基于溯源图和注意力机制的 APT 攻击检测模型构建

李元诚，罗昊，王欣煜，原洁璇
（华北电力大学控制与计算机工程学院，北京 102206）

摘 要：针对现有攻击检测方法难以应对持续时间长、攻击手段复杂隐蔽的高级持续威胁的问题，构建了基于注意力机制和溯源图的 APT 攻击检测模型。首先，基于系统的审计日志构建能够描述系统行为的溯源图；其次，设计优化算法，确保在不牺牲关键语义的前提下缩减溯源图规模；再次，利用深度神经网络（DNN）将原始攻击序列转换为语义增强的特征向量序列；最后，设计并实现了 APT 攻击检测模型 DAGCN，该模型将注意力机制应用于溯源图序列，利用该机制对输入序列的不同位置分配不同的权重并进行权值计算，能够提取较长时间内的持续攻击的序列特征信息，从而有效地识别恶意节点，还原攻击过程。该模型在识别精确率等多个指标上均优于现有模型，在公开的 APT 攻击数据集上的实验结果表明，该模型在 APT 攻击检测中的精确率达到 93.18%，优于现有主流检测模型。

关键词：溯源图；自然语言处理；APT 攻击检测；注意力机制
中图分类号：TN92
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2024039

Construction of advanced persistent threat attack detection model based on provenance graph and attention mechanism

LI Yuancheng, LUO Hao, WANG Xinyu, YUAN Jiexuan
School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China

Abstract: In response to the difficulty of existing attack detection methods in dealing with advanced persistent threat (APT) with longer durations, complex and covert attack methods, a model for APT attack detection based on attention mechanisms and provenance graphs was proposed. Firstly, provenance graphs that described system behavior based on system audit logs were constructed. Then, an optimization algorithm was designed to reduce the scale of provenance graphs without sacrificing key semantics. Afterward, a deep neural network (DNN) was utilized to convert the original attack sequence into a semantically enhanced feature vector sequence. Finally, an APT attack detection model named DAGCN was designed. An attention mechanism was applied to the traceback graph sequence. By allocating different weights to different positions in the input sequence and performing weight calculations, sequence feature information of sustained attacks could be extracted over a longer period of time, which effectively identified malicious nodes and reconstructs the attack process. The proposed model outperforms existing models in terms of recognition accuracy and other metrics. Experimental results on public APT attack datasets show that, compared with existing APT attack detection models, the accuracy of the proposed model in APT attack detection reaches 93.18%.

Keywords: provenance graph, natural language processing, APT attack detection, attention mechanism

0 引言

高级持续威胁（APT，advanced persistent threat）具有针对性强、组织严密、持久、隐蔽性强和间接攻击的特点[1-6]，攻击者通过将社会工程技术（如鱼叉式网络钓鱼）与先进的攻击技术相

收稿日期：2023−09−28；修回日期：2023−12−19

结合，从而绕过大多数广泛部署的安全技术，如地址空间随机化（ASLR）、数据执行保护（DEP）和沙盒等，实现对目标的攻击。当前，许多企业越来越依赖于主动防御技术，如入侵防御系统（IPS）、安全信息和事件管理（SIEM）工具、身份和访问管理工具以及 Web 应用防火墙（WAF）等。这些工具能应对大部分普通类型攻击，但由于系统运行中存在大量的疑似攻击事件，也会触发安全检测系统报警，这些数量巨大的报警都需要安全分析人员进行鉴别和处理，大大增加了安全分析的成本，而大量的虚假告警也可能使安全分析人员很难将真正重要的攻击与背景噪声区分开，导致攻击被漏检。因此，为有效地处理 APT 攻击，需要研究新的检测分析方法，既能够完成攻击检测，又可以对攻击的因果关系进行总结，从而使安全分析人员能快速确定是否存在重大入侵，了解攻击者最初是如何躲避安全系统的，并确定攻击行为的影响。

传统的网络攻击检测方法用于 APT 攻击检测存在如下不足：难以处理攻击跨度时间长的数据，从而无法识别恶意特征，还原攻击链；难以进行实时检测，从大量的系统日志文章筛选并检测出攻击行为，以及无法应对 0day 漏洞等。采用溯源图对 APT 攻击进行检测近年来受到广泛关注，King 等[7]提出了溯源图的构建方法，该方法对系统级对象（例如文件、文件名和进程）以及系统调用等事件进行观察，从单个检测点（例如可疑文件）开始，识别可能影响该检测点的文件和进程，之后在溯源图中显示攻击事件链。然而，这些工作是在纯粹的取证环境中进行的，因此无法应对实时执行分析的挑战。Hossain 等[8]首次利用溯源图重构 APT 攻击，提出的基于标签和传播策略的 Sleuth 模型实现了 APT 攻击检测，利用标签传播进行双向分析，搜索恶意节点和攻击路径，最终定义规则来匹配更高的威胁攻击，生成对应的分数，进行基于标签的分析后会生成并还原场景图。Milajerdi 等[9]提出 Poirot 模型，利用 CTI（cyber threat intelligence）相关性构建的查询图，将威胁搜索建模为不精确的图形模式匹配问题，能从溯源图中实现 APT 组织查询图（攻击链）匹配及对齐。董程昱等[10]使用异构图嵌入技术，将日志数据转换为具有语义的异构溯源图，通过深度学习模型，采用一种基于分层注意力的异构图注意力（HGAT）网络，来学习每个节点的 d 维向量，自动学习语义特征。Milajerdi 等[11]提出 Holmes 模型，在审计日志生成的溯源图和攻击链的映射之间引入了高级场景图（HSG），从而解决了语义鸿沟，提升了检测效果。Han 等[12]提出 Unicorn 算法，将溯源图中的节点和关系序列转化为向量序列，然后采用机器学习方法对特征向量进行聚类，从而构建系统正常行为模型。Alsaheel 等[13]从溯源图建立攻击和非攻击行为的关键模式，构建基于序列的模型识别顺序中有助于攻击的节点，恢复攻击关键步骤及还原攻击故事。

然而，在上述基于溯源图的模型中，Sleuth、Holmes 以及 Poirot 模型均依赖规则设计和先验知识，对初始数据的质量要求较高，而且无法主动学习攻击行为，检测未知实体。Atlas、Attack2VEC 模型虽采用了序列学习的方法，但其采用的方法对序列分析能力不强，从而存在无法应对模拟攻击等问题[14]。

针对以上模型在 APT 攻击检测方面的不足，本文提出了一种基于注意力机制的溯源图上 APT 攻击检测模型 DACGN（deep attention CNN-GRU network），该模型不需要引入现有的知识框架（如 ATT&CK 等），首先对系统日志进行处理生成溯源图（因果图）并对其进行优化，然后将其转化为语义增强的特征向量序列，最后构建神经网络 DACGN 对样本序列进行训练，识别恶意节点及关系，实现 APT 攻击的检测。本文的主要贡献如下。

1) 根据系统日志构建溯源图，设计算法对溯源图进行优化，在不牺牲关键语义的同时减少溯源图中冗余的节点和关系，从而减少生成序列的数量和长度，降低其复杂度。

2) 采用 Atlas 框架中提出的特征向量序列生成方法，对包含 APT 攻击的溯源图序列通过词序化和词嵌入方法进行处理，将攻击和非攻击语义模式抽象出来，最后生成实数向量序列。

3) 提出基于注意力机制的深度神经网络模型 DACGN，在公开的 APT 攻击数据集上进行训练，与基线方法和现有检测模型比较，并对结果进行评估。

1 问题描述

1.1 针对信息系统的 APT 攻击

根据多个 APT 攻击研究显示[15-20]，APT 攻击

流程通常包括如下几个阶段。

植入阶段。攻击者会对攻击目标进行分析，包括收集目标的信息、确定目标的防御机制，以及分析目标是否具有攻击价值，也会通过技术手段（如侦听）进行渗透以获取情报，从而最终决定所使用的攻击手段。根据确定的攻击手段和攻击目标的特点，开发对应的木马病毒和恶意代码，并使用 0day 漏洞、鱼叉式钓鱼、移动存储设备等手段将恶意程序植入系统。

确认阶段。攻击者设法将上一阶段构建的恶意代码植入系统中，APT 攻击多以电子邮件作为攻击载体，通过水坑攻击、鱼叉攻击等手段使受害者点击钓鱼邮件，以此获取用户账号和密码等信息，从而进行下一阶段的攻击。

扩展阶段。为了进一步访问和控制系统，攻击者建立了攻击据点，并在系统中安装后门程序和其他控件，创建软件，替换或劫持合法代码，或添加启动代码。为了进一步探索网络，攻击者还可能利用系统配置中的错误或漏洞，或使用伪造的令牌修改笔记，以及通过注册表和其他方式来增加自己的攻击权限。

移动阶段。为了进一步扩大攻击，攻击者观察网络和系统，从而窃取账号和密码，并使用这些合法凭据访问系统，创建更多账户来帮助实现他们的目标。实现攻击后，攻击者会探索攻击点周围环境以及可以操纵的单个单位。通过各种系统和账户了解环境，攻击者可以传播攻击或窃取数据。

维持阶段。攻击者会试图潜伏在系统中，避免被发现，从而维持自己在系统中的长期存在，之后攻击者会进行长时间持续性的网络渗透，逐步获取内部网络权限，以便长期潜伏在内部网络，不断收集各种信息，直至窃取到重要情报。

1.2　溯源图相关概念

1.2.1　溯源图

溯源图是从审计日志中提取的数据结构，通过对进程之间依赖关系的展示，追溯跟踪实体（如进程）和对象（如文件或连接）之间的因果关系，溯源图由表示实体和对象的节点以及表示它们之间动作的边组成，是一种有向图，其边从实体指向对象。

本节以一个溯源图为例进行说明，表 1 所示的系统日志表示进程和文件之间的操作关系，包括时间戳和实体之间的操作，其中，process 为进程，file 为文件。

表 1　　系统日志

时间戳	操作
time 0	process A creates process B
time 1	process B writes file 1
time 2	process B writes file 2
time 3	process A reads file 0
time 4	process A creates process C
time 5	process C reads file 1
time 6	process C writes file X
time 7	process C reads file 2
time 8	process A creates process D

图 1 为根据表 1 构建的溯源图，其中，带箭头的线段为边，代表实体之间的关系；数字表示时间戳；其余为实体，包括主体与客体，主体主要包括进程、线程、服务、用户等，客体主要包括文件、注册表、网络端口等。

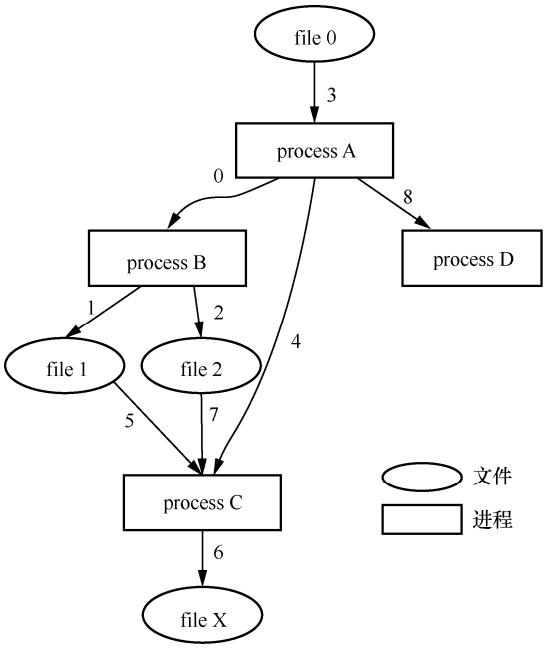


图 1　溯源图示例

溯源图中不同实体之间的边表示的关系根据定义各有不同。实体关系含义如表 2 所示，其中 Connection 为网络连接。

表 2　　实体关系含义

实体关系	含义
process → file	写文件
file → process	读文件
process → Connection	发送数据
Connection → process	接收数据
Parent process → Child process	生成新进程
process → process	进程间通信

1.2.2　实体

溯源图中的实体是指系统中的对象，如进程、文件、网络端口等，实体是从因果图中提取的唯一系统主体或对象，并用节点表示。实体可以是进程、文件和网络连接（如 IP 地址和域名），溯源图中的实体如图 2 所示，其中，C:\\dropper 为文件，129.55.12.167:8000 为一个套接字。

1.2.3　邻接图

给定一个溯源图，如果 2 个节点 u 和 v 通过一条边相连，则称这 2 个节点为邻接节点，一个节点 N_1 的邻接图是由节点 N_1 及其邻接节点和边所组成的子图，一组节点 $\{N_1, N_2, \cdots, N_n\}$ 的邻接图为包括节点 $\{N_1, N_2, \cdots, N_n\}$ 及其邻接节点和边的子图。如图 3 所示，节点 B 的邻接图包含 A、C 以及它们之间的关系，而节点组[B, C]的邻接图中同时包含两者各自的邻接关系和它们之间的关系。

1.2.4　事件

一个事件 ε 是一个四元组，由源（src）、动作（action）、目标（dest）、时间戳（t）组成，其中，源和目标是通过动作连接的 2 个实体，时间戳表示事件发生的时间。给定实体 e，它的事件可以从 e 的邻接图中提取，该图包括与 e 的邻接节点相关联的所有动作。以图 2 为例，给定实体 Firefox.exe、文件 C:\\dropper，以及 Firefox.exe 到 C:\\dropper 的动作 write，加入时间戳 t，就可以得到事件(Firefox.exe, write, C:\\dropper, t)，该事件表示 Firefox 在时间 t 对 C:\\dropper 执行写操作。

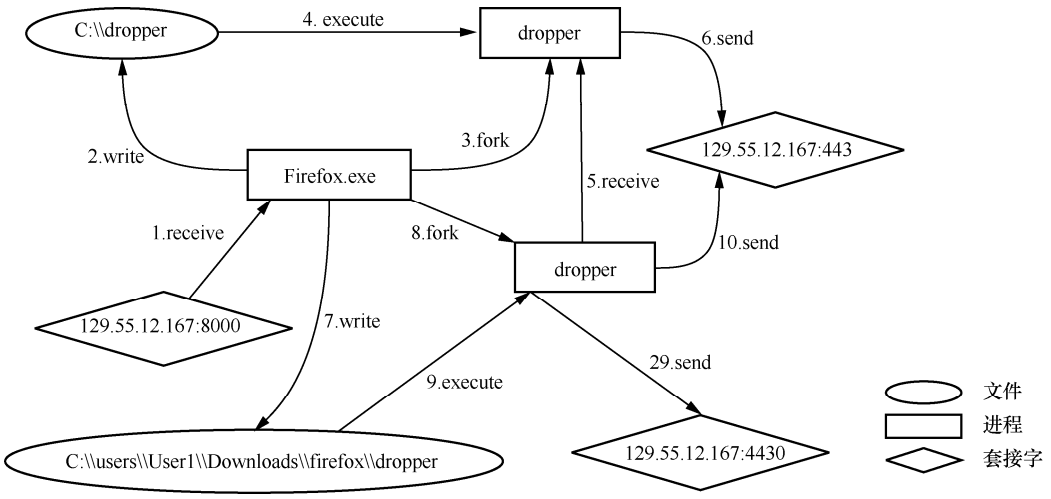
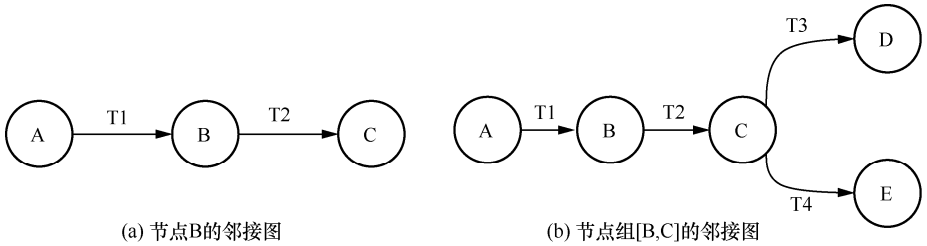


图 2　溯源图中的实体

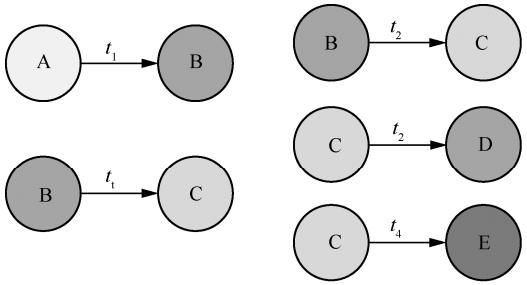


(a) 节点B的邻接图　　(b) 节点组[B,C]的邻接图

图 3　邻接图

1.2.5　序列

给定实体 e，可以从因果图中提取一个序列 S。序列 S 按时间顺序包含实体 e 的邻接图中的所有事件，表示为 $S\{e\}=\{\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_n\}$。同样地，如果给定一组实体，则可以提取包括它们邻接图中的所有事件的序列。在邻接图中提取实体 B 和实体 C 的事件序列，如图 4 所示，实体 B 的事件有 2 个，分别为 $\varepsilon_{AB}=<e_A, a_1, e_B, t_1>$和$\varepsilon_{BC}=< e_B, a_2, e_C, t_2>$，其中，$\varepsilon_{AB}$表示在时间 t_1 实体 e_A 对实体 e_B 执行了 a_1 操作，实体 C 同理。



(a) 实体B的事件　　(b) 实体C的事件

图 4　实体 B 和 C 的事件序列

1.3　基于溯源图 APT 攻击检测

溯源图也称依赖图或因果图，由实体节点和关系组成，其中，实体包括主体和客体，关系表示各种操作，如 write、read、delete 等，通过对其进行处理和分析，能够实现网络安全的各种任务，如入侵检测、攻击场景重构、空间态势感知、主动防御等。由于相较于传统检测方法，溯源图在应对长期攻击、还原入侵过程方面具有优势，利用溯源图进行攻击检测的模型有很多。Sleuth 模型利用标签和传播策略进行双向分析，分配权重，设置规则匹配攻击行为，生成攻击路径。Poirot 模型引入威胁情报中的 IOC 情报生成查询图，将其通过图对齐和图匹配技术在溯源图中查询，通过分析生成的匹配分数识别攻击。Unicorn 算法将溯源图序列转化为特征向量序列，训练模型将特征向量进行聚类，构建系统正常行为模型。Atlas 模型使用机器学习、自然语言处理（NLP）和因果关系分析的新颖组合，建立攻击和非攻击行为的关键模式，其在推断时间，给定威胁警报事件，确定因果图中的攻击症状节点。

1.3.1　溯源图生成

溯源图大多根据系统审计报告生成，审计报告通常由系统内部的审计日志，如 Linux 系统的 Audit、Windows 系统的 ETW（event tracing for Windows）以及 FreeBSD 系统的 Documentation Portal 等，也可以使用第三方日志采集工具获得。由于系统日志包括系统中的所有行为，十分繁杂，因此在初始溯源图生成后需要对其进行优化，在不损失过多语义的同时降低复杂度，从而减少系统存储负担，提高算法效率。优化分为节点压缩和边压缩，其常用的方法包括删除无用的节点和重复的边，合并事件相同的节点或边等。

1.3.2　溯源图优化

溯源图优化技术主要包括以下内容：边缩减、顶点缩减、图缩减、图形压缩和语义保留压缩。

边缩减技术包括因果关系保全缩减（CPR）、以进程为中心的因果关系逼近缩减（PCAR）和基于领域知识的缩减（DOM）。PCAR 保留因果关系并删除与目标文件无关的重复读/写操作。DOM 主要删除临时文件。完全依赖保留缩减（FDPR）和源依赖保留缩减（SDPR）是 2 种保留依赖关系的边缩减技术。顶点缩减技术关注对象的生命周期，以及 NodeMerge，它使用固定库和只读资源集来缩减系统事件数据。图缩减技术包括 PrioTracker、NoDoze 和 Rapsheet。PrioTracker 优先考虑异常依赖关系，而 NoDoze 将 PrioTracker 扩展到异常路径。Rapsheet 提供了 2 个图缩减规则。图形压缩技术包括 FD-SD、CPR 和 NodeMerge。SEAL 是一种无损压缩技术，它从系统日志生成依赖图，并压缩图的结构（顶点和边），以及边的属性（如时间戳）。语义保留压缩技术包括全局语义（GS）和可疑语义（SS）压缩策略。GS 策略删除不影响全局依赖的冗余事件。SS 策略根据取证分析的目的恢复攻击链，本文主要在边和节点方面进行溯源图优化。

1.3.3　基于溯源图的攻击检测的方法

本文方法对溯源图中的节点和边进行优化后，使用引理化和选择性采样来构建序列，从而实现对攻击的提取，并且使用嵌入式方法对序列进行向量化，从而用于学习攻击特征。

综上所述，基于溯源图的攻击检测首先利用由攻击情报生成的溯源图，能够有效地还原网络攻击的路径、时间、实体、攻击事件等关键信息，再通过结合知识框架，利用先验知识以及使用 NLP 领域的相关技术，将溯源图提取到的数据进行规范化，将其转化为易于深度学习模型训练的向量数据，从而使其获得检测网络攻击的能力。

2　基于溯源图的 APT 攻击检测

2.1　系统日志的预处理

系统日志存在大量与攻击无关的操作以及虚假告警，为了优化系统日志生成的溯源图，减少其复杂度，需要对系统日志进行预处理，从而减少冗余的操作，进而减少其生成溯源图中实体和事件的数量，溯源图缩减主要围绕边缩减、顶点缩减、图缩减、图形压缩和语义保留压缩等进行研究。

该方法使用的 3 种技术如下，其示例分别如图 5～图 7 所示，其中，P 表示进程；S 表示会话；A 表示 IP 地址；D 表示域名；T*i* 表示时间戳 T 上的第 *i* 个事件，代表事件发生的顺序。

1) 删除边缘节点

溯源图中有些节点与其他节点之间没有操作，在使用模型学习时不会使用到此类节点，故将其全部删除。如图 5 所示，P2 节点属于边缘节点，应予以删除。

2) 删除重复边

系统中存在大量的重复操作，导致节点之间有许多重复的边，删除这些冗余的边不会对整体语义造成太多损失，能够有效优化溯源图，因此删除溯源图中 2 个实体之间除首次操作之外的所有边，即只保留两者的第一次事件。如图 6 所示，P1 对 A1 执行多次连接操作，只保留最早的(T2, connect)，删除其他边。

3) 合并相同节点和边

如果溯源图中某些节点和边的组合的结构完全相同，则代表其表示发生的事件相同，可以将这些节点合并为一个节点，然后保留这些节点中最早的操作作为新节点的边。如图 7 所示，节点 S1、S2、S3 均表示 P1 到 S4 的同一种操作，故将其合并为一个节点，以(T3, bind)和(T4, send)作为边。

目前的溯源图优化方法包括节点优化和边优化，Lee 等[21]设计了一个具有垃圾收集功能的日志系统 LogGC，该方法能够有效地屏蔽日志中无效的数据关系，在没有任何压缩的情况下，LogGC 可以

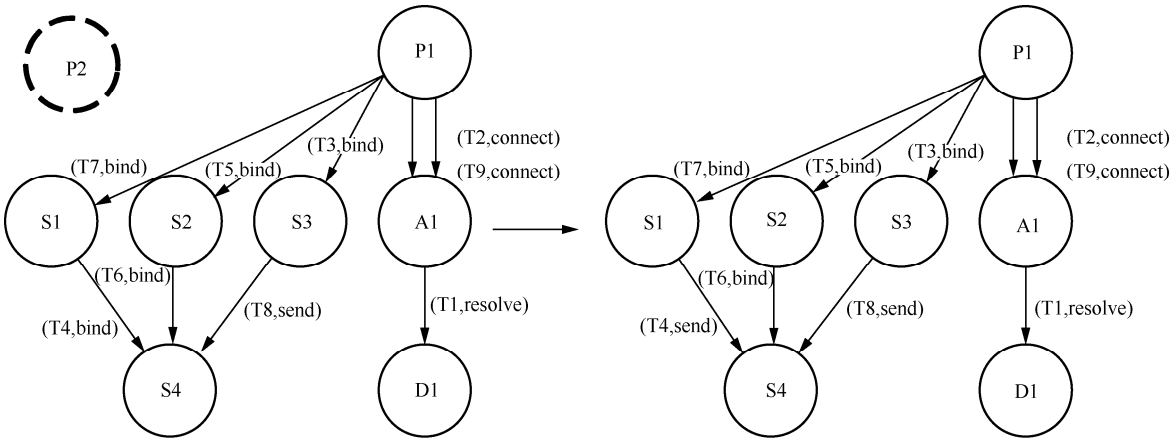


图 5　删除边缘节点示例

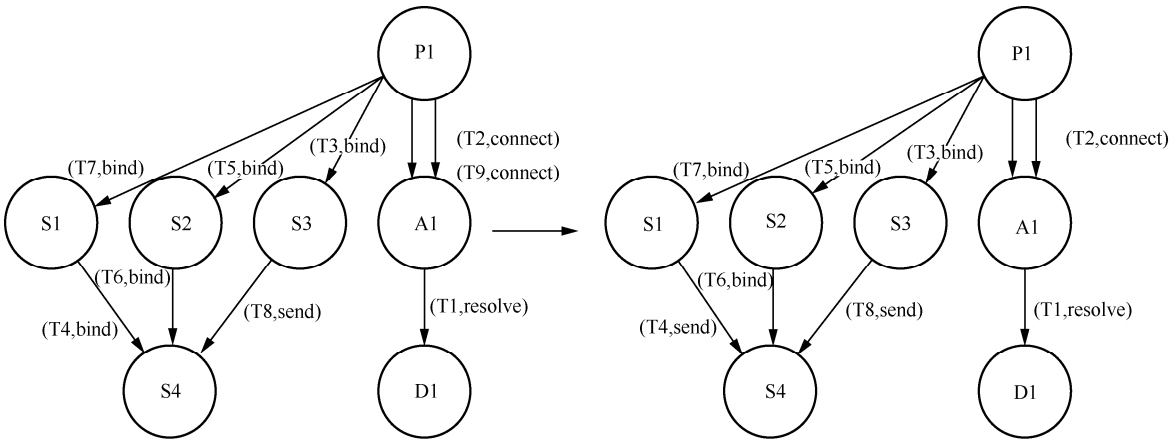


图 6　删除重复边示例

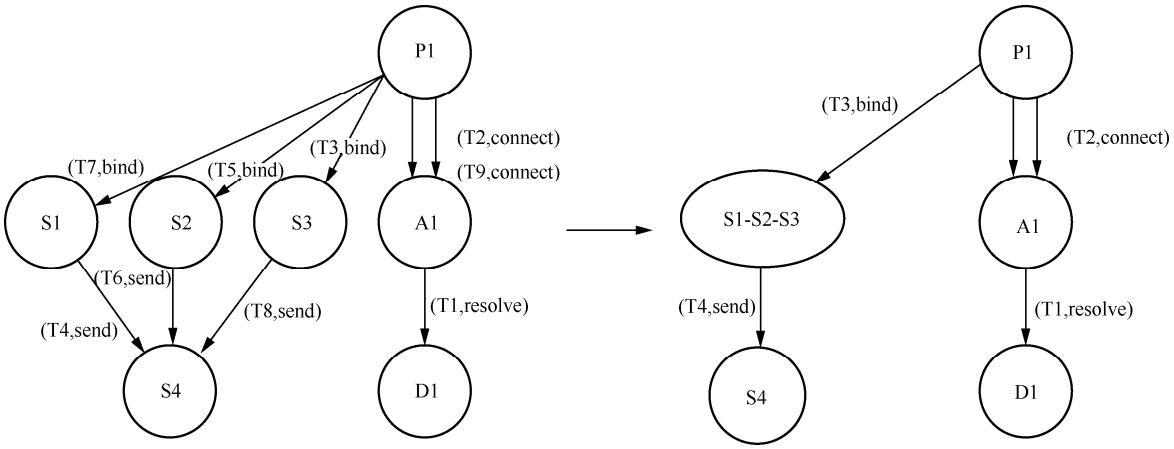


图 7　合并相同节点和边示例

将用户系统的日志大小减少至原有的 $\frac{1}{14}$，但是 LogGC 只能处理无效的节点，无法进一步处理有效但冗余的节点和关系。Hossain 等[22]使用 2 种缩减技术，在保证取证结果准确性的前提下，将完全依赖关系和源依赖关系的事件平均分别减少了 85.72%和 89.14%，但其实现比较复杂，对系统资源消耗较大。

Alsaheel[13]等通过收集模拟真实环境的 APT 攻击系统日志数据，获得包括 10 个子集的 Atlas 数据集，在该数据集上通过上述优化过程得到结果如图 8 所示，其中柱形上方的数字表示和优化前相比实体减少的百分比。

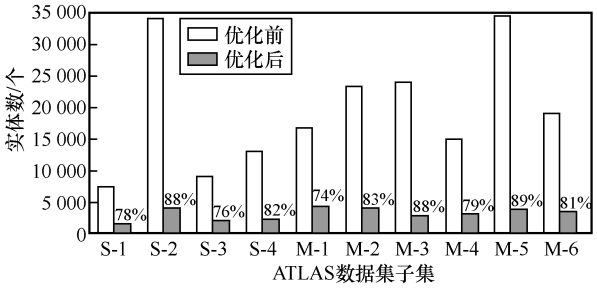


图 8　Atlas 数据集的优化结果

与优化前相比，优化后的实体和关系大量减少，实体数平均减少了 81.81%，有效地降低了系统的存储和运行成本，提高了效率。

2.2　序列构建

序列词源化。由于溯源图中涉及大量不同类型和名称的实体和操作，在直接使用它们进行模型学习时会遇到维度过高、数据稀疏等问题。因此需要对它们进行抽象化处理，并保留其语义信息。序列词源化就是将实体和操作映射到预定义好的抽象词汇上，并根据其类别进行编码。通过自然语言处理技术将序列中的实体和操作转化为抽象词汇，在保留语义的同时，使其可以用于基于序列的模型学习，这些词汇共有 30 个，分为进程、文件、网络、动作四类。用于词形还原的抽象词汇集如表 3 所示。

例如，将可执行程序读取文件的具体操作</system/process/malicious. exe read /user/secret. pdf>转化为<system_process read user_file>。

攻击序列提取。为了提取出能够描述攻击行为特征的序列，并利用深度神经网络（DNN）将其转化为可用于模型学习的词汇，从溯源图中获取全部攻击实体节点，并构成 2 个或多个的子集；然后从每个子集中提取攻击序列。具体方法如下：提取子集中每个节点的邻接图，若某事件的起始节点或目标节点为攻击节点，则称其为攻击事件，按照时间戳顺序生成事件的序列，若该序列只含有攻击序列，且包含该子集中所有攻击事件，则将该序列标

表 3　　**用于词形还原的抽象词汇集**

类别	词汇
进程	system_process, lib_process, programs_process, user_process
文件	system_file, lib_file, programs_file, user_file, combined_files
网络	ip_address,domain,url, connection,session
动作	read, write, delete,execute, invoke, fork,request,refer, bind actions receive, send,connect,ip_connect, session connect,resolve

记为攻击序列。

非攻击序列提取。与攻击序列提取类似，将一个非攻击节点添加到攻击子集中，若子集的序列与任何攻击序列都不匹配，则将其标记为非攻击序列。

2.3 DACGN 模型的设计

DACGN 模型在门控循环单元（GRU）的基础上加入深度可分离卷积机制和注意力机制，从而能够更加有效地利用序列中的语义信息，以及显著地降低模型参数的数量，提高模型的训练效率，DACGN 模型结构如图 9 所示。

1) 特征向量序列生成模块

为了便于模型的训练，需要将生成的溯源图序列转化为特征向量序列，通过一定的编码方式将序列中的单词为可训练的向量，词嵌入技术是一种将自然语言中的词语映射为数值向量的方式，使每个单词或词组在预定义的向量空间中被表示为实数向量。这样可以使相似或相关的单词在向量空间中具有相近或相似的位置。常见的词嵌入技术有 Word2Vec、GloVe、FastText 等。相较于传统的预处理编码方法（如独热编码等），词嵌入技术能够有效提取单词之间的语义关系，被大量使用于 NLP 领域中，本文采用 DNN，将由系统调用日志数据产生并优化后的溯源图序列转化成特征向量序列，记为 $\boldsymbol{S}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3,\cdots,\boldsymbol{x}_n\}$，其中，$\boldsymbol{x}_i$ 为第 i 个特征向量，其维度为 d，序列中元素的个数为 n。

2) DNN 模块

在溯源图中，相邻时刻系统所生成的向量的特征是相似的，因为系统在相邻时刻的状态往往是相似的，这导致了由溯源图提取出来的特征向量中含有大量的冗余信息，对模型的训练造成干扰。为了解决这个问题，本文使用 DNN 对序列中所有特征降维，生成低维空间中更紧凑的特征向量。DNN 是一种前馈神经网络，可以将高维输入映射到低维输出[23]。在 DNN 中使用堆栈自编码器（SAE）结构，使用编码器把高维输入编码成低维的隐变量，从而强迫神经网络学习最有信息量的特征，解码器的作用是把隐藏层的隐变量还原到初始维度，实现无监督学习。通过这种方式，可以去除冗余信息，减少计算量，提高攻击检测效率，并且使输入后续模块中的特征中有效信息的占比提高。使用 DNN 处理特征向量序列 $\boldsymbol{S}=\{\boldsymbol{x}_2,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}$，对其进行降维，通过调整编码器和解码器层数和每层神经元数量来控制输出维度。降维后可以得到更紧凑的特征向量，其包含的信息量与原始特征向量相同，但是冗余信息已被删除。

3) 深度可分离卷积层

由于溯源图中包含大量的系统日志信息，其每个节点和边都是具有特定属性和关系的对象，为了有效地分析溯源图，本文将这些节点和边转换为特征向量。然而，即使经过溯源图生成时的优化模块和深度神经网络模块的处理，生成的特征向量数量依然非常庞大，这会使模型的训练和推断变得非常困难。因此需要对卷积层进行优化，使模型能够快速且有效地处理这些溯源特征向量。与常规的卷积不同，可分离卷积能够更加有效地对其进行处理，其结构如下。

可分离卷积的结构将卷积过程分为深度卷积和逐点卷积，在深度卷积部分，以三通道为例，其卷积核的数量为 3，与通道数相等，输入向量的每一个通道分别和对应的卷积核进行卷积运算，生成同等数量的特征图（Feature Map）。逐点卷积过程和常规卷积类似，将生成的 Feature Map 与若干卷积核进行卷积运算，获得新的 Feature Map，其数量由卷积核数量决定。深度可分离卷积需要的参数运算数量较少，效率较高，因而能够将网络深度进一步增加，进而提高模型的泛化能力，从而提高溯源图分析的准确性和效率。

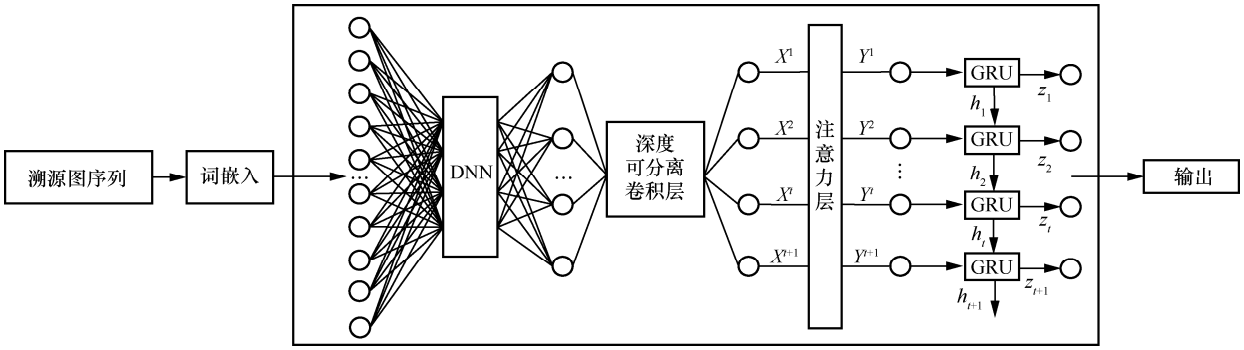


图 9 DACGN 模型结构

4) 注意力层

将经过卷积层处理后生成的特征向量序列 $\boldsymbol{X}$ 输入注意力层，将其转化为特征矩阵 $\boldsymbol{Z}=\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})$，即

$$\boldsymbol{Q}=\boldsymbol{X}\times\boldsymbol{W}^{\boldsymbol{Q}} \tag{1}$$

$$\boldsymbol{K}=\boldsymbol{X}\times\boldsymbol{W}^{\boldsymbol{K}} \tag{2}$$

$$\boldsymbol{V}=\boldsymbol{X}\times\boldsymbol{W}^{\boldsymbol{V}} \tag{3}$$

$$\text{Attention}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=\text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{4}$$

其中，$\boldsymbol{X}$ 为输入向量序列，$\boldsymbol{W}^{\boldsymbol{Q}}$、$\boldsymbol{W}^{\boldsymbol{K}}$、$\boldsymbol{W}^{\boldsymbol{V}}$ 为随机初始化矩阵，d_k 为向量维度，$\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 分别代表 Query、Key 和 Value 这 3 个不同的权值矩阵，它们是通过 3 个不同的权值矩阵由输入向量序列乘以 3 个不同的随机初始化矩阵得到的，通过将输入向量序列视为一个 query 查询向量和多个 key-value 对，根据每次计算得到不同的权重，并将其作用于 value，由于权值矩阵由输入向量序列本身计算获得，故称为自注意力机制。注意力机制根据词的上下文信息进行不同的权重计算，将重要单词赋予高权重，不重要单词赋予低权重，使其获得的权重向量包含了单词之间的相互关系。注意力机制能够有效捕捉长文本序列的特征，保留语义信息。

5) GRU 层

GRU 是一种能够处理序列数据的神经网络模型，是循环神经网络的一种，也是 LSTM 的一种简化版本，它在 RNN 的基础上进行了简化，将原来的几个运算单元合并为更新和重置门 2 个逻辑运算单元，结构如图 10 所示，其中，更新门用来控制前一个隐藏状态对当前隐藏状态的影响程度，重置门用来控制前一个隐藏状态对当前输入的影响程度。

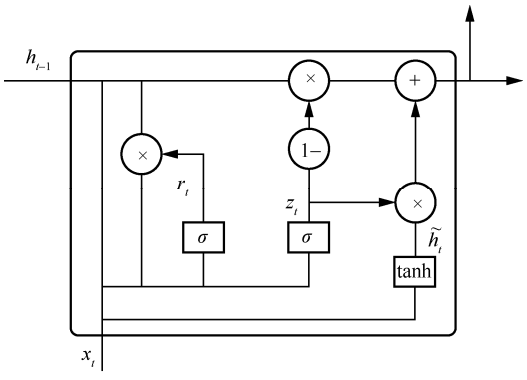


图 10　GRU 结构

在 t 时刻，GRU 中各单元的计算式如下。

更新门控制前一个隐藏状态对当前隐藏状态的影响程度，记为 z_t。

$$z_t=\text{sigmoid}W_z\left[h_{t-1},x_t\right]+b_z \tag{5}$$

重置门控制上一时刻 Memory cell 中的信息是否积累到当前时刻 Memory cell 中，记为 r_t。

$$r_t=\text{sigmoid}(\boldsymbol{W}_r\left[h_{t-1},x_t\right]+b_r) \tag{6}$$

记忆门神经元控制上一时刻 Memory cell 中的信息是否积累到当前时刻 Memory cell 中，记为 $\tilde{h}_t$。

$$\tilde{h}_t=\tanh(\boldsymbol{W}\left[r_t*h_{t-1}\right],x_t) \tag{7}$$

其中，*表示 Hadamard 积。

隐藏层输出根据更新门和候选隐藏状态计算出的最终隐藏状态，记为 h_t。

$$h_t=1-z_t*h_{t-1}+z_t*\tilde{h}_t \tag{8}$$

GRU 相比 LSTM 更加简洁高效，而且在某些情况下能够产生同样出色的结果。

2.4　基于 DACGN 模型的 APT 攻击检测

使用 DACGN 模型进行 APT 攻击检测的流程如下。

1) 对信息网络系统产生的日志进行预处理，生成原始溯源图，并对其进行优化，删除或合并不必要的边和节点，之后依据溯源图构建输入序列。

2) 将溯源图序列输入词嵌入层（即 Embedding 层），将其转化为大小为 batch_size × maxlen × embedding_size 的矩阵 $\boldsymbol{Z}$，其中，batch_size 为每批次的大小，maxlen 为最大的单词数，embedding_size 为 Embedding 层的维度，表示每个单词学习的特征数量，以此将溯源图序列转化为特征向量序列，作为之后的输入。

3) 将特征向量序列输入 DNN，通过多个隐藏层降低序列向量的维度，生成降维特征序列。

4) 将降维后的特征序列输入 Conv1d 层，即使用一个一维卷积核进行卷积，之后将结果输入深度可分离卷积模块，在逐通道（逐深度）卷积层，每个通道和对应的卷积核运算，生成相同数量的 Feature Map，之后进行逐点卷积，使用 N 个 $1\times1\times M$ 的卷积核（M 为通道数），生成 N 个新的 Feature Map，之后进入 Dropout 层处理，避免过拟合。

5) 将数据输入注意力层，令输入向量序列与随

机初始化矩阵 $\boldsymbol{W}^{Q}$、$\boldsymbol{W}^{K}$、$\boldsymbol{W}^{V}$ 分别相乘，获得 $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$，计算获得输出 Attention($\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$)。

6) 将结果输入 GRU 层进行处理，GRU 能够有效处理序列化数据，提取语义，其输出为 batch_size×N×gru_output_size，其中 gru_output_size 为输出维度，最后将结果输出到 Dense 层并进行攻击检测分类。

2.5 实验设置

为了验证本文提出的 DACGN 模型对 APT 攻击检测的有效性，本文在基于真实 APT 攻击获得的数据集上进行实验。首先，描述了实验设置，然后对模型的有效性进行分析，最后将该模型和其他 APT 攻击检测模型进行对比。

2.5.1 数据集

本文实验采用公开数据集 Atlas，由于大部分 APT 攻击检测方法的数据集都不包含系统日志，Atlas 方法基于真实世界 APT 活动的详细报告实现了 10 种攻击，并在受控测试平台环境中生成了审计日志，同时在攻击进行期间模拟了各种正常的用户活动，这些攻击包括不同的恶意软件策略，如钓鱼链接、电子邮件附件、中间进程和横向移动，以及泄露敏感数据。S-1～S-4 攻击是在单主机上执行的，M-1～M-6 攻击是在多主机上执行的。对于每个多主机攻击，在 2 个主机上进行模拟，其中第二个主机用作横向移动的目标。所有攻击都是在 Windows 7 32 位虚拟机上开发和执行的，耗时约 1 h。攻击完成后，收集 24 h 窗口内的系统日志，最后，在 24 h 内平均生成了 20 088 个唯一实体，每次攻击有 249×10^3 个事件，具体如表 4 所示，其中，PL 表示钓鱼链接，PA 表示电子邮件附件，INJ 表示注入攻击，IG 表示信息搜集，BD 表示后门，LM 表示横向移动，DE 表示数据泄露。

2.5.2 实验环境

本文采用 Ubuntu 服务器对 DACGN 模型进行训练，该服务器的设置如下：Intel core i7-9750H 处理器、NVIDIA GTX1660Ti 6G GDDR6、CUDA11.6 版本、Keras 环境（以 TensorFlow 为后端）。DACGN 模型中，Embedding 层为词嵌入层，深度可分离卷积层、GRU 层和注意力层为特征提取层，Dense 层为模型分类层。在模型的参数设置方面，Embedding 层输出维度为 128，Conv1d 层输出维度为 64，卷积核数为 5，Dropout 层的降低率为 0.2，Dense 层的神经元个数（输出维度）为 1。模型迭代次数 epoch 为 8，每个批次的大小为 80，优化器选择 Adam，学习率为 Adam，默认学习率为 0.001。

2.5.3 评价指标

攻击检测作为分类任务，较常用的指标有准确率（ACC, accuracy）、精确率（P, precision）、召回率（R, recall）和 F1 分数，它们都基于混淆矩阵进行计算，其计算式为

$$\mathrm{ACC}=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}} \tag{9}$$

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{10}$$

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{11}$$

$$\mathrm{F1}=2\frac{P\times R}{P+R} \tag{12}$$

表 4 Atlas 数据集概述

编号	攻击活动	攻击所利用的漏洞	特征							大小/MB	日志类型			总数	
			PL	PA	INJ	IG	BD	LM	DE		System	Web	DNS	实体	事件
S-1	Strategic Web compromise[17]	2015-5122	√	×	√	√	√	×	√	381	97.11%	2.24%	0.65%	7 468	95.0×10^3
S-2	Malvertising dominate[22]	2015-3105	√	×	√	√	√	×	√	990	98.58%	1.09%	0.33%	34 021	397.9×10^3
S-3	Spam campaign[39]	2017-11882	×	√	√	√	√	×	√	521	96.82%	2.43%	0.75%	8 998	128.3×10^3
S-4	Pony campaign[18]	2017-0199	×	√	√	√	√	×	√	448	97.08%	2.24%	0.68%	13 037	125.6×10^3
M-1	Strategic Web compromise[17]	2015-5122	√	×	√	√	√	√	√	851.3	96.89%	1.32%	1.32%	17 599	251.6×10^3
M-2	Targeted GOV phishing[34]	2015-5119	√	×	√	√	√	√	√	819.9	97.39%	1.36%	1.25%	24 496	284.3×10^3
M-3	Malvertising dominate[22]	2015-3105	√	×	√	√	√	√	√	496.7	99.11%	0.52%	0.37%	24 481	334.1×10^3
M-4	Monero miner by Rig[28]	2018-8174	×	√	√	√	√	√	√	653.6	98.14%	1.24%	0.62%	15 409	258.7×10^3
M-5	Pony campaign[18]	2017-0199	√	×	√	√	√	√	√	878	98.14%	1.24%	0.62%	35 709	258.7×10^3
M-6	Spam campaign[39]	2017-11882	×	√	√	√	√	√	√	725	98.31%	0.96%	0.73%	19 666	354.0×10^3
平均	—	—	—	—	—	—	—	—	—	676.5	97.76%	1.46%	0.73%	20 888	249K

表 5　　攻击实体识别结果

编号	攻击实体	TP	TN	FP	FN	精确率	召回率	F1 分数
S-1	malicious host	22	7 445	0	0	100.00%	100.00%	100.00%
S-2	leaked fifile	12	34 008	2	0	85.71%	100.00%	92.31%
S-3	malicious host	25	8 972	0	1	100.00%	96.15%	98.04%
S-4	leaked file	23	13 011	3	0	88.46%	100%	93.88%
M-1	leaked file	28	17 562	3	0	90.32%	100.00%	94.92%
M-2	leaked file	37	24 445	4	0	90.24%	100%	94.87%
M-3	malicious file	35	24 423	1	2	97.22%	93.33%	95.23%
M-4	malicious file	24	15 380	0	2	100%	92.31%	96.00%
M-5	malicious host	34	35 665	2	0	94.44%	100%	97.14%
M-6	malicious host	41	19 573	7	1	85.42%	97.62%	91.11%
平均		28	20 048	2	0.6	93.18%	97.94%	95.35%

其中，TP 表示真正类，TN 表示真负类，FP 表示假正类，FN 表示假负类，P、R 分别表示精确率和召回率。

2.6　模型有效性分析

为了验证模型在识别攻击实体和攻击事件方面的有效性，本文在上述数据集上进行了实验，攻击实体识别结果如表 5 所示。

根据表 5，模型在实体识别方面的精确率为 93.18%，召回率为 97.94%。这意味着在使用该模型进行 APT 检测时，在每 100 个攻击实体中，平均能够发现 93 个真正的攻击实体，但会误报 7 个。根据召回率，该模型能够发现 98 个攻击实体，但会漏检 2 个。而对于攻击实体的识别，由于每个实体可能与审计日志中的多个事件相关联，因此实体级别结果的误报和漏报数量比事件级别结果低得多。在这种情况下，该模型无论对实体还是事件所报告的误报（即 FP）和漏报（即 FN）数量都非常小，与真正的阳性（即 TP+FN）和阴性（即 TN+FP）数量相比较少。

2.7　实验结果

2.7.1　基线方法对比实验

为了验证本文提出模型的有效性，本文与其他模型进行对比实验，对比模型包括同样使用系统日志进行分析检测的 Atlas 框架和基线方法。由于之前使用溯源图对于 APT 检测的方法很少完全基于系统日志作为原始数据，Atlas 提出了一种基线方法，以原方法为基准，用某个方法替换模型的对应模块，观察其对精确率、召回率、F1 分数等指标的影响，替换方法包括图遍历、无优化溯源图、仅过采样模型、独热编码、支持向量机（SVM）。实验结果如表 6 所示。

表 6　　对比实验结果

方法	精确率	召回率	F1 分数
图遍历	17.82%	100.00%	30.26%
无优化溯源图	87.58%	41.55%	56.36%
仅过采样模型	91.14%	77.74%	83.28%
独热编码	92.90%	77.99%	84.66%
SVM	80.42%	87.47%	84.11%
Atlas	91.06%	97.29%	93.76%
DACGN	93.18%	97.94%	95.35%

在图遍历基线方法中，本文使用了基于 W3C 的可溯源数据模型规范构建的有向无环图来实现基线方法，该方法在可溯源图上执行向后和向前跟踪。该方法产生了 100%的召回率（即恢复了所有攻击事件），但这会导致终端溯源中的依赖爆炸问题，其平均精确率仅为 17.82%。

在无优化溯源图方法中，使用没有优化过的溯源图，其精确率、召回率和 F1 分数分别降低了 5.6%、56.39%和 38.99%。这是因为图优化从大量日志中删除不相关的事件，这些事件无法用于构建序列的语义和时间关系，且可能导致模型过度拟合。总体而言，图优化过程有助于 Atlas 提取更短的攻击/非攻击序列，并提高模型的泛化能力。

在仅过采样模型方法中，经过过采样攻击序列的过程平衡了有限数量的攻击序列和大量的非攻击序列，平衡了数据集，但是，如果没有进行欠采样，非攻击序列往往会为分类器带来更多的噪声数

据，实验结果显示，和本文中经过了基于相似性的欠采样方法相比，仅过采样方法的精确率、召回率和 F1 分数分别降低了 2.04%、20.2%和 12.07%。

在独热编码方法中，用独热编码代替了词嵌入方法，由于独热编码无法获得不同单词之间的语义关系，因此其准确率、召回率和 F1 分数分别降低了 0.28%、19.15%和 10.69%。

在 SVM 方法中，用 SVM 代替本文分类方法，使用相同的训练数据对 SVM 分类器进行了评估，并使用网格搜索来调整参数以提高分类器的准确性，线性核设置为 C=1.0 和 gamma="auto"，结果显示，其精确率、召回率和 F1 分数分别降低了 12.76%、10.47%和 12.14%。

综上所述，实验结果显示，使用图遍历、无优化溯源图、仅过采样模型、独热编码、支持向量机替换原模块都会不同程度地降低模型的效果，而本文的模型相较于 Atlas，精确率提升了 2.12%，召回率提升了 0.65%，F1 分数提升了 1.59%。

2.7.2 消融实验

选取词嵌入层、深度可分离卷积层、注意力层几个层进行消融实验，将每个层分别删除并进行比较，实验结果如表 7 所示。

表 7 消融实验

删除层	精确率	召回率	F1 分数
词嵌入层	88.90%	77.99%	84.66%
深度可分离卷积层	92.45%	89.84%	94.61%
GRU 层	93.12%	96.03%	94.55%
注意力层	91.04%	96.59%	93.30%
无	**93.18%**	**97.94%**	**95.35%**

实验结果表明，去除 4 个层中的任何一个都会对模型的检测效果产生不利影响。去除词嵌入层对模型影响最大，因为这使模型难以提取语义关系；去除 GRU 层有一定影响，因为 GRU 层也能够提取上下文关系；去除深度可分离卷积层对模型的检测结果影响最小，因为其作用主要是模型的运行效率。

2.7.3 PR 图对比实验

SeqNet[24]基于 Unicorn 算法[12]，提取溯源图的序列特征，使用聚类算法将序列特征聚为 K 类，构建系统的正常行为模型用于检测，其难以确定最佳阈值，因此使用 PR 图方法进行实验。

在 PR 图中，以召回率为横坐标，准确率为纵坐标，当某一个指标相同时，另一个指标越高，表示模型的整体表现越好，因此，综合来看，PR 曲线和坐标轴围成区域的面积 S 越大，表示效果越好。

从 Atlas 数据集中选择一个子集 S-1，该子集的数据是通过在单主机上进行 APT 攻击获取的。本文通过设置不同的置信度来计算 S-1 的准确率和召回率，从而构成 PR 图，与 SeqNet 和 Atlas 进行对比，实验结果如图 11 所示。

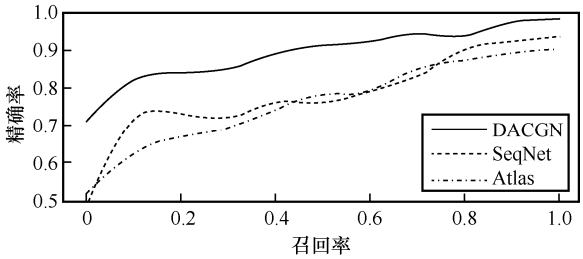


图 11 PR 图对比

由图 11 可知，在该数据集上，本文模型对应的 PR 图面积为 0.887 5，相较于 Atlas 的 0.827 提升了 7%。与 DARPA TC 数据集和 SC 数据集不同，Atlas 数据集包括在评估过程中生成的审计日志，故使用基于上述 2 个数据集训练的 SeqNet 方法的效果有所下降。综上所述，由于本文模型使用 GRU 处理特征向量序列，在 RNN 的基础上改进每个模块的结构，从而保留句子中关键的语义信息，避免梯度消失，所以能够更加有效地利用时序特征，提取序列中的语义信息。同时深度可分离卷积的使用，减少了模型参数，提高了运行速度和模型深度，注意力机制的使用有效地捕捉了长文本的语义信息，提高了模型准确率。

3 结束语

本文提出了一种基于注意力机制的溯源图上 APT 攻击检测模型 DACGN，该模型利用系统日志构建并优化溯源图序列，从而有效还原 APT 攻击的过程，检测 APT 攻击。

1) 针对溯源图文本序列的时序特点，使用自然语言处理常用的 Embedding 方法将溯源图序列转化为特征向量序列，用 DNN 方法对序列进行降维，加入注意力机制增强模型对语义的捕获和理解能力，使用 GRU 避免梯度消失问题，防止检测模型过拟合，利用深度可分离卷积，减少参数计算，提

高模型深度。

2) 与当前主流的 APT 攻击检测模型相比，DACGN 的精确率、召回率、F1 分数均优于其他模型，具有良好的 APT 攻击检测能力。

3) 与其他基于溯源图的模型相比，该模型能够有效利用审计日志，结合 NLP 和机器学习技术还原 APT 攻击过程，且在 APT 攻击检测方面拥有更高的精度。

参考文献：

[1] MANZOOR E, MILAJERDI S M, AKOGLU L. Fast memory-efficient anomaly detection in streaming heterogeneous graphs[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 1035-1044.

[2] HOFER-SCHMITZ K, KLEB U, STOJANOVIĆ B. The influences of feature sets on the detection of advanced persistent threats[J]. Electronics, 2021, 10(6): 704.

[3] BENABDERRAHMANE S, BERRADA G, CHENEY J, et al. A rule mining-based advanced persistent threats detection system[J]. arXiv Preprint, arXiv: 2105.10053, 2021.

[4] ANJUM M M, IQBAL S, HAMELIN B. ANUBIS: a provenance graph-based framework for advanced persistent threat detection[C]//Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing. New York: ACM Press, 2022: 1684-1693.

[5] CHENG X, ZHANG J L, TU Y F, et al. Cyber situation perception for Internet of Things systems based on zero-day attack activities recognition within advanced persistent threat[J]. Concurrency and Computation: Practice and Experience, 2022, 34(16): e6001.

[6] 谢丽霞, 李雪鸥, 杨宏宇, 等. 基于样本特征强化的 APT 攻击多阶段检测方法[J]. 通信学报, 2022, 43(12):66-76.
XIE L X, LI X O, YANG H Y, et al. A multi-stage detection method for APT attacks based on sample feature enhancement[J]. Journal on Communications, 2022,43(12): 66-76.

[7] KING S T, CHEN P M. Backtracking intrusions[C]//Proceedings of the nineteenth ACM symposium on Operating systems principles. New York: ACM Press, 2003: 223-236.

[8] HOSSAIN M N, MILAJERDI S M, WANG J N, et al. Sleuth: real-time attack scenario reconstruction from COTS audit data[C]//Proceedings of the 26th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2017: 487-504.

[9] MILAJERDI S M, ESHETE B, GJOMEMO R, et al. Poirot: aligning attack behavior with kernel audit records for cyber threat hunting[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2019: 1795-1812.

[10] 董程昱, 吕明琪, 陈铁明, 等. 基于异构溯源图学习的 APT 攻击检测方法[J]. 计算机科学, 2023, 50(4): 359-368.
DONG C Y, LYU M Q, CHEN T M, et al. Heterogeneous provenance graph learning model based APT detection[J]. Computer Science, 2023, 50(4): 359-368.

[11] MILAJERDI S M, GJOMEMO R, ESHETE B, et al. Holmes: real-time APT detection through correlation of suspicious information flows[C]//Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2019: 1137-1152.

[12] HAN X Y, PASQUIER T, BATES A, et al. Unicorn: runtime provenance-based detector for advanced persistent threats[J]. arXiv Preprint, arXiv: 2001.01525, 2020.

[13] ALSAHEEL A, NAN Y, MA S, et al. Atlas: a sequence-based learning approach for attack investigation[C]//Proceedings of the USENIX Security Symposium. Berkeley: USENIX Association, 2021: 3005-3022.

[14] 冷涛, 蔡利君, 于爱民, 等. 基于系统溯源图的威胁发现与取证分析综述[J]. 通信学报, 2022, 43(7): 172-188.
LENG T, CAI L J, YU A M, et al. Review of threat discovery and forensic analysis based on system provenance graph[J]. Journal on Communications, 2022, 43(7): 172-188.

[15] DONG J, LIU D R, DOU X H, et al. Key issues and technical applications in the study of power markets as the system adapts to the new power system in China[J]. Sustainability, 2021, 13(23): 13409.

[16] KHALID A, ZAINAL A, MAAROF M A, et al. Advanced persistent threat detection: a survey[C]//Proceedings of the 2021 3rd International Cyber Resilience Conference (CRC). Piscataway: IEEE Press, 2021: 1-6.

[17] LIU J X, SHEN Y, SIMSEK M, et al. A new realistic benchmark for advanced persistent threats in network traffic[J]. IEEE Networking Letters, 2022, 4(3): 162-166.

[18] KARANTZAS G, PATSAKIS C. An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors[J]. Journal of Cybersecurity and Privacy, 2021, 1(3): 387-421.

[19] CORALLO A, LAZOI M, LEZZI M, et al. Cybersecurity awareness in the context of the industrial Internet of things: a systematic literature review[J]. Computers in Industry, 2022, 137: 103614.

[20] 杨秀璋, 彭国军, 李子川, 等. 基于 Bert 和 BiLSTM-CRF 的 APT 攻击实体识别及对齐研究[J]. 通信学报, 2022, 43(6): 58-70.
YANG X Z, PENG G J, LI Z C, et al. Research on entity recognition and alignment of APT attack based on Bert and BiLSTM-CRF[J]. Journal on Communications, 2022, 43(6): 58-70.

[21] LEE K H, ZHANG X Y, XU D Y. LogGC: garbage collecting audit log[C]//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. New York: ACM Press, 2013: 1005-1016.

[22] HOSSAIN M N, WANG J, WEISSE O, et al. Dependence-preserving data compaction for scalable forensic analysis[C]//Proceedings of the 27th USENIX Security Symposium. Berkeley: USENIX Association, 2018: 1723-1740.

[23] ALI A, SEPTYANTO A W, CHAUDHARY I, et al. Applied artificial intelligence as event horizon of cyber security[C]//Proceedings of the 2022 International Conference on Business Analytics for Technology and Security (ICBATS). Piscataway: IEEE Press, 2022: 1-7.

[24] 李佳, 云晓春, 李书豪, 等. 基于混合结构深度神经网络的 HTTP 恶意流量检测方法[J]. 通信学报, 2019, 40(1): 24-33.
LI J, YUN X C, LI S H, et al. HTTP malicious traffic detection method based on hybrid structure deep neural network[J]. Journal on Communications, 2019, 40(1): 24-33.

[作者简介]

李元诚（1970− ），男，山东烟台人，博士，华北电力大学教授，主要研究方向为网络信息安全。

罗昊（1998− ），男，湖北武汉人，华北电力大学硕士生，主要研究方向为网络信息安全。

王欣煜（1998− ），男，甘肃平凉人，华北电力大学硕士生，主要研究方向为电力人工智能。

原洁璇（1998− ），男，山西晋城人，华北电力大学硕士生，主要研究方向为区块链和电力信息安全。