection via graph attention network," in *Proc. IEEE Int. Conf. Data Mining*, 2020, pp. 841–850.
- [14] Z. Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, "Learning graph structures with transformer for multivariate time-series anomaly detection in IoT," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9179–9189, Jun. 2022.
- [15] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, vol. 29, pp. 3837–3845.
- [16] W. Z. Joan Bruna, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–14.
- [17] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3837–3845.
- [18] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–14.
- [19] A. F. T. Martins and R. F. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *Proc. Int. Conf. Mach. Learn.*, 2016, vol. 48, pp. 1614–1623.
- [20] P. Qi, D. Li, and S.-K. Ng, "Mad-sgc: Multivariate anomaly detection with self-learning graph convolutional networks," in *Proc. IEEE Int. Conf. Data Eng.*, 2022, pp. 1232–1244.
- [21] J. Simeunović, B. Schubnel, P.-J. Alet, and R. E. Carrillo, "Spatio-temporal graph neural networks for multi-site PV power forecasting," *IEEE Trans. Sustain. Energy*, vol. 13, no. 2, pp. 1210–1220, Apr. 2022.
- [22] 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.
- [23] J. Goh, S. Adepu, K. N. Junejo, and A. Mathur, "A dataset to support research in the design of secure water treatment systems," in *Proc. Lect. Notes Comput. Sci.*, 2016, pp. 88–99.
- [24] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," in *Proc. SIGMOD - ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.
- [25] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. IEEE Int. Conf. Data Mining*, 2008, pp. 413–422.
- [26] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [27] A. Deng and B. Hooi, "Graph neural network-based anomaly detection in multivariate time series," in *Proc. 35th AAAI Conf. Artif. Intell.*, 2021, vol. 5A, pp. 4027–4035.
- [28] J. Zhan et al., "Stgat-Mad: Spatial-temporal graph attention network for multivariate time series anomaly detection," in *Proc. ICASSP IEEE Int. Conf. Acoust. Speech Signal Process.*, 2022, pp. 3568–3572.
- [29] S. Bhatia, A. Jain, S. Srivastava, K. Kawaguchi, and B. Hooi, "Memstream: Memory-based streaming anomaly detection," in *Proc. ACM Web Conf.*, 2022, pp. 610–621.
- [30] Y. Su, R. Liu, Y. Zhao, W. Sun, C. Niu, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 2828–2837.



Shizhong Li (Student Member, IEEE) received the B.S. degree in electrical engineering and automation from Central South University, Changsha, China, in 2018, and the M.S. degree in control science and engineering from Shandong University, Jinan, China, in 2021. He is currently working toward the Ph.D. degree in electronic information with the School of Control Science and Engineering, Zhejiang University, Hangzhou, China.

His research interests include deep learning, edge computing, and their applications in energy systems and industry.



Wenchao Meng (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2015.

He is currently with the College of Control Science and Engineering, Zhejiang University. His current research interests include adaptive intelligent control, cyber-physical systems, renewable energy systems, and smart grids.



Shibo He (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2012.

He is currently a Professor with Zhejiang University. He was an Associate Research Scientist from March 2014 to May 2014 and a Postdoctoral Scholar with Arizona State University, Tempe, AZ, USA, from May 2012 to February 2014. From November 2010 to November 2011, he was a Visiting Scholar with the University of

Waterloo, Waterloo, ON, Canada. His research interests include the Internet of Things, crowdsensing, and Big Data analysis.

Dr. He serves/served on the Editorial Board for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Peer-to-Peer Networking and Application* (Springer), and *KSII Transactions on Internet and Information Systems*, and is a Guest Editor for *Computer Communications* (Elsevier), and *International Journal of Distributed Sensor Networks* (Hindawi). He was the General Cochair for iSCI 2022, the Symposium Cochair for the IEEE/CIC ICC 2022, IEEE GLOBECOM 2020, and the IEEE ICC 2017, the TPC Cochair for i-Span 2018, the Finance and Registration Chair for ACM MobiHoc 2015, the TPC Cochair for the IEEE ScalCom 2014, the TPC Vice Cochair for ANT 2013 and 2014, the Track Cochair for the Pervasive Algorithms, Protocols, and Networks of EUSPN 2013, the Web Cochair for the IEEE MASS 2013, and the Publicity Cochair of IEEE WiSARN 2010 and FCN 2014.



Jichao Bi (Member, IEEE) received the Ph.D. degree in software engineering from Chongqing University, Chongqing, China, in 2020.

He is an Associate Research Fellow with Zhejiang Institute of Industry and Information Technology, Hangzhou, China. He was a Postdoctoral Research Fellow with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China, from 2020 to 2022. He was a Visiting Ph.D. student with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, Australia, from 2018 to 2019. He has authored or coauthored more than 20 academic papers in peer-reviewed international journals/conferences. His research interests include smart grid, cybersecurity, industrial Internet of Things, and deep learning.

Dr. Bi was the recipient of the Best Paper Award of TrustCom 2022 and YAC 2022.



Guanglun Liu (Student Member, IEEE) received the B.S. degree in electrical engineering and automation from Hohai University, Nanjing, China, in 2015, the M.S. degree in control theory and control engineering from Fuzhou University, Fuzhou, China, in 2018. He is currently working toward the Ph.D. degree in control science and engineering with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China.

His research interests include wind power systems fault diagnosis, prognosis, and Big Data analysis.

Mr. Liu is a Member of the Group of Networked Sensing and Control (IIPC-NesC).