

Research Article

Graph Feature Fusion-Driven Fault Diagnosis of Complex Process Industrial System Based on Multivariate Heterogeneous Data

Fengyuan Zhang ^(b), ¹ Jie Liu ^(b), ¹ Xiang Lu ^(b), ² Tao Li, ^{2,3} Yi Li ^(b), ⁴ Yongji Sheng, ⁴ Hu Wang, ⁵ and Yingwei Liu⁶

¹School of Civil and Hydraulic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
 ²Hubei Key Laboratory of Material Chemistry and Service Failure, School of Chemistry and Chemical Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
 ³Hubei Three Gorges Laboratory, Yichang 443000, China
 ⁴COFCO (Jilin) Bio-Chemical Technology Co., Ltd, Changchun 130000, China
 ⁵COFCO (Anhui) Bio-Chemical Technology Co., Ltd, Suzhou 234000, China
 ⁶COFCO Nutrition and Health Research Institute Co., Ltd, Beijing 102209, China

Correspondence should be addressed to Xiang Lu; luxiang_1028@163.com and Yi Li; li-yi1@cofco.com

Received 2 November 2023; Revised 23 December 2023; Accepted 5 February 2024; Published 8 March 2024

Academic Editor: Mattia Battarra

Copyright © 2024 Fengyuan Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The stable operation of the process industrial system, which is integrated with various complex equipment, is the premise of production, which requires the condition monitoring and diagnosis of the system. Recently, the continuous development of deep learning (DL) has promoted the research of intelligent diagnosis in process industry systems, and the sensor system layout has provided sufficient data foundation for this task. However, these DL-driven approaches have had some shortcomings: (1) the output signals of heterogeneous sensing systems existing in process industry systems are often high-dimensional coupled and (2) the fault diagnosis model built from pure data lacks systematic process knowledge, resulting in inaccurate fitting. To solve these problems, a graph feature fusion-driven fault diagnosis of complex process industry systems is proposed in this paper. First, according to the system's prior knowledge and data characteristics, the original multisource heterogeneous data are divided into two categories. On this basis, the two kinds of data are converted to physical space graphs (PSG) and process knowledge graphs (PKG), respectively, according to the physical space layout and reaction mechanism of the system. Second, the node features and system spatial features of the subgraphs are extracted by the graph convolutional neural network at the same time, and the fault representation information of the subgraph is mined. Finally, the attention mechanism is used to fuse the learned subgraph features getting the global-graph representation for fault diagnosis. Two publicly available process chemistry datasets validate the effectiveness of the proposed method.

1. Introduction

With the advent of the Industry 4.0 era, industrial processes are increasingly sophisticated and complex, which puts forward higher requirements for intelligent control of complex systems [1]. To achieve the goals of stable production, maximum economic profit, and green energy saving, it is increasingly important to develop an efficient and reliable health monitoring system for complex systems [2, 3]. The establishment of this system depends on a large number of sensing and monitoring equipment of complex types, and the most important task in a health monitoring system is fault detection and diagnosis (FDD) [4]. As a critical task, it has attracted more and more attention from researchers in recent years.

Machine learning (ML) based methods [5–7] and deep learning (DL) based methods [8–10] driven diagnostic methods achieve good results when performing FDD tasks in process industry systems because of their excellent nonlinear data fitting capabilities [11]. However, it is also challenged by the large number of heterogeneous sensors in complex process industry systems. These multiple data are coupled in high dimensions and cause a curse of dimensionality which weakens the DL models [12]. To reduce this influence, the feature fusion methods were applied to complex process industry FDD tasks. For example, Ye et al. constructed a feature fusion model by classifying multiple sensors and processing them separately [13]. Xu et al. on the basis of considering both the internal correlation and the distribution gap between different signals proposed a hybrid fusion network model to improve the accuracy of diagnostic tasks [14]. However, these methods ignore the prior knowledge, resulting in these feature fusion models are not closely combined with the characteristics of complex process industry systems.

In recent years, the wave of deep learning has made it possible to combine data analysis tools with process knowledge to build robust and scalable models of process industry systems [15]. For example, Venkatasubramanian added mechanistic constraints to the purely data-driven models based on first principles knowledge in the process industry [16]. Bikmukhametov and Jäschke combined machine learning and process engineering physics to enhance the accuracy and explainability of data-driven models [17]. From the point of view of physical mechanism, Ni et al. proposed a physics-informed residual network (PIResNet) that can mine the machine's potential physical characterization from measured data [18]. However, most of these methods use prior knowledge to constrain established models rather than build models directly. Meanwhile, manual selection of prior knowledge-based constraints also weakens the convenience of models. As an alternative, the edge connection between nodes in graph structure data could express the prior knowledge information of the system.

Graph-based approaches are a type of deep learning model that can convert data into graph structure or non-Euclidean space, which has drawn much attention from researchers recently [19]. Graph data contain not only data values-based information but also the structural and topological information hidden in raw data which is determined by the process knowledge of the system [20]. For example, Man et al. constructed the graph structure from the sensor layout on high-speed rail rotating machinery to mine the potential relationship between the sensor signal [21]. Liu et al. divided the timing signal into nodes and created a graph structure based on the similarity between nodes to solve the problem of fault diagnosis under unbalanced samples [22-24]. Zhang et al. mined state representations of operating units from complex time-varying operating condition information [25]. To mine fault features from the constructed graph data, the graph convolutional neural network (GCN) was used to extend convolution operations in spectral space, which is successfully used in many fields [26]. These studies also shed light on fault diagnosis in the process industry domain. The rich prior knowledge and spatial

information of the process industry provide the basis for the graph-driven approach.

In the process industry FDD tasks, graph-based approaches are beginning to get attention. There is a strong correlation between the upstream and downstream units of the process industry system, which means that the fluctuation of the signal in one link will quickly spread to the entire system [27]. As the production state of the system changes, the relationship between the sensing signals also changes. To capture this relationship, Chen et al. used different edge connection functions to construct graphs to capture fault representation information [4]. Zhang and Yu designed a pruning GCN model to reduce noise based on signal similarity composition [28]. These GCN-based models show the validity of graph theory in process industry FDD tasks, but there are still shortcomings. The direct conversion of multisource heterogeneous monitoring data into a single graph will result in high-dimensional coupling of different functions of data and reduce the quality of graph representation learning. Meanwhile, the model built from the monitoring data purely ignores the physical spatial information of the system, which is important process knowledge. According to the importance of the process, different numbers of sensors are arranged in different links. These heterogeneous sensors describe the same changing process from multiple perspectives, so it is necessary to analyze them jointly.

To solve the above problems, the original monitoring data are divided into two types and converted into graph structure data by combining the system process knowledge. Specifically, the first type is reaction process monitoring parameters, which record the working condition information of each working unit of the system, and these data information cooperate to illustrate the production state of the system. This type of data is converted to physical space graphs (PSGs) based on the integration of the physical spatial layout of the system sensors. The second category is sampling quality index parameters, which can directly determine the quality of reaction products and are affected by the mechanism of upstream and downstream processes. Pretrained networks are used to extract higher-order features from the above data to capture the mechanistic knowledge flowing through the system and convert them into process knowledge graphs (PKGs). Therefore, the research based on fused PSG and PKG for complex industrial FDD task is explored in this paper. The major contributions are summarized as follows:

- (1) The sensors in the process industrial system are transformed into graph structure by Euclidean distance measurement based on physical spatial layout, and the connections between multisource heterogeneous signals describing the same key process are established
- (2) The reaction knowledge flow in the process system is captured by the pretraining network, and the higherorder features of the internal relationships in the technical process flow are used to represent the system state

(3) Multichannel graph feature fusion (MCGFF) model is proposed to mine fault representation from two different subgraphs and then fuse subgraph features into global-graph features through an attention mechanism for fault diagnosis

The remainder of this paper is organized as follows. The graph representation learning theory is introduced in Section 2. The proposed deep graph feature learning-based diagnosis framework is introduced in Section 3. Section 4 implements a comparison between MCGFF with other models on the Tennessee Eastman process (TEP) and fedbatch fermentation penicillin process (FBFP). The discussions are given in Section 5, and the conclusion is given in Section 6.

2. Preliminary

2.1. Graph Representation. Mathematically, both undirected and directed graphs are denoted as $G = \{V, \mathbf{A}, E, \mathbf{F}\}$, as shown in Figure 1. $V = \{V_i\}$ represents the node set, which consists of the measurement time of the complex system. *E* represents the edge connections between these nodes. $\mathbf{A} = \{a_{i,j}\}$, $a_{i,j} \subseteq \{0,1\}$, is the adjacency matrix, where $a_{i,j} = 0$ represents there is no edge between node *i* and node *j*, and $a_{i,j} = 1$ represents there is an edge. **F** is an eigen matrix composed of all node eigenvalues.

2.2. Spectral Graph Convolution. Traditional convolution operations cannot be applied to graph domains, so graph convolution theory is proposed. As the basis of graph convolution theory, the classical Laplacian matrix is used for feature representation, which is denoted as follows:

$$L = I_n - D^{-1/2} W D^{-1/2}, \qquad (1)$$

where $\mathbf{L} \in \mathbb{R}^{N \times N}$ represents the Laplacian matrix, $\mathbf{I_n} \in \mathbb{R}^{N \times N}$ represents the identity matrix, $\mathbf{D} \in \mathbb{R}^{N \times N}$ denotes the degree matrix, and $\mathbf{W} \in \mathbb{R}^{N \times N}$ denotes the weight matrix. On this basis, feature extraction acting on the Laplacian matrix is used to realize convolution on graph signal $\mathbf{X} \in \mathbb{R}^{N \times S}$ [29].

$$\mathbf{Y} = \mathbf{U}\mathbf{g}_{\theta}(\mathbf{\Lambda})\mathbf{U}^{T}\mathbf{X},$$
 (2)

where $\mathbf{U} \in \mathbb{R}^{N \times N}$ represents the eigenvectors of the Laplacian matrix $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}\mathbf{T}$, $\mathbf{\Lambda}$ is the eigenvalues, $\mathbf{Y} \in \mathbb{R}^{N \times S}$ is the output of the filter, and $\mathbf{g}_{\theta}(\mathbf{\Lambda})$ is a filter parameterized by $\theta \in \mathbb{R}^{n}$.

The difficulty of calculating formula (2) led to the formulation of a new convolution formula, the Chebyshev convolution [30], which is defined as follows:

$$\mathbf{g}_{\theta}(\mathbf{\Lambda}) \approx \sum_{\mathbf{k}=0}^{K-1} \theta_k T_k(\tilde{\mathbf{\Lambda}}),$$
 (3)

where Λ is rescaled as $\overline{\Lambda} = 2\Lambda/\lambda_{max} - \mathbf{I}_n$, λ_{max} denotes the largest element of Λ , K is the order of Chebyshev polynomials, θ_k is the Chebyshev coefficient, and $T_k(\cdot)$ is the recursive Chebyshev polynomial, defined in equation (4). $T_k(\overline{\Lambda})$ denotes a function of the diagonal element of $\overline{\Lambda}$.

$$\begin{cases} T_0(x) = 1, T_1(x) = x, \\ T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x), (k \ge 2). \end{cases}$$
(4)

The mathematical definition of the Chebyshev graph convolution derived from the above equations is illustrated as follows:

$$\mathbf{Y} = \sum_{k=0}^{K-1} \theta_k \mathbf{U} T_k(\tilde{\mathbf{\Lambda}}) \mathbf{U}^T.$$
 (5)

A trainable parameterized weight matrix $\mathbf{W} \in \mathbb{R}^{S \times M}$ is introduced to implement feature matrix deformation and achieve feature transformation [31]. The output of the GCN layer $\mathbf{X}' \in \mathbb{R}^{N \times M}$ is shown as follows:

$$\mathbf{X}' = \operatorname{Cheb}\left(\mathbf{X}, \mathbf{W}\right) = \mathbf{Y}\mathbf{W},\tag{6}$$

where Cheb(,) is the Chebyshev graph convolution, and W is the trainable parameterized weight matrix.

3. Proposed Method

The prior knowledge of the complex process industrial system includes physical space layout and reaction mechanism. The physical space layout of the system reflects the monitoring information of key process control, and the reaction mechanism determines the correlation changes between signals. On the basis, this study will start from these two perspectives to extract prior knowledge to construct the graph model.

3.1. *Physical Space Graph.* In this section, physical space graphs (PSGs) are constructed from the sensor layout in the complex industrial system, as shown in Figure 2.

3.1.1. PSG Construction. According to the different system designs and sensor layouts, each chemical system has unique spatial information. On this basis, the physical spatial sensor layout in the system is transformed into a graph structure named physical space graph (PSG) to capture this spatial information. This information can explicitly express the relationship between nodes and provide a basis for spatial information and fault representation mining.

To concretely express the sensor space layout of the chemical system, the relative position of each sensor is transformed into the coordinate system by directly mapping the real system, and then the spatial coordinates of each unit are obtained. On this basis, sensors are considered nodes, and edge connection between the nodes is established with similar distances. As densely spaced sensors often describe the same important process from different perspectives, such as pressure, temperature, and power, their joint analysis is valuable. The node is connected with the closest k nodes, and the distance between the nodes is calculated by the Euclidean distance formula, as shown in the following equation:

$$D(v_{i}, v_{j}) = \sqrt{(x_{i} - x_{j})^{2} + (y_{i} - y_{j})^{2}},$$
(7)



FIGURE 1: Undirected graph and the corresponding matrix representation.



FIGURE 2: Flow diagram of PSG construction.

where $v_i(x_i, y_i)$ and $v_j(x_j, y_j)$ are two nodes in the coordinate system, and $D(v_i, v_j)$ is the Euclidean distance between the *i*-th node and *j*-th node.

3.1.2. Node Embedding. With the edge connections between the nodes determined, node features should be embedded in each node. Data normalization is operated on the original measurement data $X = (x_1, x_2, ..., x_h), x_i \in \mathbb{R}^n$, where *n* is the number of sensors. The normalized data X^{nor} can be calculated as follows:

$$x_i^{\text{nor}} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \quad i = 1, 2, ..., m.$$
 (8)

Subsequently, the unsupervised PCA algorithm is used to process the normalized data X^{nor} obtaining the data after reducing noise, as shown in the following equation:

$$X^{F} = PCA(X^{nor}).$$
(9)

The high-dimension sensor data series are processed and separated into n sensor sequences. A sliding window s is set up to divide these monitoring sequences into m sizes, and the node set $L = (l_1, l_2, \ldots, l_{n*s}), l_i \in \mathbb{R}^m$, in length m is obtained. It is worth mentioning that each time series segment m is translated into a graph with the determined edge connection.

3.2. Process Knowledge Graph. In this section, the process knowledge graphs (PKGs) are constructed to capture the relationships between nodes which are determined by the reaction mechanism in the complex industrial system. The proposed process is depicted in Figure 3.

3.2.1. Data Segments as Nodes. The *n*-dimensional monitoring signal output of each sensor unit in a chemical system can be represented as $X = (x_1, x_2, ..., x_h), x_i \in \mathbb{R}^n$.

According to the characteristics of the production line and the different needs of actual production, these signal sequences are divided into different segments $X_{seg} = (x_1, x_2, ..., x_m), X_{seg} \in \mathbb{R}^{m \times n}$ of length *m*. It is worth noting that in this subtask, all the data entered build only one graph.

3.2.2. *PKG Construction.* The monitoring signals are converted into process knowledge graphs (PKGs) under the assumption that the internal relationships between variables are reflected by the reaction mechanism and GCN can effectively mine fault representation information from these relationships. Data normalization is operated on the sliced multivariate sensor data $X = (x_1, x_2, ..., x_h), x_i \in \mathbb{R}^{m \times n}$, and the normalized data X^{nor} can be calculated as equation (8). Subsequently, the supervised linear discriminant analysis (LDA) algorithm is used to process the normalized data X^{nor} obtaining the data after dimension reduction as the node features, as follows:

$$F = \text{LDA}(X^{\text{nor}}, Y), \tag{10}$$

where LDA(,) is the linear discriminant analysis algorithm, $X^{\text{nor}} = (x_1, x_2, \dots, x_m), x_i \in \mathbb{R}^{m \times n}$, is the original vector of the input, $F = (f_1, f_2, \dots, f_m), f_i \in \mathbb{R}^{m \times d}$, is the reduced dimension vector as the node feature, and $Y = (y_1, y_2, \dots, y_k)$ is the status label for X^{nor} .

As the node features are obtained, the raw data are converted to a k-nearest neighbor graph (KNNG) by the unsupervised k-nearest neighbor (KNN) algorithm.



FIGURE 3: Proposed PKMP flow diagram.

Specifically, by calculating the feature similarity between the samples and creating edge connections between the samples and their nearest k samples, the graph processing of the associated signal is realized. Furthermore, the system's high-dimensional monitoring signals will hinder the accuracy of Euclidean distance, so the Mahalanobis distance formula is used to calculate the sample distance [26], as follows:

$$\operatorname{Dist}_{M}(v_{i}, v_{j}) = \sqrt{(v_{i} - v_{j})^{T} \sum_{i=1}^{-1} (v_{i} - v_{j})}, \quad (11)$$

where v_i and v_j are *i*-th node and *j*-th node, respectively. \sum^{-1} is the covariance matrix of multidimensional variables.

At this point, the original graph is constructed, and the edge connections between nodes will provide the connections between signals for the graph neural network to enhance the ability of the model to capture details. However, due to the inherent characteristics of the KNN algorithm, the edge connection will be calculated between each node independently, which may lead to multiple edge connections in one node. With the number of nodes and edge connection increase, the whole graph structure becomes bloated, which will greatly increase the computational burden and reduce the performance of model diagnosis. At the same time, due to the presence of noise, redundant edge connections that are established in two dissimilar nodes will provide a false representation. To simplify the original graph and make it retain real valuable information, the high-level features of nodes are extracted through the pretraining GCN layer named PKMP. The steps are illustrated as follows:

Step 1: The original graph is fed into the GCN and trained using the cross-entropy loss function.

Step 2: The higher-level features obtained from the training are regarded as node features, and the nearest neighbor graph is constructed again. The reconstruction graph G_{temp} has the same number of

nodes as the original graph with new node features and edge connections.

Step 3: The obtained reconstruction graph G_{temp} is sent to GCN training again as a new graph, and the temporary reconstruction graph G_{temp}^1 will be updated to G_{temp}^2 .

Step 4: Repeat Step 2 and Step 3 for *k* times, the ultimate output $G_{\text{temp}}^M = G_{\text{PK}}$ is the refined process knowledge graph.

The selection of parameter M is determined in training according to the characteristics of different data. Specifically, when the reconstruction of five batches cannot improve the effect, the total number of PKG reconstructions is selected as M. Unlike each PSG corresponds to a segment of time series, PKG corresponds to a ll the input data, and each node corresponds to a segment of time series.

3.3. Multichannel Graph Feature Fusion Model. As two types of subgraphs have been obtained, a multichannel graph feature fusion model (MCGFF) is designed. According to the difference of task level, the graph-level GCN and node-level GCN are used, respectively, for representation learning of subgraphs. On this basis, the learned subgraph representations are weighted and fused through the attention mechanism.

The attention mechanism allows the model to focus more on important representational information by giving the raw data a unique attention vector. The successful application of this feature has made the attention mechanism a classic enabling tool in the field of deep learning [32]. Specifically, dynamic weight parameters are used to reinforce important information while weakening useless information, and the process can be described in the following equation:

$$a_i = \operatorname{soft} \max\left(s\left(h_i, q\right)\right) = \frac{\exp\left(s\left(h_i, q\right)\right)}{\sum_{j=1}^n \exp\left(s\left(h_i, q\right)\right)},$$
(12)

where a_i is the obtained attention distribution coefficient of the vector h, h is the original input vector, q is the query vector, and $s(h_i, q)$ is the scoring function, which is defined as follows:

$$s(h,q) = v^{T} \tanh(Wh + Uq), \qquad (13)$$

where **W** and **U** are learnable parameter matrices, v is a learnable parameter vector, and tanh () is a hyperbolic tangent function.

After subgraph feature fusion, the fused global-graph representation is used for fault diagnosis. The overall flowchart of the proposed diagnosis MCGFF framework is shown in Figure 4, and the algorithm is summarized in Algorithm 1.

4. Case Study

Two public datasets of process industrial systems, including the Tennessee Eastman process (TEP) and fed-batch fermentation penicillin process (FBFP), were used for experimental verification. All algorithms were written in Python3.8 with Pytorch kit and processed by a server with an NVIDIA GeForce RTX3060 and a 16G RAM.

4.1. Case I: TEP Dataset

4.1.1. Data Description. As a classic chemical process simulation system [33], TEP is widely used in the research of process system condition monitoring [34]. The system is capable of generating a total of 41 monitoring quantities containing nonlinear relationships and producing sequential monitoring sequences at three-minute intervals. To ensure that the data analysis is more representative, only the monitoring data of the stable production stage were used for this experimental study. The schematic of TEP is shown in Figure 5.

The experiment was carried out based on mode 1 of the TEP system, and all the twelve manually controlled variables were in the initial state. Thus, ten production statuses were simulated, including normal status and nine different process disturbances which indicate typical malfunctions that could occur in real practice. It is worth noting that in addition to the six perturbations involving known variables and occurrence types, we also set up three unknown perturbations composed of two random perturbations at random times to enhance the complexity of the data, and the details are shown in Table 1.

The simulation of each batch lasted for 48 hours, and the sampling interval was 3 min, so a total of 960 data were generated. It should be noted that the upper and lower limits of reasonable operation were set for each device in the simulation model. Once a certain limit is breached, the reaction will stop to protect the complex system. In all the fault types introduced, the interruption of the *feed A loss* will trigger the complex system to stop. One hour of data were segmented as one sample and then mixed all the samples from both the abnormal state and normal state. Furthermore, we randomly take 70% samples as the training set and 30% samples as the testing set.

4.1.2. Result Analysis. Different comparison experiments based on classical data-driven methods were also set up to verify the effectiveness of the proposed MCGFF-driven method. Statistic learning methods contain PCA [5], LDA [35], and PCA + LDA [36]. Classification methods based on deep learning contain CNN [8] and standard GCN [37]. The details of the proposed MCGFF model are shown in Table 2, and the original learning rate was set to 0.01. All models were tested 10 times and analyzed for average accuracy, as shown in Figure 6.

As shown in Figure 6, due to the nonlinear characteristics of complex process industry systems, classical statistical learning methods (PCA, LDA, and PCA + LDA) cannot achieve good accuracy. The classical deep learning method CNN has a certain nonlinear capturing ability, but it was difficult to accurately diagnose faults due to the mutual coupling problem of multisource heterogeneous data. The original GCN can improve the diagnostic capability of the model, which suggests that representing signals as graph data can enhance the relevant features. The graph spatial structure obtained by signal conversion gives the process system unique state representations, but the problem of data coupling still exists, which limits the diagnostic capability of the model.

Compared with these models, MCGFF achieved the highest diagnostic accuracy of 94.33%, verifying the effectiveness of the proposed method, which further improves the accuracy of the traditional GCN by 8.69%. This effect was achieved by the following three aspects: (1) heterogeneous monitoring signals and sampled data in the system were classified to reduce the risk of model confusion, (2) the connection between multisource heterogeneous signals describing the same process was established, and thus, the state of key processes can be analyzed jointly, and (3) the reaction knowledge flow in the process system was captured by the pretraining network, and the higher-order features of the internal relationships of the process flow were used to represent the system state. All in all, the proposed MCGFF-based method can perform accurate fault diagnosis for process industrial systems and effectively improve the accuracy of the GCN-based method.

4.2. Case II: FBFP Dataset

4.2.1. Data Description. The main body of the FBFP [38] experimental unit is a fermenter, which is used to perform the fermentation task of continuous production. On this basis, two proportional, integral, and differential (PID) cycles are used to control various production indices in the tank, such as temperature acidity and hot and cold flow. The whole process of production can be divided into three



FIGURE 4: General process of the proposed diagnosis method.

(A) PSG Construction. Input: Sliced data $X = (x_1, x_2, ..., x_s), x_i \in \mathbb{R}^{m*n}$ Coordinate of sensor $v_i(x_i, y_i), i = 1, 2, ..., n$. Output: graph set $G_{PK} = \{G_{PK}^1, G_{PK}^2, ..., G_{PK}^s\}$. (1) Obtain the normalized signal X^{nor} ; (2) Calculate the feature data: $F = (f_1, f_2, ..., f_s) = PCA(X^{nor}), f_i \in \mathbb{R}^{m*n}$; (3) Separate feature data $L = (l_1, l_2, ..., l_{n*s}), l_i \in \mathbb{R}^m$; (4) Calculate the Euclidean distance: $D(v_i, v_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$; (5) Obtain the *k* closest neighbors nodes set of node $V_p: \psi(V_p) = \{\tilde{V}_{pi}\}_{i=1}^k$ if $D(V_p, \tilde{V}_{pi})$ is *k*-th smallest; (6) Establish the edge connections for every node; (7) Embed $L = (l_1, l_2, ..., l_{n*s}), l_i \in \mathbb{R}^m$, as node features; (8) Output graph set $G_{PK} = \{G_{PK}^1, G_{PK}^2, ..., G_{PK}^s\}$. (B) PKG Construction. Input: original feature matrix F^0 , training epoch *M* for the G_{PK} . Output: PKG G_{PK} with high-level feature matrix *F*. (1) Obtain the normalized signal X^{nor} ; (2) Calculate the feature matrix: $F^0 = LDA(X^{nor})$; (3) Calculate the Mahalanobis distance: Dist_M (v_i, v_j) ; (4) Obtain the *k* closest neighbors node set of node $V_p: \psi(V_p) = \{\tilde{V}_{pi}\}_{i=1}^k$, if $Dist_M (V_p, \tilde{V}_{pi})$, is *k*-th smallest;

(5) Establish the edge connections for every node; (6) Obtain original graph G_0 and original feature matrix F^0 ; (7) for i = 1, 2, ..., M: Train the GCN model for *M* epochs: $F^{i+1} = \text{PKML}(F^i) \text{ and } G^{i+1}_{\text{temp}} = \text{PKML}(G^i_{\text{temp}});$ end for (8) Output the PKG $G_{PK} = G_{temp}^M$. (C) Fault diagnosis using MCGFF. **Input:** $G_{PS} = \{G_{PS}^1, G_{PS}^2, ..., G_{PS}^s\}$ and G_{PK} . **Output:** The health label Z. (1) Divide the training set and testing set: V_{train} , V_{test} ; (2) Train the GCN model; (3) for V in V_{train} do: $MCGFF(V) \longrightarrow Z;$ $-\sum_{i=1}^{N_L} b_i \log(p_c) \longrightarrow CE$ loss; Update with backward propagation; end for (4) Output the health label: $MCGFF(V_{test}) \longrightarrow Z$.

Algorithm 1: MCGFF.



FIGURE 5: Diagram of the Tennessee Eastman process [34].

sequential stages, namely, cell growth stage, penicillin synthesis stage, and cell autolysis stage. The schematic diagram of this process is shown in Figure 7.

The system of the penicillin fermentation process contains eight kinds of manual control variables, two kinds of automatic control variables, and eleven kinds of monitoring data. Furthermore, five process disturbances are introduced into different batches. The process disturbance details are listed in Table 3 and plotted in Figure 8. In this experiment, each batch lasted for 230 h, and the sampling interval of the

	L	
Fault no.	Process variable	Туре
1	A/C ratio, B composition constant (stream 4)	Step
2	B composition, A/C ratio constant (stream 4)	Step
3	A feed loss (stream 1)	Step
4	A, B, C feed composition (stream 4)	Random variation
5	C feed temperature (stream 4)	Random variation
6	Reaction kinetics	Slow drift
7	Unknown	Unknown
8	Unknown	Unknown
9	Unknown	Unknown

TABLE 1: TE process disturbances.

TABLE 2: The details of the proposed MCGFF model.

Commonanta	Setting details			
Components	Input size	Output size	Order K	
ChebGCN_1	20 * 22	20 * 256	5	
ChebGCN_2	20 * 256	20 * 512	5	
ChebGCN_3	20 * 512	20 * 1024	5	
Graph pooling layer	20 * 1024	5 * 1024	_	
Fully connection layer_1	1024	512	_	
Fully connection layer_2	512	256	_	
Fully connection layer_3	256	9	—	



FIGURE 6: Classification results of the experimental models in TEP.

monitoring unit was 12 min; thus, 1150 original samples were contained. Each disturbance was simulated in 10 batches, and a total of 11500 anomaly samples were obtained. These abnormal samples were mixed with 100 batches of healthy samples, resulting in a total of 126,500 samples for experimental analysis.

In order to avoid possible information leakage problems, the samples were processed by mask, and then, the training set and test set were divided into 7:3 ratio.

4.2.2. Result Analysis. Comparative experiments were also carried out as same as TEP experiments in Case I.



FIGURE 7: Schematic diagram of the fed-batch fermentation penicillin process.

TABLE 3: Proce	ss disturbance	details for	different	batches in	FBFP.
----------------	----------------	-------------	-----------	------------	-------

No.	Process disturbance	Time duration (h)
1	Normal state	_
2	Aeration disturbance	(20, 24), (100, 110)
3	PH disturbance	(80, 90), (140, 170)
4	Heating/cooling water flow disturbance	(70, 135)
5	Generated heat disturbance	(30, 50), (70, 90)
6	Base flow disturbance	(30, 230)





(d) FIGURE 8: Continued.



FIGURE 8: Different disturbances in five batches: (a) aeration disturbance; (b) PH disturbance; (c) heating/cooling water flow disturbance; (d) base flow disturbance, and (e) generation heat disturbance.

The detailed settings of the proposed method were the same as in Table 2, with the output size of K=6 and K=3 replacing. The learning rate was set as 0.01, and the experimental results are shown in Figure 9.

The effect of the statistical learning model was still inferior to the deep learning model. However, because the complexity of the FBFP system is smaller than that of the TEP system, the results of all models are better than the TEP system. The diagnostic accuracy of 89.43% was obtained by using the original GCN model, which means that the system has a relatively simple spatial structure. On this basis, MCGFF improved the diagnostic accuracy to 92.46%. The less model accuracy improvement was speculated to be due to the smaller number of sensors in the FBFP and the simpler spatial layout. In addition, the improvement of the model effect is closely related to the appropriate aggregation of node information by the graph model. How to determine the most appropriate composition range and the number of convolutional nodes needs to be optimized according to different system data characteristics. However, the MCGFF model still achieves the best results, which verifies the validity of the proposed model.

5. Discussion

For graph-driven deep learning models, the quality of the constructed graph is the primary factor that determines the model effect. For this reason, three aspects that affect the effectiveness of the proposed method are discussed, including the impact of spatial aggregation range, the impact of reconstruction times, and the impact of convolutional parameter.

5.1. Impact of Spatial Aggregation Range. According to the spatial layout of different chemical process systems, the number of PSG aggregate neighbor nodes k is an important

parameter. An appropriate *k* value ensures that nodes from the same process are included without aggregating nodes too far. To this end, ten trials of experiments on different orders of k ($k = 2 \sim 7$) were conducted in both the above two datasets to compare their results, and the average results are shown in Figure 10.

As shown in Figure 10, the classification accuracy peaked at k=4 in both two datasets. This may be because the number of sensors in the critical step is usually 4, and too large k will lead to the aggregation of unrelated sensors, resulting in reduced accuracy. In the TEP dataset, the increase of k value will bring about a significant decrease in results, because it contains different reaction segments, which are quite different from each other. In contrast, sensors in FBFP monitor a single reaction process, which means that the extended aggregation range does not bring a large loss of model accuracy. Therefore, in this experiment, the parameter k of PSG was selected as 4 in both two datasets.

5.2. Impact of Reconstruction times. According to different data characteristics, the parameter M in PKMP has different best choices. Parameter M determines the number of times that the GCN layer extracts higher-lever node features and reconstructs graphs. Less M will result in more retained noise signals, and more M will lead to too slow model training and weaken the model's ability. In this experiment, M was set as 1 (that is, degraded into an ordinary GCN network) to 9 with ten times comparison test in both two datasets, and the experimental results are shown in Table 4.

It can be seen from the results in the TEP dataset that although the precision fluctuation decreases when M > 6, it will greatly increase the consumption of computing resources. This is because proper reconstruction can reduce



FIGURE 9: Classification results of 5 models in FBFP.



FIGURE 10: Experimental results of different numbers of k-nearest neighbors.

noises, but too much reconstruction will also introduce new noises, limiting the model effect to improve further. The same result happens in the FBFP dataset, with the accuracy peaking at M = 4. This is because simpler systems contain less noise and, therefore, require fewer reconstructions. To find a balance between accuracy and computational efficiency, M was set to 6 and 4 in the TEP dataset and FBFP dataset, respectively.

TABLE 4: Experimental results of comparison with different parameter M.

Parameter M	TEP dataset		FBFP dataset		
	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)	
1	83.26 ± 3.53	24	83.72 ± 3.15	17	
2	85.74 ± 2.71	51	85.24 ± 2.67	32	
3	86.91 ± 2.65	76	87.89 ± 2.13	51	
4	89.35 ± 1.88	102	92.68 ± 1.78	69	
5	92.48 ± 1.52	130	92.62 ± 1.65	85	
6	93.16 ± 1.40	164	92.57 ± 1.47	103	
7	92.79 ± 1.34	191	92.88 ± 1.55	119	
8	92.82 ± 0.98	223	92.74 ± 1.42	136	
9	92.54 ± 0.95	258	92.25 ± 1.49	152	

5.3. Impact of Convolutional Parameter. For ChebGCN, the choice of Chebyshev polynomial *K* is particularly critical, which represents the convolutional network aggregating information from the *K*-order neighbor nodes of the node. Too small *K* will limit the ability of the network to mine information from the graph structure data, and too large *K* will increase the consumption of computing resources exponentially and increase the noise. To ensure that the potential of the model is fully exploited, experiments are designed to determine the optimal *K* value, and the experimental results are listed in Table 5.

As shown in Table 5, when *K* goes from 2 to 5, the accuracy in the TEP dataset keeps going up, and as *K* goes beyond 5, the accuracy stays at about 93%. The same result happens in the FBFP dataset, with the accuracy peaking at K = 4. Considering that every increase in *K* will bring a large

Value of K	TEP dataset		FBFP dataset		
	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)	
2	77.24 ± 1.69	143	85.42 ± 2.57	97	
3	85.69 ± 1.53	187	92.33 ± 1.89	133	
4	90.17 ± 1.22	248	92.76 ± 2.12	174	
5	93.75 ± 0.86	305	92.55 ± 1.94	236	
6	93.88 ± 0.97	369	90.27 ± 2.07	288	
7	93.23 ± 1.05	415	91.58 ± 2.36	342	

TABLE 5: Selection experimental results on order K of Chebyshev convolution kernel.

amount of consumption of computing resources, *K* was set to 5 and 3 in the TEP dataset and FBFP dataset, respectively.

6. Conclusion

In this article, a graph feature fusion-driven fault diagnosis of complex process industry systems based on a multivariate heterogeneous method is proposed. First, the sensor layout of the process industrial system is transformed into a graph structure by distance measurement, and the connection between multisource heterogeneous signals that describe the same process is established. Then, the process knowledge graph is established by the similarity between the signals and refined by pretrained GCN layers. Furthermore, the multichannel graph feature fusion (MCGFF) model is proposed to mine fault representation from two different subgraphs and then fuse subgraph features into global-graph features through an attention mechanism for fault diagnosis. Two publicly available process chemistry datasets validate the effectiveness of the proposed method.

However, the neighbors of each node are equally important in this research, but this is not reasonable in practice. For the monitoring of key processes, there are often primary monitoring signals and auxiliary signals, which means that the edge connections between nodes should be weighted. To further improve the quality of the constructed graph, the information contained in the edge connection also needs to be considered and investigated in future work. At the same time, how to develop a general knowledge extraction learning framework for different complex systems also needs to be studied in the future.

Data Availability

The data that support the findings of this study are openly available at the following URL/DOI: http://depts.washington. edu/control/LARRY/TE/download.html and https://ww2. mathworks.cn/matlabcentral/fileexchange/49041-industrial-sc ale-penicillin-simulationv2.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China (nos. 52205104 and 72171096), Natural Science Foundation of Hubei Province (no. 2022CFB062), Opening/Innovation Foundation of Hubei Three Gorges Laboratory (no. SC215004), and Knowledge Innovation Program of Wuhan-Basic Research.

References

- Y. Jiang and S. Yin, "Recursive total principle component regression based fault detection and its application to vehicular cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1415–1423, 2018.
- [2] Z. Chen and Z. Ge, "Knowledge automation through graph mining, convolution and explanation framework: a soft sensor practice," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 9, pp. 6068–6078, 2022.
- [3] K. Feng, J. C. Ji, Q. Ni, Y. Li, W. Mao, and L. Liu, "A novel vibration-based prognostic scheme for gear health management in surface wear progression of the intelligent manufacturing system," *Wear*, vol. 522, Article ID 204697, 2023.
- [4] D. Chen, R. Liu, Q. Hu, and S. X. Ding, "Interaction-aware graph neural networks for fault diagnosis of complex industrial processes," *IEEE Transactions on Neural Networks* and Learning Systems, vol. 34, no. 9, pp. 6015–6028, 2023.
- [5] Z. Ge, B. Huang, and Z. Song, "Mixture semi-supervised principal component regression model and soft sensor application," *American Institute of Chemical Engineers Journal*, vol. 60, no. 2, pp. 533–545, 2014.
- [6] S. Joe Qin, "Recursive PLS algorithms for adaptive data modeling," *Computers and Chemical Engineering*, vol. 22, no. 4-5, pp. 503–514, 1998.
- [7] H. Kaneko and K. Funatsu, "Application of online support vector regression for soft sensors," *American Institute of Chemical Engineers Journal*, vol. 60, no. 2, pp. 600–612, 2014.
- [8] C. Liu, W. Hsaio, and Y. Tu, "Time series classification with multivariate convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4788–4797, 2019.
- [9] X. Yuan, L. Li, and Y. Wang, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3168–3176, 2020.
- [10] X. Yuan, B. Huang, Y. Wang, C. Yang, and W. Gui, "Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3235–3243, 2018.
- [11] D. Xie and L. Bai, "A hierarchical deep neural network for fault diagnosis on Tennessee-Eastman process," in 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), IEEE, Miami, FL, USA, December 2015.
- [12] K. Yan, Z. Ji, H. Lu, J. Huang, W. Shen, and Y. Xue, "Fast and accurate classification of time series data using extended ELM: application in fault diagnosis of air handling units," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 7, pp. 1349–1356, 2019.
- [13] F. Ye, Y. Guo, Z. Xia, Z. Zhang, and Y. Zhou, "Feature extraction and process monitoring of multi-channel data in a forging process via sensor fusion," *International Journal of Computer Integrated Manufacturing*, vol. 34, no. 1, pp. 95– 109, 2021.
- [14] Y. Xu, K. Feng, X. Yan et al., "CFCNN: a novel convolutional fusion framework for collaborative fault identification of

rotating machinery," *Information Fusion*, vol. 95, pp. 1–16, 2023.

- [15] D. Wu and J. Zhao, "Process topology convolutional network model for chemical process fault diagnosis," *Process Safety* and Environmental Protection, vol. 150, pp. 93–109, 2021.
- [16] V. Venkatasubramanian, "The promise of artificial intelligence in chemical engineering: is it here, finally?" *American Institute of Chemical Engineers Journal*, vol. 65, no. 2, pp. 466–478, 2019.
- [17] T. Bikmukhametov and J. Jäschke, "Combining machine learning and process engineering physics towards enhanced accuracy and explainability of data-driven models," *Computers and Chemical Engineering*, vol. 138, Article ID 106834, 2020.
- [18] Q. Ni, J. C. Ji, B. Halkon, K. Feng, and A. K. Nandi, "Physics-Informed Residual Network (PIResNet) for rolling element bearing fault diagnostics," *Mechanical Systems and Signal Processing*, vol. 200, Article ID 110544, 2023.
- [19] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.
- [20] K. Zhou, C. Yang, J. Liu, and Q. Xu, "Deep graph feature learning-based diagnosis approach for rotating machinery using multi-sensor data," *Journal of Intelligent Manufacturing*, vol. 34, no. 4, pp. 1965–1974, 2023.
- [21] J. Man, H. Dong, X. Yang et al., "GCG: graph Convolutional network and gated recurrent unit method for high-speed train axle temperature forecasting," *Mechanical Systems and Signal Processing*, vol. 163, Article ID 108102, 2022.
- [22] J. Liu, K. Zhou, C. Yang, and G. Lu, "Imbalanced fault diagnosis of rotating machinery using autoencoder-based SuperGraph feature learning," *Frontiers of Mechanical Engineering*, vol. 16, no. 4, pp. 829–839, 2021.
- [23] C. Yang, J. Liu, K. Zhou, and X. Jiang, "Semi-supervised machine fault diagnosis fusing unsupervised graph contrastive learning," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 8, pp. 8644–8653, 2023.
- [24] C. Yang, J. Liu, Q. Xu, and K. Zhou, "A generalized graph contrastive learning framework for few-shot machine fault diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 2692–2701, 2024.
- [25] F. Zhang, J. Liu, Y. Li, Y. Liu, M. F. Ge, and X. Jiang, "A health condition assessment and prediction method of Francis turbine units using heterogeneous signal fusion and graphdriven health benchmark model," *Engineering Applications of Artificial Intelligence*, vol. 126, Article ID 106974, 2023.
- [26] T. Li, Z. Zhou, S. Li, C. Sun, R. Yan, and X. Chen, "The emerging graph neural networks for intelligent fault diagnostics and prognostics: A guideline and a benchmark study," *Mechanical Systems and Signal Processing*, vol. 168, Article ID 108653, 2022.
- [27] Y. Wang, Z. Pan, X. Yuan, C. Yang, and W. Gui, "A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network," *ISA Transactions*, vol. 96, pp. 457–467, 2020.
- [28] Y. Zhang and J. Yu, "Pruning graph convolutional networkbased feature learning for fault diagnosis of industrial processes," *Journal of Process Control*, vol. 113, pp. 101–113, 2022.
- [29] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

- [30] T. Song, W. Zheng, P. Song, and Z. Cui, "EEG emotion recognition using dynamical graph convolutional neural networks," *IEEE Transactions on Affective Computing*, vol. 11, no. 3, pp. 532–541, 2020.
- [31] H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, "Knowledge graph convolutional networks for recommender systems," in *The world wide web conference*, pp. 3307–3313, San Francisco, CA, USA, May 2019.
- [32] S. Chaudhari, V. Mithal, G. Polatkan, and R. Ramanath, "An attentive survey of attention models," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 12, no. 5, pp. 1–32, 2021.
- [33] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers and Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.
- [34] N. Wang, H. Li, F. Wu, R. Zhang, and F. Gao, "Fault diagnosis of complex chemical processes using feature fusion of a convolutional network," *Industrial and Engineering Chemistry Research*, vol. 60, no. 5, pp. 2232–2248, 2021.
- [35] Y. Wang, D. Wu, and X. Yuan, "LDA-based deep transfer learning for fault diagnosis in industrial chemical processes," *Computers and Chemical Engineering*, vol. 140, Article ID 106964, 2020.
- [36] W. J. Park, S. H. Lee, W. K. Joo, and J. I. Song, "A mixed algorithm of PCA and LDA for fault diagnosis of induction motor," in Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence: Third International Conference on Intelligent Computing, ICIC 2007, pp. 934–942, Qingdao, China, August 2007.
- [37] F. Zhang, J. Liu, X. Lu et al., "Spatial weighted graph-driven fault diagnosis of complex process industry considering technological process flow," *Measurement Science and Technology*, vol. 34, no. 12, Article ID 125143, 2023.
- [38] S. Goldrick, C. A. Duran-Villalobos, K. Jankauskas, D. Lovett, S. S. Farid, and B. Lennox, "Modern day monitoring and control challenges outlined on an industrial-scale benchmark fermentation process," *Computers and Chemical Engineering*, vol. 130, Article ID 106471, 2019.