计 算 机 研 究 与 发 展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202330117
61(9): 2334−2346, 2024

基于增强嵌入特征超图学习的恶意域名检测方法

魏金侠[1,2] 龙 春[1,2] 付 豪[1,2] 宫良一[1,2] 赵 静[1,2] 万 巍[1,2] 黄 潘[1]
[1](中国科学院计算机网络信息中心 北京 100083)
[2](中国科学院大学计算机科学与技术学院 北京 100049)
(weijinxia@cnic.cn)

Malicious Domain Name Detection Method Based on Enhanced Embedded Feature Hypergraph Learning

Wei Jinxia[1,2], Long Chun[1,2], Fu Hao[1,2], Gong Liangyi[1,2], Zhao Jing[1,2], Wan Wei[1,2], and Huang Pan[1]
[1](*Computer Network Information Center*, *Chinese Academy of Sciences*, *Beijing* 100083)
[2](*School of Computer Science and Technology*, *University of Chinese Academy of Sciences*, *Beijing* 100049)

Abstract Attackers use the domain names to carry out various kinds of network attacks flexibly. Many scholars have put forward some malicious domain name detection methods based on statistical characteristics and association relationship. However, the two methods have shortcomings in the representation of higher-order relationship of domain name attributes, and cannot accurately present the global higher-order relationship between domains. To solve these problems, a malicious domain name detection method based on embedded feature hypergraph learning is proposed. Firstly, the domain name hypergraph structure is constructed by decision tree based on domain name spatial statistical characteristics. The output of the penultimate node of the decision tree is used as a priori condition to form a hyperedge, and the multi-order correlation between domain name traffic is quickly and clearly represented. Secondly, the character embedding features are enhanced based on the hypergraph structure features, and the hidden higher-order relationships between characters are mined from the domain name data based on the statistical characteristics of domain name space and the encoding characteristics of domain name character embedding. Finally, combined with the real domain name system traffic of China Science and Technology Network, the validity and feasibility are analyzed and evaluated, which can quickly and efficiently detect hidden malicious domain names.

Key words malicious domain name; hypergraph learning; decision tree; embedded coding; spatial statistical feature

摘 要 攻击者利用域名灵活地实施各类网络攻击,诸多学者针对性地提出了一些基于统计特征和基于关联关系的恶意域名检测方法,但这2类方法在域名属性高阶关系表示方面存在不足,无法准确呈现域间全局高阶关系.针对这类问题,提出一种基于嵌入式特征超图学习的恶意域名检测方法:首先基于域名空间统计特征利用决策树构建域名超图结构,利用决策树倒数第2层节点的输出结果作为先验条件形成超边,快速将域名流量之间的多阶关联关系清晰地表示出来;其次基于超图结构特征对字符嵌入特征进行增强编码,基于域名空间统计特征和域名字符嵌入编码特征从域名数据中挖掘出字符间隐藏的高阶关

收稿日期: 2023-04-07; **修回日期**: 2023-11-22
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, 2023181); 中国科学院战略性先导科技专项(XDC02030600)
This work was supported by the Cyber Security and Informatization Project of Chinese Academy of Sciences (CAS-WX2022GC-04), the Youth Innovation Promotion Association, Chinese Academy of Sciences (2022170, 2023181), and the Strategic Priority Research Program of Chinese Academy of Sciences (XDC02030600).
通信作者: 龙春(anquanip@cnic.cn)

系；最后结合中国科技网真实的域名系统（domain name system, DNS）流量，对有效性和可行性进行了分析与评估，能够快速高效地检测隐蔽的恶意域名.

关键词 恶意域名；超图学习；决策树；嵌入式编码；空间统计特征

中图法分类号 TP391

域名系统（domain name system, DNS）是互联网的一项重要基础设施服务，它使得域名和IP之间可以相互映射，网络中各种应用活动都与其密切相关. 但是近年来，DNS在提供正常解析服务的同时，也成为各种网络非法活动的主要攻击对象. 越来越多的攻击者通过滥用DNS来达到恶意的目的，如僵尸网络在其扩散与通信中使用DNS技术定位命令和控制（command and control, C&C）、网络钓鱼和恶意代码下载等，企图通过频繁变更域名对应的IP地址或名称服务器（name server, NS）记录隐藏背后的真实的服务器.

为了应对频发的网络攻击事件，诸多学者开展了利用DNS流量检测恶意域名的研究工作，以缓解攻击活动带来的损失和危害. 相关恶意域名检测方法主要集中于基于统计特征的方法和基于关联关系的方法，其中统计特征可以分为域名字符统计特征和DNS流量统计特征. 通过收集中国科技网骨干网2022年1月至2022年7月的域名数据构造恶意域名检测数据集，将相关算法进行实验分析得到3点结论：

1）通过域名字符串统计特征实现恶意域名检测的方法只需要字符串就可以提取，易于实现，尤其是在域名生成算法（domain generation algorithm, DGA）域名检测中效果更好一些，但是这类特征比较简单，效果有限. 将中国科技网收集的数据集应用于基于特征的自动分类模型[1]中，取得了87.45%的准确率和83.44%的召回率.

2）DNS流量统计特征包括解析内容、活动时间记录、TTL（time to live）值等信息，这些特征在良性域名和恶意域名上存在较明显的差异，虽然利用庞大的DNS流量特征可以有效检测出恶意域名，但是往往容易忽略流量攻击行为之间的关系. 将中国科技网收集的数据集应用于基于被动DNS的恶意域名检测模型[2]中，取得了90.36%的准确率和92.54%的召回率.

3）基于关联关系的方法可以充分发掘域名之间的关联性，能够通过这些关联发现更加高级隐蔽的恶意域名，这种关联是在攻击者实施攻击过程中就形成的，不容易被篡改，将中国科技网收集的数据集应用于基于异构图卷积网络的恶意域名检测模型[3]中，取得了86.58%的准确率和85.72%的召回率.

经过分析，这3类方法在准确率和召回率方面还有待于进一步提升的原因是不能够全面表示域名各个属性之间的高阶关系，无法达到多维指标的高效平衡.

为了更好地应对隐蔽攻击，提出一种基于超图学习的恶意域名检测方法. 通过对中国科技网流量进行分析，发现域名流量存在时空相似、跨节点关联的特性. 例如，对于一个恶意域名攻击事件，其存在与其关联的IP或域名节点，这些节点存在多重关联的高阶关系. 在一次攻击事件中，节点关联出现，相互配合完成一次完整的网络活动. 同时，在恶意域名请求之间也存在类似形式的多阶关联关系，不同的恶意域名在一次恶意网络活动中的时空上共同出现，很难在不影响恶意资源利用的情况下消除这种连接关系. 据此，首先通过域名内在的关联行为，从攻击事件角度出发，将域名流量之间的全局相关性进行清晰全面地表示，基于超图表示机制表征域名之间的空间高阶关系；然后利用高阶关系矩阵对域名字符嵌入特征进行变换，强化域名的深层隐含特征；最后结合分类器实现恶意域名的识别. 面临的挑战有2个：

1）如何表征真实环境下某个攻击事件中DNS流量之间的高阶关系；

2）如何充分挖掘域名间隐含的时空共现、跨节点关联特征.

针对挑战1，提出基于决策树的超图构建方法，将域名流量的注册特征、解析特征、字符统计作为基本属性，基于决策树算法形成域名流量样本的超边集，将决策路径的倒数第2层节点的分类输出作为构建超边的依据. 利用决策树构建超边集的原因主要有2个：一是考虑决策树可以在相对短的时间内能够对大型数据源做出可行且效果良好的分类结果；二是决策树对缺失值不敏感，对于输入数据要求不严格，能够很好地处理个别特征缺失给分类带来的影响.

针对挑战2，提出超图学习恶意域名检测方法，首先利用节点−边缘−节点的特征传播方式将超图结构特征进行细化，并通过超图结构收集域名间高阶

空间关联关系，形成超图表示矩阵；再利用表示矩阵强化域名字符嵌入特征以充分挖掘超图结构中的高阶相关性特征来提升模型分类效果.

与前人工作相比，本文工作共有 3 个贡献：

1)提出一种基于决策树的域名流量超图表示方法，利用决策树模型的倒数第 2 层结果作为先验条件形成超边，能够快速将域名流量之间的多阶关联关系清晰地表示出来.

2)提出一种基于增强嵌入特征超图学习的恶意域名检测模型，该模型基于域名空间统计特征和域名字符嵌入编码特征从 DNS 数据中挖掘出隐藏的高阶关系.

3)结合中国科技网连续 7 个月的真实 DNS 流量数据，对有效性和可行性进项分析与评估，实验验证本文模型能够快速高效地检测恶意域名.

1 相关工作

目前已经有很多恶意域名检测的研究工作，这些工作主要集中在基于统计特征的方法和基于关联关系的方法 2 方面. 其中，基于统计特征的方法多采用机器学习算法或者深度学习算法来构建分类器；基于关联关系的方法多采用图结构及推理算法实现恶意域名的分类.

在基于统计特征机器学习的恶意域名检测方面的相关成果较多[4-8]. 最初 Antonakakis 等人[9]提出一种未知的 DGA 发现机制，从域名字符统计特征的角度基于域名相似性以及查询这些相似域名的用户对域名进行聚类，然后利用交替决策树构建簇分类器，能够有效地检测出从未报道过的 DGA 域名. 为了能检测出更多类型的恶意域名，Schüppen 等人[1]从字符串中提取了域名的结构特征、语言特征及统计特征，利用随机森林和支持向量机(SVM)来构建恶意分类器；并在大型公司内部网络和大学校园网环境中验证了该模型的可行性，具有较高的分类精度和低误报率. Chin 等人[10]提取了处于活动状态的恶意域名 DNS 流量特征，通过分类和聚类方法构建了一个检测恶意域名的 2 级机器学习框架，在一级恶意域名分类和二级恶意域名聚类下分别达到了 95.14% 和 92.45% 的准确率. Wang 等人[11]针对恶意域名与良性域名的应答记录进行分析，分别提取了 NS 数量、MX 记录数量、NS 相似性、IP 地址数量、IP 反向解析域数量，发现恶意域名与良性域名的应答信息区别比较明显.

为了更深层次地挖掘恶意域名与良性域名在语义层面上的区别，还有一些工作利用深度学习的方法来自动提取特征实现恶意域名检测[12-13]. Anderson 等人[14]发现正常域名和恶意域名在字符统计分布方面是有差异的，并基于该发现建立了生成对抗网络的 DGA 家族检测模型. Ren 等人[15]利用神经网络和双向长短期记忆(LSTM)神经网络提取域名字符序列特征，实现恶意域名的有效识别和分类，该模型在检测常规和隐蔽的恶意域名方面取得了较高的 $F1$ 值. Ravi 等人[16]利用孪生神经网络来分析域名之间的相似性，然后建立基于深度学习的恶意域名分类，该方法可以有效地识别 DNS 同形异义词攻击，并且对常见的规避网络攻击具有弹性. Opara 等人[17]基于深度学习端到端自动网络钓鱼网页分类方法实现恶意域名识别，利用卷积神经网络(CNN)自动学习 HTML 文本内容中存在的内容统计特征. Yuan 等人[18]通过将双向独立循环神经网络、注意力机制、胶囊网络进行结合的方式提取域名相关的语义信息，以实现高效的恶意域名检测.

上述单纯通过分类实现恶意域名检测的方法大多只考虑了域名本身的特征，没有考虑到域名之间的关联关系，这增加了识别未知或者变种恶意域名的难度. 因此，诸多学者也将域名之间的关联关系作为恶意域名检测的重要特征[19-21]. Zou 等人[22]利用域-IP 关联以及域-主机关联来构建域相似关系的推理图. Zhang 等人[23]给出一种无监督的分析方法，其可以推断出参与恶意软件活动的相关服务器组. Rahbarinia 等人[24]构建一个主机到域的二分图，通过使用从大规模 ISP 网络中收集的 DNS 数据跟踪 DNS 查询行为来有效地检测新的恶意域名. Stevanovic 等人[25]通过分析 DNS 流量识别受感染的来自不同运营 ISP 网络的主机. Peng 等人[26]提出了一种恶意域名检测方法，该方法分析出通过 DNS CNAME 记录与恶意域名连接的域往往也是恶意的结论，通过计算其恶意概率来识别非法域，实验证明了该方案的高检测性能. Sun 等人[27]提出了一个名为 HinDom 的鲁棒域检测系统，该系统将 DNS 场景建模为由客户端、域、IP 地址及其不同关系组成的异构信息网络(HIN)，基于元路径的转导分类方法使 HinDom 能够仅用一小部分标记样本检测恶意域名. 结果表明，HinDom 准确、健壮，可以识别 MsraMiner 僵尸网络.

恶意域名流量往往相互关联出现，这些域名或者 IP 存在多重节点间的高阶关系，并非相互独立. 例如，攻击者通过僵尸网络发起钓鱼攻击，注册了 2 类

不同的域名 attack_fish_*.com 和 attack_bot_*.com, 这2个域名通过 DGA 域名算法生成, 因此彼此间具有字符语义上的相似性关联. 同时注册使用的信息(注册时间、注册人等)也具有一定关联性. 为了控制僵尸主机, 攻击者需要将大量域名 attack_bot_*.com 映射到同一个 IP. 在现有的研究工作中, 基于域名统计特征虽然可以检测出僵尸网络域名, 但是难以通过同样的方法同时检测出与之关联的钓鱼域名, 对这种通过僵尸网络发起钓鱼攻击行为的检测效果不理想. 基于域名关联分析的方法形成的是普通图的关联结构, 只能表示2个节点间存在某一种关联关系, 无法同时表示在网络攻击中存在的多个域名之间的高阶关联关系, 如上述攻击例子中恶意域名间的算法生成关联、注册信息关联、攻击行为关联等多重节点间的高阶关系. 当攻击者伪装或者隐藏其行为时, 普通的单一关联难以描述完整的攻击关系, 而超图可以原生表示多个节点间多重关联关系, 因此超图网络可以传播节点间的高阶信息.

因此, 借鉴现有的工作基础, 综合考虑域名空间统计特征和域名字符嵌入特征, 利用域名空间统计特征构建超图结构, 表示出域名之间存在的高阶空间关系, 包括域名−IP−注册商−注册时间−更新时间−有效时间等; 然后利用表征高阶空间关系的超图表示矩阵增强域名字符嵌入特征, 挖掘域间及域内特征的深层关系, 实现全局特征强化, 使得同类域名流量特征具有强关联性.

2 超图学习检测模型

基于增强嵌入特征超图学习的恶意域名检测方法架构如图1所示, 包括特征分析、超图构建和检测模型3部分: 1)在特征分析部分, 首先从 DNS 流量中提取域名空间统计特征, 表征域名之间的高阶关联关系, 其次利用 CNN 与 LSTM 相结合的方式提取域名字符嵌入式编码特征; 2)在超图构建部分, 利用决策树对域名空间特征进行粗分类, 将决策树倒数第2层节点输出结果作为分类依据形成超边集, 决策树分到同一个类别的样本位于一个超图区域内, 具有一条超边, 形成超图结构; 3)检测模型部分, 利用超图结构的表示矩阵对域名字符的编码特征进行变换, 进一步强化域名特征, 完整表示出域名之间的全局关联, 然后将强化之后的特征输入到神经网络检测模型中训练, 构建恶意域名分类器, 有效识别与已知恶意域名有关联的恶意域名特征, 完整表示出域名之间的全局关联, 最后将强化之后的特征输入到神经网络检测模型中训练, 构建恶意域名分类器, 有效识别与已知恶意域名有关联的恶意域名.

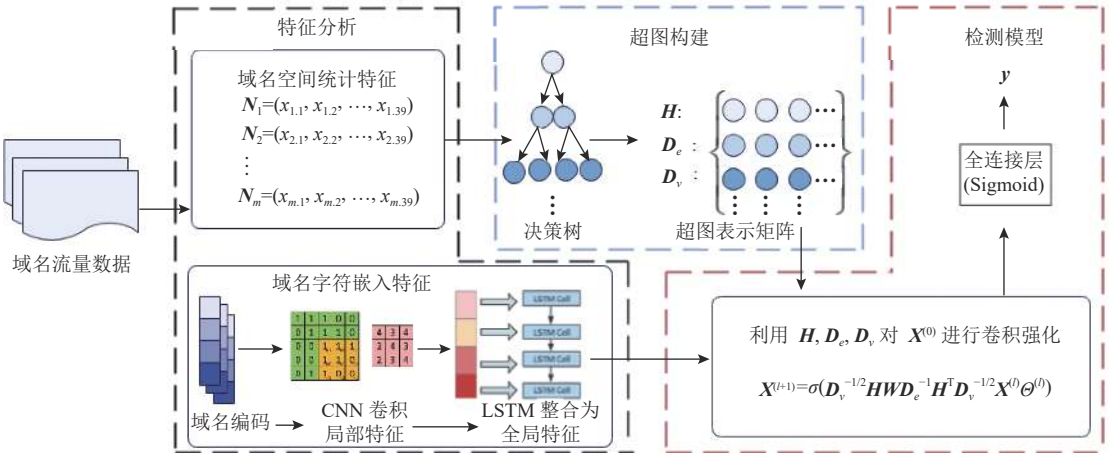

Fig. 1 Architecture for malicious domain name detection method based on enhanced embedded feature hypergraph learning

图1 基于增强嵌入特征超图学习的恶意域名检测方法架构

2.1 特征分析

1)域名空间统计特征

为了充分挖掘正常域名和异常域名的差异之处, 本文分析了3类域名空间的统计特征, 包括域名本身统计特征、Whois 查询统计特征、NS 查询记录统计特征, 共计39维. 令 $\boldsymbol{N}_p=(x_{p,1}, x_{p,2}, \cdots, x_{p,39})$ 表示一条域名样本特征向量, 其中 p 表示集合中域名样本的数量, 39 表示每一条样本的39维域名空间统计特征.

① 4维域名本身统计特征. 包括域名的长度、域名中的数字占比、域名中最长的连续数字的长度、最长连续数字序列的偏移值. 为了方便用户记忆, 便于人们访问网络服务, 良性域名简短易读且具有一

定的实际含义，在字符长度、数字占比、连续数字长度等方面规律性比较明显，而恶意域名字符特征分布很随机，这两者域名在统计特征概率分布上存在明显的差异.

② 25 维 Whois 查询统计特征. 包括注册商的数量、注册国家的数量、名称服务器的数量、注册人的数量、域名系统安全扩展(DNSSEC)是否启用、可选注册状态的数量、注册状态值、注册年、注册月、注册日、册时、注册分、注册秒、过期年、过期月、过期日、过期时、过期分、过期秒、最近更新年、最近更新月、最近更新日、最近更新时、最近更新分、最近更新秒. 通过 Whois 查询可以知道域名注册人相关信息，包括域名的注册商、注册人、注册时间、过期时间等信息. 正常域名具有域名信息的稳定特征，生命周期往往较长，提供的服务几乎都可以被访问. 相比之下，恶意域名的主要目的是为了欺骗用户开展恶意活动，为了降低被发现的概率，恶意域名的有效期通常较短. 另外，良性域名为了提高知名度，在注册时拥有较完整的注册信息，而恶意域名为了掩人耳目，注册信息随机且不完整.

③10 维 NS 查询记录统计特征. 包括 IP 数量、域名别名的数、解析服务器数量、域名服务器的数量、主要名称服务器的数量、负责人邮件地址数量、重刷新时间、重试时间、有效时间、生存时间. 良性域名通常根据需要使用负载均衡技术，在一定时间内域名会解析出多个不同的 IP，而恶意攻击者掌握的 IP 资源有限，使得恶意域名在一定时间内解析的 IP 个数比正常域名少，并且恶意域名的 NS 记录不完整，缺失值比较明显.

2)域名字符嵌入特征编码

域名特征提取是实现恶意域名检测的基础，特征选取的好坏直接影响模型的检测效果，因此，有必要挖掘出域名字符中可能存在的深层关系. 本文采用嵌入式时空特征编码的方式对域名字符深层关系进行分析：首先利用 Word2vec 对域名进行了嵌入学习，根据域名字母出现的频次对域名进行字符编码；其次利用 CNN 结合 LSTM 模型对域名进行编码以挖掘域名字符串的字符之间的相关关系，其中 CNN 的卷积和滑动窗口的操作可以提取域名字符串中字符和字符之间的局部相关性，而 LSTM 通过保留前序序列的状态来捕获各个局部之间长序列之间的语义信息，通过先细粒度的局部卷积提取再长序列全局关联的方式挖掘出单个域内的隐含语义信息.域名嵌入式时空特征编码模型如图 2 所示，该模型包括 3 个部分：域名编码层、CNN 层和 LSTM 层.

①编码层. 模型编码时，将域名数据样本输入到编码模型之后经过模型运算最终输出能表示域名并且保留域名中字符关联关系的编码向量. 首先，本文选择 Word2vec 编码模型将每个域名编码成数值向量 $\boldsymbol{e}$，将词映射成较短的向量同时也保留词和词之间的相关性. 域名经过 Word2vec 编码后的特征矩阵表示为

$$\boldsymbol{S}=(\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_l),\quad \boldsymbol{S}\in\mathbb{R}^{d\times l}, \tag{1}$$

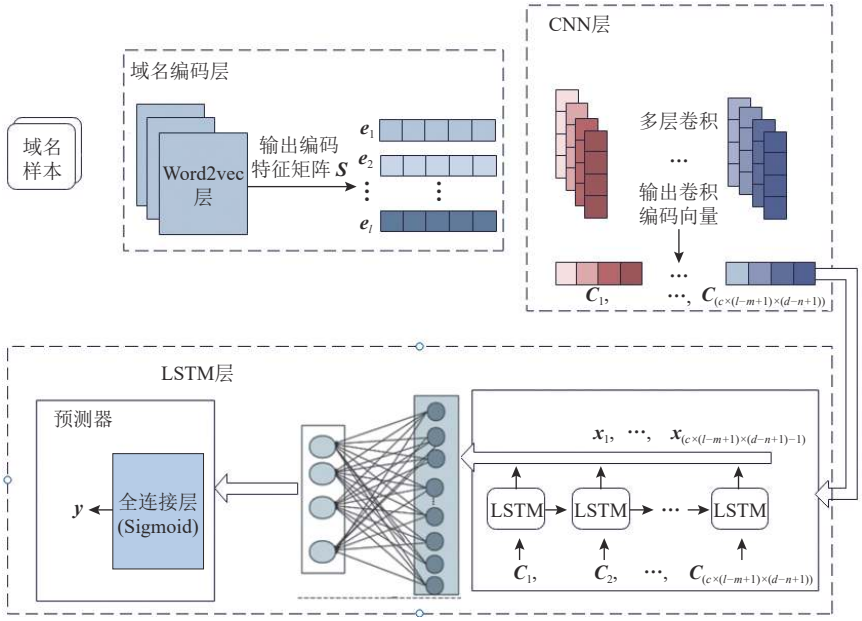

Fig. 2 Architecture of domain embedded feature encoding

图 2 域名嵌入特征编码架构

其中$\boldsymbol{e}_r \in \mathbb{R}^d$为每个域名的第$r$个字符的词袋编码向量，$d$为字符向量词袋编码的维度，$l$为字符个数.

② CNN层. 将域名词袋编码矩阵$\boldsymbol{S}$输入到卷积层，卷积层通过卷积核对域名词袋编码向量进行卷积操作，计算过程为：

$$C(i,j)=(\boldsymbol{S}\cdot\boldsymbol{K})(i,j)=f\left(\sum_{i=1}^{\ell}\sum_{j=1}^{d}S(i+m,j+n)K(m,n)+k\right), \tag{2}$$

其中i和j为卷积核的位置坐标，卷积核的大小为$m\times n$，f为激活函数，$\boldsymbol{K}$为卷积核，k为偏置项，$\boldsymbol{K}$和k通过模型训练更新. 通过多层卷积操作，捕获域名字符和字符之间的局部关联，得到域名的卷积编码向量. 词袋编码输出为$l\times d$维的矩阵，矩阵经过卷积之后变成了$c\times(l-m+1)\times(d-n+1)$维的矩阵，$c$为通道数，$m$和$d$为嵌入的大小，将卷积矩阵展开重构为$c\times(l-m+1)\times(d-n+1)$的向量$\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_{(c\times(l-m+1)\times(d-n+1))}$.

③LSTM层. 为了捕捉域名字符串之间的前后关联深层信息，将上述输出的卷积层编码向量输入LSTM层进行预训练对域名进行编码. 按照$\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_{(c\times(l-m+1)\times(d-n+1))}$的顺序依次输入到LSTM网络中，对于序列中的每个元素，每一层的运算为：

$$\begin{cases}\boldsymbol{i}_t=\sigma(\boldsymbol{W}_{ii}\boldsymbol{C}_t+b_{ii}+\boldsymbol{W}_{hi}\boldsymbol{x}_{t-1}+b_{hi}),\\ \boldsymbol{f}_t=\sigma(\boldsymbol{W}_{if}\boldsymbol{C}_t+b_{if}+\boldsymbol{W}_{hf}\boldsymbol{x}_{t-1}+b_{hf}),\\ \boldsymbol{g}_t=\tanh(\boldsymbol{W}_{ig}\boldsymbol{C}_t+b_{ig}+\boldsymbol{W}_{hg}\boldsymbol{x}_{t-1}+b_{hg}),\\ \boldsymbol{o}_t=\sigma(\boldsymbol{W}_{io}\boldsymbol{C}_t+b_{io}+\boldsymbol{W}_{ho}\boldsymbol{x}_{t-1}+b_{ho}),\\ \boldsymbol{l}_t=\boldsymbol{f}_t\odot\boldsymbol{l}_{t-1}+\boldsymbol{i}_t\odot\boldsymbol{g}_t,\\ \boldsymbol{x}_t=\boldsymbol{o}_t\odot\tanh(\boldsymbol{l}_t),\end{cases} \tag{3}$$

其中$t\in[1,c\times(l-m+1)\times(d-n+1)]$表示LSTM的单元数，$\boldsymbol{C}_t$是输入序列的第$t$个向量元素，$\boldsymbol{x}_t$表示$\boldsymbol{C}_t$在隐藏层的状态，$\boldsymbol{l}_t$是第$t$个单元所处的状态，$\boldsymbol{x}_{t-1}$是第$t-1$个元素的隐藏层状态，$\boldsymbol{i}_t$，$\boldsymbol{f}_t$，$\boldsymbol{g}_t$，$\boldsymbol{o}_t$分别是输入门、遗忘门、单元门、输出门，$\sigma$为激活函数，$\odot$为矩阵的乘法运算，$\boldsymbol{W}_i$为LSTM输入的权重矩阵，下角标$i$表示LSTM的输入内容，$b_{ii}$，$b_{if}$，$b_{ig}$，$b_{io}$为LSTM输入的偏置数，$\boldsymbol{W}_h$为单元输入的权重矩阵，下角标$h$表示单元的输入内容，$b_{hi}$，$b_{hf}$，$b_{hg}$，$b_{ho}$为单元输入的偏置数，都是训练过程中要学习的超参数. 最后利用LSTM的输出结果训练全连接层预测器.

2.2 超图构建

2.2.1 超图理论

对于一个简单图，其每条边均与2个顶点相关联，即每条边的度被限制为2，而超图则允许每一条边的度为任何非负整数，超图的简单示意如图3所示.

超图的数学定义可以表述为，超图是一个三元组$G=(V,E,\boldsymbol{W})$，其中V表示顶点集合，E表示超边集合，$\boldsymbol{W}$表示记录各条超边权重的对角矩阵. 可以用邻接矩阵$\boldsymbol{H}$来表述一个超图，$\boldsymbol{H}$是一个$|V|\times|E|$的矩阵，表示方式定义为

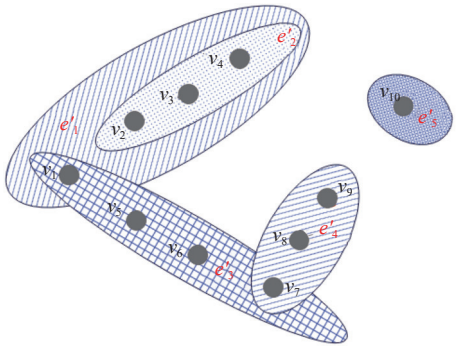

Fig. 3 A simple schematic diagram of hypergraph structure

图3 超图结构的简单示意图

$$h(v,e')=\begin{cases}1, & v\in e',\\ 0, & v\notin e',\end{cases} \tag{4}$$

其中h表示矩阵$\boldsymbol{H}$的元素，$v\in V,e'\in E$，如图3中的10个顶点$v_1\sim v_{10}$构成了集合V，5条超边$e'_1\sim e'_5$构成了集合E.

令$d(v)$和$\delta(e')$分别表示顶点的度和超边的度，$d(v)=\sum_{e'\in E}w(e')\times h(v,e')$，$\delta(e')=\sum_{v\in V}h(v,e')$. $\boldsymbol{D}_v$和$\boldsymbol{D}_{e'}$分别表示超图顶点度矩阵和超边度矩阵的对角矩阵. 超图的顶点(超边)分类问题的学习目标函数为：

$$\arg\min_{g}\{R_{\text{emp}}(g)+\Omega(g)\}, \tag{5}$$

其中$g(\cdot)$为分类函数，本文形成超图的分类函数为决策树，$\Omega(g)$为标准化损失函数，$R_{\text{emp}}(g)$为有监督的经验误差，标准化损失函数计算方法为：

$$\Omega(g)=\frac{1}{2}\sum_{e'\in E}\sum_{u,v\in V}\frac{w(e')h(u,e')h(v,e')}{\delta(e')}\left(\frac{g(u)}{\sqrt{d(u)}}-\frac{g(v)}{\sqrt{d(v)}}\right)^2, \tag{6}$$

其中u和v表示V中不同的顶点.

2.2.2 超图构建

为了呈现域名之间的空间全局关联关系，利用39维域名空间特征形成超图结构，通过超图结构把具有关联关系的域名划分到一个超边区域内. 在超图算法中，超边构建的好坏直接影响到模型的整体检测效果. 本文利用的域名空间统计特征大多是数量特征或者类别特征，比较适合决策树的分类方法. 超图构建过程分为4个阶段：

阶段1. 针对选取的28万条域名样本进行空间特征提取操作，提取39维域名空间统计特征，其中

包括 4 维域名本身统计特征、25 维 Whois 查询统计特征和 10 维 NS 查询记录统计特征，一条域名样本的特征向量可以表示为$\boldsymbol{N}_p=(x_{p,1},x_{p,2},\cdots,x_{p,39})$的形式，其中$p\in[1,280\,000]$表示集合中域名样本序号.

阶段 2. 从上述 28 万条域名样本中随机选取 1 万条数据作为训练样本对决策树模型进行训练，得到最优参数组合的决策树分类模型.

阶段 3. 将其余 27 万条数据作为测试样本输入决策树，根据决策路径倒数第 2 层叶节点的预测结果将样本分成C个类别，然后根据分类结果形成超边集E，E中包含C条超边；超边区域内的所有测试样本形成顶点集合$V=\{v_1,v_2,\cdots,v_{270\,000}\}$，即$V$中包含 27 万个元素.

阶段 4. 根据式(4)计算$h(v,e)$的值，构建超图的邻接矩阵$\boldsymbol{H}$；根据$d(v)$和$\delta(e)$的计算方法生成超图顶点度矩阵$\boldsymbol{D}_v$和超边度矩阵的对角矩阵$\boldsymbol{D}_{e'}$，这 3 个矩阵将用于式(7)中实现域名嵌入特征的增强.

2.3 恶意域名检测

构造超图结构后，利用超图神经网络完成节点分类任务来实现恶意域名检测. 首先利用超图结构的表示矩阵对域名字符的编码特征进行卷积变换，进一步强化域名特征，完整表示出域名之间全局关系. 超图的单层卷积运算公式为：

$$\boldsymbol{X}^{(l+1)}=\sigma(\boldsymbol{D}_v^{-1/2}\boldsymbol{H}\boldsymbol{W}\boldsymbol{D}_e^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{D}_v^{-1/2}\boldsymbol{X}^{(l)}\Theta^{(l)}),\quad(7)$$

其中$\boldsymbol{X}^{(l)}$为第l层神经网络的嵌入向量，$\boldsymbol{X}^{(0)}$为域名字符嵌入编码特征矩阵，σ为非线性激活函数，$\Theta_l,l\in\{1,2,\cdots,L\}$为训练过程中要学的参数，使用交差熵损失函数来反向传播更新参数Θ，L为神经网络层数.

最后将增强之后的域名嵌入特征输入到神经网络的全连接层输出预测结果，实现对恶意域名的识别，取得高效均衡的指标.

3 实　　验

3.1 数据集

本文选取了中国科技网骨干网真实的域名流量数据，收集 2022 年 1 月到 2022 年 7 月的域名流量数据并利用恶意域名威胁情报库对其进行标记，恶意样本数量为 18 万条，其中包括 200 多种恶意域名类型，正常的域名由威胁情报厂商提供，样本数量为 10 万条. 本文对恶意域名数据进行了去重，然后记录了恶意域名随时间的分布情况，对于多次出现的恶意域名只记录首次发现的时间，分布情况如图 4 所示. 由于在 2022 年 2 月进行了一次大规模数据的入库，故采集到的恶意域名数据在 2 月左右有 1 次较大的波动. 可以看出随着时间的推移，会出现少量的先前从未出现过的域名.

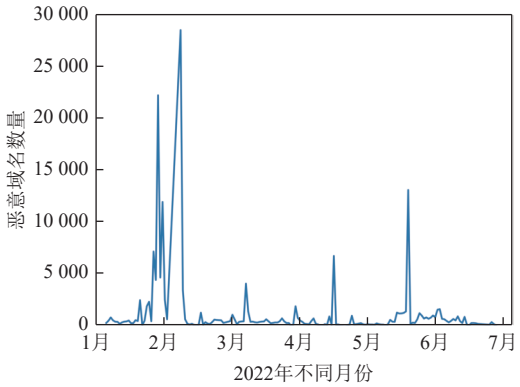

Fig. 4 Distribution of malicious domain name with time

图 4 恶意域名随时间的分布

3.2 实验设置

本文模型的实验运行在 Windows10 专业版本的工作站上，Intel® Xeon® Gold 5218 CPU，内存 128 GB，利用 Pytorch 神经网络框架，版本为 1.12.

为了评估本文方法的鲁棒性和有效性，实验选取了少量数据样本和大量数据样本 2 种情况进行分析. 同时，对于每种情况分别选取了 3 个当前主流的恶意域名检测方法作为基线，这些算法分别从不同的角度对恶意域名进行了预测：FANCI[1] 方法使用了域名的字符串特征对恶意域名进行检测，在实际情况中仅仅使用域名本身特征，具有快速检测的优点. Liu 等人[2] 利用了域名字符特征和 NSLookup 解析记录，并结合域名的一些静态特征进行恶意域名检测，提升了模型的检测效果和可解释性. 另外，还有通过异构图分析的方法提取域名节点之间的关联关系，以实现恶意域名检测的相关工作[3]，可以进一步挖掘域间隐含关系.

3.2.1 少量域名数据样本情况

本文考虑少量域名数据样本的原因主要有 2 个：1)现有的恶意域名检测相关工作已经可以取得不错的检测效果，但大多依赖大量的恶意标签，这在真实的网络场景中比较难实现，因此本文考虑了少量样本少标签的情况；2)从图 4 中可以看出随着时间的推移，往往出现数量不多的新的恶意域名，用已经训练好的模型去检测新的恶意域名容易被漏掉，而在真实网络环境下，模型需要有较快的检测速度和较高的召回率，少量样本可以加快模型的更新速度. 综合考虑上述 2 方面的问题，本文在实验中考虑了少量样本条件下模型的性能，从数据集中随机选取 100

条样本进行训练，其中良性域名样本和带有标记的恶意域名样本的比例相同，其余 27.99 万条作为测试数据集，以此验证本文方案适用于少量标记样本的场景.

3.2.2 大量域名数据样本情况

为了验证本文方法的泛化性能，也考虑了大量域名样本对模型的影响，从 28 万条样本中随机选取 10 000 条域名数据作为训练数据，其中良性域名样本和带有标记的恶意域名样本的比例相同，其他 27 万条数据作为测试数据. 一般情况下，选取训练的样本数量越多则预测效果就越好，但是经过实验发现使用过多的样本会使超图构建消耗大量的时间. 如果超图过大需要输入足够的样本构建图进行检测，从而造成了检测时延，经过实验权衡分析，选择 10 000 条数据作为训练集构建超图.

3.3 评价指标

为了定量衡量本文模型在恶意域名检测方面的效果，本文采用 4 个指标：准确率（*Accuracy*）、精确率（*Precision*）、召回率（*Recall*）、*F*1 值.

1）$Accuracy=\dfrac{TP+TN}{TP+FP+TN+FN}$；

2）$Precision=\dfrac{TP}{TP+FP}$；

3）$Recall=\dfrac{TP}{TP+FN}$；

4）$F1=\dfrac{2Precision\times Recall}{Precision+Recall}$.

其中，*TP*（true positive）表示预测是恶意而实际是恶意的样本数量；*FP*（false positive）表示实际是良性而预测是恶意的样本数量；*TN*（true negative）表示实际是良性而预测为良性的样本数量；*FN*（false negative）表示实际是恶意而预测为良性的样本数量.

3.4 模型效果分析

3.4.1 本文模型与已有模型在恶意域名检测方面的对比分析

1）根据 3.2.1 节中对实验数据进行划分，验证在少量样本训练的情况下模型的有效性，分别将本文采集的域名样本输入到 FANCI[1] 方法、Liu 等人[2] 的方法以及异构图分析[3] 方法中进行训练和预测，得到模型的性能如图 5 所示. 字符嵌入部分本文利用 Word2vec 编码输出 100×100 的向量，经过 CNN 层后输出 100×1 024 的特征矩阵，再经过 LSTM 之后域名变换成 100 个 1×100 的嵌入向量. 在预测器部分采用 2 个全连接层，输出 2 维向量. 在决策树部分，本文选择基尼指数作为划分标准构建 CART 决策树. 在超图构建部分，使用 128 大小的隐含层，进行 4 次卷积并进行全连接层输出预测结果.

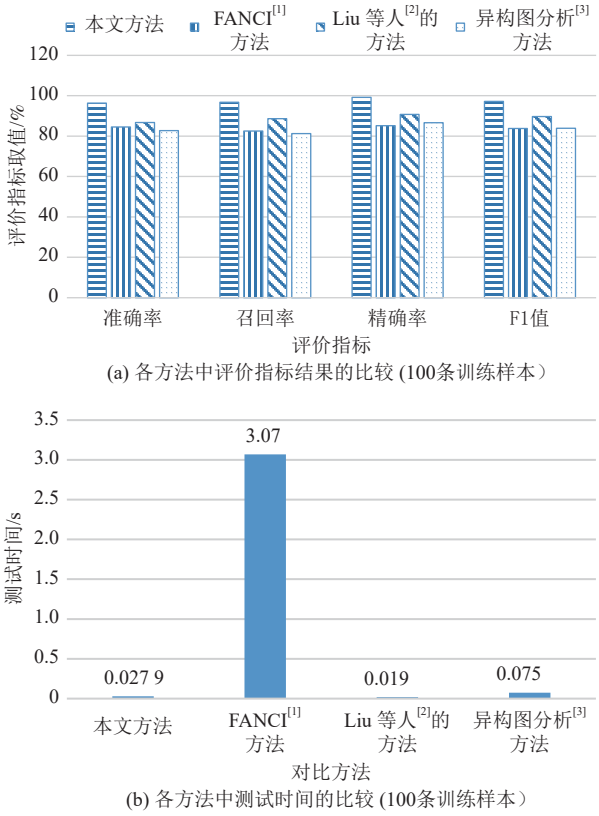

Fig. 5 Comparison of evaluating indicator results and testing time for each method (100 samples)

图 5 各方法中评价指标结果和测试时间的比较 (100 条训练样本)

在训练样本选取 100 条的情况下，本文方法取得了 96.33% 的准确率、96.81% 的召回率、99.17% 的精确率和 0.971 8 的 *F*1 值，测试时间为 0.027 9 s；FANCI 方法取得了 84.53% 的准确率、82.52% 的召回率、85.12% 的精确率和 83.8% 的 *F*1 值，测试时间为 3.07 s；Liu 等人的方法取得了 86.72% 的准确率、88.60% 的召回率、90.74% 的精确率和 89.66% 的 *F*1 值，测试时间为 0.019 s；异构图分析方法取得了 82.76% 的准确率、81.25% 的召回率、86.67% 的精确率和 83.87% 的 *F*1 值，测试时间为 0.075 s.

从图 5 可以看出，本文方法 4 个关键指标均高于其他模型，虽然在训练时间上比 Liu 等人的方法慢 0.008 9 s，但是这个不会给检测速率带来较大的影响. 本文方法在召回率、精确率、准确率和 *F*1 值方面相比其他 3 种方法取得了较好的效果. 在使用少量样本情况下训练，本文方法具有较好的泛化性能，可以通过少量样本学习到半年时间内恶意域名的内在特征.

同时，本文也对 FANCI 方法、Liu 等人的方法以及异构图分析的方法模型的适用场景进行了分析.

FANCI 方法利用 NXD 响应数据包的字符串统计特征，使用随机森林和支持向量机实现 DGA 域名检测，因此该方法在 DGA 域名识别方面效果好，而在中国科技网真实骨干网络环境中 NXD 数据包比较少，结合中国科技网网络安全应急小组实际业务需求，本文需要关注 DGA 之外的恶意域名，因此 FANCI 方法不适用于中国科技网真实网络环境中. Liu 等人的方法考虑单个域名的字符特征和解析特征，在开源数据集上验证了方法的高效性，但是没有考虑域名解析特征和注册信息的特征，也没有考虑域名空间的统计特征，在识别跨节点关联的恶意域名攻击方面效果不理想，因此，将该方法应用于本文数据集中在准确率、召回率、精确率、*F*1 值这 4 个指标上达不到较优结果. 异构图分析方法是对校园网流量进行分析，基于特征元路径的注意力和随机游走机制实现恶意域名检测，因为校园网络环境部署节点清晰，通过特征元路径的方式构建异构图模型可以取得比较理想的结果. 但是中国科技网环境比较复杂，且本文数据集在骨干网出口位置分流捕获，因此流量中没有中国科技网内部节点的往来信息，通过构建元路径的方式构建图结构容易出现孤立点.

2)根据 3.2.2 节中对实验数据进行划分，验证多样本训练的情况下模型的有效性，分别将域名样本输入到 FANCI 方法、Liu 等人方法以及异构图分析方法中进行训练和预测，得到模型的性能如图 6 所示.

在训练样本选取 10 000 条的情况下，本文方法取得了 98.29% 的准确率、98.73% 的召回率、99.95% 的精确率和 98.69% 的 *F*1 值，测试时间为 0.027 8 s; FANCI 方法取得了 87.45% 的准确率、83.44% 的召回率、89.94% 的精确率和 86.57% 的 *F*1 值，测试时间为 6.16 s; Liu 等人的方法取得了 90.36% 的准确率、92.54% 的召回率、92.62% 的精确率和 92.58% 的 *F*1 值，测试时间为 0.032 s; 异构图分析方法取得了 86.58% 的准确率、85.71% 的召回率、88.0% 的精确率和 86.84% 的 *F*1 值，测试时间为 0.183 s.

由此可以看出本文方法拥有较高的召回率，对于恶意域名的检测误报率低. 同时检测时间也比较快，在实际系统中可以满足实时检测的需求. 相对于少量样本训练的情况，本文模型召回率得到了进一步提升，说明本文模型学习到了更多的恶意域名特征，从各项指标可以看出本文模型的检测效果远好于其他对比方法. 在样本量变大的情况下，本文模型依旧可以实现较好的检测效果，泛化性能比较高. 此外，本文还利用更大数量的样本构建超图，发现使用 10 000 条样本数据作为训练集已经能够达到最优的效果. 进一步分析，由于本文使用了域名语料库训练出一个良好的嵌入，然后决策树构建的超边又可以学习到域名之间的关联特征，把域名静态特征相近的域名聚合到一个超边里面，并使用超图的卷积运算融合了超边中域名的特征，使得域名之间以及域名中每个字符串的特征得到强化，更能体现出恶意域名通用的特征，而不仅仅是每个域名的特征. FANCI 方法只使用了域名本身字符串提取的特征进行分类，需要较大的数据集训练，模型性能一般；而使用域名构建异构图的方法由于缺少足够的样本，无法学习到域名通用的特征.

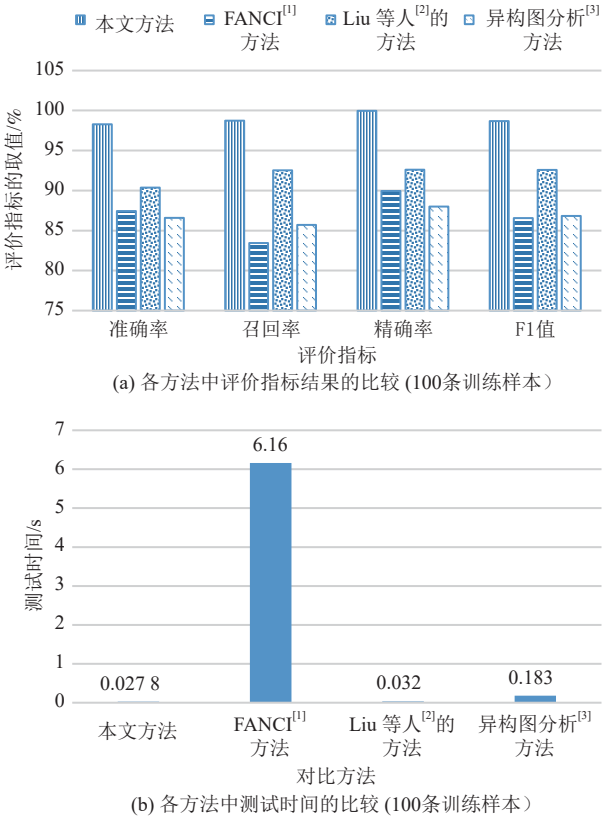

Fig. 6 Comparison of evaluating indicator results and testing time for each method (100 samples)

图 6 各方法中评价指标结果和测试时间的比较 (100 条训练样本)

3.4.2 本文模型在构建及优化过程中的实验结果分析

本文利用域名空间统计特征构建超图结构，为了找到最适合用于对本文域名空间统计特征分类的模型，应用 5 类适合处理统计特征的机器学习算法：随机森林、决策树、朴素贝叶斯、SVM、*K*近邻分别对 39 维域名空间统计特征分类，得到的准确率、召

Table 1 Efficiency Comparison of Five Machine Learning Algorithms for Classification of Statistical Features

表 1 选择 5 种机器学习算法对统计特征分类的效率比较

模型	训练样本条数	准确率/%	召回率/%	精确率/%	*F*1 值/%	测试时间/s
随机森林	100	89.62	89.81	93.93	91.82	2.97
	10 000	95.61	96.31	96.91	96.61	4.19
决策树	100	87.33	86.60	91.63	89.51	0.82
	10 000	93.04	96.85	95.35	96.59	0.81
朴素贝叶斯	100	76.01	73.53	87.52	79.91	1.30
	10 000	78.69	78.23	87.62	82.66	1.24
SVM	100	89.40	92.05	91.65	91.85	3.93
	10 000	93.08	93.10	96.13	94.59	75.42
K 近邻	100	86.67	87.48	91.61	89.50	3.80
	10 000	92.32	94.12	94.06	94.09	5.33

回率、精确率、*F*1 值及测试时间结果，如表 1 所示. 另一方面，形成超图结构阶段包括训练和超图构建 2 个过程，其中训练指的是模型参数训练的过程，超图构建指的是模型训练好之后的预测过程，表 2 给出了 2 种算法构建超图结构的效率对比结果.

Table 2 Efficiency of Malicious Domain Detection Using Hyperedge Set Constructed by Two Models

表 2 利用 2 个模型建超边集实现恶意域名检测效率

模型	训练样本条数	准确率/%	召回率/%	精确率/%	*F*1 值/%	训练时间/s	超图构建时间/s
随机森林	100	96.55	98.09	98.74	97.37	14	30.0
	10 000	98.65	99.55	99.07	98.97	300	54.0
决策树	100	96.33	96.81	99.17	97.18	4	0.1
	10 000	98.29	98.73	99.95	98.69	54	4.7

从表 1 可以看出，随机森林在统计特征分类方面效果最好. 当训练样本为 100 条时，效果较优的模型是 SVM、随机森林和决策树，其中 SVM 的准确率达到 89.4%、召回率达到 92.05%、精确率达到 91.65%、*F*1 值达到 91.85%、测试时间为 3.93 s；随机森林模型的准确率达到 89.62%、召回率达到 89.81%、精确率达到 93.93%、*F*1 值达到 91.82%、测试时间为 2.97 s；决策树模型的准确率达到 87.33%、召回率达到 86.6%、精确率达到 91.63%、*F*1 值达到 89.51%、测试时间为 0.82 s. 当训练样本数量为 10 000 条时，效果最好的 2 个模型是随机森林和决策树，其中随机森林模型的准确率达到 95.61%、召回率达到 96.31%、精确率达到 96.91%、*F*1 值达到 96.61%、测试时间为 4.19 s；决策树模型的准确率达到 93.49%、召回率达到 94.7%、精确率达到 95.23%、*F*1 值达到 94.98%、测试时间为 2.53 s；虽然 SVM 模型在准确率、召回率、精确率以及 *F*1 值方面与随机森林及决策树模型的结果相差不大，但是测试时间高达 75.42 s，在时间方面没有优势. 因此，综合考虑选择多训练样本与少训练样本 2 种情况，加上时间优势，本文选择决策树和随机森林对 39 维域名空间统计特征进行分类形成超图结构和超边集.

从表 2 可以看出随机森林和决策树这 2 种算法在选择 100 条训练样本和 10 000 条训练样本时准确率、召回率、精确率、*F*1 值 4 项指标结果相差不大，但是决策树的训练时间及超图构建时间优势很明显，100 条训练样本情况下，随机森林的训练需要 14 s，构建超图需要 30 s；而决策树的训练需要 4 s，构建超图需要 0.1 s. 10 000 条训练样本情况下，随机森林训练需要 300 s，构建超图需要 54 s；而决策树训练需要 54 s，构建超图需要 4.7 s. 时间差别比较大的原因是使用随机森林构建超边，运算量过于庞大，对超图的构建时间造成较大影响，因此在经过权衡之后选择了使用决策树的方法构建模型中的超图.

本文同时做了消融实验的分析，分别将超图表示特征和字符嵌入特征输入到超图神经网络分类器中去训练并实现恶意域名检测，检验这 2 类特征单独使用在恶意域名检测中的效果. 本文使用了全部的 28 万条域名样本数据，选定其中 70% 作为训练数据，其余 30% 作为测试数据，结果如表 3 所示.

Table 3 Analysis of Ablation Study on Hypergraph Representation Feature and Character Embedding Feature

表 3 超图表示特征和字符嵌入特征的消融实验分析 %

使用的方法	准确率	召回率	精确率	*F*1 值
仅使用字符嵌入特征训练分类器	91.98	89.29	98.16	93.48
仅使用超图表示特征训练分类器	81.13	84.21	86.09	85.14
使用超图表示和字符嵌入特征分类器（本文）	99.19	99.03	99.96	98.79

从表 3 可以看到，仅仅将字符嵌入特征输入到检测模型中时，各个指标比本文将 2 类特征增强变换后输入检测模型的效果有所下降，但是依旧有较高的精确率，说明字符嵌入可以有效地学习到恶意域名具有区分度的特征. 同样，仅将域名空间统计特征构建的超图表示矩阵输入到检测模型中去训练，在各个指标上的效果也不是很理想. 经分析发现域名空间统计特征虽然能够表示域名之间的关联关系，但是并没有学习到字符间隐含的深层特征，这样无法很好地区分字符之间的高阶距离，使得特征在融

合之后也无法得到有效地区分. 因此, 本文利用域名空间统计特征构建超图, 利用超图结构中的超边将域名之间的空间关系呈现出来, 再将表示域名空间关系的超图表示矩阵作用于域名字符嵌入特征以对字符特征进行增强, 这样既保留了域间的空间特征, 也考虑到了域内的字符特征, 使得域名时空特征进一步强化, 最终在各项指标上的结果比较均衡.

3.4.3 模型的鲁棒性分析

在实际的网络流量分析过程中, 可能会出现由于人工打标签不准确而误判的情况, 本文模型能很好地应对该问题, 并针对该情况做了充分的实验分析. 同样选择 10 000 个域名流量数据作为样本, 随机更改训练集中一部分数据的标签为错误标签以产生噪声, 然后将这些带有噪声的样本输入到超图学习模型中去训练, 得到的结果如表 4 所示.

Table 4 Efficiency Analysis of Our Proposed Method Under Labels with Different Error Ratios

表 4 本文所提方法在不同错误比率标签情况下的效率分析 %

标签错误率	准确率	召回率	精确率	*F*1 值
0	98.29	98.73	99.95	98.69
10	94.75	96.36	98.10	95.97
20	93.70	95.39	97.10	95.16
30	93.40	97.95	96.97	94.95
40	92.23	97.94	96.85	93.96
50	63.13	85.04	79.97	75.08

实验中, 本文分别制造 10%, 20%, 30%, 40%, 50% 的错误标签, 并分析这 5 种情况下模型的准确率、召回率、精确率以及 *F*1 值变化情况. 从表 4 可以看出, 在破坏少量样本标签之后精确率和准确率都有所降低, 存在对于良性域名的误报, 但是依旧具有较高的召回率. 当样本破坏量达到一定大小的时候, 即当噪声样本达到 50% 左右时, 在判断决策树叶子节点中错误样本数量占了多数后, 决策树会把更多的不相干的域名划分到一个超边中, 导致了超边中存在不相关域名的特征融合, 使得模型的精确率比没有噪声时的数据降低了 35 个百分点, 召回率降低了 13 个百分点, 精确率降低了 20 个百分点, *F*1 值降低了 23.61 个百分点, 即便是具有错误标签的样本数量达到一半, 本文模型的召回率仍然保持较高.

对于真实环境下的恶意域名检测来说, 尽可能希望将全部的恶意域名识别出来, 因此召回率是本文重点关注的一个关键指标. 所以本文模型可以很好地处理由于人工等原因在一定程度上错误标记的场景, 在少量错误标记的情况下依旧可以检测出几乎全部的恶意域名.

3.4.4 复杂度分析

在本文方案中, 超图神经网络训练过程花费的时间相对来说是最多的, 因此, 仅对图神经网络的复杂度进行分析. 在超图神经网络中, 频谱卷积需要使用拉普拉斯矩阵的特征向量, 然而对拉普拉斯矩阵做特征分解时要用到傅里叶变换, 傅里叶变换的计算代价为 $O(n^2)$. 另外分析了本文模型各部分的实际运行时间, 模型字符嵌入的时间约需要 5 min, 10 000 个样本构建超图结构需要 4.7 s, 训练 200 个遍历需要 54 s, 100 个样本构建超图结构需要 0.1 s, 训练 200 个遍历仅需要 4 s. 训练的复杂度及时间消耗主要来源于训练字符嵌入模块的时间, 但是本文训练的字符嵌入只需训练 1 次就可以供之后每次训练超图时使用.

4 总 结

域名系统(DNS)在提供正常解析服务的同时也成为了各种网络非法活动的主要攻击对象, 当前研究主要集中在基于统计特征的方法和基于关联关系的方法上. 针对相关方法无法准确呈现域间全局高阶关系的问题, 提出基于嵌入式特征超图学习的恶意域名检测方法. 首先利用决策树倒数第 2 层节点的输出结果作为先验条件形成超边, 将域名流量之间的多阶关联关系快速清晰地表示出来; 其次构建基于增强嵌入特征超图学习的恶意域名检测模型, 并基于域名空间统计特征和域名字符嵌入编码特征挖掘域名隐藏高阶关系; 最后结合中国科技网连续 7 个月的真实的 DNS 流量数据, 对本文方法的有效性和可行性进行分析与评估, 实验验证本文方法能够快速高效地检测出恶意域名.

此外, 还发现超图构建的速率受到输入样本数量的影响, 下一步将着重解决超图结构优化的问题, 打破超图网络现有的固定结构, 根据样本特点构建大规模样本自适应超图分类方法.

作者贡献声明: 魏金侠提出算法思路、调研文献、撰写论文以及完成论文修订工作; 龙春负责文献指标及论文修改; 付豪负责实验分析及论文修订; 宫良一和赵静负责论文修改; 万巍负责数据分析; 黄潘提供实验数据.

参 考 文 献

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

[2] Liu Zhenyan, Zeng Yifei, Zhang Pengfei, et al. An imbalanced malicious domains detection method based on passive DNS traffic analysis[J]. Security and Communication Networks, 2018, 2018(4): 1−7

[3] Sun Xiaoqing, Wang Zhiliang, Yang Jiahai, et al. Deepdom: Malicious domain detection with scalable and heterogeneous graph convolutional networks[J]. Computers & Security, 2020, 99(4): 102057

[4] Hou Y, Chang Yimeng, Chen T, et al. Malicious Web content detection by machine learning[J]. Expert Systems with Applications, 2010, 37(1): 55−60

[5] Rieck K, Trinius P, Willems C, et al. Automatic analysis of malware behavior using machine learning[J]. Journal of Computer Security, 2011, 19(4): 639–668

[6] Van T, Giang N. A method for detecting DGA botnet based on semantic and cluster analysis[C]//Proc of the 7th Symp on Information and Communication Technology. New York: ACM, 2016: 272–277

[7] Zang Xiaodong, Gong Jian, Hu Xiaoyan, et al. Malicious domain name detection based on AGD[J]. Journal of Communications, 2018, 39 (7): 15−25

[8] 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

[9] 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. Berkeley, CA: USENIX Association, 2012: 491−506

[10] Chin T, Xiong Kaiqi, Hu Chengbin, et al. A machine learning framework for studying domain generation algorithm (DGA)-based malware[C]//Proc of the 14th Int Conf on Security and Privacy in Communication Systems. Berlin: Springer, 2018: 433−448

[11] Wang Qing, Li Linyu, Jiang Bo, et al. Malicious domain detection based on k-means and smote[C]//Proc of the 20th Int Conf on Computational Science. Berlin: Springer, 2020: 468−481

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

[13] Selvi J, Rodríguez R J, Soria-Olivas E. Detection of algorithmically generated malicious domain names using masked n-grams[J]. Expert Systems with Applications, 2019, 124(15): 156−163

[14] Anderson H S, Woodbridge J, Filar B. Deepdga: Adversarially-tuned domain generation and detection[C]//Proc of the 2016 ACM Workshop on Artificial Intelligence and Security. New York: ACM, 2016: 13−21

[15] Ren Fangli, Jiang Zhengwei, Wang Xuren, et al. A DGA domain names detection modeling method based on integrating an attention mechanism and deep neural network[J]. Cybersecurity, 2020, 3(1): 1−13

[16] Ravi V, Alazab M, Srinivasan S, et al. Adversarial defense: DGA-based botnets and DNS homographs detection through integrated deep learning[J]. IEEE Transactions on Engineering Management, 2021, 70(1): 249−266

[17] Opara C, Wei Bo, Chen Yingke. Htmlphish: Enabling phishing Web page detection by applying deep learning techniques on html analysis[C]//Proc of the 2020 Int Joint Conf on Neural Networks. Piscataway, NJ: IEEE, 2020: 6906−6913

[18] Yuan Jianting, Liu Yipeng, Yu Long. A novel approach for malicious URL detection based on the joint model[J]. Security and Communication Networks, 2021, 2021(6): 1−12

[19] Peng Chengwei, Yun Xiaochun, Zhang Yongzheng, et al. A malicious domain name detection method based on domain name request adjoint[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)

[20] Zhang Weiwei, Gong Jian, Liu Qian, et al. Lightweight domain name detection algorithm based on morpheme features[J]. Journal of Software, 2016, 27(9): 2348−2364 (in Chinese)
(张维维, 龚俭, 刘茜, 等. 基于词素特征的轻量级域名检测算法[J]. 软件学报, 2016, 27(9): 2348−2364)

[21] Du Peng, Ding Shifei. DGA domain name detection method based on mixed word vector deep learning model[J]. Journal of Computer Research and Development, 2020, 57(2): 433−446 (in Chinese)
(杜鹏, 丁世飞. 基于混合词向量深度学习模型的 DGA 域名检测方法[J]. 计算机研究与发展, 2020, 57(2): 433−446)

[22] Zou Futai, Zhang Siyu, Rao Weixiong, et al. Detecting malware based on DNS graph mining[J]. International Journal of Distributed Sensor Networks, 2015, 2015: 1−12

[23] Zhang Jialong, Saha S, Gu Guofei, et al. Systematic mining of associated server herds for malware campaign discovery[C]//Proc of the 35th IEEE Int Conf on Distributed Computing System. Piscataway, NJ: IEEE, 2015: 630–641

[24] Rahbarinia B, Perdisci R, Antonakakis M. Segugio: Efficient behavior-based tracking of malware-control domains in large ISP networks[C]//Proc of the 45th Annual IEEE/IFIP Int Conf on Dependable Systems and Networks. New York: ACM, 2015: 403–414

[25] Stevanovic M, Pedersen J M, Alessandro D, et al. A method for identifying compromised clients based on DNS traffic analysis[J]. International Journal of Information Security, 2017, 16(2): 115−132

[26] Peng Chengwei, Yun Xiaochun, Zhang Yongzheng, et al. Discovering malicious domains through alias-canonical graph[C]//Proc of the 2017 IEEE Trustcom/BigDataSE/ICESS. Piscataway, NJ: IEEE, 2017: 225–232

[27] 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. Berkeley, CA: USENIX Association, 2019: 399–412

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.

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.

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

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

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.

Zhao Jing, born in 1987. PhD, senior engineer, master supervisor. Her main research interests include artificial intelligence-based network attack detection and security log analysis.

赵　静, 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攻击检测、安全日志分析.

Wan Wei, born in 1982. PhD, senior engineer, master supervisor. Member of CCF. His main research interests include network unknown attack detection, malicious domain name detection, network traffic analysis, and machine learning.

万　巍, 1982 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未知攻击检测、恶意域名检测、网络流量分析、机器学习.

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. 本科, 工程师. 主要研究方向为 Web 攻击检测、渗透测试和恶意域名分析.