计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2026,35(2):154−164 [doi: 10.15888/j.cnki.csa.010066] [CSTR: 32024.14.csa.010066]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于交互状态图的多粒度融合加密恶意流量检测①

李含玥, 郝俊值, 吴承荣

(复旦大学 计算机科学技术学院, 上海 200433)
通信作者: 吴承荣, E-mail: cwu@fudan.edu.cn

摘　要: 加密技术的广泛应用给恶意活动提供了藏匿的机会, 对网络安全监测体系带来了巨大挑战. 现有的加密流量检测方法主要是在单个数据包级别提取统计流量特征, 因此可能会由于潜在的 IP 分片而破坏原始连续通信行为中隐含的特征. 此外, 大多数方法对于网络流的交互模式建模粒度较粗, 未能深入挖掘对等实体间的通信意图, 难以适应新型恶意软件通信行为和通信量的变化. 本文以交互为分析粒度, 提出了方法 ISG-Net (interaction state graph-net). 该方法基于状态转换构建流量交互状态图, 并引入了融合流量时序信息的自注意力编码模型. 特别地, 本文通过交互状态图获取蕴含全局信息的交互状态表示, 然后对每次交互进行细粒度的特征提取, 以融合得到会话 (双向流) 的表示. 在 3 个数据集上的实验结果表明, 在加密恶意流量检测任务中, 本文方法在准确性、鲁棒性和容错性均优于现有算法.

关键词: 加密恶意流量检测; 流量交互模式; 网络安全; 流量分类

引用格式: 李含玥,郝俊值,吴承荣.基于交互状态图的多粒度融合加密恶意流量检测.计算机系统应用,2026,35(2):154–164. http://www.c-s-a.org.cn/1003-3254/10066.html

Multi-granularity Fusion for Encrypted Malicious Traffic Detection Based on Interaction State Graph

LI Han-Yue, HAO Jun-Zhi, WU Cheng-Rong

(School of Computer Science and Technology, Fudan University, Shanghai 200433, China)

Abstract: The widespread adoption of encryption technology has given malicious activities a chance to hide, posing a great challenge to network security monitoring systems. Existing encrypted traffic detection methods primarily extract statistical traffic features at the individual packet level. However, this may disrupt the features implied in the original continuous communication behavior, due to potential IP fragmentation. Furthermore, most approaches model the interaction patterns between network flows at a relatively coarse granularity, failing to thoroughly explore the communication intent between peer entities. This study introduces a novel method, interaction state graph-net (ISG-Net), which uses interaction as the analysis granularity. ISG-Net constructs a traffic interaction state graph based on state transitions and applies a self-attentive encoder model to capture temporal traffic information. In particular, interaction state representations containing global information are obtained through the interaction state graph. Then, fine-grained features of each interaction are extracted to obtain the representation of the sessions (bidirectional flows). Experimental results on three datasets demonstrate that the proposed method outperforms existing methods in terms of accuracy, robustness and fault tolerance in the task of encrypted malicious traffic detection.

Key words: encrypted malicious traffic detection; traffic interaction pattern; cyberspace security; traffic classification

① 基金项目: 国家重点研发计划 (2024YFC3308005)
收稿时间: 2025-07-16; 修改时间: 2025-08-13; 采用时间: 2025-09-01; csa 在线出版时间: 2025-12-19
CNKI 网络首发时间: 2025-12-22

随着用户隐私保护意识的增强和法律法规的日益严格, 加密网络流量已成为当今互联网流量的重要组成部分[1]. 根据谷歌 2025 年的调查报告[2], 其监测的网络流量中高达 95% 已采用加密技术. 加密技术有效保障了用户数据的机密性与完整性, 但同时也为网络犯罪分子提供了天然的保护伞. 恶意软件 (如僵尸网络、勒索软件、APT 攻击工具) 广泛利用加密协议 (如 SSL/TLS、QUIC) 对其通信进行加密, 将恶意活动深度隐匿于海量常规加密流量之中. 这使得依赖明文内容检测的传统网络安全防御机制, 如基于规则匹配的防火墙、深度包检测技术[3,4]近乎失效, 严重削弱了网络空间的监管与防护能力. 恶意加密流量的泛滥直接导致了数据泄露、服务中断、金融欺诈等安全事件的激增, 因此, 如何在无法解密内容的前提下, 准确地识别出混杂在合法加密流量中的恶意通信, 已成为当前网络安全领域一项亟待解决的重大挑战.

为应对这一挑战, 研究者先后提出了基于统计特征的机器学习方法[5–9]与端到端特征提取的深度学习模型[10–14]. 然而, 这些方法在实际部署中仍存在明显局限: 机器学习方法严重依赖人工特征工程, 面对不断变化的恶意行为泛化能力有限; 深度学习模型虽能自动提取特征, 但通常将流量表示为同质化向量, 忽视了其内在的层次化结构与行为语义, 难以识别复杂多阶段的攻击模式.

近年来, 基于图结构的分析方法因能够刻画网络实体间的交互关系而受到关注[15–18]. 该类方法通常以主机为节点、通信流为边构建流量图, 并试图从图拓扑中识别恶意模式, 但存在 3 个关键局限: 其一, 图结构脆弱性: 动态 IP 分配、NAT 网关及云环境导致 IP 频繁复用, 节点标识与实际网络实体持续脱节, 模型稳定性差, 在实际的公网环境中面临严重干扰, 恶意流量与正常流量在 IP 层面高度混杂, 使得在聚合通道上无法精准定位恶意来源, 而只能封锁整个公网 IP, 造成大面积误拦截. 其二, 关系表征不足: 简单的流级边连接无法表达丰富的交互语义 (如通信方向、时间相关性), 导致请求-响应模式等关键行为模式信息丢失. 其三, 时序动态性缺失: 现有静态流量建模方法存在严重时序动态性缺失问题, 忽略了恶意流量的时间规律性 (如 C&C 心跳) 和序列依赖性 (如攻击阶段转换). 因此, 构建一种能稳定表征流量、充分编码交互语义、有效捕捉时序动态的行为模式, 是提升加密恶意流量检测精度的关键.

现有基于流的建模粒度较粗, 难以刻画会话内部结构; 而包级分析无法体现软件行为级别特性, 又易受 IP 分片干扰, 导致通信意图信息丢失. 鉴于应用通信本质上以会话 (即双向流) 为基本单位, 且会话可视为由连续的“交互”构成的序列. 一次交互通常对应应用层的一次完整收发操作, 因此会话内部的交互序列能更客观地反映软件的通信行为模式. 基于此, 本研究提出一种改进的加密恶意流量检测方法 ISG-Net, 该方法以会话为基本单位, 将其解构为连续交互状态组成的序列, 进而构建交互状态图 (interaction state graph, ISG), 以刻画不同粒度下的会话结构与行为模式, 并引入融合时序特征的自注意力编码机制, 实现端到端的恶意流量检测. 本文的核心工作与贡献在于以下几方面.

(1) 多粒度融合的会话表示方法: 通过定义交互状态并构建状态转换图, 摆脱对 IP 拓扑的过度依赖, 使用自适应门控机制实现了对会话宏观状态演变和微观交互细节的层次化建模, 揭示软件通信行为不同层面的规律.

(2) 设计时序融合自注意力编码器: 通过在交互表示中显式嵌入时间戳信息并利用自注意力机制学习不同交互的权重, 有效解决了现有方法时序动态性缺失及交互相关性和重要性模糊的问题.

(3) 基于元数据的隐私保护方案: 仅利用流量元数据 (负载长度、方向、时间), 不依赖解密内容, 确保了用户数据隐私, 并天然具备对各类加密协议的鲁棒性.

(4) 实证验证的优越性能: 通过大量实验验证了所提模型 (ISG-Net) 的有效性. 结果表明, ISG-Net 在检测精度、鲁棒性和容错性方面均优于现有方法.

1 相关工作

随着网络加密流量的迅速增长, 如何有效地检测加密恶意流量已经成为一个重要的研究方向. 本文将现有的方法分为基于统计特征、端到端特征和行为模式特征这 3 大类.

基于统计特征的方法主要通过分析网络流量的统计特征来检测恶意流量. 这些特征通常包括数据包大小、时间戳、流量方向、吞吐量和序列模式等. 例如, Anderson 等人[19]发现, 在 TLS 握手阶段, 正常流量与恶意流量在像密码套件等特征上存在显著差异. de Lucia 等人[20]利用 TLS 记录的大小、类型和方向序列作为特

征实现了高效的恶意通信检测. Stergiopoulos 等人[21]利用 TCP 包的旁路信道特征来识别恶意流量. Meghdouri 等人[22]比较了几种轻量级的特征集, 发现数据包长度对于区分恶意流量和良性流量至关重要. Chen 等人[23]通过提取上下行流量特征并应用如 SMOTE 和 SVM 等机器学习算法来缓解真实网络中恶意流量样本不平衡的问题, 从而提高分类性能. Dong 等人[24]提取数据包长度序列来分析恶意流量特征, 特别是用于检测加密远程访问木马 (RAT) 通信. Dodia 等人[25]通过结合传统连接级别和全局主机网络特征, 专注于识别基于 Tor 的恶意软件通信. Fu 等人[26]利用由频域特征表示的序列信息来检测恶意流量. Yu 等人[27]基于 TLS 指纹捕获加密恶意流量特征.

基于端到端特征的方法利用深度学习技术从原始流量数据中自动提取特征, 无需进行繁琐的特征工程. 例如 Wang 等人[10]提出了一种基于卷积神经网络 (CNN) 的流量数据表示学习方法, 将原始流量数据表示为图像进行分类. Shapira 等人[14]也将原始流量数据转换成图像格式, 并使用 CNN 进行分类. 戚子健等人[28]提出结合双向 GRU 和 CNN 的恶意流量检测方法. Liu 等人[29]提出了一个端到端分类模型, 该模型包括多层编解码结构和重构机制. Lin 等人[30]通过从大规模未标记数据中预训练深层上下文化数据包表示, 在各种加密流量分类任务中实现了先进的性能. Hang 等人[31]提出了 Flow-MAE, 这是一个采用计算机视觉领域中的掩码自动编码器 (MAE) 的预训练模型, 以实现流量分类. Cai 等人[32]用 LSTM 网络捕捉网络流量时间序列特征中的长期依赖关系, 缓解概念漂移问题.

基于行为模式特征的方法主要关注网络流量的关联性和交互模式, 通常使用图模型来表示流量的行为特征, 这类方法能够捕捉网络通信中各节点和连接之间的关系信息. Wang 等人[33]结合流量中基于流的特征和图结构特征进行混合分析, 表明流量之间的相关性在恶意流量检测中至关重要. 建图方式成为影响模型效果的关键. Zhao 等人[34]建立主机与域名之间的二分图来描述恶意软件的 DNS 查询行为. Huoh 等人[35]根据流内数据包的时间关系建立边, 并使用图神经网络, 但随着噪声流量数据的存在, 该图形结构可能会变得脆弱. Fu 等人[15]利用紧凑的内存图捕获流量之间的交互模式, 通过分析图的连通性、稀疏性和统计特征发现恶意流量. Zhao 等人[36]以流为节点建立了关系多图, 并自适应地修正关系以实现威胁检测.

本文提出的方法融合了这 3 类方法的互补优势, 通过定义会话中的交互状态, 构建交互状态图以此获得交互转换的行为模式, 并显式捕获不同交互的细粒度特征, 在充分提取交互行为信息 (图表示) 和统计特征的基础上, 引入深度神经网络分类器进行会话级的加密恶意流量检测, 从而实现更高的检测精度和更强的容错性.

2 方法

图 1 展示了 ISG-Net 的整体框架, 它由 4 个模块组成. 网络流量预处理模块将收集的原始流量重组为会话, 其中会话是流量识别的粒度, 每个会话可被划分为多个交互, 交互被视为分析粒度 (第 2.1 节). 交互状态图构建模块获取交互的全局表示 (第 2.2 节). 会话嵌入模块获得交互的细粒度特征并进行多粒度的自适应融合, 从而得到会话表示 (第 2.3 节). 恶意流量检测模块基于融合时序特征的自注意力编码器模型执行最终的流量检测 (第 2.4 节).

2.1 分析粒度

当前流量分析粒度主要可分为数据包、流、会话. 数据包是网络通信的基本单位, 流是由一系列具有相同源 IP、目的 IP、源端口、目的端口和协议的数据包组成的一个逻辑单元, 会话则是由双向流组成的通信单元. 由于受到 MTU 的限制, 上层协议栈会对过长数据进行分片, 所以以数据包为粒度的分析手段并不能还原流量原本的通信目的, 还会丢失通信过程中宝贵的上下文信息; 流则只考虑单向的数据信息, 没有深入挖掘流量中体现软件通信行为的交互信息, 导致对不同行为模式的理解不完整. 因此, 本文将会话作为识别恶意流量的基本粒度, 并将会话视为一系列顺序交互的集合, 将交互中的请求和响应视为特征提取的最小分析粒度. 具体而言, 一个会话被视为由多个交互组成的序列: $Session = \{Interaction_i, i \in \mathbb{N}^+\}$, 一次交互由请求、响应组成, 请求 (响应) 指短时间内从客户端 (服务器) 到服务器 (客户端) 发送的一系列数据包集合, 具体如下:

$$Interaction = \begin{cases} Request = \{p_m^{\text{src}}, m \in \mathbb{N}^+\} \\ Response = \{p_n^{\text{dst}}, n \in \mathbb{N}^+\} \end{cases} \tag{1}$$

其中, src 和 dst 分别表示源和目的, p_m^{src} 表示源发送的第 m 个数据包.

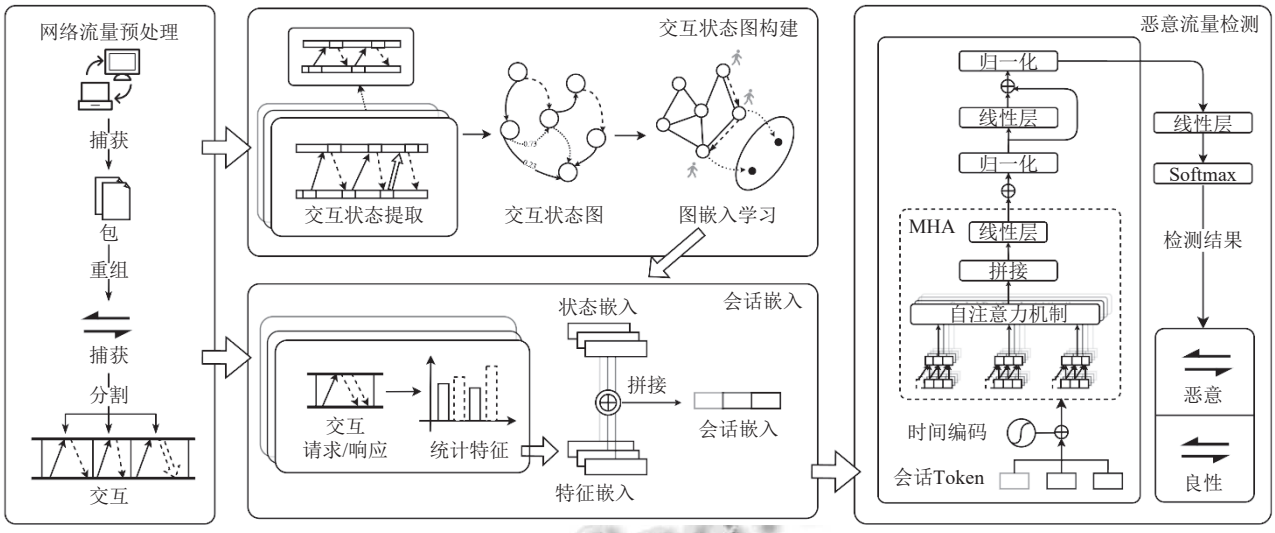

图 1 ISG-Net 流程

2.2 流量交互状态图的构建

通信实体的当前状态通常与若干先前的交互有关, 交互序列中不同状态的转换在概率方面表现出一定的规律. 本文将通信双方的交互过程视为通信过程中交互状态的转变. 为克服传统“主机-流”图模型的结构脆弱性与交互细粒度建模缺失的问题, 本文提出交互状态图 (ISG)——一种有向加权异构图 $G=(V, E, W)$, 其中 V 为交互状态集, E 为状态转移边, W 为转移权重. 该图通过将应用层交互逻辑抽象为状态转移过程, 实现了多粒度行为建模与数据流向保留, 能够充分刻画软件通信行为中蕴含的模式. 图 2 展示了 ISG 的构建过程及示例, 虚线和实线框节点分别表示由服务器、客户端发出的包集合.

(1) 节点

图中每个节点表征交互状态 IS, 定义为在短时间内沿请求或响应方向连续传输的一系列数据包的特征. 由于加密后的流量负载内容是随机的, 本文不使用任何数据包载荷的内容. 为了防止 IP 分片切割通信意图并且保证交互状态对不同网络环境的扰动性具有良好的鲁棒性, 本文选择的交互状态特征 IS 是数据包载荷长度之和, 如F_{req}或F_{resp}. 其中$F_{\text{req}}=\sum_{j=1}^{P} f\left(p_j\right)$, F_{req}是第i次交互中源发送的请求数据包载荷长度之和, $f(p_j)$表示单个数据包p_j的载荷长度. 若在交互中没有响应方向的数据包, 则使用值 0 进行填充.

为解决包长动态范围大导致的维度爆炸问题, 本文设计启发式分段映射策略: 当 $F \leqslant MTU-L_{\text{header}}$ 时按超参数 N 等距离散化 (沿用文献[17]); 当 $F>MTU-L_{\text{header}}$ 时采用指数区间划分, 显著压缩交互状态空间, 其中 MTU 表示最大传输单元, L_{header} 表示包头长度. 此设计突破了连续特征处理的局限性, 在状态分辨力与计算效率间取得平衡.

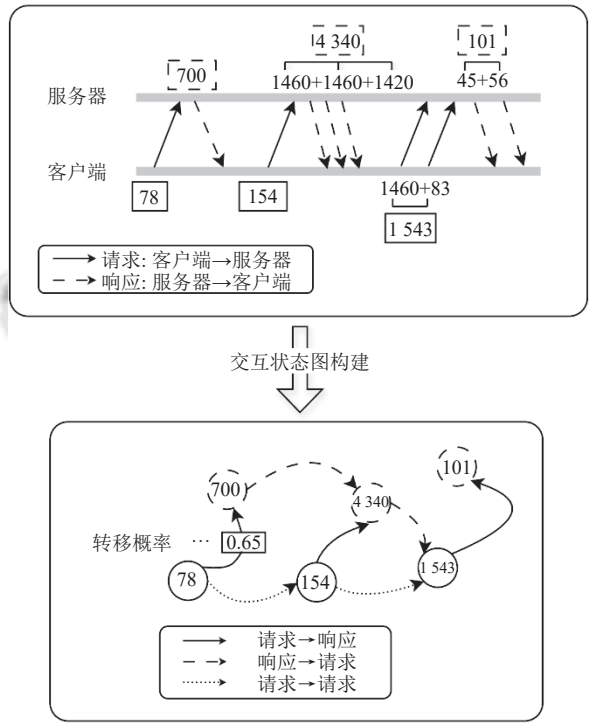

图 2 交互状态图的构建

(2) 边

为捕获交互语义依赖, 图中定义 3 类状态转移边: 1) 相邻请求的边: 建模请求序列依赖; 2) 请求至响应的

边: 表征单次交互的双向依赖; 3) 响应至请求的边: 刻画跨交互的因果链. 这 3 种边能够挖掘会话中交互内/间的依赖, 解决了图模型中关系表征不足的缺陷.

(3) 交互状态表示学习

基于构建好的交互状态图, 本文通过优化后的 DeepWalk 算法[37]学习图中节点的嵌入表示, 以表征流量的交互状态. 为了深入探讨实际互联网环境中不同交互状态之间的复杂关系, 本文基于大规模流量数据计算不同交互状态的转移概率矩阵 P, 为每条边分配相应的转移概率权重:

$$P_{uv} = \frac{w_{uv}}{\sum_{v' \in N(u)} w_{uv'}} \tag{2}$$

其中, $N(u)$ 表示与节点 u 相邻的节点集合.

从任意节点 $u_0 \in V$ 开始生成长度为 L 的随机游走序列 $\{u_0, u_1, \cdots, u_L\}$, 其中下一个节点的选择遵循转移概率矩阵中的概率分布:

$$\Pr(u_{t+1} = v \mid u_t = u) = P_{uv} \tag{3}$$

与 DeepWalk 在随机游走中的无偏性不同, 本文不是对所有相邻节点均匀采样, 而是依据边权进行偏置采样, 较高的边权重意味着在网络中状态转换更频繁, 更符合正常的交互模式. 相反, 较低的边权重表明状态转换较为罕见, 可能指示异常行为. 因此, 在游走过程中, 优先选择权重较高的边来形成游走序列. 然后通过 Skip-Gram[38]来学习节点表示, 目标是最大化游走序列中高度相关的交互状态具有更高的共现概率:

$$\max_{\theta} \sum_{u \in V} \sum_{v \in C(u)} \log \Pr(v|u;\theta) \tag{4}$$

其中, $C(u)$ 表示节点 u 在随机游走中上下文窗口内的节点集合, 预测概率定义为:

$$\Pr(v|u;\theta) = \frac{\exp(z_v^{\top} z_u)}{\sum_{v' \in V} \exp\left(z_{v'}^{\top} z_u\right)} \tag{5}$$

这样, 最终生成的节点表示向量 z_u 可以更加准确地反映整体网络环境中的流量交互状态.

2.3 多粒度融合的交互表示

会话被表示为交互序列 $\{I_1, I_2, \cdots, I_T\}$, 其中单次交互 I_t 的构建融合了宏观与微观特征.

(1) 宏观特征: 基于 ISG 生成的节点表示向量 z 拼接请求/响应状态嵌入 $e_t = \left[z_{\text{req}}; z_{\text{resp}}\right]$, 用以表征交互级的宏观语义信息.

(2) 微观特征: 针对单次交互, 提取双向包数、包长 (最大值、最小值、均值、标准差) 的统计特征, 保留包级粒度的感知能力, 经多层感知机 (multilayer perceptron, MLP) 映射为 s_t.

(3) 门控自适应特征融合: 通过门控网络自动学习宏微观特征权重, 克服了传统方法对会话内部交互结构同质化处理的局限, 最终获得交互表示 I_t.

$$g_t = \sigma(W_g[e_t; s_t]) \tag{6}$$

$$I_t = g_t \odot e_t + (1 - g_t) \odot s_t \tag{7}$$

2.4 时序增强的恶意流量检测

网络流量数据本质上是具有强时序依赖性的有序序列, 其中恶意流量通常由自动化工具或恶意软件生成, 行为模式规律性显著, 中短期序列关联尤为突出. 现有方法往往忽略交互间的时间动态特性, 且未能显式区分不同交互的重要性, 导致序列语义表示不够精准. 针对上述问题, 本文提出一种时序融合自注意力编码, 基于 Transformer 的 Encoder 架构[39]构建检测模型.

该编码器通过引入显式的时间戳嵌入机制, 将会话中每次交互的绝对位置与相对时间信息融合进其表示中, 从而增强模型对恶意流量时序模式的感知能力. 进一步, 借助自注意力机制, 模型能够自适应地学习不同交互之间的相关性并分配差异化的权重, 从而聚焦于关键交互事件, 清晰刻画序列内的依赖结构. 该方法有效解决了现有模型中时序动态性缺失与交互重要性模糊两大局限, 提升了流量表示的判别能力.

(1) 时间编码

正常流量通常具有规律性, 而恶意流量会因自动化脚本或攻击行为表现出显著的突发性和不可预测性. 因此, 精确的时间信息对于刻画流量行为模式、区分正常与异常至关重要. 然而, 标准的 Transformer 自注意力机制及其位置编码仅能捕捉序列中元素的绝对顺序, 而无法显式建模元素间实际发生的时间间隔. 为弥补这一缺陷并有效整合关键的时序上下文, 本研究提出一种改进的时间编码方案: 将每个交互事件的时间间隔转换为正弦函数的相位偏移量, 进而融合到输入表示中. 时间编码定义为:

$$\left[z\left(t_j\right)_i\right] = \begin{cases} \cos\left(t_j / 10000^{\frac{i-1}{d}}\right), & \text{if } i \text{ is odd} \\ \sin\left(t_j / 10000^{\frac{i}{d}}\right), & \text{if } i \text{ is even} \end{cases} \tag{8}$$

其中, t_j是相对于第 1 次交互的时间间隔, d是编码的维度.

在多变的网络环境下将时间单独作为一个特征使用会丧失鲁棒性. 因此本文通过引入时间编码, 将流量数据的时间信息经过单独的转化再与会话表示进行加和, 以辅助检测. 最后得到的会话表示$X=\left\{\left(s_i,z_{(t_i)}\right)\right\}_{i=1}^{L}$为:

$$X=S+Z \tag{9}$$

其中, $S=[s_1,s_2,\cdots,s_L]\in R^{N\times L}$是交互嵌入的拼接集合, $Z=[z(t_1),z(t_2),\cdots,z(t_L)]\in R^{N\times L}$是交互时间编码的拼接集合. X中的每一行对应于一个会话中的特定交互.

(2) 自注意力机制

在初始编码层之后, 本文通过自注意力模块传递X. 具体来说, 本文计算注意力输出:

$$\begin{cases} Attn=Softmax\left(\dfrac{QK^{\top}}{\sqrt{D}}\right)V \\ Q=XW_q,\ K=XW_k,\ V=XW_v \end{cases} \tag{10}$$

其中, W_q、W_k、W_v是权重矩阵. 对于每一个查询向量, 注意力权重通过计算与所有键向量的点积来获得. 本文通过使用多头自注意力机制学习不同阶段交互的注意力权重, 显式利用时序上下文识别恶意流量的阶段化行为模式.

(3) 分类器

对输出序列执行平均池化以产生会话级表征, 通过全连接层将该向量映射到二维向量, 最终由 *Softmax* 函数输出恶意流量概率.

3 实验分析

3.1 实验设置

3.1.1 数据集

为了全面评估该方法的性能, 本文在 3 个公开的流量数据集上进行了广泛的实验.

(1) DataCon-EMT[40]. 该数据集是由奇安信技术研究院的天穹沙箱捕捉生成的正常应用程序和恶意软件产生的网络流量构建而成. 大部分流量数据由 TLS/SSL 数据包组成.

(2) USTC-TFC2016[41]. 该数据集由中国科学技术大学发布, 用于恶意软件流量检测评估, 包含 10 种良性应用程序和 10 种恶意软件流量.

(3) MCFP[42]. MCFP 数据集是由捷克技术大学作为 Stratosphere IPS 项目的一部分创建的, 该项目旨在捕捉并分析持续存在的正常和恶意流量. 由于加密协议的广泛使用, 其中包含大量加密的恶意流量.

3.1.2 基线

本文在相关工作的 3 个类别中选择了 6 个具有代表性的算法, 以便更全面地评估 ISG-Net 的性能.

(1) 基于统计特征的机器学习方法

1) AppScanner[9]从数据包流中提取双向流量的多种统计特征, 然后使用随机森林分类器来识别异常流量.

2) CUMUL[43]通过提取数据包长度和时间戳的统计特征来捕获流信息, 并使用 SVM 来识别加密和匿名连接中的内容.

3) LightGBM[44]使用来自数据包序列的 61 维统计特征训练 LightGBM 进行分类.

(2) 基于深度学习的方法

1) FS-Net[29]采用双向门控循环单元 (GRU) 结合多层编解码器结构, 以端到端方式学习原始流量中的序列特性并进行分类.

2) ET-BERT[30]是一种基于预训练模型 BERT 的加密流量分类方法. 它从大规模未标记的数据中预训练流量表示, 然后在少量特定任务的标签数据上进行微调. 注意, ET-BERT 使用了数据包负载.

(3) 基于图的方法

GraphDApp[16]将会话内的单个数据包长度序列建模为图, 将加密流量识别任务转化为图分类问题进行分析.

3.1.3 评价指标

为了确保全面的比较, 本文使用精确度 (*PR*)、召回率 (*RC*)、*F*1 和 AUC 作为评估指标. 前 3 个指标的计算方法如下:

$$PR=\frac{TP}{TP+FP} \tag{11}$$

$$RC=\frac{TP}{TP+FN} \tag{12}$$

$$F1=\frac{2\times RC\times PR}{RC+PR} \tag{13}$$

其中, *TP*、*FP*、*TN* 和 *FN* 分别代表真阳性、假阳性、真阴性和假阴性. AUC 指的是 ROC 曲线下面积, 该曲线是在不同的阈值设置下绘制的真阳性率 (TPR) 与假阳性率 (FPR) 结果.

3.1.4 实现细节

在训练阶段, 本文将初始学习率设为 0.000 1 并使用带有学习率调度器的 Adam 优化器. 本文采用了早

停策略, 批大小为 512, 所有实验均使用 PyTorch 2.3.1 实现, 并在配备有 Intel(R) Xeon(R) Silver 4210R CPU@ 2.40 GHz、62 GB 内存及 NVIDIA GeForce RTX3090 GPU 的 CentOS Linux 7 服务器上进行训练.

3.2 对比实验

在 DataCon-EMT、USTC-TFC2016 和 MCFP 数据集上, ISG-Net 均取得最佳性能, 显著优于各类基线方法, 证明了其检测精度和有效性, 结果见表 1–表 3.

表 1 不同方法在 DataCon-EMT 数据集上的比较 (%)

方法	*PR*	*RC*	*F*1	AUC
AppScanner	93.81	93.82	93.81	93.82
CUMUL	93.76	93.76	93.76	93.76
LightGBM	83.91	80.36	81.86	88.65
FS-Net	91.76	91.73	91.73	91.73
ET-BERT	90.75	88.30	90.15	90.18
GraphDApp	89.99	89.45	89.49	89.53
ISG-Net (Ours)	96.33	96.35	96.35	96.35

表 2 不同方法在 USTC-TFC2016 数据集上的比较 (%)

方法	*PR*	*RC*	*F*1	AUC
AppScanner	83.21	83.21	83.21	83.21
CUMUL	81.22	81.22	81.22	80.20
LightGBM	82.92	82.92	82.91	82.93
FS-Net	86.64	86.62	86.52	87.05
ET-BERT	94.22	94.22	94.22	94.32
GraphDApp	92.72	92.73	92.73	92.73
ISG-Net (Ours)	96.67	96.64	96.67	96.67

表 3 不同方法在 MCFP 数据集上的比较 (%)

方法	*PR*	*RC*	*F*1	AUC
AppScanner	98.95	98.95	98.95	98.95
CUMUL	94.76	94.76	94.76	94.76
LightGBM	94.91	94.87	94.90	94.90
FS-Net	98.76	98.73	98.73	98.73
ET-BERT	84.60	95.27	89.62	88.02
GraphDApp	97.46	97.45	97.45	97.45
ISG-Net (Ours)	99.98	99.98	99.98	99.98

尤其值得注意的是, ISG-Net 克服了流量概念漂移挑战, 在更新的 DataCon-EMT 数据集上仍保持最优. 针对统计特征方法, AppScanner 表现最优, 但 ISG-Net 凭借融合统计特征、会话表示及宏观交互状态, 在各项指标上超越 AppScanner, *F*1 提升 2.54%. 针对深度学习方法, FS-Net 未关注交互特性, 而 ET-BERT 因仅关注负载字节相关性而效果较差, ISG-Net 通过建模正常流量交互状态变化模式, *F*1 分别超出 FS-Net 和 ET-BERT 4.63% 和 6.21%. 针对图方法 GraphDApp, 在 MCFP 数据集上次优, 但其受限于对单个数据包的关注, 在 DataCon-EMT 上表现较差. ISG-Net 通过分析请求/响应的累积特性并排除 IP 分片影响, 在 DataCon-EMT 上的 *F*1 比 GraphDApp 高 6.87%.

3.3 消融实验

为了验证所提出的每个组件对方法的贡献, 本文在 DataCon-EMT 数据集上进行了消融实验. 本文分别消融了区间的嵌入表示方法、时间编码和交互统计特征, 结果如表 4 所示. 为了便于结果展示, 本文将时间编码、位置编码和统计特征融合分别记作“TE”“PE”和“ST”. 特别地, 本文还测试了一些替代模块或操作的影响.

表 4 消融实验 (%)

方法	*PR*	*RC*	*F*1	AUC
w/o ISG	93.70	93.67	93.67	93.67
w/ GCN	91.49	92.95	92.21	92.14
w/ GraphSAGE	93.14	92.36	92.75	92.79
w/o ST	94.49	94.47	94.48	94.47
w/o TE	94.44	94.43	94.43	94.43
w/ PE	94.85	94.81	94.82	94.81
Default	96.33	96.35	96.35	96.35

(1) 交互状态表示学习: 在获得交互状态表示时, 本文不采用加权 DeepWalk 算法, 而是使用了两种基于图神经网络的节点嵌入表示方法 GCN[45]和 GraphSAGE[46]. 实验结果表明, 使用加权 DeepWalk 策略的 AUC 分别高于 GCN 和 GraphSAGE 4.21% 和 3.56%. 这表明基于状态转移序列的加权 DeepWalk 优于基于拓扑相似性的 GCN/GraphSAGE, 能更有效捕获正常流量的状态转移规律.

(2) 交互统计特征: 在会话表示中移除交互统计特征使得模型 AUC 降低 11.88%, 说明交互统计特征能增强对交互细粒度特征的捕获, 提升会话表示能力和检测有效性.

(3) 时间编码: 在获得会话的整体状态表示时, 移除时间编码导致 AUC 下降了 1.92%, 而使用原始绝对位置编码替代时间编码, 其 AUC 仍降低 1.54%, 说明时间编码对于引入流量数据的时序信息至关重要, 能有效提升模型性能.

3.4 鲁棒性分析

网络条件的变化可能导致异常情况, 包括数据包丢失和数据包到达顺序错乱. 为了评估模型在各种网络条件下所具有的鲁棒性和容错性, 本文在数据中引入了两种类型的可控噪声. (1) 数据包丢失测试. 本文在每个会话中随机丢弃 10%–80% 的数据包. (2) 数据包重新排序测试. 本文在每个会话中随机选择 10%–

80% 的数据包并打乱其顺序. 这两种因素模拟了现实世界中的网络中断, 有助于本文评估模型处理不完美网络环境的能力.

对于两种噪声类型对 DataCon-EMT 和 MCFP 数据集的实验结果分别显示在图 3 和图 4 中.

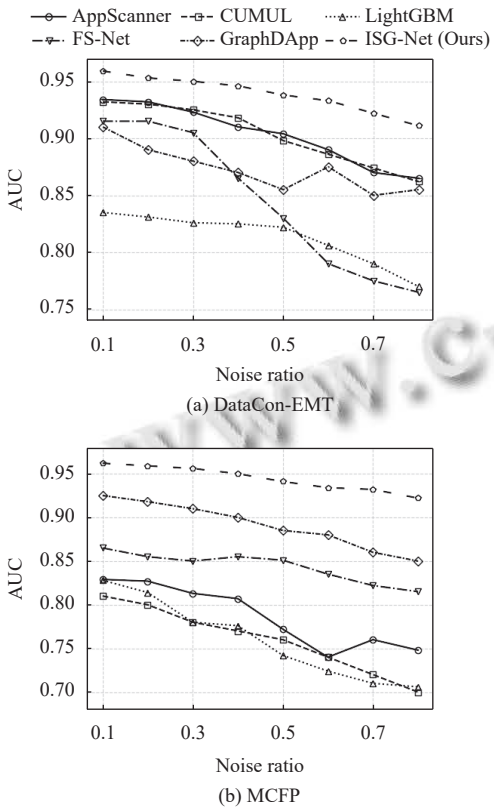

图 3 数据包丢失测试下不同模型的 AUC

结果显示, 在不同的噪声比率下, ISG-Net 表现出比其他方法更强的抗噪能力. 依赖统计特征的方法在不同程度上性能有所下降. 对于随机数据包丢失测试, ISG-Net 在 MCFP 数据集中始终表现最佳, 并且在不同噪声水平下, 在 DataCon-EMT 数据集中也表现出最佳或次优的结果. 当噪声比例达到 0.8 时, 它仍然保持高度有效性. 对于随机数据包重新排序测试, ISG-Net 在 DataCon-EMT 数据集上稳定运行, 并在高噪声比的情况下, 在 MCFP 数据集上也保证了有效性. 基于图的方法 GraphDApp 也使用图结构捕获交互信息进行流量识别. 然而, 由于其分析粒度基于单个数据包, 它对噪声更为敏感. 当噪声比为 0.8 时, 在 DataCon-EMT 数据集中, ISG-Net 的 AUC 得分比 GraphDApp 高出 14.24%. 这一结果表明, 通过基于交互粒度构建状态图, ISG-Net 具有更好的容错性.

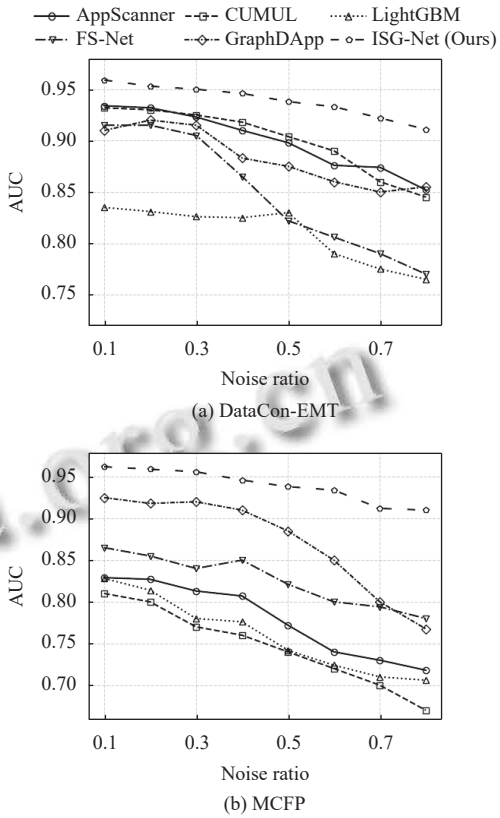

图 4 数据包重新排序测试下不同模型的 AUC

3.5 灵敏度分析

本文针对嵌入长度进行了灵敏度分析, 如图 5 所示.

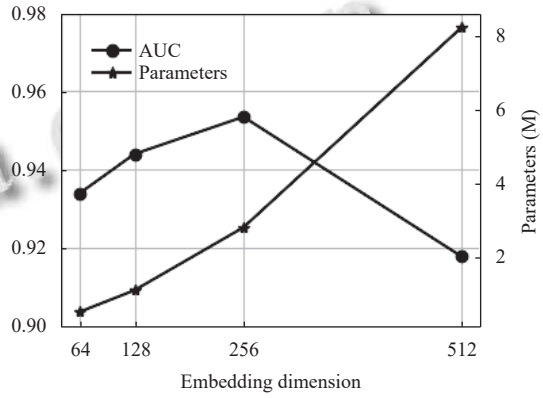

图 5 不同嵌入维度下的 AUC 和参数量

随嵌入长度增加 (64 增至 256), 模型性能提升; 但增至 512 时性能下降, 可能因过拟合影响泛化. 同时, 嵌入长度增加导致模型参数量增大. 实验表明, 嵌入长度为 256 时在性能与参数量间达到最佳平衡. 故最终选定嵌入长度为 256.

4 结语

本文提出了一种名为 ISG-Net 的模型, 用于构建

基于交互的流量状态图, 并利用元数据信息检测加密恶意流量, 该模型满足了在保护加密隐私利益的同时进行有效服务审计和监管监督的双重需求. 该模型从流量交互的角度出发, 通过建模交互状态来获取其全局表示, 然后将其与交互细粒度特征融合以生成会话的表示, 能够有效地挖掘不同会话的交互行为模式. 同时, 时序特征的引入进一步提高了方法的有效性, 从而做到精确地识别恶意流量. 在 3 个公共数据集上的评估证明了本文方法的有效性.

参考文献

1 Papadogiannaki E, Ioannidis S. A survey on encrypted network traffic analysis applications, techniques, and countermeasures. ACM Computing Surveys, 2022, 54(6): 123.

2 Google. Google transparency report. https://transparency-report.google.com/https/overview. [2025-05-01].

3 Treinen JJ, Thurimella R. A framework for the application of association rule mining in large intrusion detection infrastructures. Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection. Hamburg: Springer, 2006. 1–18.

4 Gu GF, Zhang JJ, Lee W. BotSniffer: Detecting botnet command and control channels in network traffic. Proceedings of the 15th Annual Network and Distributed System Security Symposium. 2008. 1–18.

5 Gohari M, Hashemi S, Abdi L. Android malware detection and classification based on network traffic using deep learning. Proceedings of the 7th International Conference on Web Research (ICWR). Tehran: IEEE, 2021. 71–77.

6 Vu AH, Nguyen-Khac MQ, Do XT, *et al*. A real-time evaluation framework for machine learning-based IDs. In: Balas VE, Solanki VK, Kumar R, eds. Recent Advances in Internet of Things and Machine Learning: Real-world Applications. Cham: Springer, 2022. 317–329.

7 Anderson B, McGrew D. Machine learning for encrypted malware traffic classification: Accounting for noisy labels and non-stationarity. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax: ACM, 2017. 1723–1732.

8 McGrew D, Anderson B. Enhanced telemetry for encrypted threat analytics. Proceedings of the 24th IEEE International Conference on Network Protocols (ICNP). Singapore: IEEE, 2016. 1–6.

9 Taylor VF, Spolaor R, Conti M, *et al*. AppScanner: Automatic fingerprinting of smartphone Apps from encrypted network traffic. Proceedings of the 2016 IEEE European Symposium on Security and Privacy (EuroS&P). Saarbruecken: IEEE, 2016. 439–454.

10 Wang W, Zhu M, Wang JL, *et al*. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI). Beijing: IEEE, 2017. 43–48.

11 Mirsky Y, Doitshman T, Elovici Y, *et al*. Kitsune: An ensemble of autoencoders for online network intrusion detection. Proceedings of the 25th Annual Network and Distributed System Security Symposium. San Diego: NDSS, 2018. 1–15.

12 Yu TD, Zou FT, Li LS, *et al*. An encrypted malicious traffic detection system based on neural network. Proceedings of the 2019 International Conference on Cyber-enabled Distributed Computing and Knowledge Discovery (CyberC). Guilin: IEEE, 2019. 62–70.

13 Rezaei S, Liu X. Deep learning for encrypted traffic classification: An overview. IEEE Communications Magazine, 2019, 57(5): 76–81. [doi: 10.1109/MCOM.2019.1800819]

14 Shapira T, Shavitt Y. FlowPic: A generic representation for encrypted traffic classification and applications identification. IEEE Transactions on Network and Service Management, 2021, 18(2): 1218–1232. [doi: 10.1109/TNSM.2021.3071441]

15 Fu CP, Li Q, Xu K. Detecting unknown encrypted malicious traffic in real time via flow interaction graph analysis. Proceedings of the 30th Annual Network and Distributed System Security Symposium. San Diego: NDSS, 2023. 1–18.

16 Shen M, Zhang JP, Zhu LH, *et al*. Accurate decentralized application identification via encrypted traffic analysis using graph neural networks. IEEE Transactions on Information Forensics and Security, 2021, 16: 2367–2380. [doi: 10.1109/TIFS.2021.3050608]

17 Diao ZL, Xie GG, Wang X, *et al*. EC-GCN: A encrypted traffic classification framework based on multi-scale graph convolution networks. Computer Networks, 2023, 224: 109614. [doi: 10.1016/j.comnet.2023.109614]

18 Ren GQ, Cheng G, Fu N. Accurate encrypted malicious traffic identification via traffic interaction pattern using graph convolutional network. Applied Sciences, 2023, 13(3): 1483.

[doi: 10.3390/app13031483]

19 Anderson B, McGrew D. Identifying encrypted malware traffic with contextual flow data. Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security. Vienna: ACM, 2016. 35–46.

20 de Lucia MJ, Cotton C. Detection of encrypted malicious network traffic using machine learning. Proceedings of the 2019 IEEE Military Communications Conference (MILCOM). Norfolk: IEEE, 2019. 1–6.

21 Stergiopoulos G, Talavari A, Bitsikas E, *et al*. Automatic detection of various malicious traffic using side channel features on TCP packets. Proceedings of the 23rd European Symposium on Research in Computer Security. Barcelona: Springer, 2018. 346–362.

22 Meghdouri F, Zseby T, Iglesias F. Analysis of lightweight feature vectors for attack detection in network traffic. Applied Sciences, 2018, 8(11): 2196. [doi: 10.3390/app8112196]

23 Chen ZX, Yan QB, Han HB, *et al*. Machine learning based mobile malware detection using highly imbalanced network traffic. Information Sciences, 2018, 433–434: 346–364.

24 Dong C, Lu ZG, Cui ZL, *et al*. MBTree: Detecting encryption RATs communication using malicious behavior tree. IEEE Transactions on Information Forensics and Security, 2021, 16: 3589–3603. [doi: 10.1109/TIFS.2021.3071595]

25 Dodia P, AlSabah M, Alrawi O, *et al*. Exposing the rat in the tunnel: Using traffic analysis for Tor-based malware detection. Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. Los Angeles: ACM, 2022. 875–889.

26 Fu CP, Li Q, Shen M, *et al*. Realtime robust malicious traffic detection via frequency domain analysis. Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2021. 3431–3446.

27 Yu LX, Tao J, Xu YF, *et al*. TLS fingerprint for encrypted malicious traffic detection with attributed graph kernel. Computer Networks, 2024, 247: 110475. [doi: 10.1016/j.comnet.2024.110475]

28 戚子健, 柳毅. 基于双向 GRU 和 CNN 的恶意网络流量检测方法. 计算机应用与软件, 2024, 41(12): 334–340, 366.

29 Liu C, He LT, Xiong G, *et al*. FS-Net: A flow sequence network for encrypted traffic classification. Proceedings of the 2019 IEEE Conference on Computer Communications. Paris: IEEE, 2019. 1171–1179.

30 Lin XJ, Xiong G, Gou GP, *et al*. ET-BERT: A contextualized datagram representation with pre-training Transformers for encrypted traffic classification. Proceedings of the 2022 ACM Web Conference. ACM, 2022. 633–642.

31 Hang ZJ, Lu YL, Wang YJ, *et al*. Flow-MAE: Leveraging masked autoencoder for accurate, efficient and robust malicious traffic classification. Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses. Hong Kong: ACM, 2023. 297–314.

32 Cai SH, Tang H, Chen JF, *et al*. CDDA-MD: An efficient malicious traffic detection method based on concept drift detection and adaptation technique. Computers & Security, 2025, 148: 104121.

33 Wang W, Shang YY, He YZ, *et al*. BotMark: Automated botnet detection with hybrid analysis of flow-based and graph-based traffic behaviors. Information Sciences, 2020, 511: 284–296. [doi: 10.1016/j.ins.2019.09.024]

34 Zhao D, Traore I, Sayed B, *et al*. Botnet detection based on traffic behavior analysis and flow intervals. Computers & Security, 2013, 39: 2–16.

35 Huoh TL, Luo Y, Li PL, *et al*. Flow-based encrypted network traffic classification with graph neural networks. IEEE Transactions on Network and Service Management, 2023, 20(2): 1224–1237. [doi: 10.1109/TNSM.2022.3227500]

36 Zhao J, Li Q, Han Z, *et al*. ReTrial: Robust encrypted malicious traffic detection via discriminative relation incorporation and misleading relation correction. IEEE Transactions on Information Forensics and Security, 2025, 20: 677–692. [doi: 10.1109/TIFS.2024.3515821]

37 Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online learning of social representations. Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2014. 701–710.

38 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. Proceedings of the 1st International Conference on Learning Representations. Scottsdale: OpenReview.net, 2014. 1–12.

39 Ashish V. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

40 DataCon. DataCon2020——加密恶意流量数据集. https://datacon.qianxin.com/opendata/openpage?resourcesId=6. (2021-11-11).

41 Wang W, Zhu M, Zeng XW, *et al*. Malware traffic

classification using convolutional neural network for representation learning. Proceedings of the 2017 International Conference on Information Networking (ICOIN). Da Nang: IEEE, 2017. 712–717.

42 Stratosphere. Stratosphere laboratory datasets. https://www.stratosphereips.org/datasets-overview. (2015-03-13)[2025-05-15].

43 Panchenko A, Lanze F, Pennekamp J, *et al*. Website fingerprinting at Internet scale. Proceedings of the 23rd Annual Network and Distributed System Security Symposium. San Diego, 2016. 23477.

44 Jiang MH, Gou GP, Shi JZ, *et al*. I know what you are doing with remote desktop. Proceedings of the 38th IEEE International Performance Computing and Communications Conference (IPCCC). London: IEEE, 2019. 1–7.

45 Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2017. 1–14.

46 Hamilton WL, Ying Z, Leskovec J. Inductive representation learning on large graphs. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 1025–1035.

(校对责编: 张重毅)