计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
[doi: 10.15888/j.cnki.csa.010278] [CSTR: 32024.14.csa.010278]

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

预训练模型与因果挖掘结合的多步攻击检测①

张璐璐, 杜学绘, 王文娟

(解放军信息工程大学 密码工程学院, 郑州 450001)
通信作者: 杜学绘, E-mail: dxh37139@163.com

摘　要: 针对复杂多步攻击发现中告警多源异构、表述不统一及告警间因果关系难以准确识别的问题, 提出了一种预训练模型与因果挖掘相结合的多步攻击检测方法. 首先, 利用经过 SimCSE 优化后的 BERT 模型对告警文本进行语义向量提取并进行相似性计算, 实现告警表述统一化处理, 降低告警冗余. 其次, 依据时间邻近性、地址相关性与严重性递增等关联规则构建初始因果关系图. 进一步结合时间感知图注意力网络与 NOTEARS 因果结构学习方法, 以修正告警节点间的因果关系, 从而剔除虚假关联边, 识别完整的复杂多步攻击. 实验结果表明, 所提方法能够有效提升对复杂多步攻击的检测准确率.

关键词: 多步攻击; 因果关系挖掘; 深度学习; 预训练语言模型; 图注意力网络

引用格式: 张璐璐,杜学绘,王文娟.预训练模型与因果挖掘结合的多步攻击检测.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/10278.html

Multi-step Attack Detection Combining Pre-trained Model with Causal Mining

ZHANG Lu-Lu, DU Xue-Hui, WANG Wen-Juan

(School of Cryptographic Engineering, PLA Information Engineering University, Zhengzhou 450001, China)

Abstract: To address the problems of multi-source heterogeneous alerts, inconsistent alert descriptions, and difficulties in identifying causal relationships between alerts in complex multi-step attack detection, this study proposes a detection method that combines pre-trained models with causal mining. First, a BERT model optimized with SimCSE is used to extract semantic vectors from alert texts and compute their similarity. This achieves unified alert representation and reduces alert redundancy. Second, an initial causal graph is constructed based on association rules, including temporal proximity, address correlation, and severity escalation. A time-aware graph attention network is then combined with the NOTEARS causal structure learning method to refine causal relationships among alert nodes. This eliminates false correlation edges and identifies complete complex multi-step attacks. Experimental results show that the proposed method effectively improves the detection accuracy of complex multi-step attacks.

Key words: multi-step attack; causal relationship mining; deep learning; pre-trained language model; graph attention network (GAT)

近年来, 随着全球数字化进程的不断加速和网络攻击面的持续扩张, 网络攻击在频率与复杂度上均呈现显著增长的态势. 2025 年, 国家互联网应急中心 CNCERT 发布的报告指出, 境外组织持续针对我国关键机构和企业发起网络攻击活动, 相关攻击行为不再局限于单一技术手段, 而是通过多种攻击技术的协调配合, 实施包括分布式拒绝服务 (distributed denial of service, DDoS)、高级可持续威胁攻击 (advanced persistent threat, APT) 等复杂多步攻击. 复杂多步攻击通常由多个相互关联的攻击步骤构成, 具有更强的隐蔽性以及

① 基金项目: 国家自然科学基金 (62102449)
收稿时间: 2026-03-02; 修改时间: 2026-03-30; 采用时间: 2026-04-14; csa 在线出版时间: 2026-08-21

更长的攻击执行周期, 从而导致潜在危害的加剧和检测难度的提升[1]. 因此, 准确发现复杂多步攻击行为已成为网络安全领域关注的重点, 也是业界的重点研究方向.

入侵检测系统 (intrusion detection system, IDS) 可以用于检测网络中的攻击行为. 然而, 传统 IDS 产生的告警通常仅反映单一、局部的攻击事件, 缺乏对告警间逻辑关联的深度挖掘能力, 导致难以从大量离散告警中发现隐藏其中的复杂多步攻击行为[2]. 此外, 由于不同类型的 IDS 在告警生成机制、字段结构及文本描述方式等方面存在显著差异, 所产生的告警呈现出数量庞大且多源异构的特征. 尽管这些告警在表征或描述上差异较大, 但其所反映的语义却具有相似性. 这种语义相似但表达异构的特性使得告警缺乏统一的告警语义表述, 从而阻碍了多源异构告警的语义理解与关联分析[3,4]. 因此, 复杂多步攻击的发现存在两方面的问题: 一是如何将多源异构、表述不一但语义相近的告警映射到统一的语义向量空间中, 实现告警语义的一致化表示; 二是如何挖掘告警之间的潜在逻辑关联, 从局部、零散的告警中识别出潜在的复杂多步攻击. 传统方法多依赖于基于关键词匹配的告警表述统一方法或基于规则模板的告警关联分析方法[5,6], 但这些方法侧重表层特征, 难以捕捉文本背后的深层语义联系, 也缺乏对攻击行为间因果关系的显式建模, 难以支撑对复杂多步攻击的准确发现. 随着预训练语言模型在自然语言处理领域的发展, 其卓越的语义理解与表示能力为告警语义的统一表达提供了新的技术路径[7]; 同时, 告警间因果关系的挖掘在揭示事件间逻辑依赖方面展现出独特优势[8]. 因此, 结合预训练语言模型的语义建模能力与因果关系挖掘方法, 成为探索复杂多步攻击发现的新方向. 为此, 本文提出一种预训练模型与因果挖掘相结合的多步攻击检测方法, 旨在从大量多源异构、表述不一致的告警中识别攻击者潜在的多步攻击行为. 主要贡献如下.

● 提出一种基于 BERT-SimCSE 的告警语义统一表示方法, 通过经 SimCSE (simple contrastive learning of sentence embeddings) 优化后的 BERT (bidirectional encoder representations from Transformers) 模型获得更具判别性的句向量表示. 随后, 使用优化后的 BERT 对告警文本进行深度语义编码, 并结合语义相似性计算, 将多源异构、表述不一的告警合并为统一的告警表述, 为后续告警关联分析与复杂多步攻击发现提供高质量的语义输入.

● 提出一种基于因果关系挖掘的复杂多步攻击发现方法, 首先通过将离散的告警根据预设规则构建为初始因果关系图, 随后引入基于时间感知的图注意力网络 (time-aware graph attention network, TA-GAT) 对告警所处的攻击阶段进行判别, 并采用 NOTEARS 算法对告警间的因果强度进行计算, 对因果关系图进行修正, 去除虚假关联边, 从而得到符合攻击逻辑的多步攻击场景图, 实现对攻击者真实意图及其复杂多步攻击的有效发现.

1 相关工作

1.1 告警语义统一表达

告警语义统一表达旨在将语义相近但表达不同的告警进行相似度度量, 对高相似告警进行合并实现语义一致化处理. 例如, Chen 等人[9]提出基于 GloVe 词向量的告警日志表示方法, 通过结合全局单词匹配和局部上下文信息, 在一定程度上减少了语法差异对语义提取的负面影响. 然而, Glove 作为一种静态词嵌入技术, 其局限在于无法根据具体上下文动态调整词义表示, 导致在处理多义词语和复杂语境时存在语义歧义问题. Ryciak 等人[10]提出一种改进的 LogEvent2Vec 方法, 但该方法主要依赖关键词匹配和简单的句法分析技术, 在语义理解层面较为浅显, 难以准确识别语义相似但表达方式差异较大的告警. Xiao 等人[11]则采用 fastText 模型来处理告警数据中的未知事件, 在一定程度上提升了语义相似性计算的鲁棒性. 然而, 该方法在捕获长距离上下文依赖方面仍存在不足. 随着预训练语言模型的发展, BERT 模型通过学习告警文本的上下文语义, 在告警语义建模任务中展现了显著优势. Seyyar 等人[12]将 BERT 用于网络流量告警的语义分析, 实现了对 HTTP/HTTPS 相关攻击行为的检测. Satvat 等人[13]使用 BERT 从威胁情报报告中精准提取描述攻击行为的句子, 为告警语义提取与复杂多步攻击的发现提供了高质量的语义基础. LAnoBERT[14]和 LogBERT[15], 通过学习告警日志的上下文信息, 实现了告警语义表达与攻击行为关联分析. 然而, 上述研究中直接采用标准 BERT 模型生成的向量在语义空间中往往分布不均, 难以充分区分不同语义的告警文本. 由于缺乏显式的对比学习机制, 语义相近但来源异构的告警文本在向

量空间中并不能有效聚集, 导致在进行告警语义统一表示时, 难以精准识别语义相关但描述各异的告警事件. 因此, 针对上述不足, 本文在标准 BERT 中引入 SimCSE 框架, 通过对比学习增强语义表示的判别能力, 使语义相近的告警在向量空间中更加紧密, 从而提升告警语义统一表达的效果.

1.2 告警关联关系挖掘

近年来, 告警关联关系的发现主要依赖攻击模板和数据挖掘方法进行. 基于攻击模板的方法高度依赖先验知识, 难以适应当前动态且复杂的网络环境[16]. 而基于数据挖掘的方法虽需要较少人工干预, 特别适合处理大规模数据, 但其主要依赖告警间的频繁模式来揭示相关性, 其关联结果往往仅反映统计相关性, 难以揭示攻击步骤间的因果关系[17,18].

进一步地, 为了从深层逻辑上发现复杂多步攻击, 基于因果关系挖掘的告警关联方法被广泛提出. 该方法认为攻击的不同步骤之间通常存在明确的因果联系, 攻击者的每一步行为都为下一步创造了先决条件. 因此, 其通过挖掘攻击步骤间的因果关系, 有助于识别复杂多步攻击. Barzegar 等人[19]从入侵语义的角度探讨了攻击步骤与 IDS 告警之间的逻辑关系, 但其语义推理能力较为初步. Zang 等人[20]则采用结构化的威胁信息表达对异构威胁情报进行格式化, 并设计语义推理规则以捕捉威胁情报内部的因果关系. Khosravi 等人[21]则提出了一种基于因果分析的 APT 实时检测方法, 通过计算主机感染分数并结合动态规划算法, 应对 APT 攻击的持续性特征. 然而, 这类方法普遍存在两个不足: 一是往往会忽略攻击的上下文语义信息, 对同类型告警采用统一处理方法, 难以区分其在不同攻击步骤中的实际含义. 例如, 端口扫描在攻击立足点建立阶段可能与漏洞发现或利用密切相关, 而在横向移动阶段则更可能预示数据泄露的准备活动. 二是因果图构建过程中会产生虚假因果关系. 这种虚假边通常源于时序关系处理不当或因果依赖挖掘不准确, 导致缺乏真实逻辑关联的告警节点被错误地连接. 因此, 本文通过计算因果强度来识别虚假关系, 利用基于时间感知的图注意力网络 TA-GAT 并结合 NOTEARS 因果结构学习算法对告警间的因果强度进行计算, 有效去除虚假关联边.

综上所述, 现有研究在统一告警语义表达和告警关联关系挖掘方面取得了诸多进展. 然而, 这些方法在处理动态网络环境、复杂上下文依赖以及因果关系深度挖掘等方面仍存在局限. 为此, 本文提出一种预训练模型与因果挖掘相结合的多步攻击检测方法. 首先, 利用经 SimCSE 优化的 BERT 对告警文本进行语义编码, 将多源异构告警映射到统一的语义向量空间, 实现语义相近而表述不同告警的一致化表示; 在此基础上, 进一步挖掘并分析告警事件之间的因果关系, 刻画告警间的潜在逻辑关联, 从而实现对复杂多步攻击过程中各个阶段的有效发现.

2 基于 BERT-SimCSE 的告警语义统一表达方法

为了规范处理, 本文使用一个多元组对每个告警事件进行定义.

定义 1 (告警). 告警 $a_i = (time, name, type, severity, sip, sport, dip, dport)$, 其中, *time* 表示检测到告警的时间; *name* 表示告警名称; *type* 表示告警的攻击类型; *severity* 表示告警的严重程度, 通常划分为高、中、低这 3 个等级; *sip* 和 *sport* 分别指攻击者的源 IP 地址和源端口号; *dip* 和 *dport* 分别为攻击者的目的 IP 地址和目的端口号.

在此基础上, 将告警按照时间顺序依次排列, 构建一个有序的告警序列 $A = \{a_1, a_2, \cdots, a_n\}$. 其中, a_i 表示第 i 个告警事件.

在实际网络攻击中, 攻击者通常采用多步攻击以达成其攻击目标. 因此, 一个完整的有序告警序列中往往包含了由不同攻击步骤所触发的多个告警事件. 为精准界定攻击过程、降低分析复杂度, 本文将时间窗口 Δt 设置为 1 h, 对告警序列 $A = \{a_1, a_2, \cdots, a_n\}$ 进行子序列的划分. 首先计算相邻告警 a_i 与 a_{i+1} 的时间间隔 $\Delta t_i = t_{i+1} - t_i$, 其中 t_i 为告警 a_i 的发生时间. 若 $\Delta t_i > \Delta t$, 表明前后两个告警可能隶属于不同的攻击步骤, 此时在 a_i 与 a_{i+1} 之间设置划分点. 最后按划分点拆分告警序列, 得到若干独立告警子序列.

在网络安全告警中, 告警名称 (*name*) 和告警类型 (*type*) 字段通常蕴含丰富的语义信息. 但由于其文本描述的异构性, 例如“Port scan”或“RPC portmap”, 虽然文本描述上具有差异性, 但语义上均表示扫描侦察行为; 同样地, “login”和“access”可能均表示未授权访问行为. 尽管这些告警在语义上高度相似, 但由于其文本描述各异, 传统的基于规则或关键词匹配的告警语义统一表达

方法无法捕捉文本背后的深层语义信息, 难以获取异构告警的一致化语义表达, 导致安全人员面临告警疲劳问题, 从而影响对攻击行为的快速响应和有效防御. 随着预训练语言模型在自然语言处理领域的迅速发展, 其在大规模语料库上的预训练形成的深层语义理解能力, 为解决告警文本异构性问题提供了新的技术路径. 与传统的单向语言模型不同, 基于 Transformer 架构的 BERT 模型通过双向编码机制、掩码语言模型 (masked language model, MLM) 和下一句预测 (next sentence prediction, NSP) 任务, 显著提升了模型对复杂语义及上下文关系的理解和表达能力. 然而, 在实际语义向量表示中, 直接使用标准 BERT 模型导出的句向量常呈现出各向异性分布现象, 即大量向量在语义空间中聚集于少数方向, 导致不同语义的文本表示缺乏区分度. 这种分布特性使得相似度度量失去判别性, 余弦相似度无法准确反映句子间的真实语义相关性.

针对上述问题, 引入 SimCSE 框架对 BERT 进行优化, 以实现告警语义信息的精准向量化表示. SimCSE 的核心思想是通过对比学习, 拉近语义相似样本 (正样本对) 的距离, 推开语义无关样本 (负样本对) 的距离, 从而获得分布更均匀、区分度更高的语义向量空间. 具体操作如图 1 所示.

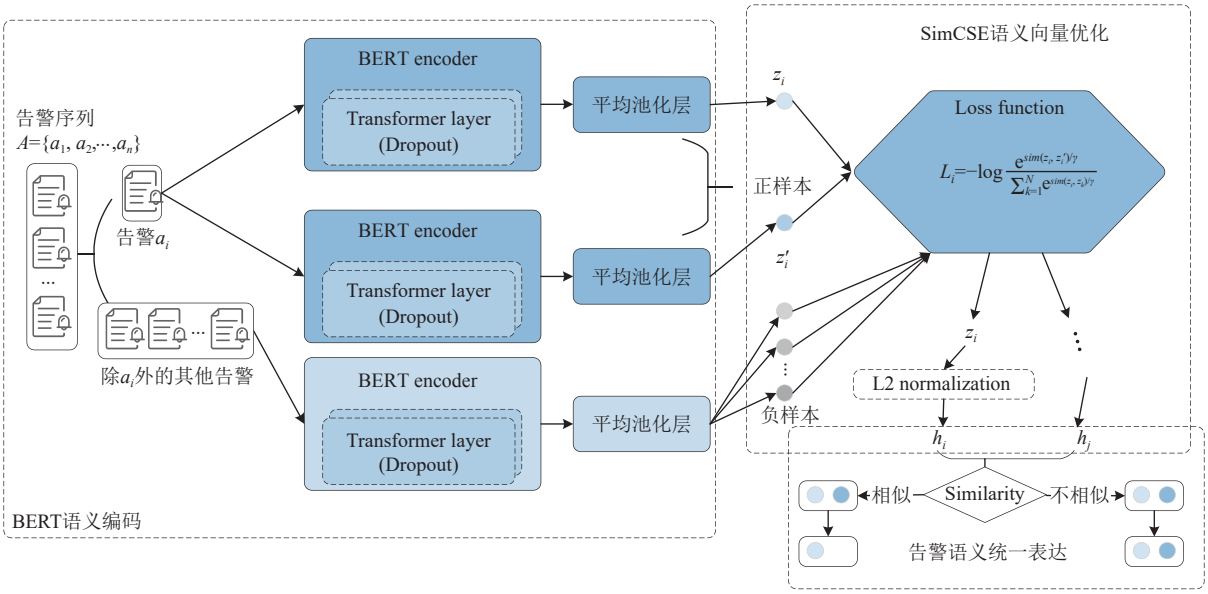

图 1 基于 BERT-SimCSE 的告警语义统一表达

(1) BERT 语义编码

对告警的名称和类型字段进行文本清洗与拼接, 构建输入文本, 作为语义编码的基础. 随后, 将其输入 BERT 模型进行编码. 为了获取包含更丰富全局信息的语义表示, 本文采用平均池化方式对 BERT 最后一层的隐藏状态进行聚合, 生成初始句向量z_i.

(2) SimCSE 语义向量优化

在获取告警文本的 BERT 编码向量后, 利用 SimCSE 的对比学习思想优化语义向量. SimCSE 通过利用 BERT 内部 Transformer 层的 Dropout (随机失活) 机制, 通过其随机性对同一输入文本进行编码, 能够生成两个存在细微差异的向量表示z_i和z_i', 二者构成正样本对. 而同一子序列内其他不同告警生成的向量则作为负样本. 模型通过最小化对比损失函数, 使正样本向量在语义空间中相互靠近、负样本远离, 从而提升语义表示的区分度.

具体地, 对于一个包含 N 个告警文本的子序列, 对于其中第 i 个告警文本, 其正样本为自身的两次随机编码结果, 负样本为其他 N–1 个样本, 损失函数定义如式 (1) 所示:

$$L_i = -\log \frac{e^{sim(z_i,z_i')/\gamma}}{\sum_{k=1}^{N} e^{sim(z_i,z_k)/\gamma}} \tag{1}$$

其中, z_i和z_i'为正样本对的向量表示, z_k表示 batch 中第 k 个文本生成的负样本向量; $sim(\cdot)$表示余弦相似度计算; γ表示温度超参数, 用于控制相似度分布的平滑

度, γ越小, 模型对正负样本距离的区分越敏感. 通过最小化该损失函数, 模型能够学习到更加紧凑且区分度高的告警语义空间. 最后, 对优化后的向量进行 L2 归一化处理, 以增强不同样本之间计算稳定性. 经过此步骤, 得到最终的告警语义向量.

(3) 语义统一表达

基于上述模型生成的语义向量, 本文采用余弦相似度度量方法来量化告警间的语义相似程度. 对于任意告警a_i和a_j, 设其经 BERT-SimCSE 优化后的语义向量分别为h_i和h_j, 相似度计算方法如式 (2) 所示:

$$sim(a_i,a_j)=\frac{h_i\cdot h_j}{\|h_i\|_2\cdot\|h_j\|_2} \tag{2}$$

设定相似度阈值θ: 当$sim(a_i,a_j)\geqslant\theta$时, 判定$a_i$和$a_j$在语义层面描述了同一或高度相关的安全事件, 并将其统一表示为一个告警; 否则, 视为语义差异较大的告警并分别保留. 该方法通过对语义相似告警进行合并, 减少语义冗余和表述差异的影响, 缓解告警来源异构带来的语义不一致问题, 为后续的关联分析提供高质量的语义输入.

综上所述, 本文采用一种基于 BERT-SimCSE 的告警语义统一表示方法. 该方法首先利用经 SimCSE 优化的预训练语言模型对告警文本进行语义编码, 并计算告警间的语义相似度; 当相似度超过预设阈值时, 认为对应告警在语义层面描述了同一或高度相关的安全事件, 将其进行合并, 从而实现语义一致化表示.

3 基于因果关系挖掘的复杂多步攻击发现

为了进一步发现复杂多步的攻击, 亟需在此基础上发现告警之间的潜在因果关系, 并通过有效关联揭示攻击路径.

3.1 因果关系初建

在复杂多步攻击中, 各告警之间往往并非相互独立, 而是存在一定的因果依赖关系, 即前一攻击步骤是后一攻击步骤的前提, 后一攻击步骤则是前者的结果. 这种因果关系不仅体现在攻击行为的逻辑先后顺序上, 也体现在时间、空间等维度上的紧密关联. 基于此, 本文从时间、空间和严重性这 3 个方面对告警间的依赖关系进行建模. 若两个告警a_i和a_j (a_i发生于a_j之前) 同时满足以下 3 条关联规则, 则认为两者之间存在潜在的因果关系.

● 时间 (*time*) 邻近性: 攻击者的多步攻击行为往往遵循一定的时间顺序, 即后续攻击步骤通常紧随前一攻击步骤而发生, 其时间间隔集中于特定范围内. 因此, 若两者之间的时间间隔 t 小于预定义的时间窗口Δt, 则构成因果关系的必要条件.

● 地址 (IP 地址) 相关性: 攻击者通常会利用已攻陷主机的 IP 地址作为跳板进而攻击其他目标. 因此, 若告警a_i的目的 IP 地址 *dip* 与a_j的源 IP 地址 *sip* 相同, 则可能表明告警a_i的攻击目标是a_j, 即攻击可能从当前告警的目的地传播到下一个告警的源.

● 严重性 (*severity*) 递增: 在多步攻击过程中, 攻击者通常会遵循从低严重性到高严重性的攻击路径. 这种行为模式使得攻击路径中后续告警的严重性往往高于前序告警. 因此, 若a_j的告警严重性等级高于a_i, 则两者之间可能存在因果关联.

综上所述, 当两个告警满足以上 3 个关联规则时则认为二者之间可能存在因果关系.

基于上述关联规则, 本文采用有向图模型来形式化描述多步攻击场景中的潜在因果依赖.

定义 2 (初始因果关系子图). 初始因果关系子图 $G=(V,E)$. 其中节点集合 V 代表告警事件, 边集合 E 则表示告警之间潜在的因果关联.

当任意两个节点V_i和V_j同时满足以上关联规则, 则在它们之间添加一条有向边E_{ij}, 边的方向反映了告警间的因果逻辑, 即作为前因的节点V_i指向作为后果的节点V_j, 以此表明这两个节点之间存在潜在的因果关系. 通过遍历所有告警并应用上述关联规则, 便可以构建一个包含所有潜在攻击路径的初始因果关系子图 G. 然而, 依据这些规则构建的初始图也可能会导致一些不相关的告警被错误连接. 因此, 需要对初始关联结果再进行判断和优化.

尽管初始因果关系子图已经初步勾勒出攻击路径, 但其中仍存在由良性系统活动或设备误报产生的无关告警节点, 导致其在构建因果关系图的过程中偶然满足关联规则而形成虚假边. 此外, 单个告警的威胁性依赖于其在攻击序列中的上下文语义, 例如漏洞扫描告警, 若后续为管理员的合规性检查或漏洞修复, 则属于正常行为; 但若后续涉及利用漏洞提升权限, 则被判定为恶意攻击. 这些由无关节点与上下文误判产生的虚假边, 会严重干扰对复杂多步攻击的精确发现. 因此, 需

对因果关系进行修正, 以剔除干扰并识别真实的攻击.

3.2 因果关系修正

由于图注意力网络 (graph attention network, GAT) 可对图中的节点进行阶段分类, 捕捉节点间的潜在关联性和上下文依赖性, 从而实现攻击路径的精确刻画. 相较于图采样聚合算法 (graph sample and aggregate, GraphSAGE) 或图卷积神经网络 (graph convolutional network, GCN) 等其他图神经网络 (graph neural network, GNN) 变体, GAT 通过自注意力机制为邻居节点分配不同的权重, 更好地捕捉节点关联性. 考虑到告警数据的时间序列特性, 时间上更接近的节点往往具有更强的因果关联性, 传统的 GAT 在计算注意力权重时仅依赖于节点特征和图结构信息, 忽略了时间维度对告警关联的重要性. 为此, 本文提出一种基于时间感知的图注意力网络 (time-aware graph attention network, TA-GAT), 通过将时间间隔作为注意力计算的附加因素, 优先关注时间接近的告警节点, 从而增强模型对攻击序列的捕捉能力和阶段分类的准确性. 具体而言, 该机制在计算节点间的注意力权重时, 引入节点间的时间间隔作为惩罚因子. 对于时间间隔较小的节点对, 注意力权重会得到提升, 以反映其更高的潜在因果关联; 反之, 对于时间间隔较大的节点对, 注意力权重会被适当降低, 以减少无关告警对模型预测的干扰. 核心步骤如下所述.

(1) 计算注意力系数

对于初始因果关系子图$G=(V,E)$中的任意节点V_i与其任一邻居节点V_j, 其时间间隔$\Delta t_{ij}=\left|t_i-t_j\right|$. 设计一个时间衰减函数$f\left(\Delta t_{ij}\right)=\exp\left(-\lambda\cdot\Delta t_{ij}\right)$, 其中, λ是一个可调参数, 用于控制时间间隔对注意力的衰减速度. 如图 2 所示.

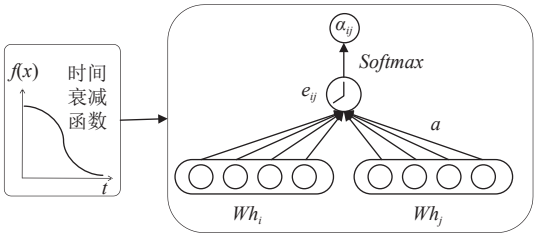

图 2 引入时间间隔的注意力系数

模型首先计算一个代表V_j对V_i重要性的原始注意力分数$e_{ij}=a\left(\left[Wh_i,Wh_j\right]\right)$. 其中, h_i和h_j分别是节点V_i和V_j的特征向量; a是一个可学习的注意力权重向量; W为权重矩阵.

为了保留图中的结构信息, 不需要计算节点之间的所有注意力分数, 而是利用 *Softmax* 函数对一个节点的所有邻居的注意力分数进行归一化处理, 得到最终的时间感知注意力系数, 如式 (3) 所示:

$$\begin{aligned}\alpha_{ij}&=Softmax_j\left(e_{ij}\right)\\&=\frac{\exp\left(LeakyReLU\left(e_{ij}\right)+\log\left(f\left(\Delta t_{ij}\right)\right)\right)}{\sum_{k\in N_i}\exp\left(LeakyReLU\left(e_{ik}\right)+\log\left(f\left(\Delta t_{ik}\right)\right)\right)}\end{aligned}\tag{3}$$

其中, $\log\left(f\left(\Delta t_{ij}\right)\right)=-\lambda\cdot\Delta t_{ij}$将时间间隔转化为对注意力系数的惩罚项. 时间间隔$\Delta t_{ij}$越大, $\log\left(f\left(\Delta t_{ij}\right)\right)$越小, 从而降低$\alpha_{ij}$的值, 使模型更加关注时间上接近的节点; N_i是节点V_i在图中所有邻居节点的集合.

(2) 加权求和

根据计算好的不同节点之间的注意力系数α_{ij}来得到每个节点的输出向量, 为了稳定注意力系数的学习过程, 使用 K 头注意力机制来更新特征, 之后将得到的结果进行拼接, 得到新的特征向量h_i'. 计算过程如式 (4) 所示:

$$h_i'(K)=\Big\|_{k}^{K}\sigma\left(\sum_{j\in N_i}\alpha_{ij}^{k}W^{k}h_j\right)\tag{4}$$

其中, α_{ij}^{k}是第 k 组注意力机制计算得出的权重系数. N_i是节点 i 的邻居节点集合; h_i'是节点 i 更新后的特征表示; W 是可学习的权重矩阵, 用于对邻居节点的特征h_j进行线性变换; σ是一个非线性激活函数. 对注意力特征取平均值得到输出结果h_i'. 计算过程如式 (5) 所示:

$$h_i'=\sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in N_i}\alpha_{ij}^{k}W^{k}h_j\right)\tag{5}$$

以 K=3 为例, 多头注意力结构如图 3 所示. 每个节点在通过 TA-GAT 网络聚合邻居信息得到对应的富含上下文的特征表示h_i'后, 利用输出层将节点分类为不同的攻击阶段或标记为非攻击阶段, 为后续分析提供重点. 然而, 尽管当前因果关系图已经明确了攻击节点与正常节点, 但其边之间并非严格的因果关系, 无法直接还原攻击的完整路径. 为此, 本文引入因果发现方法对图进行精炼分析, 以挖掘攻击步骤间的真正因果关系并确定事件间的因果顺序.

(3) 识别虚假边

NOTEARS 算法作为一种基于连续优化的因果结

构学习方法, 相较于 PC (Peter-Clark) 算法等基于条件独立性测试的方法, 无需遍历大量条件集进行独立性检验, 而是通过优化权重矩阵直接量化节点间的因果强度, 同时通过无环约束函数自动保证输出图的有向无环特性, 能够高效且精准地挖掘告警节点间的因果依赖. 具体而言, 本文结合 TA-GAT 提供的节点阶段分类结果、注意力权重, 将其融入 NOTEARS 的权重矩阵优化过程, 使模型优先聚焦攻击节点间的因果关系, 最终输出一个精确且全面的攻击因果结构.

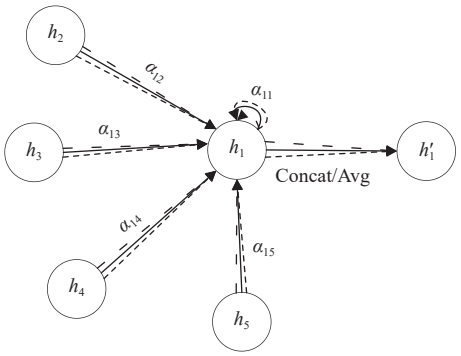

图 3 多头注意力结构图

对于初始因果关系子图$G=(V,E)$, 定义节点间的因果权重矩阵$W\in\mathbb{R}^{n\times n}$ (n 为节点数), 其中W_{ij}表示节点V_i对V_j的因果影响强度, 若$W_{ij}\neq 0$, 则说明V_i对V_j存在直接因果关系; W_{ij}越大, 因果关联越强.

结合 TA-GAT 的输出特征, 将 NOTEARS 的损失函数设计为注意力引导的损失函数, 如式 (6) 所示:

$$L(W)=\frac{1}{2n}\sum_{j=1}^{n}\left\|h_j'-\sum_{j=1}^{n}W_{ij}h_i'\right\|_2^2 +\lambda_1\sum_{i=1}^{n}\sum_{j\in N_i}\left(1-\alpha_{ij}\right)\left|W_{ij}\right|+\lambda_2\|W\|_1 \tag{6}$$

其中, h_i'为 TA-GAT 输出的节点V_i特征向量, α_{ij}为 TA-GAT 计算的节点V_j对V_i的时间感知注意力权重, N_i为节点V_i的邻居集合, λ是两个损失项的平衡系数.

为避免因果关系图出现循环, 引入 NOTEARS 的无环检测函数作为硬约束, 如式 (7) 所示:

$$h(W)=\mathrm{tr}(\exp(W\circ W))-n=0 \tag{7}$$

其中, $\mathrm{tr}(\cdot)$为矩阵迹运算, $\exp(\cdot)$为矩阵指数运算, $\circ$为哈达玛积. 当且仅当$h(W)=0$时, 权重矩阵W对应的图为无环图.

结合损失函数与无环约束, 构建最终优化目标, 如式 (8) 所示:

$$\min_w L(W)\quad \text{s.t. } h(W)=0 \tag{8}$$

上述约束优化问题采用 NOTEARS 中提出的增强拉格朗日优化策略进行求解, 通过引入对偶变量和惩罚参数, 将原问题转化为一系列无约束子问题并迭代优化. 当无环约束函数值收敛且目标函数不再显著变化时, 终止优化过程, 得到最终的因果权重矩阵W^*.

基于 TA-GAT 的节点分类结果, 对最优权重矩阵W^*进行过滤, 对于任意攻击节点V_i、V_j, 若$\left|W_{ij}^*\right|<\varepsilon$, ε为设定的因果强度阈值, 则说明二者无直接因果关系, 移除边E_{ij}.

通过上述过程, 成功对初始因果关系子图进行因果关系修正, 将虚假相连的边进行去除, 得到能反映真实因果关系的图.

3.3 因果关系合并

通过上述操作, 在划分过的告警子序列中挖掘出了相对应的因果关系子图, 然而单个的因果关系子图往往仅反映了攻击的一个局部阶段. 一方面, 多步攻击通常具有长周期的特点, 攻击行为可能分散到了不同的子图中; 另一方面, 攻击者会通过横向移动从初始访问节点向网络内部渗透, 最终跳跃到目标主机, 导致攻击行为会分散在不同子图中. 为获得对多步攻击行为的整体刻画, 需将这些因果关系子图进行合并, 合并规则如下.

如果对于因果关系子图G_i、G_k满足以下任一条件, 则在两者间创建一条新的有向边, 从G_i中的最后一个告警节点指向G_k中第 1 个告警节点.

条件 1: 如果两个子图相对应的 IP 对完全相同, 并且后一种攻击是在前一种攻击结束后的特定时间内开始, 后一个子图的告警严重性也比前一个子图的严重性高.

条件 2: 前一个因果关系子图的目标 IP 与后一个因果关系子图的源 IP 一致, 并且后一种攻击是在前一种攻击结束后的特定时间内开始.

此外, 需要对告警节点不超过两个的子图进行过滤, 因为像 DDoS 攻击、APT 攻击应该是一个渐进的过程, 不应该只产生少量的告警.

根据上述规则对所有因果关系子图进行合并, 最终构建出完整的多步攻击场景图, 发现完整的复杂多步攻击行为.

综上所述, 本文总体框架如图 4 所示. 首先利用基于 BERT-SimCSE 的告警语义统一表达方法对告警进

行统一表示, 之后对告警进行因果关系挖掘, 以发现复杂多步攻击.

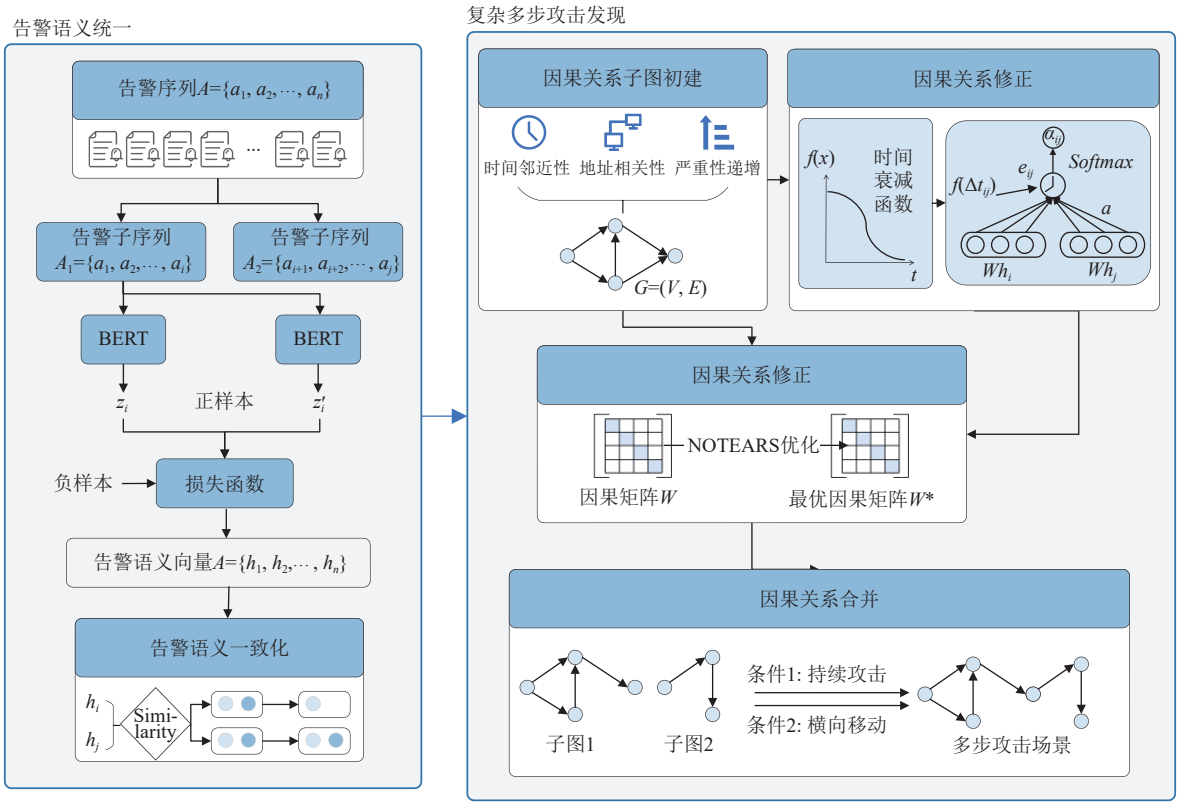

图 4 预训练模型与因果挖掘相结合的多步攻击检测方法

4 实验分析

本文在公开的多步攻击入侵数据集 DARPA2000 LLDoS 1.0[25] (简称 LLDoS 1.0) 和 CICIDS2017[26]上进行实验, 以验证本文方法在多步攻击场景重构方面的有效性与泛化能力. LLDoS 1.0 数据集描述了典型的 5 阶段多步攻击过程, 包括扫描主机、发送漏洞查询报文、利用漏洞攻击主机、在目标主机内植入 DDoS 软件、通过 DDoS 软件发起 DDoS 攻击 5 个阶段. CICIDS-2017 数据集包含正常流量和多类现代网络攻击流量, 按采集日期划分为多个场景, 为网络安全研究提供丰富多样的环境. 本次实验选择了 CICIDS2017 数据集中第 4 天的数据, 其中包含暴力破解、跨站脚本、SQL 注入和网络渗透等攻击类型.

4.1 评价指标

在实际环境中, 很多告警仅因为合法用户误操作而触发, 例如, INFO TELNET login incorrect. 这种告警可能是良性的误操作, 通常不包含多步骤攻击的语义. 因此, 除了多步攻击阶段之外, 还有一个正常类别. 准确率、精确率、召回率和 $F1$ 分数是在检测问题中经常使用的 4 个关键评估标准, 这些指标能够提供关于模型的全面信息. 因此本文采用这 4 个评估指标对模型进行评估, 对每个数据集的不同阶段分别进行了评估指标计算, 最后计算平均值作为模型的最终结果. 计算指标如式 (9)–(12) 所示:

$$Accuracy=\frac{\sum_{i=0}^{n}\frac{TP_i+TN_i}{TP_i+TN_i+FP_i+FN_i}}{n} \tag{9}$$

$$Precision=\frac{\sum_{i=0}^{n}\frac{TP_i}{TP_i+FP_i}}{n} \tag{10}$$

$$Recall=\frac{\sum_{i=0}^{n}\frac{TP_i}{TP_i+FN_i}}{n} \tag{11}$$

$$F1=2\times\frac{Precision\times Recall}{Precision+Recall} \tag{12}$$

其中, TP_i表示第 i 阶段告警被正确识别为对应阶段的数目, FP_i表示其他阶段告警被错误识别为第 i 阶段的

数目, FN_i表示第 i 阶段告警被错误识别为其他阶段的数目, TN_i表示其他阶段的告警被正确识别为对应阶段的数目.

4.2 实验结果与分析

在实验中, 本文使用 Snort IDS 重放了 LLDoS 1.0 和 CICIDS2017 数据集中的网络流量, 产生了大量的原始告警. 对告警进行预处理后的结果如表 1 所示.

表 1 经预处理后的告警

元素	原始告警	预处理后
time	03/07 12:09:51	03/07 12:39:59
name	RPC sadmind UDP PING	BAD-TRAFFIC loopback traffic
type	Attempted administrator privilege gain	Potentially bad traffic
severity	1	2
sip	202.77.162.213	127.11.8.18
sport	54792	7325
dip	172.16.115.20	131.84.1.31
dport	32773	10886

(1) 对比实验

多步攻击往往伴随大量语义相似但表述不一致的冗余告警. 为验证本文所提方法能够从告警消息字段中学习到可靠的攻击语义信息, 并通过语义相似性实现告警语义一致化, 本文选取 Word2Vec、Doc2Vec 等模型作为对比方法进行实验. 实验结果表明: 本文所使用的 BERT-SimCSE 模型能够更有效地学习告警字段所隐含的攻击语义信息, 相较于其他模型在语义表征能力上具有显著优势. 具体而言, 传统的 Word2Vec 或 Doc2Vec 仅能进行静态的词向量映射, 无法结合告警上下文理解深层语义; 而 BERT-SimCSE 模型则能利用上下文感知的动态嵌入机制, 对告警文本进行语义增强与深度表征, 从而精准捕捉告警字段背后的真实攻击意图. 表 2 给出了告警语义统一表达处理前后两个数据集中告警数据的对比结果. 本文方法在保证语义信息完整性的前提下, 在 LLDoS 1.0 数据集中实现了 42.60% 的语义冗余消除, 并在 CICIDS2017 数据集中实现了 38.18% 的语义冗余消除, 表明其能有效降低告警数据的语义冗余程度, 显著降低了数据处理复杂度.

通过比较多种图神经网络模型在攻击阶段分类任务上的性能, 证明了本文方法能够准确地识别和分类攻击阶段, 从而提升多步攻击检测的精确性. 除了本文方法 TA-GAT 外, 本文还选择了其他几种常见的 GNN 模型进行比较, 包括图卷积网络 (GCN)[22]、图同构网络 (GIN)[23]、K-NN[11]和 GraphSage (AGCM)[24]等. 表 3 给出了两个数据集的实验结果, 实验结果表明本文方法表现出色, 其中在 LLDoS 1.0 上的准确率达到 99.15%, 在 CICIDS2017 数据集上准确率达到 96.52%, 表现均为最佳. 相比于仅关注节点局部特征的 K-NN 或常规图卷积网络, TA-GAT 能够动态为不同邻居节点分配不同的注意力权重, 识别出不相关的噪声告警, 因此本文方法完全能够对复杂多步攻击进行检测, 准确地感知攻击者的意图.

表 2 告警语义统一表达

数据集	Model	告警数量	语义统一后的告警数量	约简率
LLDoS 1.0	BERT-SimCSE	1561	896	0.4260
	Word2Vec	1561	1025	0.3433
	Doc2Vec	1561	1134	0.2735
CICIDS2017	BERT-SimCSE	2216	1370	0.3818
	Word2Vec	2216	1554	0.2987
	Doc2Vec	2216	1718	0.2247

表 3 复杂多步攻击发现结果对比实验

数据集	Model	*Accuracy*	*Precision*	*Recall*	*F*1
LLDoS 1.0	TA-GAT	0.9915	0.9919	0.9915	0.9916
	Alert-GCN	0.9787	0.9816	0.9787	0.9795
	GIN	0.8936	0.9290	0.8936	0.8972
	AGCM	0.9234	0.9196	0.9234	0.9185
	K-NN	0.8809	0.9215	0.8809	0.8867
CICIDS2017	TA-GAT	0.9652	0.9664	0.9645	0.9654
	Alert-GCN	0.9412	0.9435	0.9408	0.9421
	GIN	0.8754	0.8812	0.8740	0.8776
	AGCM	0.9105	0.9088	0.9123	0.9105
	K-NN	0.8521	0.8644	0.8515	0.8579

基于本文所提方法, 最终构建的 LLDoS 1.0 和 CICIDS2017 的攻击场景图分别如图 5 和图 6 所示. 两幅图均较为清晰地反映了攻击活动的阶段演化过程, 其中, LLDoS 1.0 展现了较为典型和完整的多阶段攻击链, 而 CICIDS2017 则体现了由 Web 攻击向渗透入侵逐步演化的复合攻击场景.

对于 LLDoS 1.0 数据集, 构建结果较为完整地还原了其攻击场景的 5 个阶段.

① 主机扫描: 从远程站点利用 IPSweep 进行 IP 扫描, 寻找网络环境中的活跃主机.

② 服务识别: 对上一步扫描到的活跃主机进行探测, 查询主机是否运行 sadmind 进程.

③ 漏洞利用: 尝试利用 sadmind 缓存区溢出漏洞实现提权, 获取主机的 root 权限.

④ 命令与控制: 攻击者获取主机的 root 权限后就

可以进行远程登录, 安装 mstream 守护进程控制该主机成为跳板机, 为后续攻击活动奠定基础.

⑤ 攻击执行: 攻击者控制跳板机, 隐藏自己的真实 IP, 使用大量虚假 IP 实施 DDoS 攻击.

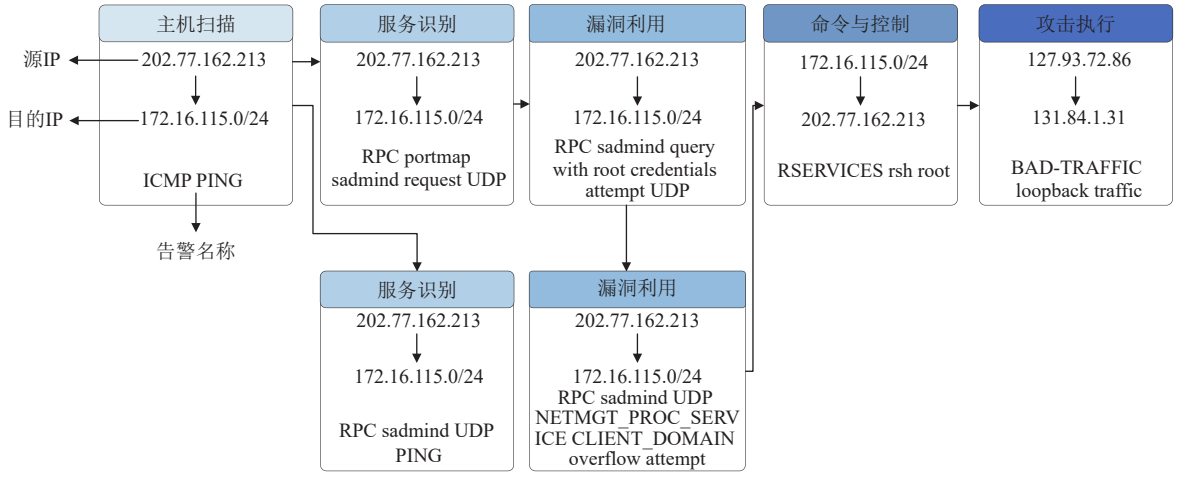

图 5 LLDoS 1.0 复杂多步攻击场景图

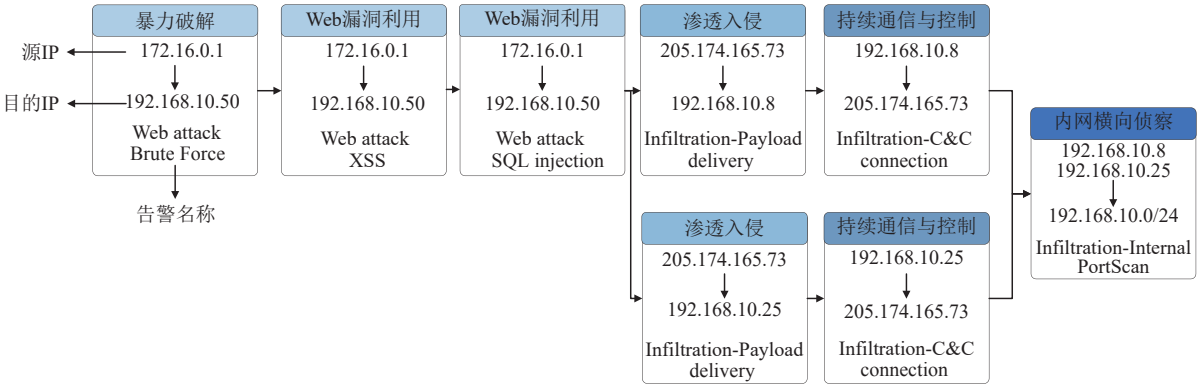

图 6 CICIDS2017 复杂多步攻击场景图

对于 CICIDS2017 数据集, 构建结果反映了攻击者由应用层攻击逐步向渗透入侵扩展的过程, 具体如下.

① 暴力破解: 攻击者针对内网服务器发起持续请求, 并实施 Brute Force 攻击, 尝试获取初始访问权限.

② Web 漏洞利用: 攻击者进一步发起 XSS 与 SQL 注入攻击, 通过恶意脚本注入和 SQL 语句构造利用 Web 应用漏洞.

③ 渗透入侵: 在 Web 层攻击基础上, 攻击行为进一步发展为渗透入侵, 表现为恶意载荷传递和异常连接建立.

④ 持续通信与控制: 攻击者与受害主机之间形成异常通信关系, 使攻击者获得了对内网节点的控制权.

⑤ 内网横向侦察: 在获取内网立足点后, 攻击者以该沦陷主机为跳板, 对内网其他资产发起端口扫描, 企图进一步扩大渗透范围.

综合来看, 本文方法不仅能够较好地还原 LLDoS 1.0 这类具有明确阶段边界的典型攻击链, 也能够对 CICIDS2017 这类包含多种攻击行为的复杂场景进行有效重建, 说明所提方法在不同类型网络攻击场景下均具有较好的适用性.

(2) 消融实验

为验证本文所提方法中 BERT 模型在告警语义信息提取的有效性及其对告警语义一致化处理和告警阶段分类任务的贡献, 设计并进行了实验对其进行评估. 从表 4 中可以看出, 本文所采用的 BERT 模型能够从告警字段中提取到可靠的攻击语义信息, 有效识别语义高度相似的告警事件, 从而减少语义冗余, 并显著提升告警阶段分类的准确性.

表 4 BERT 对实验结果的影响

数据集	方法	*Accuracy*	*Precision*	*Recall*	*F*1
LLDoS 1.0	引入BERT	0.9915	0.9919	0.9915	0.9916
	不引入BERT	0.8809	0.9100	0.8809	0.8903
CICIDS2017	引入BERT	0.9652	0.9664	0.9645	0.9654
	不引入BERT	0.8543	0.8652	0.8415	0.8532

为验证本文提出的基于时间感知注意力机制的图注意力网络 (TA-GAT) 中时间间隔因子的有效性及其对模型性能的具体影响, 设计并进行了实验对其进行评估, 实验结果如表 5 所示. 实验结果表明, TA-GAT 的性能显著优于不使用 GAT 和传统 GAT 模型, 引入时间感知注意力机制能够有效捕捉时间维度上的因果关联性, 增强模型对攻击序列的识别能力.

表 5 时间间隔因子对实验结果的影响

数据集	方法	*Accuracy*	*Precision*	*Recall*	*F*1
LLDoS 1.0	不使用GAT	0.9447	0.8968	0.9447	0.9200
	传统GAT	0.9872	0.9886	0.9872	0.9875
	TA-GAT	0.9915	0.9919	0.9915	0.9916
CICIDS2017	不使用GAT	0.8935	0.8850	0.8950	0.8900
	传统GAT	0.9412	0.9425	0.9398	0.9411
	TA-GAT	0.9652	0.9664	0.9645	0.9654

综上所述, 本节实验从多个方面验证了本文所提方法在复杂多步攻击的发现上的有效性、完整性和准确性.

(3) 敏感性分析

为进一步验证本文所提方法中关键超参数设置的合理性, 本文对时间衰减控制参数λ和因果强度阈值ε进行了敏感性分析. 其中, λ用于控制节点间时间间隔对注意力权重的衰减程度, ε用于筛选 NOTEARS 输出的因果边, 当边权小于该阈值时将被视为弱关联并予以删除. 由于这两个参数分别作用于攻击阶段分类和因果关系修正过程, 因此都会对最终攻击场景图的构建质量产生直接影响.

① 时间衰减控制参数λ的影响

在其他实验设置保持不变的条件下, 本文分别取$\lambda \in \{0.000\,1, 0.000\,2, 0.000\,3, 0.000\,4\}$, 比较不同取值下模型在两个数据集上的分类性能, 实验结果如表 6 所示.

实验结果显示, 随着时间衰减控制参数λ的增加, 模型在两个数据集上的各项性能指标均呈现先上升后下降的趋势, 并在$\lambda = 0.0003$时达到最优值. 当取值较小 (如 0.000 1) 时, 由于时间衰减作用不足, 模型难以有效区分时间跨度较大的无关告警, 导致性能受限; 而当取值过大 (如 0.000 4) 时, 过强的时间惩罚会削弱模型对长跨度语义关联的建模能力, 导致指标出现回落. 这表明$\lambda = 0.0003$能够有效平衡时间邻近性与语义相关性, 使模型获得最佳的分类效果.

表 6 时间衰减控制参数λ对结果的影响

数据集	λ	*Accuracy*	*Precision*	*Recall*	*F*1
LLDoS 1.0	0.0001	0.9452	0.9418	0.9465	0.9441
	0.0002	0.9789	0.9776	0.9792	0.9784
	0.0003	0.9915	0.9919	0.9915	0.9916
	0.0004	0.9823	0.9815	0.9828	0.9821
CICIDS2017	0.0001	0.9215	0.9187	0.9223	0.9205
	0.0002	0.9538	0.9524	0.9542	0.9533
	0.0003	0.9652	0.9664	0.9645	0.9654
	0.0004	0.9587	0.9573	0.9591	0.9582

② 因果强度阈值ε的影响

由于因果强度阈值ε主要作用于因果边筛选阶段, 其影响主要体现在最终攻击场景图中因果关系的保留与删除上. 为分析不同阈值设置对攻击图构建质量的影响, 本文采用完整度 *Completeness* 和正确度 *Soundness* 作为评价指标, 并在不同ε取值下比较最终结果. 完整度和正确度计算公式如式 (13)、(14) 所示:

$$Completeness = \frac{\#CCA}{\#RA} \tag{13}$$

$$Soundness = \frac{\#CCA}{\#CA} \tag{14}$$

其中, *#CCA*表示被正确识别的因果告警对数量, *#RA*表示数据集中真实存在因果关系的告警对总数, *#CA*表示算法最终输出的所有因果告警对数量.

实验中, 设置$\varepsilon \in \{0.4, 0.5, 0.6, 0.7\}$对 LLDoS 1.0 数据集进行测试, 结果如表 7 所示.

表 7 因果强度阈值ε对结果的影响

ε	完整度	正确度
0.4	1.00	0.77
0.5	1.00	0.83
0.6	1.00	0.92
0.7	0.91	0.95

从表 7 可以看出, 随着因果强度阈值ε的增大, *Soundness* 持续提升. 当ε从 0.4 增加到 0.7 时, *Soundness* 从 0.77 逐步提升至 0.95, 这表明设置更高的阈值能够更有效地过滤掉弱关联或错误的因果边, 从而提高重建攻击场景图的准确性.

与此同时, *Completeness* 的变化呈现出不同的趋

势. 在ε从 0.4 增加到 0.6 的过程中, *Completeness* 始终保持为 1.00, 这说明在这一阈值范围内, 算法能够完整地识别出所有真实存在的因果关系, 没有发生漏报. 然而, 当阈值进一步提高至 0.7 时, *Completeness* 下降至 0.91. 这表明过高的阈值虽然可以使结果更准确 (*Soundness* 达到 0.95), 但却牺牲了完整性, 导致部分真实的因果关系被误判为非因果关系而被移除.

综合考虑两项指标, 当$\varepsilon = 0.6$时, *Completeness* 仍然保持 1.00, 同时 *Soundness* 达到了 0.92. 这一设置在确保所有真实攻击路径都被完整恢复的前提下, 最大化了重建结果的准确性. 因此, 本文最终选择 0.6 作为因果强度阈值, 以达到完整性与准确性之间的最佳平衡.

5 结论与展望

在网络安全领域, 告警语义相似但表达不一致的问题和潜在因果关系难发现问题严重阻碍了复杂多步攻击检测的发展, 而对复杂多步攻击场景进行刻画是提升防御能力的关键. 针对上述挑战, 提出了一个系统性的解决方案. 首先, 通过基于 BERT-SimCSE 的告警语义统一表达方法, 实现了对告警文本的深度语义理解, 通过语义相似度度量实现语义一致化处理, 降低告警语义冗余与分析复杂度. 其次, 利用时间邻近性、地址相关性、严重性递增这 3 个关联规则构建初始的因果关系子图. 最后, 结合 TA-GAT 模型和 NOTEARS 算法从初始因果关系子图中挖掘告警间的因果依赖关系, 对告警因果子图进行修正, 剔除虚假关联边, 得到能反映真实因果关系的复杂多步攻击场景图. 在 LLDOS 1.0 与 CICIDS2017 数据集上显示, 所提方法可以准确地发现并重构复杂多步攻击.

在此基础上, 未来的研究工作将侧重于对该框架计算效率的提升与性能的优化. 一方面, 计划构建面向高并发流式告警的仿真测试环境, 系统性地量化评估本方法在不同并发负载下的吞吐量极值与推理延迟分布; 另一方面, 将探索模型轻量化策略, 旨在保障攻击场景重构精度的前提下, 有效降低整体计算开销, 从而进一步提升该方法在大规模实时告警分析场景中的适用性与处理效能.

参考文献

1 Navarro J, Deruyver A, Parrend P. A systematic survey on multi-step attack detection. Computers & Security, 2018, 76: 214–249. [doi: 10.1016/j.cose.2018.03.001]

2 Ingale S, Paraye M, Ambawade D. A survey on methodologies for multi-step attack prediction. Proceedings of the 4th International Conference on Inventive Systems and Control. Coimbatore: IEEE, 2020. 37–45. [doi: 10.1109/ICISC47916.2020.9171106]

3 Hassan WU, Guo SJ, Li D, *et al*. NoDoze: Combatting threat alert fatigue with automated provenance triage. Proceedings of the 2019 Network and Distributed System Security Symposium. San Diego: Internet Society, 2019. 1–15. [doi: 10.14722/ndss.2019.23349]

4 Yao YY, Wang ZQ, Gan C, *et al*. Multi-source alert data understanding for security semantic discovery based on rough set theory. Neurocomputing, 2016, 208: 39–45. [doi: 10.1016/j.neucom.2015.12.127]

5 Tao XL, Shi L, Zhao F, *et al*. A hybrid alarm association method based on AP clustering and causality. Wireless Communications and Mobile Computing, 2021, 2021(1): 5576504. [doi: 10.1155/2021/5576504]

6 Sartiano D, Attardi G, Deri L, *et al*. Suspicious network event recognition leveraging on machine learning. Proceedings of the 2019 IEEE International Conference on Big Data. Los Angeles: IEEE, 2019. 5915–5920. [doi: 10.1109/BigData47090.2019.9006178]

7 Zhou P, Zhou GY, Wu DK, *et al*. Detecting multi-stage attacks using sequence-to-sequence model. Computers & Security, 2021, 105: 102203. [doi: 10.1016/j.cose.2021.102203]

8 Li T, Liu YT, Liu YB, *et al*. Attack plan recognition using hidden Markov and probabilistic inference. Computers & Security, 2020, 97: 101974. [doi: 10.1016/j.cose.2020.101974]

9 Chen R, Zhang SL, Li DW, *et al*. LogTransfer: Cross-system log anomaly detection for software systems with transfer learning. Proceedings of the 31st IEEE International Symposium on Software Reliability Engineering. Coimbra: IEEE, 2020. 37–47. [doi: 10.1109/ISSRE5003.2020.00013]

10 Ryciak P, Wasielewska K, Janicki A. Anomaly detection in log files using selected natural language processing methods. Applied Sciences, 2022, 12(10): 5089. [doi: 10.3390/app12105089]

11 Xiao FR, Chen SW, Yang J, *et al*. GRAIN: Graph neural network and reinforcement learning aided causality discovery for multi-step attack scenario reconstruction. Computers & Security, 2025, 148: 104180. [doi: 10.1016/j.cose.2024.104180]

12 Seyyar YE, Yavuz AG, Unver HM. An attack detection framework based on BERT and deep learning. IEEE Access, 2022, 10: 68633–68644. [doi: 10.1109/ACCESS.2022.3185748]
13 Satvat K, Gjomemo R, Venkatakrishnan VN. Extractor: Extracting attack behavior from threat reports. Proceedings of the 2021 IEEE European Symposium on Security and Privacy. Vienna: IEEE, 2021. 598–615. [doi: 10.1109/EuroSP51992.2021.00046]
14 Lee Y, Kim J, Kang P. LAnoBERT: System log anomaly detection based on BERT masked language model. Applied Soft Computing, 2023, 146: 110689. [doi: 10.1016/j.asoc.2023.110689]
15 Guo HX, Yuan SH, Wu XT. LogBERT: Log anomaly detection via BERT. arXiv:2103.04475, 2021.
16 Bryant BD, Saiedian H. Improving SIEM alert metadata aggregation with a novel kill-chain based classification model. Computers & Security, 2020, 94: 101817. [doi: 10.1016/j.cose.2020.101817]
17 Rahman MA, Al-Saggaf Y, Zia T. A data mining framework to predict cyber attack for cyber security. Proceedings of the 15th IEEE Conference on Industrial Electronics and Applications. Kristiansand: IEEE, 2020. 207–212. [doi: 10.1109/ICIEA48937.2020.9248225]
18 Li D, Cheng X. A first-out alarm detection method via association rule mining and correlation analysis. Entropy, 2024, 26(1): 30. [doi: 10.3390/e26010030]
19 Barzegar M, Shajari M. Attack scenario reconstruction using intrusion semantics. Expert Systems with Applications, 2018, 108: 119–133. [doi: 10.1016/j.eswa.2018.04.030]
20 Zang XD, Gong J, Zhang XC, *et al*. Attack scenario reconstruction via fusing heterogeneous threat intelligence. Computers & Security, 2023, 133: 103420. [doi: 10.1016/j.cose.2023.103420]
21 Khosravi M, Ladani BT. Alerts correlation and causal analysis for APT based cyber attack detection. IEEE Access, 2020, 8: 162642–162656. [doi: 10.1109/ACCESS.2020.3021499]
22 Cheng QM, Wu CM, Zhou SY. Discovering attack scenarios via intrusion alert correlation using graph convolutional networks. IEEE Communications Letters, 2021, 25(5): 1564–1567. [doi: 10.1109/LCOMM.2020.3048995]
23 Hu WH, Liu BW, Gomes J, *et al*. Strategies for pre-training graph neural networks. arXiv:1905.12265, 2019.
24 Lyu H, Liu J, Lai YX, *et al*. AGCM: A multi-stage attack correlation and scenario reconstruction method based on graph aggregation. Computer Communications, 2024, 224: 302–313. [doi: 10.1016/j.comcom.2024.06.016]
25 Mao BF, Liu J, Lai YX, *et al*. MIF: A multi-step attack scenario reconstruction and attack chains extraction method based on multi-information fusion. Computer Networks, 2021, 198: 1–15. [doi: 10.1016/j.comnet.2021.108340]
26 Sharafaldin I, Lashkari AH, Ghorbani AA. A detailed analysis of the CICIDS2017 data set. Proc. of the 2018 International Conference on Information Systems Security and Privacy. 2018. 172–188. [doi: 10.1007/978-3-030-25109-3_9]
27 Mikolov T, Sutskever I, Chen K, *et al*. Distributed representations of words and phrases and their compositionality. arXiv:1310.4546, 2013.
28 Le QV, Mikolov T. Distributed representations of sentences and documents. arXiv:1405.4053, 2014.

(校对责编: 张重毅)