计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(9): 2298−2314, 2025

基于语义图学习的恶意域名检测技术

付 豪[1,2] 龙 春[1,2] 宫良一[1,2] 魏金侠[1,2] 黄 潘[1] 林延中[3] 孙德刚[1,2]

[1](中国科学院计算机网络信息中心 北京 100083)

[2](中国科学院大学 北京 101408)

[3](广东盈世计算机科技有限公司 广州 510006)

(fuhao@cnic.cn)

Malicious Domain Detection Technology Based on Semantic Graph Learning

Fu Hao[1,2], Long Chun[1,2], Gong Liangyi[1,2], Wei Jinxia[1,2], Huang Pan[1], Lin Yanzhong[3], and Sun Degang[1,2]

[1] (*Computer Network Information Center*, *Chinese Academy of Sciences*, *Beijing* 100083)

[2] (*University of Chinese Academy of Sciences*, *Beijing* 101408)

[3] (*Coremail*, *Guangzhou* 510006)

Abstract Malicious domain name detection is a critical component of network intrusion detection systems, enabling the rapid identification of network attacks through domain name requests. Machine learning methods overcome the limitations of blacklist mechanisms and improve detection accuracy. However, challenges such as the high variability of domain name structures and the complexity of real-world environments lead to low detection efficiency and poor robustness in practical applications. To address these issues, a malicious domain name detection technology based on domain name semantic graph learning is proposed, leveraging semantic graph association analysis for efficient detection. Specifically, 12 months of domain request data from China Science and Technology Network are first collected, encompassing 3.33 billion access records, including more than 6.5 million malicious domain name entries across 284 attack types. Semantic analysis reveals significant differentiation between domain categories, yet considerable feature overlap in certain regions degrades classifier performance. To tackle this, a domain association graph model based on character-level semantic similarity is proposed. By integrating features of neighboring domains, the model enhances semantic representations in overlapping regions, thereby improving detection performance. The method includes filtering noise characters through structural similarity analysis, constructing a dynamic domain semantic graph using an online aggregation algorithm, and training a multi-head attention-based message-passing graph model with node-degree-weighted samples. Finally, a multi-layer neural network classifier is employed for malicious domain detection. Experimental results demonstrate that the proposed method achieves an average precision rate of 0.96 and a recall rate of 0.97 on the dataset of different types of malicious domain names. Furthermore, the model exhibits strong online adaptability, achieving high detection rate and robustness.

Key words malicious domain; semantic graph learning; attention mechanism; graph neural network; self-evolution

摘 要 恶意域名检测是网络入侵检测系统中重要的组成部分,能够通过域名请求快速发现网络攻击.基于机器学习的恶意域名检测能够克服黑名单机制缺陷,提升对恶意域名的识别精度,然而由于域名构造

收稿日期: 2024-05-31; **修回日期**: 2025-02-10

基金项目: 中国科学院网络安全和信息化专项(CAS-WX2022GC-04); 中国科学院青年创新促进会项目(2022170)

This work was supported by the Cyber Security and Informatization Project of Chinese Academy of Sciences (CAS-WX2022GC-04) and the Project of Youth Innovation Promotion Association, Chinese Academy of Sciences (2022170).

通信作者: 孙德刚(anquanip@cnic.cn)

差异性大，实际环境域名复杂多变，应用过程中检测效率低、鲁棒性差. 为此，提出一种基于域名语义图学习的恶意域名检测技术，利用语义图关联分析来实现高效的恶意域名检测. 具体而言，首先收集了中国科技网 12 个月的域名请求数据，共 33.3 亿访问记录，其中包括超过 650 万条恶意域名记录，涉及 284 个攻击类型. 通过对不同类别域名的语义特征分析，发现不同类别域名之间具有明显的语义区分度，但存在较大的特征分布重叠区间，重叠的域名数据降低了分类器性能. 因此，提出一种基于字符语义相似性的域名关联图模型，通过融合邻居域名特征增强重叠区间域名语义特征，进而提升检测性能. 首先，通过分析域名结构的相似性过滤域名中吻合度较高的噪声字符以消除域名固有结构造成的检测干扰；其次通过提取域名字符的语义相似性特征构造域名语义图模型，进而通过在线聚合算法构建动态的域名语义图，以基于节点度权重抽样经验池获取的样本集为基础，训练得到基于样本语义权重的多头注意力消息传播图模型；最后使用多层神经网络分类器实现恶意域名检测. 实验结果表明，提出的恶意域名检测技术在不同类型恶意域名的数据集上取得了平均 0.96 的精确率和 0.97 的召回率，并且该模型能够在线进行自演进，具有较高的识别率和鲁棒性.

关键词 恶意域名；语义图学习；注意力机制；图神经网络；自演进

中图法分类号 TP391

DOI: 10.7544/issn1000-1239.202440375 **CSTR:** 32373.14.issn1000-1239.202440375

大部分网络攻击通过域名系统来实现灵活的资源访问和命令传输功能，恶意域名识别也成为网络入侵检测系统中用于发现网络攻击事件的重要组成部分[1]. 恶意域名作为重要的安全威胁情报指标被配置到网络入侵检测系统中，通过对域名请求内容的匹配与分析来快速识别恶意域名. 目前常见的网络攻击[2]，例如僵尸网络、恶意软件、钓鱼攻击等都需要利用恶意域名来实施攻击. 一旦发现恶意域名请求，网络入侵检测系统将拦截恶意域名请求，对网络攻击流量实施阻断. 为了躲避网络入侵检测系统的拦截，攻击者会利用 DGA 域名生成算法短时间内生成大量恶意域名，且仅使用其中小部分域名与资源服务器进行通信. 例如飞客蠕虫病毒 Conficker.C 每天能够生成 5 万个域名[3]，将其分布在 113 个顶级域名（TLD）中. 攻击者只需要选择一个或者一些生成域名来实施攻击，但对于网络入侵检测系统来说，想要拦截攻击则需要阻止所有的域名，这对于网络入侵检测来说是一个巨大的挑战[4].

为了能够检测复杂多变的恶意域名，研究人员探索利用机器学习技术来对域名特征进行分析与识别[5]. 研究人员根据域名自身特点，从域名内容中提取语义特征、字符特征[6]等，有的工作则进一步提取域名的解析特征与注册特征，通过对域名多维特征的融合来试图检测恶意域名. 现有的检测方法按特征可分为基于字符的方法[7]、基于注册信息的方法[8]和基于网络数据包的方法[9]. 基于字符的方法利用合法域名与恶意域名在字符分布上存在的差异进行检测；基于注册信息的方法使用域名的注册时间、别名记录等注册信息进行检测；基于网络数据包的方法分析网络环境中域名系统（domain name system, DNS）流量的检测. 然而基于注册信息的方法需要查询域名的 WHOIS 信息和域名解析信息，基于网络数据包的方法需要收集网络环境中的流量包，两者均需要较大的时延导致其难以用于实时监测系统. 其次域名注册者可按个人意愿隐藏域名相关信息使其不可被查询，难以收集到全部域名的完整数据信息[9]. 近些年来，随着人工智能的发展，大量学者也尝试利用深度学习和图学习技术来检测恶意域名. 然而在实践过程中，智能化恶意域名检测精度往往无法达到令人满意的程度，与实验室测试结果存在较大的差距.

由于网络安全领域数据的稀缺性，现有恶意域名检测方法大多针对某种类型的网络攻击，实验数据集来自小型网络环境或模拟环境[5]，缺乏对不同网络攻击中存在的恶意域名语义特征的分析和检测[6]. 为此，我们在中国科技网环境下开展针对不同网络攻击的恶意域名检测研究. 中国科技网是我国骨干网之一，是学术性、非盈利的科研网络. 中国科技网每天面临着大量网络攻击，在过去的 1 年中，共收集到 33.3 亿条 DNS 域名访问记录，其中被威胁情报标记的有 650 万条恶意域名记录，涉及 284 类网络攻击.

基于上述数据集，本文提出一种基于域名字符语义学习的恶意域名检测技术. 相比于注册特征和解析特征，基于字符语义特征的恶意域名检测无需

查询额外信息,恶意域名实时检测效率更高,同时深度学习等技术能够帮助挖掘更多的域名语义信息.基于域名字符语义特征的恶意域名检测技术因其较快地识别响应,能够在发现恶意域名的第一时间进行防御,及时阻断进一步网络攻击,适用于大规模高速网络环境下实时恶意域名检测场景.通过对大规模数据集中恶意域名的字符语义特征分析,我们也发现不同类别恶意域名之间具有明显的语义区分度,同种类别的恶意域名具备字符构造上的相似性.因此,本文探索基于语义图学习模型来实现高效的恶意域名检测.

本文首先提出一种基于字符语义相似性的域名关联图模型,该图采用基于域名差异的图消息传播机制.具体而言,我们发现同类家族恶意域名往往使用相同的生成算法批量产生恶意域名,使得该类域名在语义上具有一定的相似性,而不同类家族恶意域名拥有不同的语义特征.鉴于此,提取了各节点域名的12种语义特征,利用域名相似关系来构建关联图模型.同时在图消息传播时使用多头注意力机制学习域名语义图节点不同维度的语义特征.

进一步,利用基于Transformer的编解码器对域名字符语义进行关键特征提取.由于恶意域名具有较长的字符构造,传统的字符编码器往往仅捕获域名字符的局部关系,难以对于长域名字符关系进行有效的表示.本文提出的编码器不仅能够捕获域名字符语义的局部关系,也能够捕获域名字符语义的全局关系,实现对各类域名字符语义的级联表示,从而将注意力集中在域名的关键特征上.

最后,利用关键特征实现基于注意力图神经网络的恶意域名检测模型.具体的,关键特征被输入到基于注意力的图神经网络以获得节点的嵌入表示向量,进而将嵌入表示向量输入到由卷积神经网络(CNN)、长短期记忆神经网络(LSTM)以及稠密层网络(DENSE)组成的分类器.利用激活函数softmax计算每一个域名类型的概率值,将概率较高的输出值作为最终的分类结果.

为了验证本文模型的高效性,利用大规模数据集开展实验评估,从AlexTop域名数据集中采集到了10万个白名单域名,从中筛选出9万个合法域名,从科技网流量数据集中选取了8万个左右的黑名单域名来进行模型训练.实验结果显示本文提出的恶意域名检测模型在不同类型恶意域名的数据集上可以取得平均96%的准确率和97%的召回率,相比于其他相同领域最好的恶意域名检测模型,能够提升2%的准确率和2.5%的召回率.此外,实验过程中引入了模型自演进机制,短时间内检测模型能够利用迁移学习思路来对参数进行局部训练,避免模型老化问题,增强了检测系统的鲁棒性.

本文的主要贡献有3个:

1)提出基于语义图学习的恶意域名检测技术,利用域名字符语义相似性原理来构建域名语义图模型,实现对不同网络攻击类型的恶意域名高效检测.

2)利用中国科技网环境中的大规模恶意域名数据集进行实验,结果显示本文提出的恶意域名检测模型在准确率和召回率方面有较高的精度,可以应用于大规模高速网络环境下的网络攻击入侵检测系统中.

3)提出一种自演进的恶意域名检测模型,利用迁移学习思想来提升恶意域名检测模型的自我更新、自我学习能力,增强了网络入侵检测系统的鲁棒性.

1 相关工作

近年来,如何更有效地对恶意域名进行检测得到了诸多学者的关注[10].恶意域名检测算法根据其原理可以分为3类:基于域名语义信息的机器学习检测算法、基于域名通信特征的检测方法和基于深度学习的检测方法.

1.1 基于域名语义信息的恶意域名检测

基于域名字符进行语义分析的检测方法是最早的也是目前最成熟的检测方法,其原理是分析域名的字符组成特点,比较合法域名和恶意域名之间的字符级特性来实现恶意域名检测.

在基于语义的检测方法中,Yadav等人[11]利用字母数字字符和映射到同一个IP地址集的所有域中的双字符分布来提取KL距离、编辑距离及Jaccard度量,以实现在ISP级的数据检测.Cucchiarelli等人[12]从域名字符串中提取*n*-gram特征,测试了不同机器学习算法的检测效果,发现仅利用域名的词汇特征便可达到很高的准确性.Zhao等人[13]提取了11个URL统计特征,集成了多个弱分类器提高了恶意域名检测的泛化能力.Nguyen等人[14]依靠域名特征分布的相似性来消除噪声并对相似的域名进行分组,使用了一种基于协同过滤和密度的聚类算法进行分组,实现了对DNS流量日志数据的恶意域名分类和检测.此外,Nguyen等人[15]提出一种新的检测思路,对域名数据选择融合中性集合,用于对合法域名、恶意域名和不确定性域名的分类,减少了对合法域名的

错误检测的情况. 但是上述算法特征维度小、分类器简单、适应性不强.

1.2 基于域名通信特征的恶意域名检测

用户端恶意软件在和 C&C 服务器进行通信时，其攻击流程会呈现规律性的生命周期和查询模式. 因此一些研究设计了将域名解析信息及通信行为进行结合的检测算法.

Bilge 等人[16] 提出恶意域名检测系统 EXPOSURE，通过被动分析 DNS 数据，总结了基于时间、流量包、TTL、域名字符统计 4 组特征，之后使用决策树进行分类. EXPOSURE 系统在 17 个月的运行时间内检测出超过 10 万个恶意域名. Manadhata 等人[17] 利用恶意软件通信的固有结构建立主机域图，并使用图推理的方法实现恶意域名检测. Sun 等人[18] 对 DNS 场景进行建模，构建异构信息网络，实现了对未知恶意域名的检测. Cheng 等人[19] 主动探测域名的 WHOIS 注册信息并提出使用一种基于 AdaBoost 的轻量级恶意域名检测方法，在域名注册阶段实现防御. Antonakakis 等人[20] 考虑到不存在域名查询将产生 NXDomain 响应，提出结合了聚类和分类的检测算法. 上述算法依赖于更多的辅助信息，但无论是在线还是离线检测，很多信息都难以完全收集.

1.3 基于深度学习的恶意域名检测

近年来，深度学习在网络安全领域的应用也越来越广泛. Vinayakumar 等人[21] 通过分析局域网中的 DNS 数据日志，评估了递归神经网络、长短期记忆网络（LSTM）和其他传统的机器学习分类器. 与经典的机器学习分类器相比，基于深度学习的方法表现较好. 深度学习可以从域名本身等提取深层次特征，从自然语言处理的角度来看，开展基于深度学习的恶意域名检测已经成为恶意域名检测最流行的发展方向.

Park 等人[22] 从精心标记的数据集中挑选特征，并使用自编码器方法以无监督的方式实现恶意域名检测. Ma 等人[23] 针对域名语义信息提取的问题，提出了一种基于 Doc2vec 和混合网络的恶意域名检测模型 DLR. DLR 使用优化的编码构造词向量，之后将双向 LSTM 网络和双向 RNN 网络进行串联融合，提高了检测精度. Jiang 等人[24] 提出了 GNN-GRU-Attention 检测模型，使用 CNN 提取域名空间特征，然后利用 GRU 提取域名时间特征，最后利用注意力机制提高域名的检测速度. Yang 等人[25] 利用隐蔽生成算法 SDGA 域名字符级特征，提出了一种用于检测恶意域名的异构深度神经网络框架 HDNN. 该算法采用改进的并行 CNN 架构和基于注意力的双向 LSTM 来检测隐藏度较高的恶意域名.

综上所述，现有检测方法已经对某些类型的恶意域名产生良好的检测效果，但是近年来随着僵尸网络、DNS 隧道等攻击类型的增加，现实的网络流量场景中出现了越来越多的新型恶意域名. 不同的恶意域名之间存在较大差异，如欺骗用户点击链接的钓鱼域名与合法域名在构造上具有相似性，而恶意软件域名具有较大的长度和字符信息熵，与合法域名不同. 关于如何对不同域名进行区分目前还缺乏有效的方法，尤其是针对大规模网络环境中的多种攻击类别域名的检测，需要根据其不同的语义特点构建检测模型.

2 科学测量与分析

2.1 数据采集与处理

为了对不同攻击类型和犯罪团队所使用的恶意域名语义特征进行分析，本文从 2022 年 4 月至 2023 年 4 月采集到了 33.3 亿次 DNS 域名请求记录. 利用威胁情报库对监测到的恶意域名进行标记，共识别出约 650 万条恶意域名记录，涉及 284 种攻击类型. 基于商业情报库的恶意域名检测难免存在一定的误报率和漏报率，这主要是因为商业情报库中的恶意域名往往来源于多源安全产品和数据，数据质量存在一定的差异. 因此，本文对恶意域名进一步校准，对访问量大的恶意域名类型进行了多源威胁情报对比校准，最终选择出 180 811 个恶意域名，涉及四大类恶意域名. 具体包括 71 603 个钓鱼类域名、20 206 个恶意软件类域名、82 231 个垃圾邮件类域名以及 6 771 个木马类域名. 同时，选择前 10 万的 AlexTop 域名，对这些域名进行连通性测试，过滤掉当前未使用的合法域名. 本文使用了 2 种连通性测试方法：向测试域名发送 PING 报文，收到响应报文的域名作为潜在合法域名；通过 NsLookup 查询域名 IP 地址，对 IP 地址使用 Nmap 探测，存在开放端口的为潜在合法域名. 最后对潜在合法域名进行校验，使用微步在线情报、奇安信情报、腾讯威胁情报中心对潜在合法域名进行情报匹配，获得共 9 万个合法域名.

2.2 数据分析与发现

域名语义特征提取是恶意域名检测的基础，本文调研了流行的恶意域名检测语义特征，王伟等人[26] 提取了五大类特征，包括 9 个字符子特征，并使用 GBDT 方法进行分类；刘善玲等人[27] 通过分析提取了 4 个基本字符特征，使用随机森林分类；蒋鸿玲等

人[28]通过统计分析，提取了6个字符特征，使用多种机器学习算法分类；张洋等人[29]提取了11个词法特征，使用随机森林进行分类. 通过对数据集中的域名语义进行统计分析，本文选择语义区分度较大的12维特征，如表1所示.

Table 1 Classification Features Statistics

表1 分类特征统计

特征	说明	来源
长度	所有字符长度	文献 [26–29]
数字占比	数字数量占总长度比例	文献 [26–29]
连续数字最大长度	连续的数字的最大长度	文献 [26，29]
数字字母转换次数	数字-字母，字母-数字比例	文献 [29]
元音占比	元音数量占总长度比例	文献 [26，28]
连续元辅音占比	元音-辅音，辅音-元音比例	文献 [26]
辅音占比	辅音数量占总长度比例	文献 [26]
连续辅音占比	连续的辅音字符的比例	文献 [26]
特殊字符占比	除去数字和字母的比例	文献 [29]
唯一字符占比	不重复字符的比例	文献 [28]
分级数	按点分割的字符串数量	文献 [29]
信息熵	字符计算的信息熵	文献 [26–29]

统计发现，合法域名和恶意域名在不同统计特征的分布上具有一定的区分度，同时不同类型域名在各个特征上的区分度不同，如图1所示.

1)域名长度. 为了可读性和方便记忆，合法域名的长度一般不会太长，并且具有明确的定义；而恶意域名则由随机算法辅助生成具有较大的随机性. 此外，由于注册时系统会对同名域名进行冲突检测，攻击者为了避免和正常域名冲突，一般会将恶意域名设置较长的字符串. 如图1(a)所示，大规模数据集测量结果显示，合法域名的字符串长度均值为13，恶意域名字符串长度均值则超过20，其中恶意软件和木马类攻击的恶意域名长度均值超过了50.

2)域名中数字占比. 自动化域名生成算法往往采用字符与数字组合生成原则，因此恶意域名中包含大量随机生成的无意义数字，而合法域名中数字占比往往较小. 图1(b)中通过对各类域名数字占比的分析可以发现合法域名数字占比为仅1%，而恶意域名数字占比高达7%，其中钓鱼类和垃圾邮件类的恶意域名占比在10%左右.

3)域名中连续数字最大长度. 正常域名为表达一定的含义，往往会包含部分数字，而恶意域名中连续数字的最大长度通常较大，并且没有特定含义. 图1(c)中合法域名连续数字长度为0.1，而恶意域名连续数字长度在1.5~2.0.

4)字母和数字的转换次数. 正常域名中数字和字母需要表达意义，但是连续的数字和字母转换不符合语义特征. 恶意域名的字符随机生成，转换次数偏多. 如图1(d)所示，合法域名有很少的反转情况，恶意域名平均具有1~2次的反转.

5)域名中元音占比. 为了满足可读性，合法域名通常包含单词或者单词的组合，而恶意域名是随机生成的，包含的元音字母则相对较少. 如图1(e)所示，实验结果显示合法域名的元音占比超过30%，而恶意域名元音占比为20%~25%.

6)域名中连续元辅音占比. 恶意域名通常会遵循某种特定的命名模式，可能包含一些常见的字符串片段、特殊字符，或者具有特定长度和结构的字符序列. 而合法域名为了更好的可读性，包含较多的元音和辅音组合. 如图1(f)所示，实验结果显示合法域名的元辅音占比超过40%，而恶意域名元音占比为20%~35%.

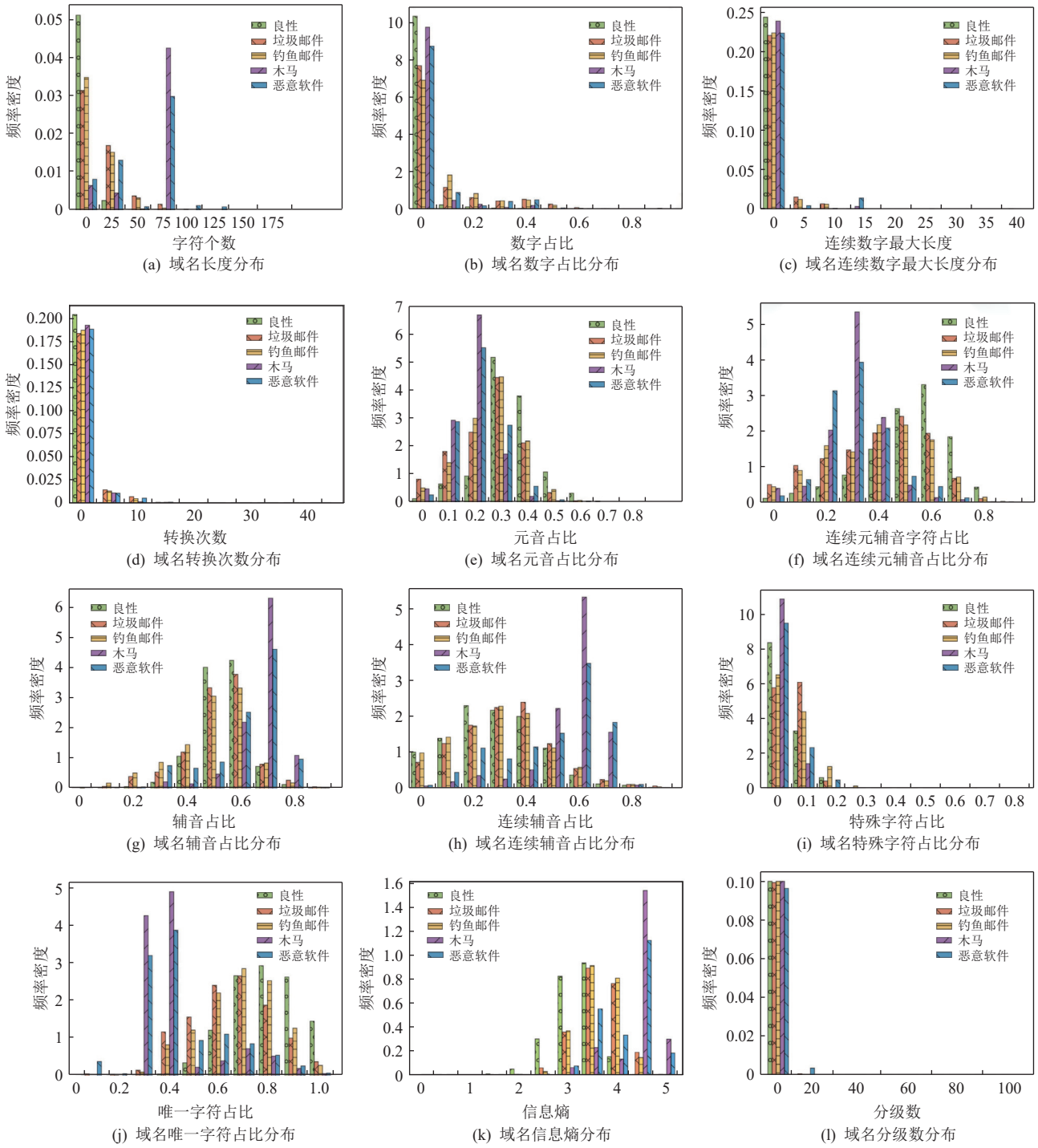
7)域名中辅音占比. 合法域名通常为英文单词组合且容易记忆的字符串，其辅音和元音的占比是相对均衡的. 而恶意域名倾向于采用特定的命名方式，包含了大量的辅音字符. 如图1(g)所示，实验结果显示合法域名的辅音占比不超过60%，而恶意域名元音占比为60%~90%.

8)域名中连续辅音占比. 算法生成的恶意域名中，连续辅音的分布可能会显示不均匀性，而连续辅音在合法域名中出现的频率更低. 如图1(h)所示，实验结果显示合法域名的连续辅音占比不超过50%，而恶意域名连续辅音占比为50%~75%.

9)域名中特殊字符占比. 为模仿合法域名混淆用户，恶意域名倾向于使用包含更多的特殊字符. 而大多数合法域名通常只包含少量的特殊字符. 如图1(i)所示，实验结果显示合法域名的特殊字符占比不超过10%，而恶意域名特殊字符占比为10%~30%.

10)域名中唯一字符占比. 恶意域名倾向于包含更多的唯一不重复字符以用于隐藏其真实企图. 为了满足可读性，合法域名通常包含单词或者单词的组合，其不重复字符相对较多. 如图1(j)所示，实验结果显示合法域名的唯一不重复字符占比超过60%，而恶意域名元音占比为30%~50%.

11)域名信息熵. 合法域名通常具有一定的规律性和结构性，如公司名称、品牌、地点等，而恶意域名倾向于采用更加随机和混淆的命名模式以增加其隐蔽性. 如图1(k)所示，实验结果显示合法域名的信



(a) 域名长度分布 (b) 域名数字占比分布 (c) 域名连续数字最大长度分布

(d) 域名转换次数分布 (e) 域名元音占比分布 (f) 域名连续元辅音占比分布

(g) 域名辅音占比分布 (h) 域名连续辅音占比分布 (i) 域名特殊字符占比分布

(j) 域名唯一字符占比分布 (k) 域名信息熵分布 (l) 域名分级数分布

Fig. 1 Distribution of domain name semantic

图 1 域名语义分布

息熵不超过 3.5, 而恶意域名信息熵为 3.5~5.

12)域名分级数. 为了实施随机子域名等攻击, 恶意域名往往会有更多的级数, 而合法域名的子域名则相对较少. 如图 1(l)所示, 大部分合法域名往往只有 1 个分级, 而恶意域名则拥有 2~4 个分级.

虽然合法域名和恶意域名具有一定区分度, 但不同类型的域名语义区分度有所偏重. 合法域名具有较小的字符串长度、较短的最大连续数字长度以及最小的数字字母反转次数; 恶意软件域名具有较大的域名字符串长度、较大的分级数以及较大的连续数字最大长度. 钓鱼攻击类域名具有较大的数字占比以及较多的数字字母反转次数; 木马攻击类域名则具有最小的元音占比. 从统计结果来看, 各类域名特征都有较大的标准差, 使得待检测恶意域名在某些语义

特征下区分度不高. 针对某类恶意域名检测较好的算法往往对其他类型恶意域名检测效果差. 因此对不同类型的恶意域名需要关注不同类型的语义信息.

2.3 研究动机和方法

统计分析发现, 域名数据总体在不同语义特征上区分度明显, 同类域名存在相似性语义分布, 域名数据之间也存在着重叠的置信度区间. 复杂域名的数据分布情况对检测模型提出了更高的要求, 训练样本依赖较大参数量模型以拟合不断变换的特征. 模型需长时间训练以拟合复杂样本输入. 同时复杂的模型结构带来了现实检测时解释性差、难以迁移的问题.

本文分析域名的语义特征, 通过域名语义相似度聚合构建域名语义图. 节点融合同类域名特征以实现特征增强效果. 增强的域名特征输入分类器以实现高效的恶意域名检测. 具体而言, 本方案优点如下:

1)通过语义相似性划分, 同类域名构成邻居节点, 增强了图神经网络域名语义特征, 同时增加模型的可解释性及检测效果.

2)待检测域名样本与语义图节点实时构建连接, 缓解了新型恶意域名样本少造成的模型拟合难问题.

3)数据分布变化时, 可以通过分类器微调实现快速演进的效果.

即使同样存在基于域名语义相似度进行恶意域名检测的研究, 大多也通过简单域名字符间距离度量方法以及深度学习模型区分域名语义结构. Yadav等人[11]计算合法域名字符与恶意域名字符相似度进行分类, 发现 Jaccard 指数在多数情况下表现最佳, 尤其在处理较大域名集时. Ma 等人[23]利用 Doc2Vec 算法对域名进行向量化, 通过子域名出现频率关系度量不同域名相似性, 之后利用 Bi-LSTM 和 Bi-RNN 进行分类. 相较而言, 本文利用域名语义相似度构建域名语义图, 语义图结构直接反映域名间相似度, 避免了复杂的深度神经网络融合. 同时语义图节点编码域名的整体语义特征, 增强了恶意域名检测的鲁棒性. 快速的域名语义关系构建存在噪声, 但本文利用样本经验池技术对构建的动态语义图加权采样, 关注频繁出现的域名相似度关系, 缓解了噪声干扰.

3 系统设计与实现

3.1 系统总体设计与描述

本文考虑同种家族的恶意域名在字符构成上可能具有相似性, 而利用这种相似性进行特征聚合可以提升该类恶意域名的检测效果. 因此, 首先提出一种基于语义图学习的恶意域名检测模型 SGNN (semantic graph neural networks).SGNN 首先过滤掉合法域名和恶意域名构造的相同字符以减少噪声; 其次提取域名语义特征, 聚合相似域名节点构建域名语义图; 接着基于自注意力机制对域名进行深度编码, 获得域名语义嵌入, 并使用多头注意力图神经网络对 DNS 流量中的域名进行特征融合; 最后使用深层分类器网络对融合特征分类识别. 实际上, SGNN 主要包括域名字符噪声去除、域名语义图构建、多头注意力图神经网络和分类识别 4 个模块, 系统框架图如图 2 所示.

3.2 系统模块设计与实现

3.2.1 域名字符噪声去除

恶意域名由加密算法生成的一些伪随机字符串组成. 这些域名具有随机性, 用于逃避恶意域名的黑名单检测技术, 其形态和云服务负载均衡合法域名具有很大相似性. 因此基于黑名单规则匹配的检测方法缺乏混淆域名检测能力. 为了降低误报, 采用域名结构相似度的度量进一步优化域名语义图节点, 过滤与恶意域名类似的合法域名结构.

1)Jaro 距离

本文计算域名字符串间的距离, 将相似度较大的结构进行过滤. 在现有的距离度量算法中使用 Jaro 距离相似性度量方法.Jaro 距离适用于短字符串的实时检测, 计算方法如式(1)所示.

$$Jaro(s_1,s_2)=\begin{cases}0, & m=0,\\ \dfrac{1}{3}\left(\dfrac{m}{|s_1|}+\dfrac{m}{|s_2|}+\dfrac{m-t}{m}\right), & \text{其他,}\end{cases} \tag{1}$$

其中$|s_i|$是字符串s_i的长度, m是匹配的字符数量, t是字符转换的次数. 当且仅当s_1和s_2的字符匹配, 且距离小于匹配窗口时, 认为存在匹配.Jaro 算法匹配滑动窗口中相同字符, 强调局部相似度, 适用于计算不同长度域名间存在的局部相似性.

2)域名相似度计算

对于域名d_1和域名d_2, 计算两者间相似度为$Jaro(d_1, d_2)$. 恶意域名家族 emkei 的恶意域名“emkei.cz”和合法域名“emkei.com”共享超过 70% 的字符. 高相似度域名结构导致域名语义混淆, 因此过滤域名中的相似字符后提取语义特征. 具体而言, 设置消除阈值, 消除相似度大于消除阈值域名中的匹配字符序列. 合适的消除阈值通过训练获得.

3.2.2 域名语义图构建

本文利用同类型恶意域名家族之间相关性进行

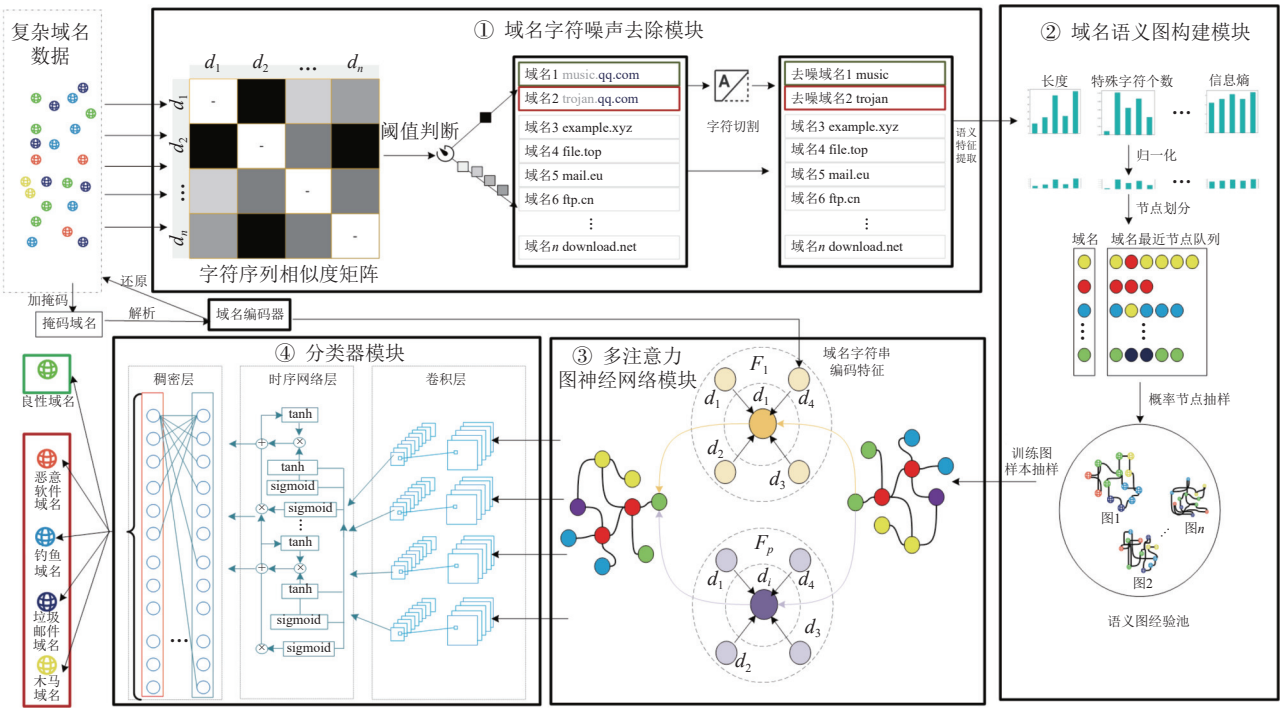


Fig. 2 Architecture for SGNN system

图 2 SGNN 系统架构

节点聚合. 通过将域名和现有域名图节点对比, 实时构建域名语义图, 如算法 1 所示.

算法 1. 基于相似度的语义图构建算法.

输入: 域名 d, 子图大小 M;

输出: 语义图节点列表 $\boldsymbol{G}$, 边矩阵 $\boldsymbol{E}$.

① Gid=($\boldsymbol{G}$←d);
② if |$\boldsymbol{G}$|=1 then
③ return $\boldsymbol{G}$, $\boldsymbol{E}$;
④ end if
⑤ if |$\boldsymbol{G}$|=2 then
⑥ dst=$calDistance$(G_0, G_1);
⑦ $\boldsymbol{E}$←(0, 1, dst);
⑧ $\boldsymbol{E}$←(1, 0, dst);
⑨ return $\boldsymbol{G}$, $\boldsymbol{E}$;
⑩ end if
⑪ for $nodeInfo$ in $\boldsymbol{E}$ do
⑫ for 周边节点 $childnodeInfo$ in $\boldsymbol{E}$ do
⑬ $childnodeids$, $childdist$=$childnodeInfo$;
⑭ dst=$calDistance$(Gid, $childnodeids$);
⑮ if 周边节点数量小于 M then
⑯ $nodeInfo$←按照距离插入(Gid, dst);
⑰ end if
⑱ loc=$calLoc$(dst, $\boldsymbol{E}$);
⑲ if loc<M then
⑳ $nodeInfo$←loc(Gid, dst);
㉑ 删除 $nodeInfo$ 最后一项;
㉒ end if
㉓ end for
㉔ end for
㉕ for $nodeids$ in $\boldsymbol{G}$ do
㉖ $nodeFea$=$nodeids$ 特征;
㉗ dst=$calDistance$(d, $nodeFea$);
㉘ if 周边节点数量小于 M then
㉙ $nodeInfo$←按距离插入(Gid, dst);
㉚ continue;
㉛ end if
㉜ $nodeGinfos$ 等于周边节点索引及距离;
㉝ loc=$calLoc$(dst, $nodeGinfos$);
㉞ if loc<M then
㉟ $nodeGinfo$←loc(Gid, dst);
㊱ 删除 $nodeGinfo$ 最后一项;
㊲ end if
㊳ end for
㊴ return $\boldsymbol{G}$, $\boldsymbol{E}$.

算法 1 旨在实时计算域名节点的相似周围节点列表. 首先, 新的域名被添加到域名分组的节点列表中. 域名节点具有周围最近节点分组, 存储最近域名节点以及节点间的距离信息. 距离 $calDistance$ 计算方

法为特征归一化后的欧氏距离, 如式(2)所示. 其中 d_c 表示当前节点, d_e 表示环境节点, n 表示环境节点数量. 若域名周围节点数小于最大存储大小 M, 则直接插入节点; 若大于则按序插入并删除最远周围节点. 同时算法计算新加入节点和当前节点间距离以构建新节点的最近周围节点分组. 算法运行时, 可实时获得每个节点的相似周围节点.

$$calDistance = \sum_{k=0}^{n}(d_c^k - d_e^k)/n. \tag{2}$$

现实环境产生大量不相干的域名数据导致短时间内采集的域名距离较大. 为了增加模型的鲁棒性, 构建全局的域名动态语义图, 将新节点实时加入域名语义图. 同时从全局域名语义图中以度为权重随机抽取域名节点, 利用抽取节点作为中心节点构建域名语义图. 新节点加入时, 同样将其作为中心节点构造语义图. 最后将构建的语义图放入经验池作为模型训练样本. 训练时, 将从经验池中批量抽样域名语义图训练.

3.2.3 多注意力图神经网络

高相似度域名实体属于同类型域名家族的可能性更大. 以此为基础构建以域名相关性为特征传播权重的图神经网络.

3.2.3.1 域名编码

为了便于记忆, 合法域名往往使用英文简写或者汉语拼音构造, 域名的字符上下文存在相关性. 为了提取域名上下文特征, 本文探索基于 Transformer 的编码模型[30], 相较于其他模型(one-hot[31], word2vec[32]等), Transformer 的特征抽取能力强, 且拥有自编码上下文双向建模的功能. Transformer 编码器由自注意力层和前馈网络层组成, 先将域名特征输入自注意力层编码, 同时关注于句中其他的词. 解码器中包括自注意力层, 编解码自注意力层和前馈网络层, 增加对句子的理解. Transformer 网络可以并行处理数据, 而常见的 RNN 网络无法并行展开, 效率较慢. 此外 Transformer 的网络输入和输出来自同一序列, 更好地描述了全局消息. 本文结合 Transformer 模型结构提出一种基于自注意力机制的域名编码方法.

首先, 对域名中的字符概率生成掩码, 对掩码位置的域名字符进行替换, 生成带有掩码的域名. 利用带有掩码的域名生成原始域名, 模型具有更好的编码效果. 其次, 将原始域名作为解码器的输入, 掩码域名作为编码器输入训练. 最后, 将训练后的编码器用作域名的编码器. 以百度域名为例: ①对于域名 www.baidu.com 中的字母 b 和 u 进行概率掩码处理, 生成 [0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0], 得到掩码域名 www.[mask]aid[mask].com. ②将掩码编码输入到编码结构中, 获得输出 o. ③在解码器字符串前添加前置符号 [bos], 将 [bos] 输入解码器第 1 层, 并使用原始域名作为预测输出. 对于第 1 层输入 [bos] 以及 o, 预测域名首个字符“w”, 对第 2 层输入 [bos], 字符“w”以及 o, 预测域名的第 2 个字符“w”. ④以此类推, 最后 1 层输出结果作为原始域名预测. ⑤通过输出和标签对比计算模型损失. ⑥反向传播更新模型参数. 如图 3 所示.

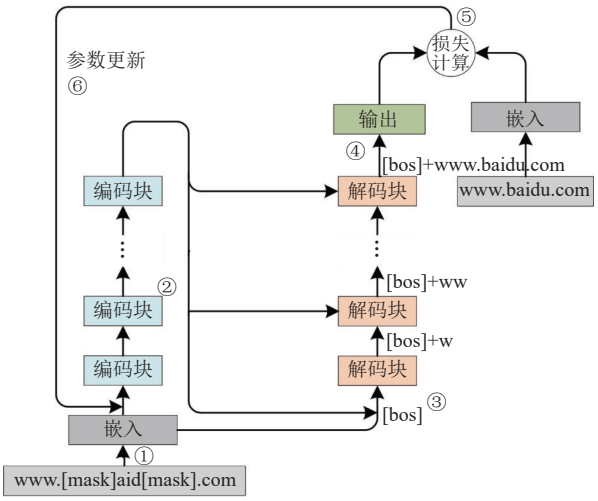


Fig. 3 Encoding process of domain name

图 3 域名编码流程

尤其针对字典类恶意域名, 模型使用部分字符预测出字典其他字符, 提升了模型编码的鲁棒性. 之后使用图神经网络对域名编码向量进行特征增强.

3.2.3.2 消息传播

通过基于语义权重的多头注意力图神经网络对域名特征进行卷积, 网络结构如图 4 所示.

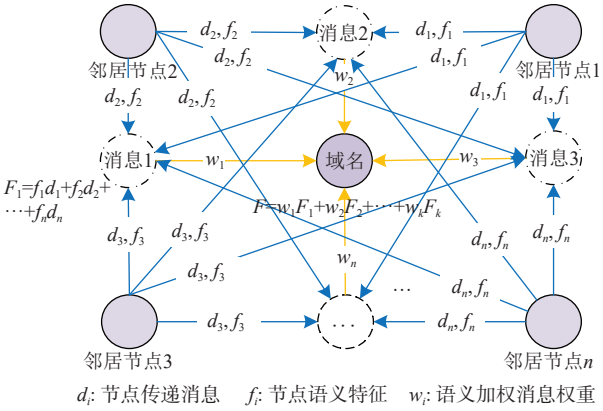


Fig. 4 Aggregating weighted attention

图 4 聚合加权注意力

1)路径权重

对域名d, 计算相似性获得其邻居节点集合$\{d_1,d_2,\cdots,d_n\}$, 以及相似度集合$\{s_1,s_2,\cdots,s_n\}$. 相似度大

小决定了消息更新的比重. 相似度越高的域名，期望学习更多特征. 定义特征传播函数 $H(d)=\sum(d\times s)$，作为传递消息.

2)多头语义消息

为丰富域名节点的邻居节点 d 的传递消息类型，计算 d 的多个语义特征 $\{d_{f_1},d_{f_2},\cdots,d_{f_n}\}$. 之后对节点进行加权求和，得到语义加权消息，即 $F_k=H(d)\times d_{f_k}$. F为邻居节点传递消息，f为域名的语义特征. 语义加权消息作为权重进行消息传递增加了域名语义信息.

3)注意力聚合

不同类恶意域名对语义加权消息感知不同. 比如木马域名较长，但是其数字占比低于钓鱼域名. 因此，不同语义加权消息对于节点卷积结果影响不同. 以此为基础，采用注意力机制，对节点d的语义加权消息F_k，计算其注意力系数$e=a(d,F_k)$，其中e表示节点d对于消息F_k的关注度，a表示注意力卷积. 最终对注意力系数进行归一化处理，如式(3)所示.

$$A=softmax(e)=\exp(e)/\sum\exp(e'), \tag{3}$$

其中A表示归一化后的注意力，e'表示注意力. 最后使用注意力权重计算语义加权消息作为卷积后的节点特征.

3.2.4 分类识别

图神经网络消息传播融合了语义图中同类域名特征，具有更好的嵌入效果. 以此为基础，利用不同的网络结构进行卷积分类，如图 5 所示.

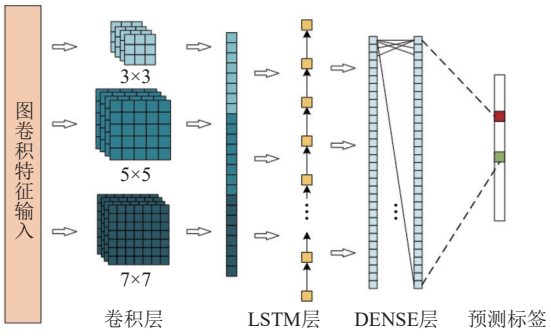


Fig. 5 Classifier structure

图 5 分类器结构

首先以尺寸为 3, 5, 7 的卷积核对域名嵌入进行卷积特征提取，以获取域名字符特征间局部特征. 局部特征扁平化处理后输入 LSTM 网络以提取前后长距离依赖，最后使用稠密层输出样本预测类别标签.

3.2.5 模型迁移

在现实环境中，新恶意域名使得当前分类器性能下降. 为了解决高速流量环境下模型快速更新的难题，使用基于模型的迁移学习方法加快训练速度，如图 6 所示. 由于新恶意域名样本少，基于模型的迁移学习方法利用现有模型中语义图权重达到高精度训练效果.

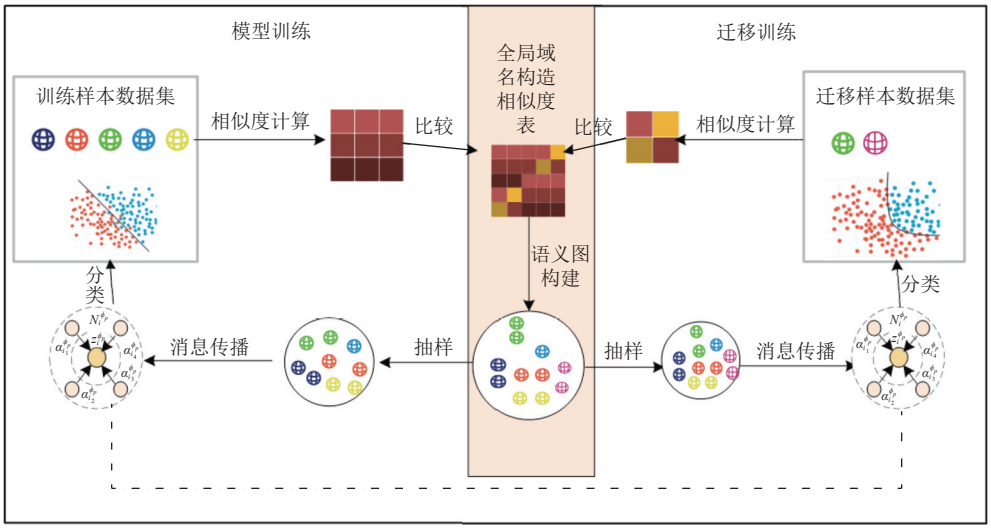


Fig. 6 Transfer learning structure

图 6 迁移学习结构

迁移训练和全训练共享全局的语义构造相似度矩阵以及域名语义图. 首先，利用新样本与全局的语义构造相似度矩阵计算去除噪声. 其次，添加新样本到全局域名语义图中构建关联关系，增强新样本节点特征. 迁移训练时，从全局域名语义图中抽样新节点为中心节点构造域名语义图样本，之后输入图神经网络进行特征融合. 迁移训练模型和全训练模型的图结构共享权重，通过训练分类器以适应新样本.

4 实验部署与评估

4.1 系统总体设计与描述

实验数据集由合法域名数据集和恶意域名数据集2部分组成. 合法域名数据集从亚马逊 Alexa Top 域名列表中提取排名前10万的域名，之后人工对域名进行进一步校验，从中选择9万个合法域名. 恶意域名数据集包含现实环境采集的恶意域名. 在2022年4月到2023年4月的1年时间内，捕获中国科技网某个出口的DNS流量并提取流量数据包中的域名信息，之后使用置信度较高的商业威胁情报库对其进行标记，获得了来自200多个威胁种类的共18万条恶意域名. 采集的数据仅包括DNS报文中的域名请求，不涉及用户设备IP等隐私数据. 数据集信息如表2所示.

Table 2 Number of Samples and Categories of the Dataset

表2 数据集样本数量和类别

类别	具体分类	数量
合法域名	AlexTop	约90 000
恶意域名	垃圾邮件类	82 160
	钓鱼邮件类	70 853
	恶意软件类	20 213
	木马类	6 774

为了验证本文模型SGNN的效果，与主流的恶意域名检测方法进行了对比实验分析. 由于SGNN基于域名语义，对比方法同样基于域名语义特征. 选择同领域最好的3个相关工作进行对比，包括FANCI, *n*-CBDC, TF-IDF. FANCI[33]基于恶意域名和合法域名之间的字母、数字和字符分布差异检测. 从域名中提取出3组21个不同特征，即结构特征、语言特征和统计特征，之后通过实验比较选择随机森林方法分类. FANCI摆脱了研究中需要的额外上下文特征，从单个域名中提取特征并且具有较高的准确性，是机器学习方法在恶意域名检测的主流实践. Xu等人[34]提出了*n*-CBDC检测模型，在字符级卷积神经网络的基础上，开发出2维卷积方法应用字符级处理，设计基于*n*-gram的组合字符嵌入模型，使用深度神经网络分类. Le等人[35]提出使用TF-IDF测量域名中最相关的*n*元语法频率，最终对比各种机器学习和深度学习模型，使用带嵌入层的LSTM分类.

为了评价模型的检测效果，采用准确率(accuracy)、精确率(precision)、召回率(recall)、*F*1分数(*F*1-score)进行评价. 其中，准确率表示模型的效果，精确度表示模型的可信度，召回率反映模型的漏报情况，*F*1分数验证模型的综合表现，这些指标越高效果越好.

4.2 实验结果评估与分析

分别设计多类恶意域名和单类恶意域名检测实验，实验结果如表3所示.

Table 3 Classification Results of Domain Name

表3 域名分类结果

模型	多类恶意域名数据集				单类恶意域名数据集											
					垃圾邮件域名数据集				钓鱼域名数据集				恶意软件域名数据集			
	准确率	精确率	召回率	*F*1分数	准确率	精确率	召回率	*F*1分数	准确率	精确率	召回率	*F*1分数	准确率	精确率	召回率	*F*1分数
FANCI	0.878 9	0.944 9	0.864 9	0.903 2	0.900 5	0.848 8	0.925 2	0.885 3	0.885 5	0.928 2	0.785 8	0.851 1	0.982 2	0.973 3	0.918 4	0.945 1
n-CBDC	0.940 0	0.961 4	0.946 0	0.953 3	0.987 2	**0.985 1**	0.986 7	0.985 9	0.964 5	0.979 6	0.898 0	0.937 0	0.984 3	0.958 2	0.948 7	0.953 4
TF-IDF	0.937 4	**0.986 5**	0.916 2	0.948 3	0.970 8	0.980 1	0.954 6	0.967 2	0.967 5	**0.983 4**	0.904 8	0.942 4	0.979 3	**0.978 5**	0.897 4	0.936 2
SGNN（本文）	**0.961 4**	0.969 6	**0.971 3**	**0.970 3**	**0.988 0**	0.982 5	**0.991 2**	**0.986 8**	**0.984 0**	0.971 2	**0.974 5**	**0.972 8**	**0.993 0**	0.974 6	**0.984 2**	**0.979 4**

注：检测不同类型相同标准检测结果中，黑体数值表示最佳结果.

4.2.1 多类恶意域名检测实验

按照训练集和测试集7∶3的比例训练，从多类恶意域名数据集中随机抽取21万个域名作为训练数据集，6万个域名作为测试数据集.

根据主流模型与SGNN对于合法域名和恶意域名的检测结果，SGNN可以实现0.961 4的准确率、0.969 6的精确率、0.971 3的召回率、0.970 3的*F*1分数，整体上具有最好的检测效果，尤其是模型召回率优于其他模型. 基于机器学习算法的FANCI使用域名的人工提取特征进行分类且分类算法简单，无法应对多类域名环境. FANCI模型准确率低于其余深度学习算法超过5.85个百分点，召回率低于其他算法超过5.13个百分点，在所有对比实验中检测效果最差. 由于合法域名具有较好的*n*元语法特征，*n*-CBDC算法使用的*n*-gram模型提取到丰富的域名字符间关系特征，具有较高的精确率以及较高的召回率. TF-

IDF 算法进一步分析了域名的词频特征，同时使用序列化模型，具有最高的检测精确率，但召回率低于 SGNN 5.51 个百分点. TF-IDF 同样对恶意域名中存在的噪声数据不敏感. 相较而言 SGNN 在取得了最高的召回率的同时兼顾了检测精确率，综合 *F*1 分数最高.

4.2.2　单类恶意域名检测实验

现有的检测模型大多对单类恶意域名检测效果较好，面对不同类型恶意域名的混合数据时检测效果较差. 为了验证检测模型对某类域名的检测效果，本文从恶意数据集中选择数量较多的垃圾邮件域名、钓鱼域名、恶意软件域名 3 类数据作为单类恶意域名检测实验中的恶意域名数据集，从 9 万个合法域名中随机抽取同等比例合法域名作为单类恶意域名检测实验中的合法域名数据集，构建的数据集情况如表 4 所示. 按照训练集和测试集 7∶3 的比例训练，3 类恶意域名数据分别进行二分类实验，实验结果如表 3 所示.

Table 4　Overview of Single Datasets

表 4　单一数据集概况

类别	垃圾邮件域名数据集	钓鱼域名数据集	恶意软件域名数据集
良性	约 8 万	约 7 万	约 2 万
恶意	82 160	70 853	20 213

1）垃圾邮件域名数据集分类实验

由于垃圾邮件形式多样，基于域名粗粒度统计特征的 FANCI 算法检测效果最差. 其余使用深度学习的模型都取得了较好的效果，SGNN 精确率比其余对比实验中的最好算法高 0.08 个百分点，召回率高 0.45 个百分点.

2）钓鱼域名数据集分类实验

钓鱼攻击企图通过伪装可信实体诱导目标点击恶意链接，其域名与合法域名具有一定的相似性. 与垃圾邮件恶意域名检测类似，FANCI 算法对比其余深度学习模型检测效果差，同时对钓鱼类域名召回率偏低. SGNN 相比其余对比实验模型准确率提高 1.65 个百分点，召回率提高 6.97 个百分点.

3）恶意软件域名数据集分类实验

恶意软件域名为 C&C 服务器通信的 DGA 域名，机器学习和深度学习方法取得较好的检测效果. SGNN 召回率高于其他模型超过 3.55 个百分点.

二分类实验表明现有工作可以取得好的检测效果，分别具有 0.987 2, 0.967 5, 0.984 3 的准确率. 但现有工作无法在不同实验中保持较高的召回率. *n*-CBDC 算法对 spam 具有 0.986 7 的召回率，但检测钓鱼域名时，召回率降低至 0.898 0，可见现有检测算法对不同类型恶意域名检测效果差别大. SGNN 针对不同恶意域名检测均保持了超过 0.98 的准确率及 0.97 的召回率.

通过多类恶意域名检测实验和单类恶意域名检测的对比实验可以看到，各算法检测效果均有所下降，说明不同类域名对算法检测造成干扰.SGNN 对域名进行多类语义特征学习，使用注意力机制融合不同语义信息，保持了超过 0.97 的召回率.

4.3　消融实验

为了验证模型中编码模块、语义分析模块以及图神经网络模块的优势，分别对模型结构替换进行对比实验.

实验 1. 为了验证本文编码模型的作用，替换编码模块为 one-hot 编码处理域名.

实验 2. 为了验证语义模块的作用，删除噪声消除方法以及语义相似度聚合节点构建语义图方法，按照邻近时间序列节点构图.

实验 3. 为了验证分类器效果，使用全连接神经网络分类.

实验结果如图 7 所示.

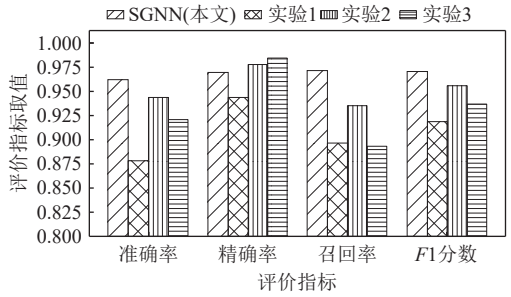


Fig. 7　Results of ablation experiments

图 7　消融实验结果

通过实验结果分析发现编码模块对分类起着重要的作用，相似度计算以及域名语义特征融合增强了编码特征. 替换编码模块后，模型准确率下降 8.35 个百分点，召回率下降 7.51 个百分点. 由于分类器针对域名编码检测，编码方法对检测效果至关重要. one-hot 编码缺乏对域名词素的学习，弱编码分类性能损失最为严重. 删除了语义模块后，难以检测隐藏度高的相似恶意域名，模型的召回率下降了 3.62 个百分点.

使用全连接神经网络分类，虽拥有更高的精确率，超过原模型 1.44 个百分点，但是召回率下降了 7.77 个百分点. 高精确率取决于健壮的域名字符编码模型. 缺乏域名间特征融合的模型对恶意域名拟合效果差.

4.4 模型自演进实验

中国科技网环境需要模型快速更新以应对出现的新型恶意域名,然而新型域名构造与原有域名数据不同导致其连通性存在差异. 因此 SGNN 实际部署时,对模型的实际检测效果进行周期性的人工二次验证,当模型性能明显下降时,迁移学习训练.

将数据集中的域名数据划分为 4 个时间段,使用时间段 1 的域名数据训练模型,对时间段 2~4 的域名数据使用迁移学习微调训练分类器. 实验结果图 8 所示.

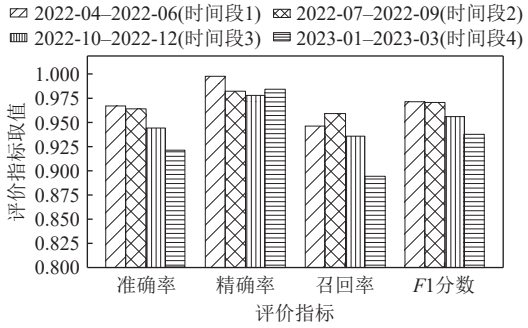


Fig. 8 Results of migration experiments

图 8 迁移实验结果

实验结果表明,迁移的模型可以在 3 个月的多类域名数据集上保持较稳定的检测效果,检测准确率均超过 0.96.

4.5 现实环境部署

现实高速网络环境域名数据和实验数据分布不同[36]. 首先,现实环境域名请求存在大量混杂的合法域名及少量分散的恶意域名,恶意域名比例小且分布不均;其次,现实网络环境会不断产生新型域名,数据分布发生偏移. 为了测试上述训练模型在现实环境的检测效果,采集 2023 年 4 月的实际 DNS 请求数据验证.

为验证时效性,数据采集以 1 周为单次采集时间窗口,连续采集 4 周域名数据. 每个时间窗口采集 10 万域名数据,最终获得共计 4 个时间窗口的 40 万域名数据. 为了评估模型性能,采用公开威胁情报及专家知识对数据集进行类别标注. 检测结果如图 9 所示.

通过检测结果发现,随着检测时间逐步增长,检测精确率保持在了 0.97 左右,但召回率下降了约 15 个百分点. 进一步分析发现,现实环境新型域名和模型训练域名差异大导致查全率降低. 短时间内,SGNN 在真实环境中取得了可用的效果,但应对新型域名分布能力差.

根据 4.4 节讨论,可以自适应地更新模型以实现

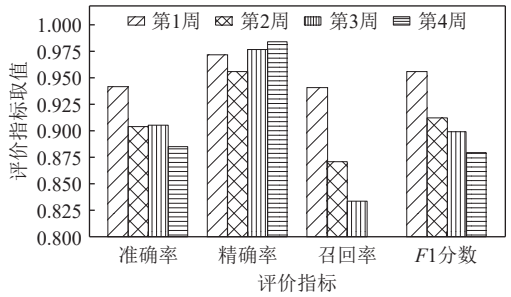


Fig. 9 Results of data drift

图 9 数据漂移结果

模型的性能优化,更新后的模型保持一定的检测能力. 为检验该自演进方案在现实环境中的效果,按 4.5 节相同采集方法再次采集中国科技网 2023 年 5 至 7 月之间的域名数据,形成共 8 个时间窗口的 80 万域名数据集. 为了研究分类器微调对检测效果的影响,使用 2023 年 4 月数据训练的模型在新数据集上测试. 图 10 展示了在各个数据集上原有分类器及迁移训练分类器的实验结果.

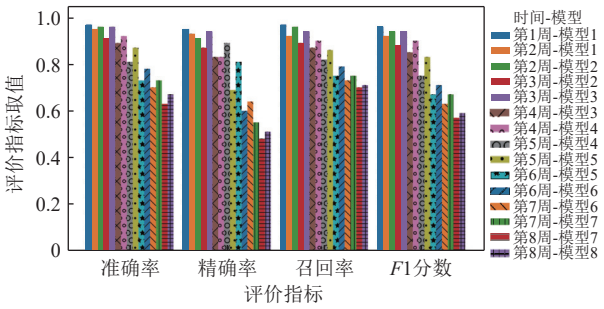


Fig. 10 Results of evolutionary experiment

图 10 演进实验结果

迁移训练可以将 1 个月内的模型检测召回率维持在 90% 之上,保留了对新型域名的检测能力.1 个月后的数据与原有数据产生较大偏差,需要全训练模型. 现实部署时,模型迁移训练满足快速调整要求. 将全训练周期设置在 1 个月左右,满足性能检测需求.

此外测试了现实部署的运行时间,针对图 10 中原始全训练模型和 7 个迁移训练模型的训练和检测过程测试性能,结果如图 11 所示. 实验结果表明,自演进过程中方法的调整对于检测性能影响较小,自演进模型训练和检测的误差在 10 s 之内. 在中国科技网环境中,SGNN 平均 1 min 的时间可完成约 1 万个域名的检测,基本满足网络入侵工作中的实时性要求. 与目前流行的语义检测大模型进行对比,虽然大模型在处理海量域名数据时达到了更高的精确率,但产生了更高的分类时延. 本文对比了 BERT 类通用语义预训练大模型在相同条件下的分类实验,显示 1 万个

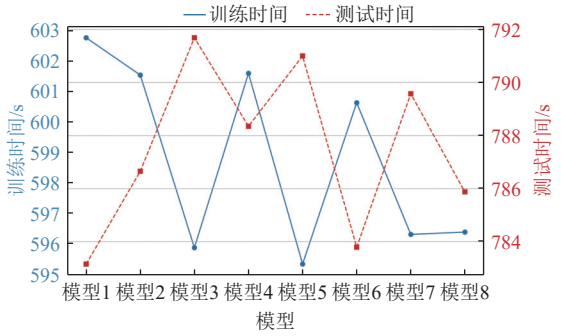


Fig. 11 Performance Test Result of evolutionary

图 11 自演进性能测试结果

域名的检测时间超过了 38 min, 无法满足恶意域名检测系统的实时性要求. 相较而言, 语义图算法可以在保证模型较高检测精确率的条件下实现快速分类.

4.6 附加实验

4.6.1 同类研究对比实验

为对比利用域名语义相似度进行恶意域名检测的同类研究, 本文选择 Yadav 等人[11]利用 Jaccard 指数计算语义相似度并直接分类的方法以及 Ma 等人[23]提出的 DLR 模型深度学习的方法进行对比实验. 对比实验使用同样的恶意域名数据集并以 7∶3 的比例划分训练数据集和测试数据集. 实验结果如图 12 所示.

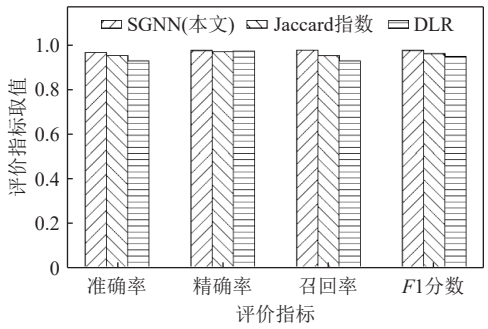


Fig. 12 Comparative test results of similar studies

图 12 同类研究对比测试结果

结果表明 SGNN 相较于其他最好对比实验方法, 准确率提高了 1.43 个百分点, 精确率提高了 0.36 个百分点, 召回率提高了 2.48 个百分点, *F*1 分数提高了 1.48 个百分点. Jaccard 指数方法通过相同的解析 IP 将域名进行分组, 精确率高于 DLR 方法 0.14 个百分点, 但由于可分组的样本相较于 DLR 更少, 召回率降低 2.27 个百分点. SGNN 根据语义构建了更多的边, 并通过抽样及注意力关注了更准确的域名关系, 提升了检测效果.

4.6.2 性能测试

为具体比较各模型的运行性能, 本文测试各模型运行时间. 使用数据集中 27 万个域名, 并按照 7∶3 的比例划分训练数据集和测试数据集. 针对神经网络结构的 *n*-CBDC, TF-IDF, DLR, SGNN 模型, 统一训练 20 次 epoch. 实验结果如图 13 所示.

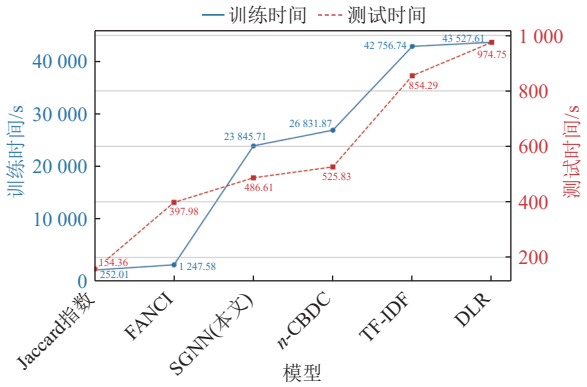


Fig. 13 Results of performance test

图 13 性能测试结果

Jaccard 指数统计分析数据后划分阈值用时最少. FANCI 方法使用机器学习, 相较于神经网络方法训练时间更短. SGNN 直接构建语义关联并通过图神经网络进行语义融合. 虽然相较于机器学习算法 SGNN 训练时间更长, 但 *n*-CBDC, TF-IDF, DLR 神经网络模型产生更大的训练和测试时延. DLR 模型比 SGNN 多使用 2 倍左右的训练和测试时间. SGNN 在实现最优检测 *F*1 分数的同时保持了较高的检测性能.

4.6.3 参数敏感性分析

SGNN 超参数为噪声消除阈值和图节点数. 使用相同划分数据集进行超参数对比实验, 结果如图 14 所示.

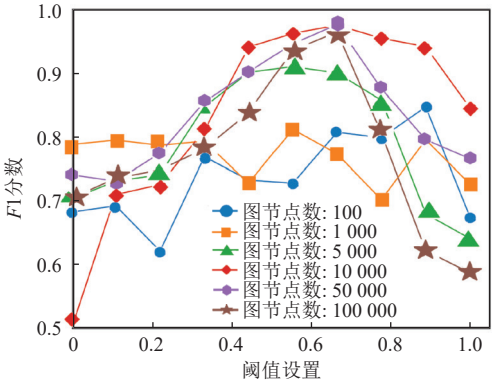


Fig. 14 Results of sensitivity analysis

图 14 敏感性分析结果

实验结果表明, 当图节点数为 100 或者 1 000 时, *F*1 分数震荡较大, 无法产生稳定的分类. 这是由于关联节点数量少难以学习全局性特征. 当图节点数大

于 5 000 后 $F1$ 分数较为稳定. 图节点数这一超参数的选择具有较强的鲁棒性, 图节点数对检测 $F1$ 分数影响不超过 1 个百分点. 当节点数超过 10 000 时, 最佳阈值超参数选择为 0.7 左右. 当阈值超参数变化在 0.2 以下时, 对 $F1$ 分数影响在 3 个百分点以下, 因此噪声阈值超参数选择同样具有较强的鲁棒性.

4.6.4 公开数据集对比实验

中国科技网内的域名采集环境相近, 实验域名样本具有更多的语义相似性. 为了验证 SGNN 的泛化能力, 采用新的公开域名数据集[①]实验. 该数据集的合法域名来源于 Alexa, 共 603 387 个. 恶意域名来源于 20 多类不同的域名生成算法的域名, 共 832 276 个. 按照 7∶3 的比例将该数据集随机划分为训练集和测试集, 并进行对比实验. 实验结果如图 15 所示.

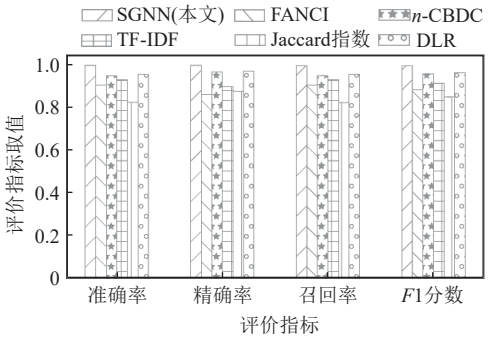


Fig. 15 Comparative results of public datasets

图 15 公开数据集对比结果

实验结果表明, SGNN 取得了最好的效果, 可以实现 0.992 4 的准确率、0.992 7 的精确率、0.991 9 的召回率、0.992 3 的 $F1$ 分数. 同样基于域名语义相似度算法的 DLR 取得了对比方法中次好的效果. DLR 的双层双向深度神经网络融合了域名语义的全局特征以及长距离局部特征, 但相较于 SGNN, 准确率下降 4.2 个百分点、精确率下降 2.65 个百分点、召回率下降 4.14 个百分点、$F1$ 分数下降 3.41 个百分点. 对比实验基于 Jaccard 指数的方法效果最差, 取得低于 0.85 的 $F1$ 分数. 由于实验数据集存在大量 DGA 域名, 导致 IP 分组关联产生了大量的孤立节点. n-CBDC 方法采用 n-gram 对域名进行字符集卷积, 提取了域名丰富的语义特征, 其和 DLR 方法的 $F1$ 分数相差 0.51 个百分点. TF-IDF 和 FANCI 方法效果较差, $F1$ 分数低于 0.91. 这 2 类算法依赖域名的字符分布以及词频, 难以应对 DGA 算法生成的伪装性样本.

5 结 论

针对大规模网络环境多类别大量恶意域名检测难以取得令人满意效果的问题, 提出一种基于域名语义学习的恶意域名检测技术, 并收集现实域名请求数据开展大规模实验. 基于对不同类恶意域名间具有明显的语义区分度, 同类别恶意域名具备字符构造相似性的发现, 提出基于字符语义相似性的域名关联图模型. 该图模型采用字符噪声过滤方法以及域名语义相似度在线聚合方法, 通过基于语义差异的图消息传播机制检测关键特征. 实验结果表明提出的恶意域名检测技术在多类恶意域名的数据集取得平均 0.96 的精确率和 0.97 的召回率, 并且该模型能够进行自演进, 具有较高的识别率和鲁棒性.

作者贡献声明: 付豪提出算法思路、撰写论文、分析实验以及完成论文修订工作; 龙春负责文献整理及论文修改; 宫良一指导论文写作及修订; 魏金侠负责论文修改; 黄潘提供实验数据; 林延中负责数据分析; 孙德刚负责论文修改.

参 考 文 献

[1] Versign. Domain names: Introducing the all new dnib. com [EB/OL]. (2024-12-07)[2024-12-25]. https://www.verisign.com/en_US/domain-names/dnib/index.xhtml

[2] Zhang Jianwu, An Yanjun, Deng Huangyan. A survey on DNS attack detection and security protection[J]. Telecommunications Science, 2022, 38(9): 1−17 (in Chinese)
(章坚武, 安彦军, 邓黄燕. DNS 攻击检测与安全防护研究综述[J]. 电信科学, 2022, 38(9): 1−17)

[3] Porras P, Saïdi H, Yegneswaran V. A foray into conficker's logic and rendezvous points[C/OL] //Proc of the 2nd USENIX Conf on Large-scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More. Berkeley, CA: USENIX Association, 2009[2025-01-22]. https://dl.acm.org/doi/10.5555/1855676.1855683

[4] Gong Liangyi, Li Zhenhua, Wang Hongyi, et al. Overlay-based android malware detection at market scales: Systematically adapting to the new technological landscape[J]. IEEE Transactions on Mobile Computing, 2021, 21(12): 4488−4501

[5] Zhao Fan, Zhao Hong, Chang Zhaobin. Small sample malicious domain names detection method based on transfer learning[J]. Computer Engineering and Design, 2022, 43(12): 3381−3387 (in Chinese)

① https://github.com/Juhong-Namgung/Malicious-URL-and-DGA-Domain-Detection-using-Deep-Learning

(赵凡, 赵宏, 常兆斌. 基于迁移学习的小样本恶意域名检测[J]. 计算机工程与设计, 2022, 43(12): 3381−3387)

[6] Gong Liangyi, Li Zhenhua, Qian Feng, et al. Experiences of landing machine learning onto market-scale mobile malware detection[C/OL]//Proc of the 15th European Conf on Computer Systems. New York: ACM, 2020[2025-01-22]. https://doi.org/10.1145/3342195.3387530

[7] Zhang Qing, Zhang Wenchuan, Ran Xingcheng. Malicious domain names detection based on CNN-BiLSTM and attention mechanism[J]. Journal of China Academy of Electronics and Information Technology, 2022, 17(9): 848−855 (in Chinese)
(张清, 张文川, 冉兴程. 基于 CNN-BiLSTM 和注意力机制的恶意域名检测[J]. 中国电子科学研究院学报, 2022, 17(9): 848−855)

[8] Yuan Fuxiang, Wang Zheng, Liu Fenlin, et al. Malicious fast-flux domains detection algorithm based on IP distribution and request response time[J]. Journal of Information Engineering University, 2017, 18(5): 601−606 (in Chinese)
(袁福祥, 王琤, 刘粉林, 等. 基于 IP 分布及请求响应时间的恶意 fast-flux 域名检测算法[J]. 信息工程大学学报, 2017, 18(5): 601−606)

[9] Peng Chengwei, Yun Xiaochun, Zhang Yongzheng, et al. Detecting malicious domains using co-occurrence relation between DNS query[J]. Journal of Computer Research and Development, 2019, 56(6): 1263−1274 (in Chinese)
(彭成维, 云晓春, 张永铮, 等. 一种基于域名请求伴随关系的恶意域名检测方法[J]. 计算机研究与发展, 2019, 56(6): 1263−1274)

[10] Gong Liangyi, Lin Hao, Li Zhenhua, et al. Systematically landing machine learning onto market-scale mobile malware detection[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 32(7): 1615−1628

[11] Yadav S, Reddy A K K, Reddy A L N, et al. Detecting algorithmically generated malicious domain names[C]//Proc of the 10th ACM SIGCOMM Conf on Int Measurement. New York: ACM, 2010: 48−61

[12] Cucchiarelli A, Morbidoni C, Spalazzi L, et al. Algorithmically generated malicious domain names detection based on *n*-grams features[J]. Expert Systems with Applications, 2021, 170: 114551

[13] Zhao Hong, Chen Zhiwen, Yan Rongjing. Malicious domain names detection algorithm based on statistical features of URLs[C]//Proc of the 25th IEEE Int Conf on Computer Supported Cooperative Work in Design (CSCWD). Piscataway, NJ: IEEE, 2022: 11−16

[14] Nguyen T D, CAO T D, Nguyen L G. DGA botnet detection using collaborative filtering and density-based clustering[C]//Proc of the 6th Int Symp on Information and Communication Technology. New York: ACM, 2015: 203−209

[15] Can N V, Tu D N, Tuan T A, et al. A new method to classify malicious domain name using neutrosophic sets in DGA botnet detection[J]. Journal of Intelligent & Fuzzy Systems, 2020, 38(4): 4223−4236

[16] Bilge L, Sen S, Balzarotti D, et al. EXPOSURE: A passive DNS analysis service to detect and report malicious domains[J]. ACM Transactions on Information and System Security, 2014, 16(4): 1−28

[17] Manadhata P, Yadav S, Rao P, et al. Detecting malicious domains via graph inference[C]//Proc of the 2014 Workshop on Artificial Intelligent and Security Workshop. New York: ACM, 2014: 59−60

[18] Sun Xiaoqing, Tong Mingkai, Yang Jiahai, et al. HinDom: A robust malicious domain detection system based on heterogeneous information network with transductive classification[C]// Proc of the 22nd Int Symp on Research in Attacks, Intrusions and Defenses (RAID 2019). Berkeley, CA: USENIX Association, 2019: 399−412

[19] Cheng Yanan, Chai Tingting, Zhang Zhaoxin, et al. Detecting malicious domain names with abnormal whois records using feature-based rules[J]. The Computer Journal, 2022, 65(9): 2262−2275

[20] Antonakakis M, Perdisci R, Nadji Y, et al. From throw-away traffic to bots: Detecting the rise of DGA-based malware[C]//Proc of the 21st USENIX Security Symp (USENIX Security 12). Berkeley, CA: USENIX Association, 2012: 491−506

[21] Vinayakumar R, Soman K P, Poornachandran P. Detecting malicious domain names using deep learning approaches at scale[J]. Journal of Intelligent & Fuzzy Systems, 2018, 34(3): 1355−1367

[22] Park K H, Song H M, Do Yoo J, et al. Unsupervised malicious domain detection with less labeling effort[J]. Computers & Security, 2022, 116: 102662

[23] Ma Donglin, Zhang Shuhuan, Kong Fanqi, et al. Malicious domain name detection based on Doc2Vec and hybrid network[C]//Proc of the 8th Annual Int Conf on Geo-Spatial Knowledge and Intelligence. Princeton, NJ: IOP Publishing, 2021: 12089

[24] Jiang Yanshu, Jia Mingqi, Zhang Biao, et al. Malicious domain name detection model based on CNN-GRU-attention[C]//Proc of the 33rd Chinese Control Decision Conf (CCDC). Piscataway, NJ: IEEE, 2021: 1602−1607

[25] Yang Luhui, Liu Guangjie, Dai Yuewei, et al. Detecting stealthy domain generation algorithms using heterogeneous deep neural network framework[J]. IEEE Access, 2020, 8: 82876−82889

[26] Wang Wei, Luo Pengyu. DGA malicious domain detection based on machine learning modeling[J]. Communications Technology, 2022, 55(6): 753−761(in Chinese)
(王伟, 罗鹏宇. 基于机器学习建模的 DGA 恶意域名检测[J]. 通信技术, 2022, 55(6): 753−761)

[27] Liu Shanling, Qi Zhenghua. Malicious domain detection based on diversified characteristics[J]. Journal of Nanjing University of Posts and Telecommunications: Natural Science Edition, 2021, 41(6): 95−100 (in Chinese)
(刘善玲, 祁正华. 基于特征多样化的恶意域名检测[J]. 南京邮电大学学报: 自然科学版, 2021, 41(6): 95−100)

[28] Jiang Hongling, Dai Junwei. DGA malicious domain name detection method[J]. Journal of Beijing Information Science & Technology University: Natural Science Edition, 2019, 34(5): 45−50 (in Chinese)
(蒋鸿玲, 戴俊伟. DGA 恶意域名检测方法[J]. 北京信息科技大学学报: 自然科学版, 2019, 34(5): 45−50)

[29] Zhang Yang, Liu Tingwen, Sha Hongzhou, et al. Malicious domain detection based on multiple-dimensional features[J]. Journal of Computer Applications, 2016, 36(4): 941−944 (in Chinese)
(张洋, 柳厅文, 沙泓州, 等. 基于多元属性特征的恶意域名检测[J]. 计算机应用, 2016, 36(4): 941−944)

[30] Vaswani A, Shazeer N, Paramar N, et al. Attention is all you need[C]//Proc of the 31st Int Conf on Neural Information Processing Systems (NIPS'17). New York: ACM, 2017: 6000−6010

[31] Yang Luhui, Liu Guangjie, Wang Jinwei, et al. A semantic element representation model for malicious domain name detection[J]. Journal of Information Security and Applications, 2022, 66: 102662

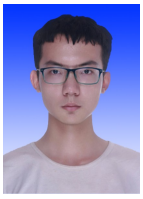
[32] Mikolov T, Chenkai, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint, arXiv: 1301.378, 2013

[33] Schüppen S, Teubert D, Herrmann P, et al. FANCI: Feature-based automated NXDomain classification and intelligence[C]//Proc of the 27th USENIX Security Symp (USENIX Security 18). Berkeley, CA: USENIX Association, 2018: 1165−1181

[34] Xu Congyuan, Shen Jizhong, Du Xin. Detection method of domain names generated by DGAs based on semantic representation and deep neural network[J]. Computers & Security, 2019, 85: 77−88

[35] Le F, Ortiz J, Verma D, et al. Policy-based identification of IoT devices' vendor and type by DNS traffic analysis[J/OL]. Policy-Based Autonomic DataGovernance, 2019: 180−201[2025-01-22]. https://doi.org/10.1007/978-3-030-17277-0_10

[36] Wei Jinxia, Long Chun, Fu Hao, et al. Malicious domain name detection method based on enhanced embedded feature hypergraph learning[J]. Journal of Computer Research and Development, 2024, 61(9): 2334−2346 (in Chinese)
(魏金侠, 龙春, 付豪, 等. 基于增强嵌入特征超图学习的恶意域名检测方法[J]. 计算机研究与发展, 2024, 61(9): 2334−2346)



Fu Hao, born in 1999. PhD candidate. His main research interests include malicious domain name detection, network traffic analysis, and machine learning.

付 豪, 1999 年生. 博士研究生. 主要研究方向为恶意域名检测、网络流量分析、机器学习.

Long Chun, born in 1979. PhD, senior engineer, PhD supervisor. Member of CCF. His main research interests include artificial intelligence based network unknown attack detection, malicious domain name detection, and network traffic analysis.

龙 春, 1979 年生. 博士, 正高级工程师, 博士生导师. CCF 会员. 主要研究方向为基于人工智能的网络未知攻击检测、恶意域名检测、网络流量分析.

Gong Liangyi, born in 1987. PhD, senior engineer, master supervisor. Member of CCF. His main research interests include network attack detection, malicious domain name detection, Web attack analysis, and machine learning.

宫良一, 1987 年生. 博士, 高级工程师, 硕士生导师. CCF 会员. 主要研究方向为网络攻击检测、恶意域名检测、Web 攻击分析、机器学习.

Wei Jinxia, born in 1987. PhD, senior engineer, master supervisor. Her main research interests include artificial intelligence-based network unknown attack detection, malicious domain name detection, and network traffic analysis.

魏金侠, 1987 年生. 博士, 高级工程师, 硕士生导师. 主要研究方向为基于人工智能的网络未知攻击检测、恶意域名检测、网络流量分析.

Huang Pan, born in 2000. Bachelor, engineer. His main research interests include Web attack detection, penetration testing, and malicious domain name analysis.

黄 潘, 2000 年生. 学士, 工程师. 主要研究方向为 Web 攻击检测、渗透测试、恶意域名分析.

Lin Yanzhong, born in 1973. Master, vice president of Coremail Technology. His main research interests include email scaling and anti-phishing emails.

林延中, 1973 年生. 硕士, Coremail 技术副总裁. 主要研究方向为邮件扩容、反钓鱼邮件.

Sun Degang, born in 1970. PhD, senior engineer, PhD supervisor. His main research interests include communication security, network architecture, and network and system security.

孙德刚, 1970 年生. 博士, 正高级工程师, 博士生导师. 主要研究方向为通信安全、网络体系结构、网络与系统安全.