计算机科学
COMPUTER SCIENCE

基于贝叶斯网的故障根因分析

刘华帅, 陶厚国, 岳昆, 段亮

引用本文

刘华帅, 陶厚国, 岳昆, 段亮. 基于贝叶斯网的故障根因分析[J]. 计算机科学, 2026, 53(3): 143-150.

LIU Huashuai, TAO Houguo, YUE Kun, DUAN Liang. Bayesian Network Based Fault Root Cause Analysis [J]. Computer Science, 2026, 53(3): 143-150.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.241200100

基于贝叶斯网的故障根因分析

刘华帅　陶厚国　岳　昆　段　亮
云南大学信息学院　昆明 650500
(avalon@mail.ynu.edu.cn)

摘　要　故障根因分析旨在找到导致特定问题、故障或事件发生的原因，是多个领域中追踪溯源的重要支撑技术，但现有方法在效率、准确性和稳定性等方面仍不能满足故障根因分析任务的实际需求。对此，将贝叶斯网作为相关属性之间依赖关系表示和推理的知识框架，提出基于贝叶斯网的故障根因分析方法。首先，针对高维数据和稀疏样本带来的挑战，提出基于向量量化自编码器的高维属性约简算法，并给出 α-BIC 评分准则，高效地学习根因贝叶斯网(Root Cause Bayesian Network，RCBN)。随后，基于贝叶斯网嵌入技术实现 RCBN 的高效推理，高效计算各原因条件下故障产生的可能性，进而使用因果模型中的 Blame 机制度量各原因对给定故障的贡献度，从而实现故障根因分析。在 3 个公共数据集和 3 个合成数据集上的实验结果表明，所提方法的平均检测准确性和效率明显优于对比方法，在 CHILD 数据集上精度提升了 7%，运行时间快了 60%。

关键词：故障根因分析；贝叶斯网；向量量化自编码器；贝叶斯信息准则；根因贡献度
中图分类号　TP391

Bayesian Network Based Fault Root Cause Analysis

LIU Huashuai，TAO Houguo，YUE Kun and DUAN Liang
School of Information Science and Engineering，Yunnan University，Kunming 650500，China

Abstract　Fault root cause analysis is to find the occurrence cause of specific problems，faults and events，becoming the important technique for origin tracing in several paradigms. However，existing methods still cannot satisfy practical requirements of efficiency，accuracy and stability. BN(Bayesian network) is used as the knowledge framework for representing and inferring the dependencies among relevant attributes. Specifically，the vector quantized variational autoencoder algorithm for attribute reduction is proposed at first. Then，the α-BIC scoring metric is adopted to learn RCBN efficiently. Following，efficient inferences in RCBN are implemented by BN embedding by calculating the probabilities of fault occurrence for given causes. Finally，the Blame mechanism in causal model is adopted to evaluate the contribution of causes w. r. t. given faults and fulfill fault root cause analysis. Experimental results on 3 public datasets and 3 synthetic datasets show that the average accuracy and efficiency of the proposed fault detection are better than current representative methods，such that the precision is 7% higher and the running time is 60% faster than the comparison methods.

Keywords　Fault root cause analysis，Bayesian network，Vector quantized variational autoencoder，Bayesian information criterion，Root cause contribution

1　引言

故障根因分析[1-2](Fault Root Cause Analysis)旨在找到导致特定问题、故障或事件发生的原因，是计算机信息安全保障和系统运维管理等实际应用中一项基础且具有挑战性的任务。围绕质量保障、成本管理、效率提升等目标，故障根因分析在化工工业、制造业、无线通信网络等智能运维领域发挥着重要作用。快速发现问题并准确进行故障定位，可有效提升运维质量的保障能力。尤其在故障诊断领域，无论是工业设备、软件系统还是移动基站等，故障往往会带来运行中断、设备损坏、经济损失，甚至安全风险[3]。因此，故障根因分析成为学术界和工业界关注的热点问题。

针对故障产生的可能原因，准确高效地度量其造成故障的贡献程度，进而为故障原因的定量分析提供参考，是准确定位故障根因的重要保证。然而，传统故障根因分析技术，包括基于聚类的方法[4]和基于分类的方法[5]，由于缺乏对根因贡献度的考虑，无法保证高效的根因贡献度计算，不能精准高效地完成实际中的故障根因分析任务。此外，基于启发式搜索

到稿日期：2024-12-13　返修日期：2025-03-08
基金项目：云南省重大科技专项(202202AD080001)；云南省“兴滇英才支持计划”青年人才项目(C6213001195)
This work was supported by the Major Project of Science and Technology of Yunnan Province(202202AD080001) and Xingdian Young Talent Program of Yunnan Province(C6213001195).
通信作者：段亮(duanl@ynu.edu.cn)

的方法[6]需根据不同数据设置合适的初值和启发式策略才能得到最优解，使得其在稳定性和准确性方面仍然存在不足。

贝叶斯网(Bayesian Network，BN)[7]能够定量地描述变量之间的依赖关系，因此本文使用 BN 来描述可能产生故障的各变量之间的依赖关系，为根因贡献度计算提供模型基础，称其为根因贝叶斯网(Root Cause Bayesian Network，RCBN)，并重点研究 RCBN 的学习和基于 RCBN 的根因算法。

故障诊断领域中的数据通常具有高维稀疏的特点，直接使用 BN 现有的学习方法会导致时间复杂度随属性维度增加呈指数增长[7]，且高维数据导致的样本稀疏性，使得现有学习方法难以准确提取属性间存在的依赖关系，从而无法保证根因贡献度计算的高效性和准确性。针对数据高维度的特点，本文考虑到不同属性所属的不同类别并将其作为约束，采用能够得到低维离散特征的向量量化变分自编码器(Vector Quantized Variational Auto-Encoder，VQVAE)[8]以降低数据的维度，提出基于类别约束的属性约简方法，通过向量量化技术和 VQVAE 的训练实现属性约简，将高维数据转换为离散型的低维表示，以保证 RCBN 学习的高效性。针对数据样本稀疏的问题，直接使用 BIC 评分计算的似然函数值可能存在较大误差，导致学习到不准确的数据分布，影响故障根因分析准确度。本文通过 Dirichlet 分布引入先验概率来改进 BIC 评分中似然函数的计算，提出一种新的评分准则(称为 α-BIC)，进而减小数据稀缺性带来的估计偏差，更有效地适应稀疏样本且更准确地表达变量间的依赖关系，最终确保了 RCBN 学习的准确性和稳定性。

针对使用条件概率来衡量故障原因的贡献度导致故障根因分析不准确的问题，引入 Blame 的概念[9](即责任度量)，用来刻画原因对结果事件贡献度的期望值，以定量分析不同原因对造成故障的贡献度。本文将故障根因与 Blame 的含义进行类比，提出基于 Blame 机制的故障根因分析算法。由于 Blame 值的计算涉及多次条件概率的计算，为保证故障原因贡献度计算的高效性，本文使用 Qi 等[10]提出的贝叶斯网嵌入算法，给出基于 RCBN 的概率推理算法，将 RCBN 映射到低维向量空间，基于嵌入向量实现条件概率的近似计算，进而快速计算各原因节点对给定故障的贡献度，实现故障根因的高效精准定位。

综上，本文的主要贡献如下：

1)针对 RCBN 学习存在的效率低下和稀疏样本的问题，提出基于类别约束的属性方法来提升学习效率，并提出 α-BIC 评分准则来准确表达产生故障的各变量之间的依赖关系；

2)针对条件概率难以有效衡量故障原因的贡献度，基于 Blame 思想提出了根因贡献度的计算方法，并将 RCBN 嵌入至低维向量，以实现贡献度的高效计算；

3)在真实世界数据集及合成数据集上进行对比实验和消融实验，结果表明，本文方法在效率和有效性方面明显优于其他对比方法。

2 相关工作

典型的故障根因分析方法主要包括基于聚类、基于分类、基于启发式和基于 BN 的方法。

基于聚类的方法利用基于距离的聚类挖掘算法，评估输入数据间或输入数据与参考数据之间的相关性，作为故障根因分析的依据。例如，Xue 等[4]采用 K 近邻算法实现两层宏基站-微基站异构网络的故障检测；Liu 等[11]通过计算变化分数对各个关键指标的异常程度进行标准化，使用 DBSCAN 密度聚类算法对异常的簇进行排序，实现故障根因分析。但这类方法的簇数量在实际应用中难以确定，无法发现真实的故障原因。

基于分类的方法将训练数据按故障原因分类，进而训练分类模型判断故障原因。基于径向基函数支持向量机和 Hilbert-Huang 变换等传统分类方法的故障诊断[12]，通过训练模型建立分类器来确定可能的故障根因。近年来，研究人员提出基于机器学习和深度神经网络模型的故障根因分析方法。例如，Wang[13]提出一种基于联邦学习的故障检测与诊断方法，能进行跨故障等级和跨系统故障检测与诊断；Liang[14]提出使用堆栈稀疏自编码器和 XGBoost 梯度提升树进行电力变压器故障诊断的方法；Jiang 等[15]提出警报传播图神经网络(Alarm Propagation Graph Neural Network，APGNN)，用于识别真实故障的警报模式；Yan 等[16]提出基于深度神经网络的跨度量多维度(Cross-Metric Multi-dimensional)根因分析方法，通过图神经网络和遗传算法，在检测到关键绩效指标异常后进行异常维度定位。然而，这类方法面临解释性和透明性差、需要大规模训练数据等问题，在应对高维数据和多种故障原因的场景时，决策空间的复杂度会显著提高，限制了其有效应用和泛化能力。

基于启发式的方法通过定义目标函数来评估故障原因的贡献度，进而找出产生故障的原因。例如，Ranjita 等[17]采用基于自回归移动平均模型(Auto Regressive Moving Average)的异常检测方法，将异常归因于维度及其对应指标；Liu 等[18]针对车联网问题，提出了一种基于航位推算的层次式的故障检测方法；Li 等[19]基于涟漪效应(Ripple Effect)并利用蒙特卡罗树启发式搜索，提出 Squeeze 方法以定位根因元素组合；Budhathoki 等[6]引入通过概率校准的信息论异常值分数，使用博弈论中的 Shapley 值识别异常值的原因。然而，这类方法需要根据不同数据设置合适的初值和启发式策略才能得到最优解，在实际的故障根因分析应用中缺乏稳定性。

基于 BN 的方法通过学习依赖关系并进行概率推理，以判断故障的产生原因。例如，Chen 等[20]通过评估节点影响，结合场景中的观测实例将其与 BN 集成，提出基于可达影响(Reachable Influence)的根因分析算法；Matsuo 等[21]针对罕见故障设备，提出由 BN 和扩展推理算法构成的根因诊断方法；Wee 等[22]通过从数据中学习 BN 来构建因果知识边缘模型，将 BN 的后验推理用于根因诊断；Wunderlich 等[23]提出基于 BN 的洪水预警根因方法。这些方法通过条件概率的大小判断故障原因，但与故障无关的原因也可能具有较高的条件概率，因此仅靠条件概率难以准确反映故障原因。

3 根因贝叶斯网学习

RCBN 学习是一个数据驱动的图模型构建过程，主要包

括结构学习和参数学习两个关键步骤。针对故障诊断数据中高维属性和稀疏样本带来的挑战，通过基于类别约束的属性约简方法对数据中的高维属性进行约简，进而基于约简后的数据学习 RCBN。针对稀疏样本的问题，提出 α-BIC 评分准则，用于 RCBN 中依赖关系的准确表达。

3.1 基于类别约束的属性约简

故障诊断数据通常为实际中设备或系统的历史运维数据，包含的属性往往涉及不同类别（即多个属性可能导致一类故障发生）。使用 $S=\{s_1,s_2,\cdots,s_n\}$ 表示故障诊断数据集 $\{U_1,U_2,\cdots,U_l\}$ 中的属性，共 n 个属性及 l 组数据，涉及 k 个类别 $\{C_1,C_2,\cdots,C_k\}$，$\{U_{1C_i},U_{2C_i},\cdots,U_{lC_i}\}$ 表示根据类别 $C_i(1\leqslant i\leqslant k)$ 划分后的故障诊断数据集。

VQVAE 模型由编码器、编码表和解码器 3 部分组成[8]。编码器负责从原始数据中提取特征，得到数据的潜在表示；编码表对这些提取的特征进行映射，找到与之最相似的向量特征；解码器负责重构数据，从简化的特征表示中恢复出原始数据。该过程结合了特征提取、特征映射和数据重构，使得 VQVAE 能在降维的同时保留关键信息和原始数据的类别约束。本节将类别约束引入 VQVAE 的训练，通过模型训练实现故障诊断数据中属性的约简。

将约简后的属性称为根因节点集 $\boldsymbol{Q}_q=\{Z_{C_1},Z_{C_2},\cdots,Z_{C_k}\}$，其中，$Z_{C_i}$ 代表约简后的根因节点。在模型训练时，首先对输入数据样本 U_{iC_i} 进行特征提取；然后通过编码器输出特征编码向量 $\boldsymbol{Z}_e$，并将 $\boldsymbol{Z}_e$ 输入编码表；最后通过最近邻搜索算法找到与之最相近的特征向量，从而得到离散的特征向量 $\boldsymbol{Z}_c$。计算式如下：

$$\boldsymbol{Z}_e=encode(U_{iC_i}) \tag{1}$$

$$\boldsymbol{Z}_c=e_m \tag{2}$$

$$m=\arg\min_j \|\boldsymbol{Z}_e-\boldsymbol{e}_j\|_2 \tag{3}$$

其中，$\|\cdot\|_2$ 表示欧几里德范数，$\boldsymbol{e}_j$ 表示编码表中的特征向量。

再通过最近邻搜索将 $\boldsymbol{Z}_c$ 输入解码器模块，进行数据重构，整个模型的损失函数如下：

$$L=\|\boldsymbol{X}-decode(\boldsymbol{Z}_e+sg(\boldsymbol{Z}_c-\boldsymbol{Z}_e))\|_2^2+\|sg(\boldsymbol{Z}_e)-\boldsymbol{Z}_c\|_2^2+\beta\|\boldsymbol{Z}_c-sg(\boldsymbol{Z}_e)\|_2^2 \tag{4}$$

$$sg(\boldsymbol{Z}_e)=\begin{cases}\boldsymbol{Z}_e, & \text{正向传播时}\\ 0, & \text{反向传播时}\end{cases} \tag{5}$$

式(4)中，第一项为重构损失；第二项为用于更新编码表的编码表损失；第三项为编码器的输出损失，使输出更接近于编码表；sg 表示梯度停止操作，β 为超参数($0.1\leqslant\beta\leqslant2.0$)。

基于 VQVAE 模型进行属性约简的关键在于向量量化模块。因此，本文利用训练好的 VQVAE 模型对故障诊断数据中的属性进行约减。首先，按类别顺序选择故障诊断数据集 U_{C_i}。然后，对于类别 C_i 和相应的数据 U_{iC_i}，通过编码器处理得到连续的特征编码向量 $\boldsymbol{Z}_e$，再将其输入编码表，用式(3)中的最近邻搜索算法得到离散的类别向量 $\boldsymbol{Z}_c$。重复执行这一过程 k 次，得到类别约束下约简后的数据集，进而生成根因节点集 $\boldsymbol{Q}_q=\{Z_{C_1},Z_{C_2},\cdots,Z_{C_k}\}$。以上思想如算法 1 所示。

算法 1 基于类别约束的属性约简

输入：故障诊断数据集 $U=\{U_1,U_2,\cdots,U_l\}$，类别数量 k，迭代次数 T

输出：约简后的数据集 $U_{OC_1},U_{OC_2},\cdots,U_{2C_k}$，根因节点集 $Q_q=\{Z_{C_1},Z_{C_2},\cdots,Z_{C_k}\}$

1. 根据所属类别生成 $U_{C_i}=\{U_{1C_i},U_{2C_i},\cdots,U_{2C_i}\}$
2. 初始化 VQVAE 模型
3. for t=1 to T do
4. for i=1 to k do
5. $\mathbf{Z}_e\leftarrow$encode(U_{iC_j}) // 生成连续特征编码向量
6. $\mathbf{Z}_c\leftarrow e_m$// 生成离散特征向量
7. 将离散特征向量$\mathbf{Z}_c$输入解码器进行数据重构
8. 使用式(4)的损失函数进行模型训练
9. end for
10. end for
11. for each U_i in U do
12. for i=1 to k do
13. 根据式(1)计算U_i在C_j类别的连续特征向量$\mathbf{Z}_e$
14. 根据式(2)将$\mathbf{Z}_e$转换为离散类别向量$\mathbf{Z}_c$
15. end for
16. end for
17. return $U_{OC_1},U_{OC_2},\cdots,U_{OC_k}$ Q_q

3.2 基于 α-BIC 的根因贝叶斯网学习

根据 BN 学习的一般步骤[7]，RCBN 的学习以用于描述故障诊断数据中的各属性之间依赖关系的有向无环图（Directed Acyclic Graph，DAG）构建为核心，即选择与数据拟合程度最高的模型。传统的 BIC 准则[7]通过衡量模型对数据的拟合程度与模型自身复杂度之间的平衡，以评估并选择最优模型。对于含有 n 个随机变量的 BN 结构 $\mathcal{B}$ 和 l 个样本数据 D 的情形，m_{ijk} 为给定父节点 X_j 下 $X_i=k$ 的样本数量，q_i 为父节点的组合数，r_i 为当前节点 X_i 的取值个数。BIC 的计算式如下：

$$BIC(\mathcal{B}|D)=\sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i}m_{ijk}\log\frac{m_{ijk}}{m_{ij*}}-\sum_{i=1}^{n}\frac{q_i(r_i-1)}{2}\log l \tag{6}$$

然而，BIC 准则在样本稀疏时的惩罚过于严厉，导致模型倾向于选择更简单的结构，这忽略了数据中的依赖关系，导致欠拟合。为了更好地表达稀疏样本中变量之间的依赖关系，进而构建支持根因贡献度计算的 RCBN，基于 Dirichlet 分布拟合多项式分布参数的先验分布，扩展 BIC，提出了 α-BIC 评分函数，其计算式如下：

$$S(G_c)=\sigma(\sum_{i=1}^{n}\sum_{j=1}^{p_i}\sum_{w=1}^{r_i}m_{ijk}\log\frac{m_{ijw}}{m_{ij*}}-\frac{\log l}{2}\sum_{i=1}^{n}p_i(r_i-1))+(1-\sigma)(\sum_{i=1}^{n+1}\sum_{j=1}^{p_i}(\log\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+m_{ij})}\sum_{w=1}^{r_i}\log\frac{\Gamma(\alpha_{ijw}+m_{ijw})}{\Gamma(\alpha_{ijw})})+\log P(G_c)) \tag{7}$$

其中，$\sigma(0\leqslant\sigma\leqslant1)$ 为调整因子；n 为 RCBN 中节点的数量；p_i 为节点 Z_{C_i} 父节点的取值组合数；r_i 为节点 Z_{C_i} 取值组合的数目；m_{ijw} 为故障诊断数据集中所有满足 $Z_{C_i}=\mathcal{H}$ 和 $\pi(Z_{C_i})=j$ 的样本数，且 $m_{ij}=\sum_{\mathcal{H}=1}^{r_i}m_{ij\mathcal{H}}$；$\Gamma$ 代表 Gamma 函数；α_{ijw} 表示 Dirichlet 分布中的超参数取值且 $\alpha_{ij}=\sum_{w=1}^{r_i}\alpha_{ijw}$；$P(G_c)$ 为 G_c 的先验概率。

针对根因节点集 $\boldsymbol{Q}_q=\{Z_{C_1},Z_{C_2},\cdots,Z_{C_k}\}$ 和故障诊断数据

集$\{U_{OC_1}, U_{OC_2}, \cdots, U_{OC_k}\}$，采用基于 α-BIC 评分准则和爬山法进行 RCBN 最优结构的搜索。初始时，RCBN 为无边的图结构，同时也是当前最优结构；从当前的最优结构开始，在每一步通过单次加边、减边、反转边操作生成一组新的候选结构，按式(6)计算各候选结构的评分，将评分最高者作为新的最优结构。重复上述过程直至评分不再增加，从而得到最优的 RCBN 结构 G_R。最后采用最大似然估计(Maximum Likelihood Estimation)算法，计算 RCBN 中每个节点 Z_{C_i} 的条件概率参数 $\theta^*_{\mathcal{H}_{i,r_i}|q_{i,m}}$：

$$\theta^*_{\mathcal{H}_{i,r_i}|q_{i,m}} = P(Z_{C_i} = \mathcal{H}_{i,r_i} | \pi(Z_{C_i}) = q_{i,m}) = \frac{\#(Z_{C_i} = \mathcal{H}_{i,r_i}, \pi(Z_{C_i}) = q_{i,m})}{\#(\pi(Z_{C_i}) = q_{i,m})} \tag{8}$$

其中，$q_{i,m}$ 为节点 Z_{C_i} 的父节点集 $\pi(Z_{C_i})$ 中的第 m 个取值组合，$m = 1, 2, \cdots, M_i$，M_i 为父节点集 $\pi(Z_{C_i})$ 中节点取值组合数，$\#(\cdot)$ 为满足条件约束的样本数。以上思想如算法 2 所示。

算法 2 基于 α-BIC 的 RCBN 学习

输入：根因节点集 Q_q，总迭代次数 T，故障诊断数据集 $\{U_{OC_1}, U_{OC_2}, \cdots, U_{OC_k}\}$

输出：最优 RCBN 结构 G_R 和条件概率分布参数 θ

1. 初始化 $\mathcal{B}$ 为以 Q_q 为节点集的无边图
2. for t=1 to T do
3. 对 $\mathcal{B}$ 进行加边、减边、反转边操作，生成候选结构
4. 将 α-BIC 评分最高的候选结构作为当前最优结构 G_R
5. end for
6. for i=1 to k do
7. $\theta^*_{\mathcal{H}_{i,r_i}|q_{i,m}} \leftarrow \frac{\#(Z_{C_i} = \mathcal{H}_{i,r_i}, \pi(Z_{C_i}) = q_{i,m})}{\#(\pi(Z_{C_i}) = q_{i,m})}$ //式(7)
8. end for
9. return G_R, θ

4 基于根因贝叶斯网的故障根因分析

针对多次条件概率计算开销过大的问题，即多次基于 RCBN 的概率推理，本章给出基于嵌入向量的条件概率近似计算方法，实现高效的故障根因分析。针对直接使用条件概率衡量根因贡献度不准确的问题，给出基于 Blame 机制的根因贡献度计算方法，实现精确的故障根因分析。

4.1 根因贝叶斯网的概率推理

故障往往由多种原因造成，例如，在包含多个服务器、网络设备和存储系统的数据中心，造成服务器崩溃的原因可能是网络延迟、硬件故障或软件错误等。为此，需基于构建的 RCBN 计算不同故障原因的可能性，即在给定证据变量 Z_{C_e} 取值为 err 的条件下多次计算查询变量 Z_{C_q} 取值为 q 的条件概率 $P(Z_{C_q} = q | Z_{C_e} = err)$。为了避免多次概率推理过程中大量中间结果的重复计算，本文基于 BN 嵌入的思想，通过计算嵌入向量之间的相似度，近似计算变量间的条件概率[9]。具体地，基于 RCBN 的嵌入向量，使用如下欧氏距离衡量节点嵌入向量之间的相似度：

$$S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i\mathcal{H}^b}) = \left(\sum_{\ell=1}^{d} (\boldsymbol{y}^{\ell}_{i\mathcal{H}} - \boldsymbol{y}^{\ell}_{b_i\mathcal{H}^b})^2\right)^{\frac{1}{2}} \tag{9}$$

其中，$\boldsymbol{y}_{i\mathcal{H}}$ 和 $\boldsymbol{y}_{b_i\mathcal{H}^b}$ 分别表示 RCBN 中的节点 Z_{C_i} 取值为 $\mathcal{H}$、其邻居节点取值为 $\mathcal{H}^b$ 的嵌入向量，d 为嵌入向量的维数，ℓ 表示嵌入向量的第 ℓ 个分量。

节点嵌入向量之间的距离越短，它们之间的关联性越强。基于式(8)，对于每个节点 Z_{C_i} 的 N_i 个邻居，可得到与其所有邻居节点之间的平均相似度：

$$S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i})_{\text{avg}} = \frac{S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i\mathcal{H}^1}) + \cdots + S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i\mathcal{H}^{N_i}})}{N_i} \tag{10}$$

其中，$\boldsymbol{y}_{b_i}$ 代表 $\boldsymbol{y}_{b_i\mathcal{H}^1}, \cdots, \boldsymbol{y}_{b_i\mathcal{H}^{N_i}}$，即将每个邻居节点可能取值的嵌入向量作为输入。

之后，使用 $S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i})_{\text{avg}}$ 与节点 Z_{C_i} 及其邻居所有不同取值情形下的平均相似度之和的比值，作为节点 Z_{C_i} 取值为 $\mathcal{H}$ 的边缘概率，计算式如下：

$$P(Z_{C_i} = \mathcal{H}) = \frac{S(\boldsymbol{y}_{i\mathcal{H}}, \boldsymbol{y}_{b_i})_{\text{avg}}}{\sum_{Z_{C_i}\text{所有取值}\mathcal{H}'} \sum_{Z_{C_i}\text{邻居所有取值}\mathcal{H}'} S(\boldsymbol{y}_{i\mathcal{H}^x}, \boldsymbol{y}_{b_i\mathcal{H}^y})_{\text{avg}}} \tag{11}$$

基于式(11)的边缘概率计算方法，可高效实现条件概率 $P(Z_{C_q} = q | Z_{C_e} = e)$ 的多次计算，为故障根因分析奠定定量计算的基础。

4.2 根因贡献度计算

本节使用因果模型中的 Blame 机制[10]来定量计算具有依赖关系的各节点对故障的贡献度。直观地，若 RCBN 中一个节点取值的变化影响相应的条件概率值，则该节点对故障发生应具有较大的影响。若 Z_{C_i} 可能取值为 x 和 y，则分别计算节点 Z_{C_i} 在取值为 x 和 y 时条件概率值 $P(Z_{C_i} | Z_{C_e})$，之后通过式(12)衡量二者之间的差异，记为 $vd(Z_{C_i})$：

$$vd(Z_{C_i}) = |P(Z_{C_i} = x | Z_{C_e}) - P(Z_{C_i} = y | Z_{C_e})| \tag{12}$$

若节点 Z_{C_i} 对故障发生的影响高于 Z_{C_j}，则 Z_{C_i} 值的改变将使 $vd(Z_{C_i})$ 的概率高于 Z_{C_j} 值的改变时 $vd(Z_{C_i})$ 的概率。基于这一性质，给出责任度(Responsibility Degree)的计算式，用于度量节点 Z_{C_i} 取值为 x 时对故障发生的影响程度：

$$dr(Z_{C_i} = x) = \lceil \max\{vd(Z_{C_i})\} \rceil \times \frac{P(Z_{C_i} = x | Z_{C_e})}{P(Z_{C_i} = x | Z_{C_e}) + P(Z_{C_i} = y | Z_{C_e})} \tag{13}$$

式(12)和式(13)通过 RCBN 的多次概率推理得到每个节点在特定状态下对故障的影响。将 Z_{C_i} 在所有可能取值下对故障责任度的期望值称为 Z_{C_i} 对故障的贡献度，记为 $db(Z_{C_i})$，其大小反映了 Z_{C_i} 作为故障原因的可能性：贡献度越高，作为故障根因的可能性越大。$db(Z_{C_i})$ 的计算式如下：

$$db(Z_{C_i}) = \frac{\sum_{i=1}^{r_i} (P(Z_{C_i} = x | Z_{C_e}) * dr(Z_{C_i} = x))}{\sum_{i=1}^{r_i} P(Z_{C_i} = x | Z_{C_e})} \tag{14}$$

式(14)综合考虑了节点 Z_{C_i} 在各种可能取值情形下对故障的影响。为了有效定位故障原因，将 RCBN 中的所有节点按照其 db 值从高到低排序，根据故障根因分析实际需求，选取排名靠前的节点作为分析结果，也可以进一步设置贡献度的阈值选取满足需求的原因节点。上述思想如算法 3 所示。

算法 3 基于 Blame 机制的故障根因分析

输入：RCBN 中的根因节点集 $Q_q = \{Z_{C_1}, Z_{C_2}, \cdots, Z_{C_k}\}$，$Z_{C_i}$ 的取值 x_i，给定证据节点 Z_{C_e} 下的查询节点 Z_{C_q}

输出：原因节点集 J

1. J←∅ // 初始化按贡献度降序排序的原因节点

2. for each Z_{C_i} in Q_q do

3. $y_i \leftarrow \bar{x}_i$ //以布尔型原因节点为例

4. $Vd(Z_{C_i}) \leftarrow |P(Z_{C_i}=x_i|Z_{C_e})-P(Z_{C_i}=y_i|Z_{C_e})|$

5. $dr(Z_{C_i}=x_i) \leftarrow \lceil \max\{vd(Z_{C_i})\} \rceil \times \frac{P(Z_{C_i}=x_i|Z_{C_e})}{P(Z_{C_i}=x_i|Z_{C_e})+P(Z_{C_i}=y_i|Z_{C_e})}$

6. $db(Z_{C_i}) \leftarrow \frac{\sum_{i=1}^{r_i}(P(Z_{C_i}=x_i|Z_{C_e}) * dr(Z_{C_i}=x_i))}{\sum_{i=1}^{r_i}P(Z_{C_i}=x_i|Z_{C_e})}$

7. end for

8. for i=1 to k do

9. J←J∪{Z_{C_i}} //按节点Z_{C_i}的贡献度放入 J

10. end for

11. 依据节点Z_{C_i}对应 db 值对 J 进行降序排序

12. return J

5 实验及结果分析

5.1 实验设置

5.1.1 数据集

本文实验采用 3 个公共数据集(CWRU，TELECOMMUNICATION，HCV)和 3 个合成数据集(CHILD，HAILFINDE 和 PATHFINDER)作为测试数据集。

CWRU 是公开的轴承故障检测数据集；TELECOMMUNICATION 是在 2022 年 ICASSP 论文中公开的根因分析数据集[24]；HCV 是在 UCL 数据库中公开的丙型肝炎患者诊断和根因分析的公共数据集。CHILD，HAILFINDE 和 PATHFINDER 这 3 个真实世界的 BN 作为测试数据集，分别代表规模较小、中等和较大的 BN，具体特征和参数如表 1 所列。

表 1 合成数据集统计信息

Table 1 Statistics of synthetic datasets

数据集	节点数	边数	参数个数
CHILD	20	25	230
HAILFINDER	56	66	2656
PATHFINDER	109	195	72079

5.1.2 评价指标

为了测试本文方法的有效性和高效性，使用精确度(Precision)、召回率(Recall)、F1 分数、平均倒数排名(Mean reciprocal rank，MRR)、归一化折损累计增益(Normalized Discounted cumulative gain，NDCG)和执行时间作为评价指标。其中，精确度、召回率和 F1 分数为准确性指标；MRR 是对排序列表进行评价的指标，主要衡量针对给定查询返回排序结果的质量；NDCG 是用于评估排序模型性能的指标，旨在考虑排名的影响。

5.1.3 对比算法

KNN[25]：一种非参数分类方法，通过 k 个最近邻居的投票对样本进行分类。

CatBoost[26]：梯度提升算法的开源实现，可处理分类变量，通过信息增益比构建决策树，且已证明该算法优于其他梯度增强决策树算法。

Trans-FL[27]：基于 Transformer 和 Focal Loss 的故障根因分析方法，通过 Transformer 学习不同根因概率，通过 Focal Loss 增加稳定性。

Squeeze[19]：基于广义波纹效应，将潜在的异常维度值组合分组到聚类中，然后基于聚类中的广义潜在得分寻找原因。

CMMD[5]：由两个关键步骤组成，一是关系建模，利用图神经网络对历史数据中度量之间的未知复杂计算和维度之间的聚合函数进行建模；另一个是根因定位，采用遗传算法，在检测到 KPI 异常后，高效地深入原始数据并定位异常维度。

BN-BI[20]：结合 BN 评估节点的影响，并使用场景中的观测实例将其与 BN 集成，最后基于 BN 中故障节点的后验概率进行根因定位。

5.1.4 实验环境

本文实验环境为：Intel i5-13500H CPU，RTX 2080Ti GPU，16 GB 内存，Windows 11 操作系统，编程语言为 Python 3.9。

5.2 实验结果

5.2.1 有效性测试

1)不同方法的故障根因分析结果

为了测试本文方法的有效性，将本文方法与 6 个对比方法在 3 个公共数据集上进行故障根因分析结果比较，结果如表 2 所列，其中最佳结果和次优结果分别用加粗和下划线标出。

表 2 公共数据集上的根因分析结果

Table 2 Comparison of root cause analysis results on public datasets

方法	CWRU			TELECOMMUNICATION			HCV		
	Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*	*Precision*	*Recall*	*F1*
KNN	0.30	0.64	0.41	0.73	0.51	0.60	0.89	0.85	0.87
CatBoost	0.67	0.53	0.59	0.81	0.58	0.66	0.98	0.93	0.95
Squeeze	0.56	0.64	0.60	0.82	0.56	0.67	0.96	0.90	0.93
CMMD	0.41	0.47	0.44	0.86	0.66	0.75	0.96	0.93	0.94
BN-BI	0.68	0.62	0.65	0.84	0.73	0.78	0.97	0.91	0.94
Trans-FL	0.59	0.58	0.58	0.87	0.71	0.78	0.97	0.94	0.95
RCBN	**0.76**	**0.71**	**0.73**	**0.89**	**0.77**	**0.83**	**0.99**	**0.95**	**0.97**

可以看出，本文方法在 3 个数据集上的 3 个指标均优于对比方法。具体而言，在 CWRU 数据集上，本文方法的各项指标比 KNN，CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 平均高出 39%；在 TELECOMMUNICATION 数据集上，本文方法的各项指标比 6 个对比方法平均高出 18%；在 HCV 数据集上，本文方法的各项指标比 6 个对比方法平均高出 4%。上述结果表明，本文提出的故障根因分析方法可以有效地发现数据中导致故障发生的原因。

为了进一步测试本文方法的有效性，将本文方法与 6 个对比方法，以及基于 BIC 评分准则使用本文的故障根因分析方法

(记为 BN-BIC)在 3 个不同规模的合成数据集上进行故障根因分析结果的比较。首先,利用前向采样算法从 3 个 BN 上分别采样出规模为 1000 的数据集。然后,随机选择故障变量,并将其父节点作为真实原因,把数据集中除去故障变量取值外的部分作为特征数据集。最后,将特征数据集和故障变量作为 6 个对比方法的输入,并对所输出的故障变量与其他变量的特征重要性进行降序排序,将取值最大的特征重要性所对应的变量作为预测的原因。真实原因和预测原因的比较结果如表 3 所列。

为了测试本文方法在实际应用中的可行性,由于故障根因分析结果通常包括多个潜在的原因,本节进一步在 3 个合成数据集上测试 MRR 和 NDCG 指标。采用与前述在公共数据集上相同的测试方法,其中故障变量的多个父节点被认为是多个潜在的根因,真实原因和预测原因的比较结果如表 3 和表 4 所列。

表 3 合成数据集上的根因分析结果比较

Table 3 Comparison of root cause analysis results on synthetic datasets

方法	CWRU			TELECOMMUNICATION			HCV		
	Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*	*Precision*	*Recall*	*F1*
KNN	0.90	0.46	0.61	0.34	0.42	0.38	0.49	0.48	0.48
CatBoost	0.91	0.69	0.78	0.39	0.46	0.42	0.51	0.51	0.51
Squeeze	0.91	0.45	0.60	0.64	0.71	0.67	0.32	0.43	0.37
CMMD	0.90	0.45	0.60	0.72	0.66	0.69	0.73	0.74	0.73
BN-BI	0.93	0.78	0.85	0.83	0.75	0.79	0.82	0.70	0.76
Trans-FL	0.91	0.46	0.61	0.68	0.63	0.65	0.75	0.71	0.73
RCBN	**0.98**	**0.88**	**0.93**	**0.97**	**0.84**	**0.90**	**0.91**	**0.82**	**0.86**

表 4 合成数据集上 MRR 和 NDCG 指标的比较

Table 4 Comparison of MRR and NDCG on synthetic datasets

方法	CHILD		HAILFINDER		PATHFINDER	
	MRR	*NDCG*	*MRR*	*NDCG*	*MRR*	*NDCG*
KNN	0.50	0.36	0.42	0.33	0.36	0.25
CatBoost	0.65	0.53	0.61	0.49	0.47	0.43
Squeeze	0.43	0.47	0.52	0.28	0.39	0.41
CMMD	0.61	0.43	0.57	0.41	0.52	0.35
BN-BI	0.73	0.67	0.63	0.52	0.60	0.55
BN-BIC	0.77	0.69	0.75	0.59	0.69	0.63
Trans-FL	0.67	0.51	0.73	0.62	0.72	0.66
RCBN	**0.82**	**0.75**	**0.77**	**0.63**	**0.73**	**0.67**

从表 3 和表 4 可以看出:

(1)在 3 个合成数据集上,本文方法的精确度、召回率和 F1 分数均优于其他对比方法。

具体地,本文方法在 CHILD 上的各项指标比 6 个对比方法平均高出 39%,在 HAILFINDER 上的各项指标比 6 个对比方法平均高出约 62%,在 PATHFINDER 上的各项指标比 4 个对比方法平均高出约 56%。上述结果表明,本文提出的故障根因分析方法能够准确发现故障的原因。

(2)在 3 个合成数据集上,本文方法的 MRR 和 NDCG 指标均优于 6 个对比方法。具体地,本文方法在 CHILD 上的各项指标比 6 个对比方法平均高出 43%;在 HAILFINDER 上的各项指标比 6 个对比方平均高出约 39%;在 PATHFINDER 上的各项指标比 6 个对比方法平均高出约 51%。以上结果表明,本文提出的故障根因分析方法能够准确发现实际应用中存在的多个根因。

(3)在 3 个合成数据集上,本文方法的各项指标均优于 BN-BIC。具体地,在 CHILD,HAILFINDER 和 PATHFINDER 上,本文方法的各项指标比 BN-BIC 方法平均高出 7.59%,4.72%和 6.07%。上述结果表明,本文提出的 α-BIC 评分准则能够克服基于 BIC 评分准则的故障根因分析方法在稀疏训练样本下易出现欠拟合的问题,从而准确地从稀疏数据中发现故障原因。

2)变量规模对故障根因分析结果的影响

为了测试本文方法对变量规模的可扩展性,基于 3 个合成数据集中变量规模最大的 PATHFINDER,将其变量规模分别设置为 20,40,60,80 和 100,利用前向采样算法从这 5 个生成的合成数据集上分别采样出规模为 1000 的训练集。在这 5 个生成的合成数据集上进行故障根因分析的结果如图 1 所示。可以看出,本文方法的各项指标均优于对比方法。此外,随着变量规模增加,本文方法的各项指标均无明显下降,而 KNN,CatBoost,Squeeze,CMMD,BN-BI 和 Trans-FL 的各项指标分别有 13.16%~24.89%,20.33%~29.49%,23.76%~56.12%,7.96%~18.57%,3.46%~14.42%和 4.22%~15.00%的下降。上述结果表明,本文提出的故障根因分析方法对变量规模的变化具有稳定性。

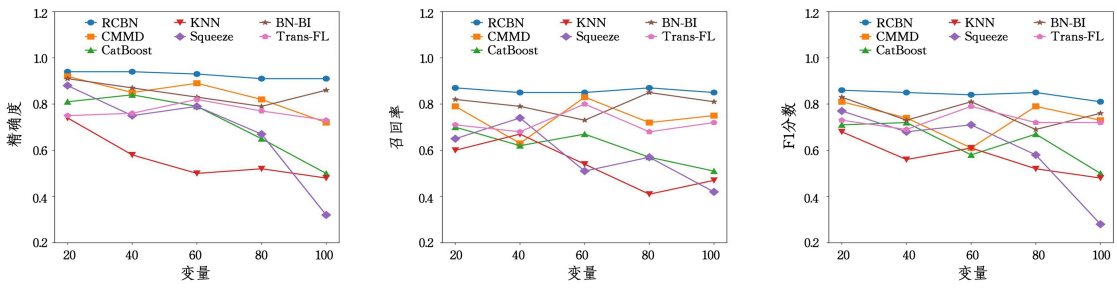

图 1 不同变量规模的根因分析结果比较

Fig. 1 Comparison of root cause analysis results with various scales of variables

5.2.2 效率测试

为了测试本文方法的效率，本小节在 6 个数据集上对本文方法从两个方面进行测试：不同故障根因分析方法的执行时间，以及变量规模对故障根因分析执行时间的影响。

1）不同故障根因分析方法的效率比较

为了测试本文方法的效率，在 3 个公共数据集和 3 个合成数据集上将本文方法与 KNN，CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 方法的故障根因分析时间进行了比较，结果如图 2 所示。可以看出，本文方法在 CWRU，HCV，HAILFINDER 和 PATHFINDER 这 4 个数据集上的故障根因分析时间最短。本文方法在 CWRU 上比 6 个对比方法分别快 40%，680%，300%，110%，120%和 180%；在 HCV 上比 6 个对比方法分别快 17%，79%，96%，110%，53%和 160%；在 HAILFINDER 上比 6 个对比方法分别快 39%，190%，140%，130%，100%倍和 750%；在 PATHFINDER 上比 6 个对比方法分别快 56%，110%，350%，540%，170%和 500%。

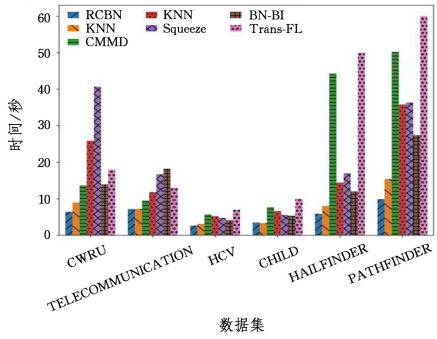

图 2 不同数据集上根因分析方法效率比较

Fig. 2 Comparison of efficiency of root cause analysis methods on various datasets

此外，在 CHILD 和 TELECOMMUNICATION 这 2 个数据集上，本文方法的故障根因分析时间比 CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 少。具体而言，在 CHILD 上，本文方法比 CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 分别快 69%，97%，130%，60%和 180%；在 TELECOMMUNICATION 上，比 KNN，CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 分别快 0.6%，110%，50%，20%，130%和 80%。需要说明的是，虽然本文方法在 CHILD 上的故障根因分析时间比 KNN 慢 4.6%，但在这个数据集上的有效性测试结果中，本文方法的各项指标比 KNN 平均高出 29.67%和 26.00%。上述结果表明，本文方法能在不同领域、不同规模的数据集上实现高效的故障根因分析。

2）变量规模对故障根因分析效率的影响

为了测试变量规模对故障根因分析效率的影响，将根据 PATHFINDER 生成的 5 个变量规模分别为 20，40，60，80 和 100 的合成数据集以及对应的训练集作为输入，不同故障根因分析方法的执行时间如图 3 所示。可以看出，所有方法的执行时间随着变量规模增加均呈线性增长，但本文方法的时间均短于其他方法。具体地，本文方法的时间比 KNN，CatBoost，Squeeze，CMMD，BN-BI 和 Trans-FL 平均快 27%，210%，200%，270%，88%和 350%。上述结果表明，本文方法在变量规模增加时依然可以保持故障根因分析的高效性。

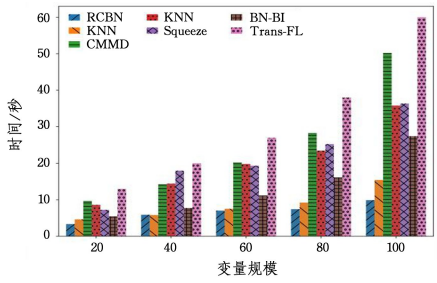

图 3 不同变量规模下的根因分析方法的效率比较

Fig. 3 Comparison of efficiency of root cause analysis methods with various scales of variables

结束语 本文围绕事件追踪溯源的实际需求，研究数据和知识驱动的故障根因分析方法，从故障诊断数据的特点和故障根因分析的实际需求出发，以 BN 作为数据中各属性之间依赖关系表示和推理的基本框架，借鉴因果模型中的 Blame 机制，提出基于概率推理的责任度和贡献度的概念和定量计算方法，实现高效、准确、稳定的故障根因分析，实验测试结果展示了本文方法的良好性能。

本文方法通过深度神经网络的训练实现高维属性的约简，一定程度上保证了故障根因分析的高效性；在约简后的属性集和有标注的数据集上学习 BN，忽略了故障根因分析任务所涉及属性及数据的局部性。如何有效表示故障根因分析任务、考虑故障根因分析的实时性需求，将根因任务的表示和局部数据选取相结合，研究更高效的故障根因分析方法，是目前我们正在研究的问题。此外，如何发挥基于 BN 推理结果不断完善模型的优点，根据运维数据随时间推移和系统运行产生的变化，对模型进行增量更新、对数据进行局部清洗和质量提升管理，也是未来要开展的工作。

参 考 文 献

[1] CHENG Y，WANG L，ZHAO X Y. A Review of Root Cause Analysis Research [J]. Computer Application Research，2023，40(4)：961-966.

[2] WANG L，ZHANG C Y，DING R M，et al. Root cause analysis for microservice systems via hierarchical reinforcement learning from human feedback [C] // Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York：ACM，2023：5116-5125.

[3] JIA T，LI Y，WU Z H. A review of fault diagnosis in distributed software systems based on log data [J]. Journal of Software，2020，31(7)：1997-2018.

[4] XUE W，PENG M，MA Y，et al. Classification－based approach for cell outage detection in self-healing heterogeneous networks [C] // Proceedings of the IEEE Wireless Communications and Networking Conference. Piscataway，NJ：IEEE，2014：2822-2826.

[5] YANG S，SHAN C，YANG W，et al. CMMD：Cross-metric multi-dimensional root cause analysis [C] // Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York：ACM，2022：4310-4320.

[6] BUDHATHOKI K, MINORICS L, BLOBAUM P, et al. Causal structure-based root cause analysis of outliers [C]// Proceedings of 39th International Conference on Machine Learning. New York: ACM, 2022: 2357-2369.

[7] ZHANG L W, GUO H P. An introduction to Bayesian networks [M]. Beijing: Science Press, 2006: 30-192.

[8] VAN DEN OORD A, VINVALS O. Neural discrete representation learning [C]// Proceedings of the 31st Advances in Neural Information Processing Systems. Massachusetts: MIT Press, 2017: 6306-6315.

[9] CHOCKLER H, HALPERN J Y. Responsibility and Blame: a structural-model approach [J]. Journal of Artificial Intelligence Research, 2004, 22: 93-115.

[10] QI Z W, YUE K, DUAN L, et al. Matrix factorization based Bayesian network embedding for efficient probabilistic inferences [J]. Expert Systems With Applications, 2020, 169: 114294.

[11] LIU P, CHEN Y, NIE X, et al. Fluxrank: A widely-deployable framework to automatically localizing root cause machines for software service failure mitigation [C]// Proceedings of the 30th International Symposium on Software Reliability Engineering. New York: IEEE, 2019: 35-46.

[12] CUNHA P, RODRIGO H, GOEDTEL A, et al. A comprehensive evaluation of intelligent classifiers for fault identification in three-phase induction Motors [J]. Electric Power Systems Research, 2015, 127: 249-258.

[13] WANG X, YAN K. Fault Detection and Diagnosis of HVAC System Based on Federated Learning [J]. Computer Science, 2022, 49(12): 74-80.

[14] LIANG H Y. Fault Diagnosis of Power Transformer Based on Stacked Sparse Autoencoder and XGBoost [J]. Journal of Chongqing Technology and Business University(Natural Science Edition), 2024(6): 65-71.

[15] JIANG W B, BAI Y B. APGNN: Alarm propagation graph neural network for fault detection and alarm root cause analysis [J]. Computer Networks, 2023, 220: 322-327.

[16] YAN S, SHAN C, YANG W, et al. CMMD: Cross-Metric Multi-Dimensional Root Cause Analysis [C]// Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining(KDD). 2022: 4310-4320.

[17] RANJITA B, RAHUL K, RAMACHANDRAN R, et al. Adtributor: Revenue debugging in advertising systems [C]// Proceedings of the 11th USENIX Symposium on Networked Systems Design and Implementation. 2014: 43-55.

[18] LIU J X, WU N, DING F. Fault Detection Based on Dead Reckoning in VANETs [J]. Computer Science, 2022, 49(12): 319-325.

[19] LI Z, LUO C, ZHAO Y, et al. Generic and robust localization of multi-dimensional root causes [C]// Proceedings of the 30th International Symposium on Software Reliability Engineering(ISSRE). 2019: 47-57.

[20] CHEN B, LI J, WEI J. A graph-based algorithm for root cause analysis of faults in telecommunication networks [C]// Proceedings of the 19th International Conference on Automation Science and Engineering. 2023: 1-7.

[21] MATSUO Y, NAKANO Y, WATANABE A, et al. Root-cause diagnosis for rare failures using Bayesian network with dynamic modification [C]// Proceedings of the IEEE International Conference on Communications. 2018: 1-6.

[22] WEE Y Y, CHEAH W P, TAN S C, et al. A method for root cause analysis with a Bayesian belief network and fuzzy cognitive map [J]. Expert Systems with Applications, 2015, 42(1): 468-487.

[23] WUNDERLICH P, NIGGEMANN O. Structure learning methods for Bayesian networks to reduce alarm floods by identifying the root cause [C]// Proceedings of the 22nd IEEE International Conference on Emerging Technologies and Factory Automation. 2017: 1-8.

[24] ZHANG T, CHEN Q, JIANG Y, et al. Root cause analysis for wireless network fault localization [C]// Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. 2022: 9301-9305.

[25] ZHANG Y, GAO G, WANG B, et al. A novel ensemble method for k-earest neighbor [J]. Pattern Recognition, 2019, 85: 13-25.

[26] ROKHORENKOVA L, GUSEV G, VOROBEV A, et al. CatBoost: Unbiased boosting with categorical features [C]// Proceedings of the 32nd Advances in Neural Information Processing System. 2018: 6638-6648.

[27] LYU Z, LIU Y, WANG X, et al. A knowledge-enhanced Transformer-FL method for fault root cause localization [C]// Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. 2024: 1607-1616.

LIU Huashuai, born in 2003, postgraduate. His main research interest is data and knowledge engineering.

DUAN Liang, born in 1986, Ph.D, associate professor, is a member of CCF (No. 95258M). His main research interests include graph analysis and knowledge engineering.

(责任编辑:何杨)