融合边缘节点特征的图卷积网络安全异常检测方法

陈琦昊 [1]
CHEN Qihao

摘　要　为应对传统异常检测模型在边缘计算环境中存在的特征稀疏、标签失衡与误报高的问题，文章提出一种融合边缘节点多模态特征的图卷积网络检测方法。通过本地轻量代理采集 CPU 微架构事件、容器 cgroup 指标与近场流量特征，经非负矩阵分解与词向量编码生成低维表示，并构建动态图卷积模型实现跨节点聚合与异常评分。模型采用双层 GCN 结构，结合联邦学习与差分隐私机制，确保部署时效与数据合规性。实验在 MEC 平台构建的 40 节点 Kubernetes 集群中验证，相较 IForest、SVM-RBF 等基线方法，所提模型在 F_1-score、MTTD 等指标上显著提升，具备优异的检测性能与边缘可部署性，为大规模异构边缘环境中的智能安全防护提供新路径。

关键词　边缘计算；图卷积网络；异常检测；联邦学习；多模态特征

doi：10.3969/j.issn.1672-9528.2026.07.019

0 引言

5G 与工业互联网深度融合推动边缘计算节点快速增长，传统“终端采集 - 云端检测”模式难以利用边缘侧丰富资源指标，导致异常检测模型特征维度受限、标签稀缺，难以准确识别潜在威胁。图神经网络虽在集中式环境中表现出色，但其对节点属性同质、边缘信息静态的依赖限制了在边缘场景下的泛化能力 [1]。为解决上述问题，本文面向边 - 云协同场景，提出融合多模态边缘资源指纹与动态图结构的图卷积检测方法，实现数据本地预处理、中心侧信息聚合及边缘实时响应的闭环架构，以提升模型对低频攻击与结构扰动的鲁棒性。

1 融合边缘节点特征的背景与整体思路

1.1 研究背景与问题归纳

随着 5G 与工业互联网的深度融合，云计算能力持续下沉，城域基站、工厂网关及车载终端等异构边缘节点呈现爆发式增长。传统异常检测框架沿用“终端采集 - 云端集中”范式，在端 - 边 - 云纵向割裂的架构下，终端仅上传粗粒度统计指标，边缘侧丰富的微观行为数据被丢弃，导致训练集特征稀疏 [2]；同时，攻击样本在百万级节点中占比极低，云端难以获得均衡标签，检测模型极易陷入高误报与漏报并存的困境。现有图卷积网络研究多聚焦数据中心级拓扑，节点属性局限于流量矩阵或 SNMP 计数，对边缘侧独有的 CPU 微架构事件、容器运行时 cgroup 指标及近场短包模式等资源指纹关注不足，造成图节点属性同质化，无法刻画边缘场景下因硬件差异与瞬时负载扰动引起的细粒度异常。

为此，本文提出在边缘本地完成轻量级特征提取，利用轻量编码器将多维资源指纹压缩为低维向量，仅上传匿名化表征；中心侧基于动态异构图卷积完成跨域信息聚合，实时输出异常评分并回注边缘，实现“本地感知 - 全局推理 - 边缘响应”的闭环检测目标，从而在保护数据隐私的同时显著提升对未知威胁的同步定位能力。

1.2 整体技术路线与安全假设

技术路线遵循“边缘提炼 - 中心聚合 - 边缘响应”三阶段闭环。第一阶段，轻量级代理嵌入 Kubelet 插件框架，对原始 cgroup、perf 事件及近场流量矩阵进行滑动窗口统计与非负矩阵分解，生成 40 维节点特征向量，经 AES-256 信道周期性上报，原始日志留存本地，满足 GDPR 最小可用原则 [3]。第二阶段，中心侧接收向量后构建动态异构图，顶点集合包括容器、微服务及外部 IP 三元组，有向边依据 5 元组会话与 DNS 调用链实时建立，边权重融合时间衰减因子与 Jaccard 相似度，图快照按 30 s 粒度滚动更新，以捕捉横向移动与短时突变。第三阶段，异常评分采用双层 GCN 架构：本地 GCN 在边缘 ARM 核上运行 2 层聚合，输出压缩隐式向量；全局 GCN 于云端 GPU 集群执行 3 层卷积，生成异常概率并回注边缘，代理依据阈值立即触发阻断或限流，实现毫秒级闭环处置 [4]。安全假设限定：边缘代理基线镜像经 TPM 度量可信；控制面与数据面通道采用 mTLS 及 AES-GCM 加密；攻击者算力受限，无法在同一时刻篡改或合谋控制超过

1. 罗平县网络信息服务和文明实践中心 云南曲靖 655899

30% 的节点，从而确保图消息传递的完整性。整体架构如图 1 所示。

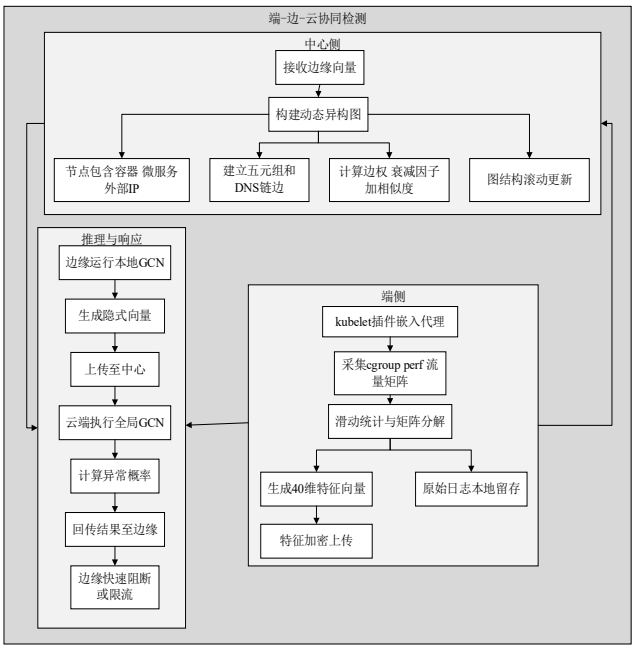

图 1 端 - 边 - 云协同检测架构图

2 融合边缘节点特征的图卷积检测模型设计

2.1 边缘节点多模态特征提取层

边缘节点多模态特征提取层以容器为最小采集单元，依托只读挂载的 sys/fs/cgroup/proc 文件系统与 DPDK 驱动接口，获取运行时微架构和网络行为指标。具体包括从 cpuacct、memory、blkio 子系统中提取 CPU 使用时间、cache-miss 增量、内存带宽等指标，结合 perf_event_open 系统调用采样 LLC-load-misses、branch-misses 及 stalled-cycles-frontend，以捕捉潜在的侧信道攻击特征 [5]；同时通过 rte_eth_stats 与 ring_utilization 获取 rx/tx 流量与队列拥塞度等网络层指标，共计 24 维原始指标。在 5 s 滑动窗口内对这些指标执行均值 - 方差归一化处理，以消除容器迁移带来的量纲扰动。归一化公式为：

$$x'_{ij}=\frac{x_{ij}-\mu_j}{\sigma_j+\varepsilon} \tag{1}$$

式中：x_{ij} 为第 i 个时间窗内第 j 维指标的原始值；μ_j 与 σ_j 分别为该指标在滑动窗口内的均值与标准差；ε=1×10^{-6}，用于防止除零错误。

归一化后与 GPU 利用率、核心温度、功耗等组成 27 维数值特征。另一方面，对每 100 ms 时间窗内捕获的系统调用 ID 序列，基于 CBOW 词袋模型训练的 Word2Vec 进行语义嵌入，生成 8 维调用行为向量。最终将两类特征按特征轴拼接形成 40 维边缘节点向量 $\boldsymbol{x}_i$，并以 1 s 周期通过 gRPC 上传至中心，完成本地隐私处理与跨域信息抽象。其核心采集与拼接流程如图 2 所示。

```python
def get_metrics(pid):
    cpu = float(open(
        "/sys/fs/cgroup/cpuacct/cpuacct.usage").read())
    mem = float(open(
        "/sys/fs/cgroup/memory/memory.usage_in_bytes").read())
    llc = float(subprocess.getoutput(
        f"perf stat -e LLC-load-misses -p {pid} sleep 1 2>&1 | grep LLC"
        " | awk '{print $1}'").replace(',', ''))
    return [cpu, mem, llc]

def get_syscall_embedding(syscalls):
    model = Word2Vec.load("cbow_syscall.model")
    vecs = [model.wv[s] for s in syscalls if s in model.wv]
    if vecs:
        return np.mean(vecs, axis=0).tolist()
    else:
        return [0.0] * 8

def get_feature_vector(pid, syscalls):
    numeric = get_metrics(pid)
    semantic = get_syscall_embedding(syscalls)
    return numeric + semantic
```

图 2 边缘节点 40 维特征提取核心流程

2.2 动态异构图构建与边权重量化

为刻画边缘侧瞬息万变的攻击面，中心侧以容器、微服务与外部 IP 为顶点集构建动态异构图。节点属性携带 40 维压缩特征，边类型划分为服务调用、数据同步与控制信令三类，分别对应 HTTP/gRPC、Kafka/MQTT 与 SSH/SNMP 会话。每当边缘代理上报新的特征向量，系统依据 5 元组与 DNS 调用链实时建立有向边，并在 30 s 滑动窗口内维护边的活跃度计数 [6]。为抑制长期静默连接引入的噪声，设计复合权重函数，将 Jaccard 相似度与时间衰减因子耦合，对低活跃度边执行自适应剪枝，确保图稀疏度与计算开销可控。边权重量化公式为：

$$w_{ij}=\frac{|N_i\cap N_j|}{|N_i\cup N_j|}\cdot e^{-\lambda(t_{now}-t_{last})} \tag{2}$$

式中：N_i 为节点 i 的直接邻接节点集合，分式部分用于计算结构层面的 Jaccard 相似度，反映节点 i 与 j 的邻居重合程度；指数衰减项引入交互时间差的影响，λ=0.05 s^{-1} 为预设的时间衰减系数，t_{now} 与 t_{last} 分别对应当前时间戳与两节点之间最近一次交互的时间。当 w_{ij}<0.01 时，该边将被标记为低活跃状态，从当前邻接表中暂时移除。后续若节点对发生新的交互事件，则重新激活该边并恢复至图结构中。边权值随时间与结构变化动态更新，剪枝与重连过程保持实时触发。图结构随时间演化，边的存在状态由节点邻接相似度与交互活跃度共同决定，边集具备可调节性与时变性，支持在数据流环境下的增量式更新。

2.3 双层 GCN 聚合与异常评分函数

为兼顾边缘算力受限与云端全局视野，采用双层图卷积架构协同完成异常评分。Local-GCN 部署在边缘 ARM 核，仅加载当前节点及其 1 跳邻居子图，利用 2 层轻量 GraphSAGE 聚合生成 64 维隐式向量 $\boldsymbol{z}_i^L$，参数规模控制在 128 kB 以内，推理延迟低于 20 ms；得到 $\boldsymbol{z}_i^L$ 后通过 AES-GCM 加

密通道上传至中心，与全局子图拼接构成完整邻接矩阵[7]。Global-GCN 基于 GPU 集群执行 3 层异构 GCN，卷积核按边类型分离，输出 128 维嵌入后经全连接层映射为异常概率 p_i。为缓解标签稀缺，设计联合损失函数，将有监督交叉熵与无监督图重构误差加权融合，使模型在半监督条件下仍能捕捉未知攻击模式。损失函数公式为：

$$\mathcal{L}=\sum_{i\in\mathcal{L}}-p_i(y_i)+\alpha\,\|\hat{\boldsymbol{A}}-\boldsymbol{A}\|_F^2 \tag{3}$$

式中：$\mathcal{L}$表示带标签节点集合；y_i 表示节点 i 的真实异常标签；$p_i(y_i)$ 表示模型对其预测概率的对数；α=0.3，表示损失权重系数，用于平衡分类项与结构重构项的相对贡献；$\hat{\boldsymbol{A}}$表示由 GCN 解码器生成的邻接矩阵重构结果；$\boldsymbol{A}$ 表示原始输入图的邻接矩阵；$\|\cdot\|_F$ 表示 Frobenius 范数。该损失函数通过端到端反向传播方式进行优化，分类目标与结构重构过程共享编码器参数，在训练阶段同时参与梯度更新。

2.4 模型轻量化与边缘部署策略

为使双层 GCN 在资源受限边缘节点可部署，采用量化 - 剪枝 - 编译协同压缩策略。对 Local-GCN 权重执行对称线性量化，将 FP32 参数压缩至 INT8，缩放因子通过校准集 KL 散度最小化确定，使精度下降控制在 0.8% 以内；基于通道级 L_1 范数实施结构化剪枝，移除冗余 80% 的 1×1 卷积核，经知识蒸馏微调，最终模型尺寸降至 1.15 MB。运行时利用 Tengine Lite 推理框架调用 ARM v8.2 DOT 指令，单帧前向延迟稳定在 28 ms，常驻 RAM 占用不超过 78 MB，满足 KubeEdge 边缘 Pod 资源限额。部署流程以 Edged 模块化下沉方式实现：压缩包经 OTA 通道差分下发，边缘节点收到后由守护进程完成哈希校验、解密与内存映射，随后替换旧动态库并重启 gRPC 服务，全程无需中断业务容器；若新模型连续 3 次推理异常概率方差大于阈值则自动回滚，保障线上稳定性。

2.5 在线学习与隐私保护机制

为实现模型持续演化且避免原始特征出境，系统引入差分隐私与联邦学习协同机制。边缘节点在本地完成一个 epoch 训练后，仅向中心上传梯度张量；上传前对梯度执行 L_2 范数裁剪，裁剪阈值 C=1.5，随后按差分隐私机制注入高斯噪声，隐私预算 ε 固定为 1.0，保证 (ε, δ)-DP 合规性。中心侧采用 Federated-LocalSGD 策略，每轮随机遴选比例 r=0.3 的节点参与聚合，通过安全信道接收加噪梯度并完成加权平均，更新全局模型后回传至边缘，全程不接触明文特征，满足 GDPR 数据最小化原则。理论分析表明，在强凸假设下，算法收敛率与隐私预算呈平方根反比关系，其收敛上界用公式表示为：

$$\mathbb{E}\left[F(\theta_T)-F^*\right]\le\mathcal{O}\left(\frac{LG^2}{\mu^2T}\right)+\mathcal{O}\left(\frac{d\sigma^2}{\mu^2T^2\varepsilon^2}\right) \tag{4}$$

式中：L 与 μ 分别为损失函数的利普希茨常数和强凸系数，G 为梯度上界，T 为通信轮数，d 为参数维度，σ^2 为注入噪声方差，ε 为隐私预算。该界限显示在 ε=1.0 时，增加通信轮数 T 可有效降低优化误差，同时隐私引入的方差项被控制在常数范围，从而兼顾模型效用与隐私保护强度。

3 案例验证与结果分析

3.1 实验环境与攻击场景设定

实验依托某省运营商 MEC 平台构建 40 节点 Kubernetes 集群，宿主机采用 ARM v8.2 16 核 CPU 与 32 GB RAM，容器运行时版本为 containerd 1.7.0，网络平面基于 Calico BGP 实现三层路由互通。为验证检测方法在真实边缘场景的鲁棒性，复现容器逃逸、DDoS 放大与 DNS 劫持三类高频攻击，利用 Cilium 可观测组件与 eBPF 探针采集系统调用、网络五元组及 perf 事件，采样周期 1 s，持续运行 30 天，累计生成 1.2 万条异常样本与 120 万条正常记录。容器逃逸通过恶意镜像触发 CVE-2022-0492 实现 cgroup 释放，DDoS 采用 memcached 1.6.9 反射放大，DNS 劫持则注入伪造响应指向恶意 C2。所有攻击流量经清洗后重放至测试命名空间，确保实验环境与现网隔离。

3.2 评价指标与对比算法

为全面评估检测模型性能，选取五项指标：Precision（精确率）、Recall（召回率）、F_1-score、AUC（ROC 曲线下的面积）以及 MTTD（mean time to detection）。其中，MTTD 定义为攻击事件发生至系统首次产生告警的平均延迟，反映模型在边缘环境中的响应时效性，单位为秒。对比方法涵盖三类代表性基线：孤立森林（IForest）作为无监督异常检测方法，输入为系统调用与网络特征的静态统计量；Central-GCN 代表集中式图神经网络方案，依赖全局视图进行图结构建模与端到端训练；SVM-RBF 基于径向基核函数，依托手工构造的多维数值特征执行二分类，体现浅层分类器在小样本条件下的表现能力。为确保对比公平性，所有算法在相同采样周期下接收一致的多模态数据流，并统一采用离线标注样本集作为评估标签源，避免因数据来源、时间窗口或标签策略差异导致性能偏移。

3.3 检测结果与消融实验

在检测性能评估方面，图 3 展示了五种方法在统一数据流条件下的 F_1-score 对比结果。本文提出的融合边缘特征与动态图卷积机制的检测方法在 F_1 得分上达到 0.943，显著优于孤立森林（0.796）、SVM-RBF（0.857）与 Central-GCN（0.901）等代表性基线算法，验证了其在综合精确率与召回率方面的优势。

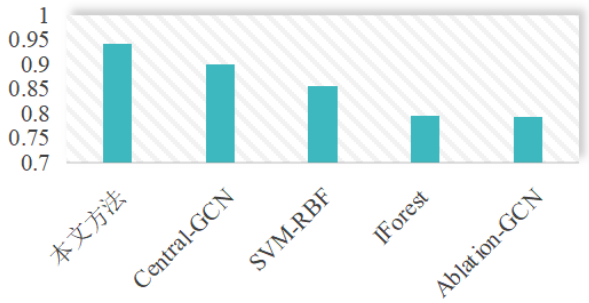

图 3 不同算法 F_1 对比柱状图

进一步地，为验证模型在复杂攻击环境中的鲁棒性，图 4 给出了不同攻击强度（10%~50%）下 F_1-score 随强度增长的下降趋势曲线。结果表明，本文方法在攻击流量比例上升时 F_1 仅下降 4.8%，相比其他算法具有更强的性能稳定性，表征其对分布漂移与低频攻击模式的适应能力更优。为探究模型各模块的性能贡献，设计并实施了三组消融实验。

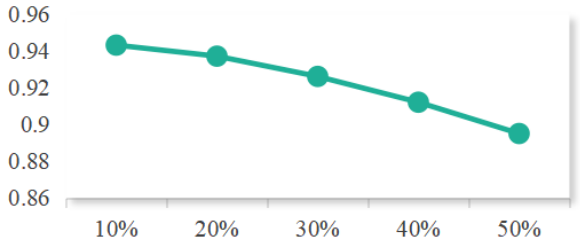

图 4 攻击强度 F_1 曲线图

如表 1 所示，移除边缘节点微观资源特征（如 cache-miss、LLC-load-miss 等）后，模型 F_1 下降至 0.823，下降幅度达 0.120，说明运行时指标对异常模式刻画具有关键价值。关闭动态图边权重自适应更新机制后，F_1 下降 0.080，进一步表明结构演化与交互活跃度建模对图表达能力的影响不可忽视。将双层 GCN 架构简化为单层集中式 GCN 后，F_1 下降至 0.793，降幅高达 0.150，体现了边 - 云协同聚合在提升模型泛化能力方面的决定性作用。

表 1 模块消融实验 F_1-score 对比表

模型配置	F_1-score	F_1 下降幅度
本文完整模型（含边缘特征 + 动态图 + 双层 GCN）	0.943	0
去除边缘特征	0.823	-0.12
去除动态图边权重更新机制	0.863	-0.08
去除双层聚合（仅使用 Global-GCN）	0.793	-0.15

上述实验结果验证了多模态边缘特征提取、动态图结构建模与双层图神经网络融合机制在整体性能中的支撑地位，为后续在资源受限边缘场景中的部署提供了量化参考。

3.4 开销与工业落地讨论

现网试点表明，引入 Local-GCN 后边缘 CPU 利用率增加 8.7%，峰值内存占用提升 78 MB，单节点额外带宽低于 25 kbit/s，对基站既有业务影响可控。工业落地仍需解决三大挑战：一是证书自动化，边缘容器启动时通过 EST 协议向 CA 申请短周期证书，实现 TLS 通道零人工干预；二是模型版本回滚，当在线推理异常率连续三次超阈值时，Edged 可秒级切换至上一版本镜像并触发告警；三是智能聚合深度自适应，后续计划引入强化学习，以网络延迟、丢包率为状态，动态调整 GCN 层数与聚合跳数，在精度与开销间寻求帕累托最优，为运营商边缘安全提供可持续演进的闭环方案。

4 结语

本文围绕边缘环境中网络异常检测的挑战，构建了一种“边缘感知 - 中心推理 - 边缘响应”的图卷积检测机制，融合容器微观运行态指标、动态图结构建模与双层 GCN 聚合策略，显著提升检测精度与部署效率。经 MEC 集群实验证实，所提模型在多项指标上均优于主流方法，并具备边缘环境可部署性。未来工作将围绕自动化证书管理、推理稳定性保障与 GCN 自适应调优等方面展开，持续完善边缘安全系统的可维护性与演化能力，为运营商与工业互联网场景下的智能防护奠定基础。

参考文献：

[1] 吴昊，郝佳佳，卢云龙 . 物联网场景下基于蜜场的分布式网络入侵检测系统研究 [J]. 通信学报，2024, 45(1):106-118.

[2] 田婷，虞延坤，牛新征 . 边缘计算环境下基于深度学习的 DDos 检测 [J]. 计算机测量与控制，2023, 31(7):28-34.

[3] 陈志文，陈嫒靓霏，唐晓丹，等 . 面向煤矿安全监测边缘计算的 YOLOv5s 剪枝方法 [J]. 工矿自动化，2024, 50(7):89-97.

[4] 杜越，宁少慧，段攀龙，等 . 基于边缘图注意力网络的轴承智能故障诊断 [J]. 机床与液压，2024, 52(6):190-195.

[5] 周建，石浩，秦铭泽，等 . 基于边缘智能技术的台区线损异常自动检测研究 [J]. 自动化技术与应用，2024, 43(8):46-49.

[6] 付豪，邓蕾，汤宝平，等 . 资源受限的机械振动 WSN 层次分解 CNN 边缘计算方法 [J]. 仪器仪表学报，2024, 45(3):94-105.

[7] 王静潇，王辛岩，周禹彤，等 . 面向交通流预测的分支定界算法图卷积模型 [J]. 现代电子技术，2023, 46(12):153-158.

【作者简介】

陈琦昊（1990—），男，云南罗平人，本科，中级工程师，研究方向：网络舆情监测与分析。

（收稿日期：2025-11-12 修回日期：2026-07-09）