

一种面向安全生产过程的知识图谱故障检测模型

李志勇¹, 李景宇², 秦晓波³

(1. 河北省军民融合发展促进中心, 石家庄 050011;

2. 张家口市气象局, 河北 张家口 075000; 3. 河北省石家庄市气象局, 石家庄 050011)

摘要: 考虑安全生产过程的多样化和分布需求, 提出一种知识图谱故障检测模型; 该模型采用互信息和嵌入变量相似性构建安全生产过程知识图谱, 提出基于图注意力网络全局特征提取和双向长短期记忆网络的局部特征提取方法, 并设计了子块融合协同预测模型; 通过实验证明: 所提模型不仅具有良好的实际应用性, 检测精度和误报率优于现有基准方法, 还具有更有效的参数敏感性, 为安全生产过程故障检测提供了技术支持。

关键词: 安全生产过程; 知识图谱; 分布式故障检测; 图注意力网络; 双向长短期记忆网络

A Knowledge Graph Fault Detection Model for Safety Production Processes

LI Zhiyong¹, LI Jingyu², QIN Xiaobo³

(1. Hebei Provincial Military-Civilian Integration Development Promotion Center, Shijiazhuang 050011, China;

2. Zhangjiakou Meteorological Bureau, Zhangjiakou 075000, China;

3. Shijiazhuang Meteorological Bureau, Shijiazhuang 050011, China)

Abstract: To meet the diversity and distributed requirements of safety production processes, a fault detection based on knowledge graph is proposed, which adopts the mutual information (MI) and embedded variable similarity (VES) to construct a knowledge graph for safety production processes. A global feature extraction based on graph attention networks and a local feature extraction based on bidirectional long short-term memory (Bi-LSTM) networks are proposed, with a sub-block fusion collaborative prediction model and fault detection designed. Experimental results show that the proposed model not only has good practical applicability, but also outperforms existing benchmark methods in terms of detection accuracy and false alarm rate. Moreover, it has more effective parameter sensitivity, providing a technical support for fault detection in safety production processes.

Keywords: safety production processes; knowledge graph; distributed fault detection; graph attention network; Bi-LSTM network

0 引言

安全生产强调对生产过程的故障检测和诊断, 其通过最大限度地提高系统性能、优化维护策略, 确保安全生产过程的寿命和弹性^[1]。现有安全生产故障检测应用研究很大程度上依赖于对状态监测的数据分析^[2], 但由于安全生产过程由多个操作单元和数百个空间分布的控制回路组成^[3], 缺乏对故障检测结果解释和过程分解。当前大部分研究基于先验知识^[4]进行过程分解, 但先验

知识通常分布于不同生产过程系统中, 导致知识很难集成或融合, 同时安全生产过程的互连结构和耦合关系对于分布式检测模型构建至关重要, 其单一的建模方式容易忽略模型的局部信息。

随着知识图谱^[5]技术的深入应用, 基于知识驱动方法通过知识匹配定位故障问题并探索故障原因, 可以集成多源异构知识, 为故障检测提供有效解决方案。安全生产过程知识具有复杂的层次结构和关系, 可以生成过程分解、子块建模和决策融合^[6]。文献 [7] 构

收稿日期:2025-03-24; 修回日期:2025-05-07。

基金项目:河北省军民融合发展研究课题(HB24JMRH002)。

作者简介:李志勇(1977-),男,大学本科,工程师。

通讯作者:李景宇(1978-),男,大学本科,高级工程师。

引用格式:李志勇,李景宇,秦晓波.一种面向安全生产过程的知识图谱故障检测模型[J].计算机测量与控制,2026,34(4):10

建了一种基于有向知识图谱的压水堆故障诊断, 用于揭示故障传播路径和解释故障原因。文献 [8] 开发了一种具有双层池的知识驱动图注意力自动编码器, 很好地描述生产过程因果关系的关键层次特征, 用于故障检查和原因定位。但这些方法侧重于将安全生产过程数据视为孤立样本, 忽略了数据中的时间序列属性^[9]。文献 [10] 提出了一种知识驱动的时空图注意力神经网络, 用于解决动态和时空相关性导致的质量相关故障检测的不稳定性。但该方法没有充分考虑不同节点之间相关程度的差异性, 使在特征提取时难以获取复杂的特征依赖性并解释节点之间的特征交互^[11]。文献 [12] 开发了一种知识图谱引导的多源交互域自适应网络用于旋转机械故障诊断, 以促进知识在各个领域之间的传播和交互。文献 [13] 提出了一种基于知识图谱的增强型变压器网络复杂过程故障检测, 利用图注意力网络从时间和变量两个维度挖掘数据信息进行联合时空学习。但由于生产过程是连续行为或非线性发生, 对于每个故障检测子模型, 需要在知识图谱结构上提取非线性或顺序动态过程^[14]。

基于上述问题, 提出了一种知识图谱故障检测模型。该模型不仅可以融合领域专家知识和历史过程数据构建知识图谱, 还可以通过图注意力网络整合安全生产过程中变量的全局特征, 利用双向长短期记忆网络提取子块的局部特征, 从而提高故障检测性能。

1 框架设计

本文所述安全生产过程是一种针对军用一次性食品安全生产的过程, 涉及原料采购与存储、生产设备安全、生产过程中的危害控制、自热装置生产、环境保护与废弃物处理、人员安全培训、质量监控与检测等, 将这些过程构建为一个具有结构化形式化的语义知识网络, 形成安全生产过程知识图谱。该知识图谱的构建在传统建模基础上, 定义安全生产过程概念之间的相互关系, 它由头部实体、关系和尾部实体的三元组结构组成。其中, 头部实体和尾部实体表示设备的概念或对象, 关系表示头部和尾部实体之间的有向连接。将安全生产过程知识图谱定义为 3 个核心要素: 1) 节点。每个节点由一系列变量组成, 即每个节点包含一系列的传感器、传动器等设备介质, 可以概括为一个子过程, 所有节点集合形成一个知识图谱; 2) 头部实体和尾部实体。主要为生产过程中的变量, 这些变量是制造过程中发生的故障检测措施; 3) 关系。实体知识通过关系进行表示和连接, 通过领域专家明确的规则获取显性关系表示头部实体、关系、尾部实体, 利用互信息^[15] (MI, mutual information) 和变量嵌入相似性^[16] (VES, variable embedding similarity) 的方法对隐性关系进行识别,

增加实体和关系的可解释性。图 1 描述了一个具有 3 个节点的安全生产过程知识图谱。

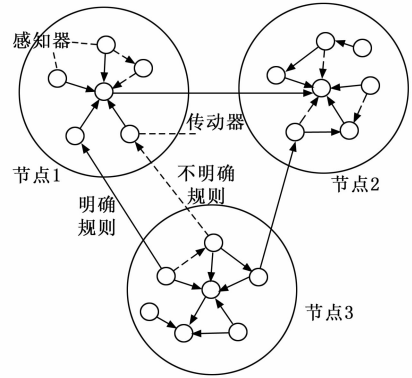


图 1 安全生产过程知识图谱描述

故障检测主要按照生产标准, 设置乳制品巴氏杀菌不足、果蔬加工农药残留、罐头杀菌釜压力失控、油炸食品过氧化值超标等故障类型。为实现故障检测, 如图 2 所示, 设计了一个分布式故障检测框架。该框架包括数据处理、全局特征提取、局部特征提取及故障检测 4 个模块。其中: 1) 数据预处理模块用于对原始数据进行归一化和窗口化处理; 2) 全局特征提取模块主要基于图注意力网络^[17] (GAN, graph attention network), 对邻近节点的特征进行加权求和, 通过图注意力网络节点间的关系, 动态确定每个邻居节点的重要性, 使模型在处理图数据时更具可解释性; 3) 局部特征提取模块主要基于双向长短期记忆网络^[18] (Bi-LSTM, bidirectional long short term memory network), 用于挖掘安全生产过程知识的潜在相关性特征, 通过结合正向和反向的 LSTM 层, 分别处理输入序列的正向和反向信息, 然后将这两个方向的输出合并, 以增强模型对上下文信

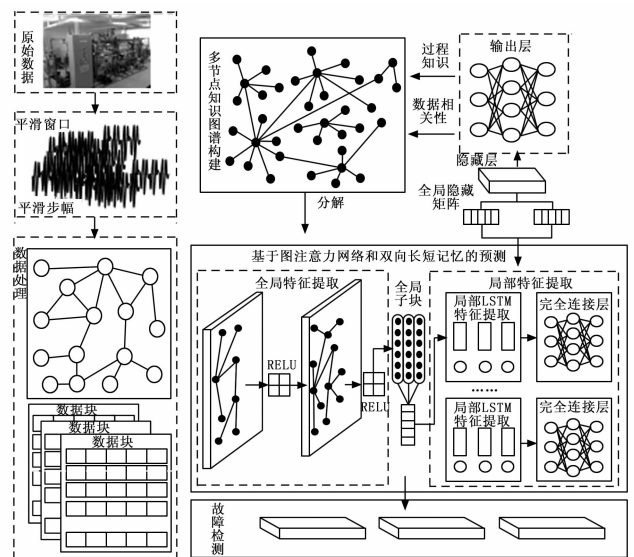


图 2 故障检测框架

息的捕捉能力；4) 故障检测模块用于确定在每个时刻是否发生故障，计算特征融合和结果预测。

2 知识图谱构建

2.1 数据预处理

对于安全生产过程中的原始数据，需要将原料安全检查数据、设备运转数据、化学添加剂使用数据、化学反应产生热量数据、废气废水污染物数据等多个变量过程进行表示，表示为 $DS \in \mathbb{R}^{M \times N}$ ，其中 N 为变量的数量， M 为时间戳的数量。为确保数据范围的一致性，采用最大-最小归一化方法^[19]计算每列每个元素相对于最小值和最大值之间范围的比例，然后将该比例投影至最终的归一化结果中，计算为：

$$X_{std} = \frac{X_{ini} - \min(X_{ini})}{\max(X_{ini}) - \min(X_{ini})} \quad (1)$$

$$X_{scaled} = X_{std} * [\max(X_{std}) - \min(X_{std})] + \min(X_{std}) \quad (2)$$

式中， $\min(X_{ini})$ 和 $\max(X_{ini})$ 分别为每列中识别的最小值和最大值， X_{std} 为标准化结果， X_{scaled} 为归一化结果。按照这一过程，归一化数据利用基于模式中值策略^[19]进行采样，其大小减少至原始数据量的 10%，从而减少个别异常值或噪声对整体数据的影响；然后对数据进行窗口化处理，不仅可以提高故障检测模型计算效率，还能提供较为准确的上下文信息。数据预处理的窗口切片定义为 $S_w \in \mathbb{R}^{S \times N}$ ， S_x 表示窗口长度，通过窗口化处理得到数据输入，计算为：

$$D = SPL(X_{scaled}) = \{x^{T_s+1}, x^{T_s+2}, \dots, x^{T_w}\} \quad (3)$$

式中， SPL 为分割操作， x^{T_j} 表示在时间戳 j 的输入。首先对数据进行拓扑化，为使模型更全面地获取数据信息，将变量本身的语义信息引入输入节点，设置节点向量为两部分拼接：第一部分为向量的时间序列，第二部分为随着模型训练而不断更新的语义向量。其次，将知识图谱中的关系初始化为完全相关，随着训练不断更新，对测试数据集进行相同的预处理。

2.2 关系识别

安全生产过程中每个过程节点之间蕴含着链接关系，如超高压灭菌和压缩成型之间的参数关系等，判断是否会产生对应的故障类型。根据安全生产过程知识图谱的核心要素，将节点定义为变量的集合，表示为：

$$Node = \{Node_i, M \times Q_j, i \in (1, 2, \dots, N_{num}), j \in (1, 2, \dots, B_{num})\} \quad (4)$$

式中， num 为节点的总数， Q_j 为节点 i 的特征维度数， B_{num} 为最终子系统数量。节点的划分是后续子系统分解的基础，变量之间的关系识别对于知识图谱构建至关重要，由于知识图谱只利用专家知识（如文本、表格）获得显式关系，大多数显性关系无法完全表示变量之间的所有相关性。因此，提出基于互信息 MI 和

嵌入变量相似性 VES 的隐性关系识别方法。MI 反映两个变量之间的强依赖性，利用这种关系作为隐性关系的筛选指标之一。对于两个变量 V_i 和 V_j ，其互信息 MI 计算为：

$$I_{ij} = I(V_i, V_j) = \sum_{v_i \in V_i} \sum_{v_j \in V_j} p(v_i, v_j) \log \left(\frac{p(v_i, v_j)}{p(v_i)p(v_j)} \right) \quad (5)$$

式中， $p(v_i, v_j)$ 表示变量的联合概率分布 V_i 和 V_j ， $p(v_i)$ 和 $p(v_j)$ 分别表示边缘概率分布，对于所有变量，计算互信息矩阵为：

$$I_M = \begin{bmatrix} I_{11} & I_{12} & \dots & I_{1N} \\ I_{21} & I_{22} & \dots & I_{2N} \\ \vdots & \vdots & \dots & \vdots \\ I_{N1} & I_{N2} & \dots & I_{NN} \end{bmatrix} \quad (6)$$

为平衡模型的操作效率与关系选择的合理性，为每个变量选择与互信息 MI 最相关的前 K 个变量作为关系筛选结果。因此，互信息 MI 筛选结果矩阵计算为：

$$I_{TopK} = K_{max}(I_M) \quad (7)$$

式中， K_{max} 表示取前 K 个最大值的操作，且 K 不大于 $N/2$ 。由于变量嵌入是持续训练性，其包含语义和数据信息，对于两个变量 V_i 和 V_j ，变量嵌入相似度 VES 计算为：

$$Sim_{ij} = Sim(E_{V_i}, E_{V_j}) = \frac{E_{V_i} \cdot E_{V_j}}{\|E_{V_i}\| * \|E_{V_j}\|} \quad (8)$$

式中， E_{V_i} 和 E_{V_j} 分别为 V_i 和 V_j 的变量嵌入， Sim_{ij} 为它们之间的相似度。对于所有变量，计算变量嵌入相似度 VES 矩阵为：

$$I_E = \begin{bmatrix} Sim_{11} & Sim_{12} & \dots & Sim_{1N} \\ Sim_{21} & Sim_{22} & \dots & Sim_{2N} \\ \vdots & \vdots & \dots & \vdots \\ Sim_{N1} & Sim_{N2} & \dots & Sim_{NN} \end{bmatrix} \quad (9)$$

与处理互信息 MI 矩阵的方式相同，为每个变量选择最相关的 P 个变量作为筛选结果，筛选结果矩阵为：

$$I_{TopP} = P_{max}(I_E) \quad (10)$$

式中， P_{max} 表示取前 P 个值的操作。通过公式 (7) 和 (10)，可以推导出隐式关系集合；然后通过融合显性和隐性关系，得到最终的关系集合。构建安全生产过程知识图谱如算法 1 所示。

算法 1：安全生产过程知识图谱构建

输入：实体节点变量 E_w ，前置知识 P_k ， Top_k 数量 K ， Top_p 数量 P

输出：安全生产过程知识图谱

for each 实体变量 E_w in entities do

通过公式(5)、(6)计算 MI 矩阵 I_M

通过公式(8)、(9)计算 VES 矩阵 I_E

$I_{TopK} = K_{max}(I_M)$

$$\mathbf{I}_{\text{Top}P} = K_{\max}(\mathbf{I}_E)$$

对于每个实体变量 E_w 由 P_k 明确定义规则

end for

return 安全生产过程知识图谱

算法 1 中, 首先, 通过遍历实体变量 E_w , 计算出互信息矩阵 \mathbf{I}_M 和变量嵌入相似矩阵 \mathbf{I}_E ; 然后, 计算每个变量选择与互信息 MI 最相关的前 K 个变量作为关系筛选结果 $\mathbf{I}_{\text{Top}K}$ 和为每个变量选择最相关的 P 个变量作为筛选结果 $\mathbf{I}_{\text{Top}P}$; 最后, 对于每个实体变量 E_w 由 P_k 明确定义规则, 从而形成具有头部实体、关系、尾部实体的安全生产过程知识图谱。

3 故障检测

3.1 特征提取

在构建了知识图谱后, 考虑变量之间的非线性关系, 比如超高压灭菌采用 HPP 技术灭活芽孢杆菌与残留菌落数之间, 需要挖掘数据的全局特征。因此, 提出一种基于图注意力网络 GAN 的全局特征提取方法, 将当前时刻变量 V_i 节点特征计算为:

$$\text{Fea}_{V_i}^T = \text{LeakyReLU}[\alpha_{V_i, V_j} * \mathbf{W}_5 * (\mathbf{x}_i^T \oplus E_{V_i}) + \sum_{j \in N_i} \alpha_{V_i, V_j} * \mathbf{W}_6 * (\mathbf{x}_j^T \oplus E_{V_j})] \quad (11)$$

式中, \mathbf{x}_i^T 表示当前时刻变量 V_i 的输入矩阵数据, \mathbf{x}_j^T 表示当前时刻变量 V_j 的输入数据, \mathbf{W}_5 和 \mathbf{W}_6 为可训练的矩阵。 α_{V_i, V_j} 为 V_i 和 V_j 的注意力特征权重, N_i 表示变量 V_i 的邻接变量集合, LeakyReLU 为激活函数用于更新中心节点变量特征。通过变量 V_i 和 V_j 的输入矩阵数据与 E_{V_i} 和 E_{V_j} 的嵌入矩阵数据进行异或运算, 其结果与可训练的矩阵、注意力权重数据相乘, 分别进行数据更新变量更新和求和, 最终得出当前时刻变量节点的特征值。对于每个 α_{V_i, V_j} 计算为:

$$\alpha_{V_i, V_j} = \frac{\exp[\text{LeakyReLU}(e_{V_i, V_j})]}{\sum_{V_i \in N_j} \exp[\text{LeakyReLU}(e_{V_i, V_j})]} \quad (12)$$

式中, 对于每个更新中心节点变量的特征, 其均通过节点与所有邻居节点的概率。注意力网络节点特征如图 3 所示, $V_1 \sim V_5$ 表示邻接节点变量, V_0 表示中心节点变量, V' 表示中心节点变量特征, 从而形成全局特征网络 $gNet$ 。

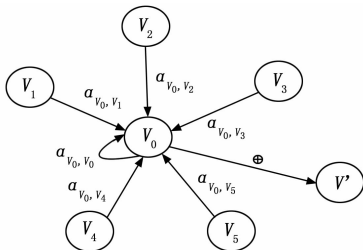


图 3 注意力网络节点变量特征

α_{V_i, V_j} 还可以通过可训练矩阵与输入矩阵数据和嵌入矩阵数据的异或运算进行计算:

$$\alpha_{V_i, V_j} = \mathbf{W}_7 \{ [\mathbf{W}_8 (\mathbf{x}_i^T \oplus E_{V_i})] \oplus \mathbf{W}_9 (\mathbf{x}_j^T \oplus E_{V_j}) \}, j \in E_i \quad (13)$$

式中, \mathbf{W}_7 、 \mathbf{W}_8 、 \mathbf{W}_9 为可训练矩阵, E_v 为嵌入变量矩阵, \oplus 表示连接操作 (异或运算)。

由于知识图谱包含丰富的变量关联关系, 直接使用节点变量进行全局特征获取较难获取其内在的特征关系。为此, 对于知识图谱中识别的节点 $Node$, 将其节点内部变量 V_i 可作为子块 i 变量的一部分, 而节点外部变量则通过以下规则确定:

$$R_{N_{m-i}, N_{e-j}} = \begin{cases} 1, \text{Num}_{N_{m-i}, N_{e-j}} > TH_r \\ 0, \text{Num}_{N_{m-i}, N_{e-j}} < TH_r \end{cases} \quad (14)$$

式中, N_{m-i} 和 N_{e-j} 分别为 $Node_i$ 的内部变量 V_i 和外部变量 V_j 。 $\text{Num}_{N_{m-i}, N_{e-j}}$ 为节点之间关系的重复次数, 1 表示变量之间存在关系。 $\text{Num}_{N_{m-i}, N_{e-j}}$ 通过叠加由变量嵌入相似性 VES 和互信息 MI 贡献的关系数量获得。 TH_r 是一个在优化搜索后获得的参数, 其大小不超过节点内关系数量的 30%, 否则会导致模型关系冗余。根据公式 (14) N_{num} 个节点被转换为 B_{num} 个子块, 每个子块包含的变量数量为 Q_i 。为使模型更多地关注子块内高相关性的变量, 对于每个子块 B_i , 构建基于双向长短期记忆网络 Bi-LSTM 的局部特征提取器, 子块 B_i 在时间步 T_i 的特征计算为:

$$\begin{aligned} \text{FeaBL}_{B_i}^T &= \text{sigmoid}(\mathbf{V}\mathbf{S}^T + \mathbf{V}'\mathbf{S}'^T) \\ \mathbf{S}^T &= \text{Tanh}(F_i \mathbf{x}^T + \mathbf{W}_{10} \mathbf{S}^{T-1}) \\ \mathbf{S}'^T &= \text{Tanh}(I_i \mathbf{x}^T + \mathbf{W}_{11} \mathbf{S}'^{T-1}) \end{aligned} \quad (15)$$

式中, \mathbf{W}_{10} 、 \mathbf{W}_{11} 为可训练矩阵, \mathbf{S}^T 和 \mathbf{S}'^T 分别为正向 LSTM 和反向 LSTM 单元在时刻 T_i 的隐藏层输出; F_i 为遗忘门, 用于调节前一个隐藏层的信息流, 输出越接近 0 表示被遗忘的数据越多; I_i 为保留门, 用于保留当前节点的信息。如图 4 所示, 描述了一个双向长短期记忆网络, 将当前 $t-1$ 时刻的节点特征矩阵 \mathbf{V}_{t-1} 输入至提取器中的输入门, 集成节点状态通过 \oplus 连接操作至输出门, 输出门中通过正反向 LSTM 输出隐藏层 H_i 状态, 并更新状态, 最终形成局部特征网络 lNet。

3.2 子块协同预测

为提高模型的准确性, 选择一个多层感知器作为输出层计算预测结果, 输出层将子块降级到初始节点的维度, 节点 $Node_i$ 在时间 T_i 的特征计算为:

$$\text{FeaFinal}_{\text{Node}_i}^T = \text{Tanh}(\mathbf{W}_{12} * \text{FeaBL}_{B_i}^T + B_5) \quad (16)$$

式中, \mathbf{W}_{12} 为一个可训练矩阵, B_5 为一个偏置项, 对于剩余 $B_{\text{num}-1}$ 个子块执行相同的操作, 以在时间 T_i 获得最终特征, 计算为:

$$\begin{aligned} \text{FeaFea}^T &= \text{Concat}(\text{FeaFea}_{B_i}^T), \\ i \in (1, \dots, B_{\text{num}}), \text{FeaFea}^T &\in \mathbb{R}^{1 * N} \end{aligned} \quad (17)$$

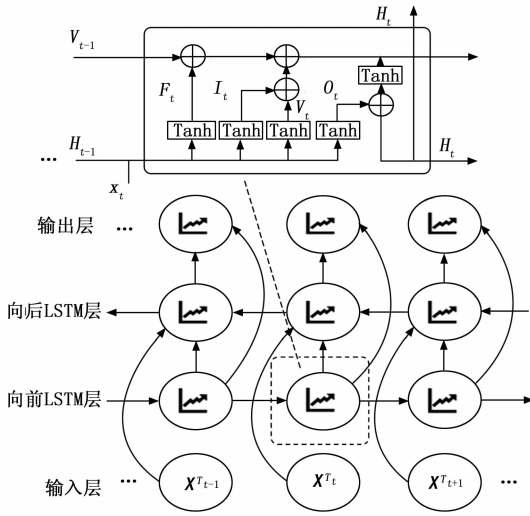


图 4 双向长记忆网络

激活后，获取时间点 T_i 所有变量的预测，形成融合网络 oNet，计算为：

$$Pre^T_i = LeakyReLU(FeaFea^T_i * E_{V_i}) \quad (18)$$

对于所提模型，损失函数通过预测值和真实值之间的平均误差来定义：

$$L = \frac{1}{N - s_z} \|Pre^T_i - x^T_i\|^2 \quad (19)$$

子块协同预测如算法 2 所示。

算法 2：子块协同预测

输入：训练数据集 $X_{train} = \{x^T_i, i \in (s_z + 1, s_z + 2, \dots, M)\}$ ，迭代次数 E_p ，全局特征网络 $gNet$ ，局部特征网络 $lNet$ ，输出融合网络 $oNet$

输出：训练后的 $gNet, lNet, oNet$

初始化参数 $gNet, lNet, oNet$

设 $n \leftarrow 1$

repeat

for x^T_i in X_{train} do

通过公式(12)计算全局特征 $h_{gr} = gNet(x^T_i)$

通过公式(15)计算局部特征 $h_{li}^i, i \in (1, 2, \dots, B_{min}) \leftarrow$

$lNet(h_{gr})$

通过公式(16)输出 $h_{oi} \leftarrow oNet(x^T_i), i \in (1, 2, \dots, B_{min})$

通过公式(19)计算损失，根据损失更新模型参数化 $gNet,$

$lNet, oNet$

end for

设 $n \leftarrow n + 1$

Until $n = E_p$

return 训练后的模型 $gNet, lNet, oNet$

3.3 检测算法

为实时进行检测故障，设计的故障检测流程如图 5 所示，对于训练后的模型数据，先提取整个过程的全局特征，将全局特征分解为子块局部特征，再进行特征融合和预测。通过计算每个时刻的每个变量计算异常指

数，从而反映异常程度，帮助确定故障。

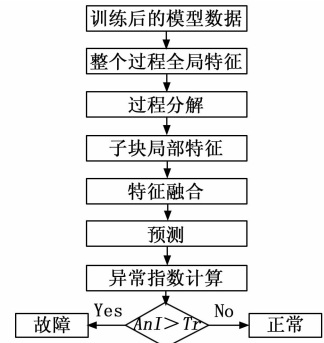


图 5 故障检测流程

具体来说，通过比较预测值和实际值来制定异常指数，由于每个变量的测量尺度不同，直接使用原始的异常指数可能会导致某个特定变量的异常值过大，可能导致模型错过真正异常的变量。为避免这种情况，异常指数需要进行平滑处理^[20]，计算为：

$$AnI^T_i = \left| \frac{(Pre^T_i - x^T_i) - Mean(Pre^T_i - x^T_i)}{Iqr(Pre^T_i - x^T_i)} \right| \quad (20)$$

式中， $AnI^T_i \in \mathbb{R}^{1 \times N}$ 为在特定时间点 T_i 时所有变量的异常指数， $Mean$ 表示预测值与实际值之间的平均差异， Iqr 表示四分位误差。平滑后的异常指数可以平等地反映每个变量的异常情况，并增强模型的鲁棒性，选择 AnI^T_i 在 T_i 时的最大值作为的异常指数：

$$Index^T_i = \max(AnI^T_i) \quad (21)$$

式中， \max 表示列的最大值。为了检测故障，需要确定正常和故障条件下的控制限，采用网格搜索策略来确定控制限，该策略可以系统地遍历多个参数组合，以找到给定评估标准的最优参数。这种方法优化了深度学习模型的参数，旨在在给定参数空间内定位最优参数。首先，定义控制限的搜索空间为：

$$Index^T_i \{i \in (1, 2, \dots, L_{test})\} \quad (22)$$

式中， L_{test} 表示测试数据的总长度。然后，遍历搜索空间以找到最优的控制限：

$$T_r = Optimize_{gd} \{Index^T_i, i \in (1, 2, \dots, L_{Test})\} \quad (23)$$

式中， $Optimize$ 表示从寻找最优值的操作。如果异常指数超过控制限，则认为发生了故障；反之，则认为没有发生故障。故障检测的计算为：

$$Fault^T_i = \begin{cases} 1, & Index^T_i > T_r \\ 0, & Index^T_i < T_r \end{cases} \quad (24)$$

算法 3：检测算法

输入：窗口长度 S_z ，训练后的 $gNet, lNet, oNet$ ，超参数 P_q ，训练数据集 $X_{train} = \{x^T_i, i \in (s_z + 1, s_z + 2, \dots, M)\}$ ，测试数据

集 $X_{test} = \{x^T_i, i \in (s_z + 1, s_z + 2, \dots, M)\}$

输出：故障检测结果 $Y = \{y^T_i, i \in (s_z + 1, s_z + 2, \dots, M)\}$

for x^T_i in X_{train} do

通过公式(12)计算全局特征 $h_{gr} = gNet(x^T_i)$

通过公式(15)计算局部特征 $h'_i, i \in (1, 2, \dots, B_{\text{num}}) \leftarrow \text{lNet}(h_{gr})$

通过公式(16)输出 $h_{gr} \leftarrow \text{oNet}(x^i), i \in (1, 2, \dots, B_{\text{num}})$

通过公式(18)计算变量预测 Pre^T

end for

设 $T_r \leftarrow \text{Grid_threshold_selection}(X_{\text{train}})$

设 $Y \leftarrow \Phi$

for x^T in X_{test} do

通过公式(12)计算全局特征 $h'_{gr} = \text{gNet}(x^T)$

通过公式(15)计算局部特征 $h'_i, i \in (1, 2, \dots, B_{\text{num}}) \leftarrow \text{lNet}(h'_{gr})$

通过公式(16)输出 $h_{gr}' \leftarrow \text{oNet}(x'_i), i \in (1, 2, \dots, B_{\text{num}})$

通过公式(18)计算变量预测 Pre^T

通过公式(20)进行平滑处理 AmT^T

通过公式(21)计算异常数据 $Index^T$

if $Index^T > T_r$ then $y^T \leftarrow 1$

Else $y^T \leftarrow 0$

end if

add y^T to Y

end for

Return Y

4 实验结果分析

4.1 实验设置

4.1.1 数据集

结合课题研究实际,以野战单兵自热压缩干粮生产线为具体对象,其安全生产流程有4个核心环节:

- 1) 原料预处理:高能配方混合(碳水化合物:蛋白质:脂肪=6:2:2)、水分活度控制($Aw \leq 0.6$)。
- 2) 超高压灭菌:采用HPP技术(600 MPa/5 min)灭活芽孢杆菌,残留菌落数 ≤ 10 CFU/g。
- 3) 压缩成型:四柱液压机(压力 ≥ 50 MPa)压制为25 mm \times 25 mm \times 5 mm模块,密度 ≥ 1.8 g/cm³。
- 4) 真空镀膜封装:铝塑复合膜氧透过率 ≤ 0.1 cm³/(m²·d),耐温范围-40~80 ℃。实验包含449,920个样本,测试集包含496800个样本,正常数据占比86%。在测试数据集中,有58000个故障点,共有微生物污染、化学污染类、物理异物混入、过程控制失效、包装完整性等5个类别和乳制品巴氏杀菌不足、果蔬加工农药残留、罐头杀菌釜压力失控、油炸食品过氧化值超标等26种故障类型。

模型训练采用ReLU激活函数与批归一化。训练阶段设置初始学习率为0.001,使用Adam优化器($\beta_1 = 0.9, \beta_2 = 0.999$)并配合余弦退火学习率调度策略($T_{\text{max}} = 50$)。训练周期(Epoch)为200。数据集按6:3:1比例划分为训练集、验证集与测试集,并采用分层采样确保子集类别分布一致性。所有实验均在NVIDIA Tesla V100 GPU上重复运行5次以消除随机

性,最终结果取均值 \pm 标准差形式报告。

4.1.2 评估指标

使用精确度(Precision)、召回率(Recall)、错误接受率(FAR)、标准度量(F_1)和准确率(Acc)评估模型的性能^[21]。其中, P 指标用于模型识别为正例的实例中真正是正例的比例; R 指标用于模型准确地确定为正例的真实正例的百分比,它作为模型在识别正例方面的能力的指标; FAR 指标用于为在实际为负的样本中,被错误地判定为正的样本中,被错误地判定为正的样本中, F_1 指标用于全面评估模型的性能,解决 P 和 R 之间的权衡; Acc 指标用于评估分类模型整体有效性,正确预测与所做预测整体范围之间的平衡。

4.1.3 基准模型

1) 主成分分析PCA^[22]通过识别数据中最具显著方差的正交轴来保留数据的基本信息,能够确定数据中正常波动的主要方向。

2) 自编码器AE^[23]获取正常数据中存在的模式,旨在学习数据的低维表示的模型。

3) 深度自编码器高斯混合模型^[24](DAGMM, deep autoencoder gaussian mixture model)使用深度AE学习数据的低维表示及其相关的重建误差,通过计算高斯混合模型中每个高斯分布的概率来为每个数据生成异常分数,然后利用这些信息进行异常检测。

4) 多变量异常检测生成对抗网络^[25](MAD-GAN, multivariate anomaly detection generative adversarial network)使用生成对抗网络识别不同时间尺度上的数据属性。

5) 多变量时间序列异常检测与图注意力网络^[26](MTAD-GAT, multivariate time series anomaly detection and graph attention network)将每个单一指标时间序列作为特征,然后使用并行图注意力网络层学习多个时间序列之间的时序和特征依赖性。

6) 融合稀疏自编码器和图网络^[23](FuSAGNet, fusion of sparse autoencoder and graph network)通过合并稀疏自编码器和图神经网络在高维时间序列中检测故障。

7) 节拍生成对抗网络^[25](BeatGAN, beat generation adversarial network)采用对抗学习重建主要是正常的时间序列数据。

4.1.4 参数设置

实验基于Python3.7框架,使用PyTorch1.7,实验服务器配备Intel Ice Lake Gold 3.24 @ 2.9 GHz处理器和NVIDIA 2090图形卡。

4.2 检测结果

如表1所示,描述了所提模型与7种基准模型在 P 、 R 、 FAR 、 F_1 、 Acc 评估指标的比较。表2描述了

所提模型与 7 种基准模型的训练周期数、每个周期的训练时间、总训练时间和测试时间，每个指标的最高值采用实线加粗进行标记，排序第二的参数采用下划线进行标记。平均值指数表示每个度量的平均值，增益指数表示所提模型与基准模型相比平均提高的幅度。

表 1 数据集上的故障检测结果 %

模型	评估指标				
	<i>P</i>	<i>R</i>	<i>FAR</i>	F_1	<i>Acc</i>
PCA	24	21	10	22	82
AE	73	53	3	61	92
DAGMM	28	70	25	39	74
MAD-GAN	95	69	0.5	80	96
MTAD-GAT	97	69	0.2	81	96
FuSAGNet	96	70	0.2	82	96
BeatGAN	84	67	0.2	75	94
所提模型	97	76	0.2	86	96
增益	28	16	5	24	8

所提模型在 *P*、*R*、*FAR*、 F_1 、*Acc* 评估指标上表现最佳，相比其他 7 种基准模型增益 *P* 提高了 28%，*R* 增加了 16%，*FAR* 降低了 5%， F_1 提高了 24%，*Acc* 提高了 8%。在基准模型中，PCA 模型由于捕获非线性数据特征的能力较差，仅获得了 22% 的 F_1 ，AE 模型由于其出色的非线性拟合能力，获得了 61% 的 F_1 。相比之下，DAGMM 模型在基准模型中获得排序第二高的 *R* 值，但其 *P* 值仅为 28%。主要是这些模型缺乏明确的时间信息映射，且仅依赖单一的 GRU，没有利用时间序列窗口；MTAD-GAT 模型采用注意力机制专注于特定数据模式，通过调整神经网络权重获取长期依赖特征，提供了接近所提模型的检测结果，但该模型仅利用局部窗口作为输入，导致只提取了全局特征，忽略了局部属性；MAD-GAN 模型作为另一种与性能相互对齐的模型，其鉴别器和生成器部分均使用 LSTM-RNN，为时间序列数据提供了有效的特征提取。然而，它忽略了变量的固有信息。所提模型融合了全局和局部特征，考虑了时间数据和变量的固有特征，从而实现最佳检测结果。FuSAGNet 模型在 *P* 和 *FAR* 指标上表现出与所提模型相近的性能指标，但在准确性方面略逊于所提模型，这种差异主要源于它仅依赖数据进行特征输入。

如表 2 所示，所提模型的总训练时间比 FuSAGNet 大约高出 20.88%，主要是所提模型将知识与数据相结合，不仅增强了其预测能力，还提高了 GNN 在有效提取和表达特征方面的实用性，特别是在稀疏潜在特征表示的环境下。图 6 描述了一个故障检测过程的表示效果，其异常指数均保持在控制限以下。总之，所提模型由于其捕获变量相关性和时间特征的能力，超越了所有

基准模型。

表 2 训练时间和测试时间比较

模型	评估指标			
	训练次数	每个训练时间/s	总训练时间/s	测试时间/s
PCA	—	—	0.18	0.04
AE	120	1.32	84.48	0.56
DAGMM	200	18.51	3 702	2.90
MAD-GAN	100	27.79	2 779	1.9
MTAD-GAT	30	103.92	3117.60	9.66
FuSAGNet	65	2.28	114.6	3.15
BeatGAN	11 002	0.28	38 010	2.59
所提模型	36	3.89	138.1	4.19
平均值	—	23.18	5 801.44	3.24
增益	—	19.22	5 602.72	1.18

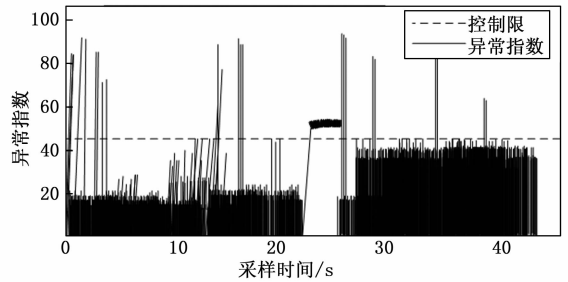


图 6 故障检测结果

4.3 消融实验

为验证所提模型中每个模块的作用，对所提模型进行消融研究分析。1) 实验 1：将双向 LSTM 改为单向 LSTM；2) 实验 2：移除双向 LSTM 模块；3) 实验 3：移除注意力网络，改用平均权重系数。

表 3 消融实验分析 %

方法	评估指标				
	<i>P</i>	<i>R</i>	<i>FAR</i>	F_1	<i>Acc</i>
所提模型	98	75	0.2	85	97
实验 1	97	62	0.3	76	95
实验 2	42	76	15	53	84
实验 3	91	72	0.9	80	95

表 3 描述了每个消融的实验结果，所提模型性能优于其他 3 种实验结果。具体来讲，当实验 1 双向 LSTM 转变为单向时， F_1 下降了 9%。这是由于移除单向 LSTM 后，单向 LSTM 只能从一个方向捕捉这种依赖关系，而双向 LSTM 通过正向和反向的传播，充分考虑到过去和未来的信息，更有效地捕捉长距离依赖关系，同时双向 LSTM 通过融合前后向信息，能够更好地理解每个位置的上下文，从而更准确地对序列进行建模。当实验 2 移除双向 LSTM 模块后， F_1 下降了 32%。这是由于移除所有 LSTM 后，所提模型只能简

单地根据当前节点和邻居节点的判断故障点信息, LSTM 网络的引入可以增强了上下文信息特征的提取, 提高了所提模型的预测准确性, 同时安全生产过程知识图谱分隔的子块, 允许对高度相关的变量给予极大的关注, 从而以更高的效率提取特征。当实验 3 移除注意力网络后, F_1 下降了 5%。这是由于移除图注意力网络 GAN 模块后, 所提模型无法对中心节点变量特征进行实时更新, 失去了根据输入数据的特征和上下文动态分配权重的能力。增加 GAN 后, 使所提模型能够有效地关注与当前任务相关的信息, 加入扩大了其提取数据总体特征的能力。

4.4 敏感性分析

为进一步验证所提模型的性能, 如图 7 所示, 进行了平滑步长大小、LSTM 层中隐藏向量维度、训练周期数、平滑窗口大小、变量嵌入维度等 5 个超参数在 F_1 、 P 、 R 评估指标的敏感性分析。如图 7 (a) 所示, 当平滑步长设置得太小时或太大时, 模型的准确率将降低, 这是由于平滑步长直接影响输入数据的量, 较小的步长可能导致数据重复, 而较大的步长可能导致大量数据丢失。这两种情况都影响了模型整体性能; 图 7 (b) 描述了模型对 LSTM 隐藏层维度大小的敏感性, 该参数决定模型特征提取能力, 较小的维度限制了模型的表达能力, 而较大的维度增加了复杂性, 可能导致过度拟合; 图 7 (c) 描述了训练周期设置不当可能导致模型过拟合和训练资源的浪费, 为避免这些问题, 所提模型选择 35 次训练周期。图 7 (d) 描述了模型对窗口大小变化的响应, 该参数是一个确定输入数据段维度, 对异常指数有一定影响, 一个小窗口有效地阻碍了上下文信息提取, 而一个大窗口延长了训练时间, 并可能掩盖较短的异常。因此, 所提模型选择了 15 作为窗口大小, 以在效率和准确性之间保持平衡; 图 7 (e) 描述了变量嵌入维度对模型的影响, 该参数控制全局特征向量的维度, 如果选择得过小, 可能会稀释嵌入权重, 从而影响变量的独特特征。为了避免这种情况, 在本文中将该维度设置为 256。

5 结束语

本文提出了一种面向安全生产过程的知识图谱故障检测模型, 该模型将知识图谱、图注意力网络、双向长

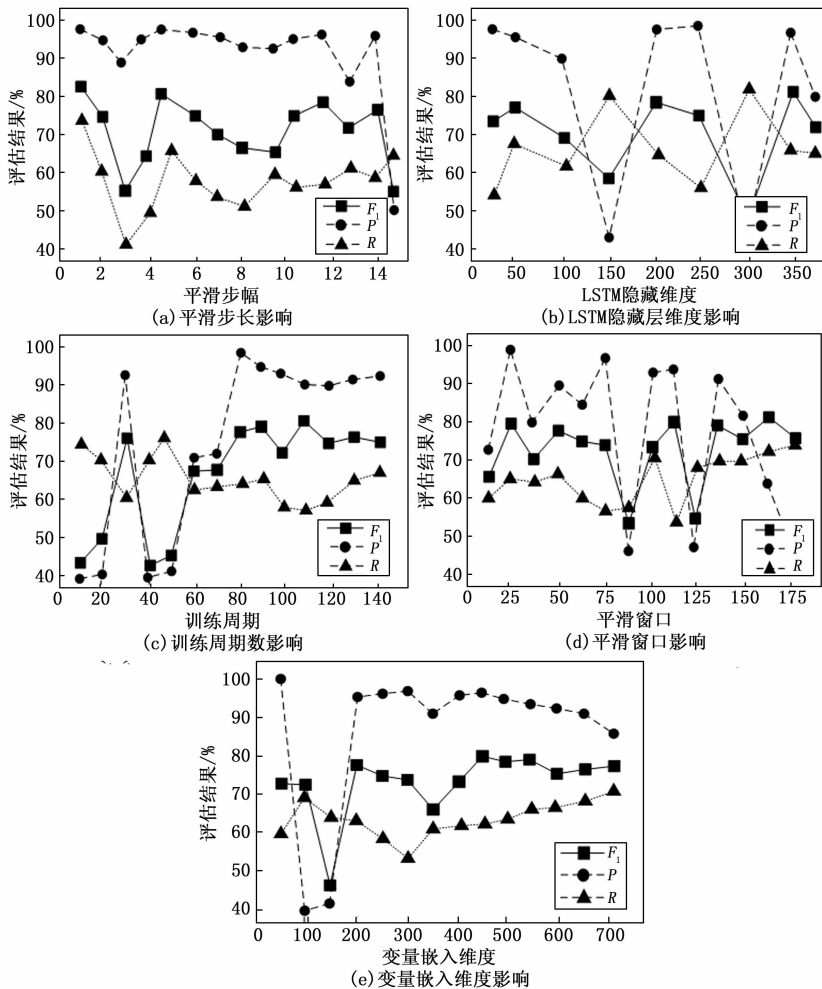


图 7 敏感性分析

短期记忆网络相融合, 旨在解决安全过程中故障的多样化和分布需求。在该模型中, 变量的全局特征通过图注意力网络从非线性数据中提取, 并通过知识图谱分解为子块; 变量的局部特征通过双向长短期记忆网络提取, 并通过异常指数进行故障检测。然而, 所提模型在离线建模阶段会消耗大量资源的局限性, 当设备新增故障类型时, 全量数据重新训练将产生高昂的时间与经济代价, 同时现有模型难以直接适配资源受限的边缘设备 (如嵌入式 PLC 或工业网关), 制约了实时故障诊断系统的落地。下一步, 一是设计轻量化模型网络, 引入知识蒸馏技术将现有模型压缩至 1/10 参数量级, 保留 95% 以上的原始精度将修改模型架构以加快训练速度; 二是部署边缘-云协同推理机制, 构建基于模型分片的层级化诊断系统, 将特征提取层部署至边缘端, 分类器置于云端, 以降低单点资源负载。

参考文献:

[1] 刘俊. 基于机器学习的工业过程故障诊断 [D]. 杭州: 浙江大学, 2020.

- [2] VELASCO C C, DE MAYA B, MOLINA C M, et al. Recent advancements in data-driven methodologies for the fault diagnosis and prognosis of marine systems: a systematic review [J]. *Ocean Engineering*, 2023, 284: 1152 - 1177.
- [3] ZHU Q X, WANG X W, LI K, et al. Enhanced multicorrelation block process monitoring and abnormality root cause analysis for distributed industrial process: a visual data-driven approach [J]. *Journal of Process Control*, 2022, 118: 1 - 15.
- [4] ZHANG J X, RANGAIAH G P, DONG L C, et al. An improved industrial fault diagnosis model by integrating enhanced variational mode decomposition with sparse process monitoring method [J]. *Reliability Engineering & System Safety*, 2025, 253: 1104 - 1192.
- [5] FAN L J, SUN Y Y, XU F, et al. Knowledge graph embedding based on semantic hierarchy [J]. *Cognitive Robotics*, 2022, 2: 147 - 154.
- [6] XIAO Y Z, ZHENG S, SHI J C, et al. Knowledge graph-based manufacturing process planning: a state-of-the-art review [J]. *Journal of Manufacturing Systems*, 2023, 70: 417 - 435.
- [7] MA Z G, DENG S G, ZHOU Z R, et al. Expert knowledge modelling software design based on signed directed graph with the application for PWR fault diagnosis [J]. *Annals of Nuclear Energy*, 2024, 196: 110 - 120.
- [8] LIU Y, XU Z, ZHAO J, et al. Hierarchical fault propagation path recognition method based on knowledge-driven graph attention autoencoder with bilayer pooling for large-scale industrial system [J]. *Advanced Engineering Informatics*, 2025, 63 (102): 9 - 30.
- [9] SHIRSHAHI A, MOSHIRI B, ALIYARI-SHOOREHDELI M. Identification of propagation path and root cause of faults based on generative adversarial networks in industrial systems [J]. *Process Safety and Environmental Protection*, 2024, 187: 1606 - 1617.
- [10] GUO L, SHI H, TAN S, et al. A knowledge-driven spatial-temporal graph neural network for quality-related fault detection [J]. *Process Safety and Environmental Protection*, 2024, 184: 1512 - 1524.
- [11] ZENG L, JIN Q, LIN Z, et al. Dual-attention LSTM autoencoder for fault detection in industrial complex dynamic processes [J]. *Process Safety and Environmental Protection*, 2024, 185: 1145 - 1159.
- [12] WU Z, JIANG H, WANG X, et al. Knowledge correlation graph-guided multi-source interaction domain adaptation network for rotating machinery fault diagnosis [J]. *ISA Transactions*, 2023, 142: 663 - 682.
- [13] CAO Y, TANG X, DENG X, et al. Fault detection of complicated processes based on an enhanced transformer network with graph attention mechanism [J]. *Process Safety and Environmental Protection*, 2024, 186: 783 - 797.
- [14] 钱金传. 基于深度学习贡献度的非线性过程故障诊断 [D]. 杭州: 浙江大学, 2022.
- [15] 郑静, 熊伟丽. 基于互信息的多块 k 近邻故障监测及诊断 [J]. *智能系统学报*, 2021, 16 (4): 717 - 728.
- [16] HAN Y, LIU J. Adaptive instance similarity embedding for online continual learning [J]. *Pattern Recognition*, 2024, 149: 11023 - 11032.
- [17] 李悦, 阴彦磊, 易斌. 融合双重注意力机制与 TCN 的复杂生产过程工艺质量精准预测 [J]. *计算机集成制造系统*, 2025, 31 (12): 4724 - 4738.
- [18] 高玉才, 付忠广, 王诗云, 等. 基于 Bi-LSTM 和自注意力机制的旋转机械故障诊断方法研究 [J]. *中国工程机械学报*, 2022, 20 (3): 273 - 278.
- [19] LIMA F T, SOUZA V M A. A large comparison of normalization methods on time series [J]. *Big Data Research*, 2023, 34: 10040 - 10047.
- [20] LIU F, WANG H, JIANG M, et al. Balance-blended adversarial distribution and smooth-suppressed labels refinement network for partial transfer fault diagnosis [J]. *Engineering Applications of Artificial Intelligence*, 2024, 135: 1087 - 1110.
- [21] 李文卿. 数据驱动的复杂工业过程统计过程监测 [D]. 杭州: 浙江大学, 2018.
- [22] ZHANG S, WANG S. Spectral radius-based interval principal component analysis (SR-IPCA) for fault detection in industrial processes with imprecise data [J]. *Journal of Process Control*, 2022, 114: 105 - 119.
- [23] MOKHTARI K E, MCARTHUR J J. Autoencoder-based fault detection using building automation system data [J]. *Advanced Engineering Informatics*, 2024, 62 (C): 1028 - 1038.
- [24] ZHAO Z, XU Y, ZHANG J, et al. A semi-supervised Gaussian mixture variational autoencoder method for few-shot fine-grained fault diagnosis [J]. *Neural Networks*, 2024, 178: 1064 - 1082.
- [25] SHIRSHAHI A, MOSHIRI B, ALIYARI S M. Identification of propagation path and root cause of faults based on generative adversarial networks in industrial systems [J]. *Process Safety and Environmental Protection*, 2024, 187: 1606 - 1617.
- [26] GOSWAMI U, RANI J, KODAMANA H, et al. A graph embedding based fault detection framework for process systems with multi-variate time-series datasets [J]. *Digital Chemical Engineering*, 2024, 10: 1001 - 1035.