李成宁, 孙连山. 基于注册信息与网页内容特征的钓鱼网页检测模型[J]. 智能计算机与应用,2025,15(6):90-96. DOI:10.20169/j.issn.2095-2163.250613

基于注册信息与网页内容特征的钓鱼网页检测模型

李成宁, 孙连山
(陕西科技大学 电子信息与人工智能学院, 西安 710021)

摘 要: 网络钓鱼攻击者在部署钓鱼网站时,至少需要设计一个与攻击目标品牌内容相似的网页,并通过 url 的方式诱使用户点击。故钓鱼攻击者在一段时间内批量部署的钓鱼网页在注册信息和网页内容上,有一定的关联关系和相似特征。针对该现象,本文以钓鱼 url 注册信息中的域名、注册商、注册时间、解析 NS 服务器和存活时间作为节点基于注册信息的特征,以网页源码中链接的总数量、链接种类数量、外部链接数量、无效链接数量和输入框数量作为节点基于网页内容的特征,以解析 ip、title 和 cmtcc 值作为节点的边关系,提出了一种基于注册信息与网页内容特征的钓鱼网页检测模型 GRIWC。实验结果表明,该方法可通过图神经网络和注意力机制从少量的注册信息和网页内容特征中提取关键信息,用一段时间内已知的钓鱼 url 检测后续出现的钓鱼 url。

关键词: 网络钓鱼检测; 注册信息; 网页内容; 图注意力网络
中图分类号: TP393.08;TP18 **文献标志码**: A **文章编号**: 2095-2163(2025)06-0090-07

Phishing webpage detection model based on registration information and Web content features

LI Chengning, SUN Lianshan
(School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi′an 710021, China)

Abstract: Phishing attackers, when deploying phishing websites, typically design a webpage that resembles the content of the targeted brand and entices users to click through url manipulation. Consequently, phishing webpages deployed in batches over a period of time exhibit certain correlations and similar features in registration information and webpage content. Addressing this phenomenon, this paper introduces a phishing webpage detection model, GRIWC, based on features derived from registration information, including domain, registrar, registration time, resolved NS servers, and survival time, and features derived from webpage content, including the total number of links, the number of link types, the number of external links, the number of invalid links, and the number of input fields. Edge relationships between nodes are established based on resolved ip, title, and cmtcc values. Experimental results demonstrate that the proposed method, leveraging graph neural networks and attention mechanisms, effectively extracts critical information from a small set of registration information and webpage content features. The model is capable of detecting subsequent phishing URLs based on known phishing urls over a period of time.

Key words: phishing detection; registration information; webpage content; graph attention network

0 引 言

网络钓鱼攻击作为一种严重的网络犯罪行为,通过误导用户进入伪造的钓鱼网页,以诱使用户输入账号和密码[1-3]。在近年来的网络安全环境中,网络钓鱼攻击呈现出一定的增长态势。

当前,网络钓鱼检测方法主要分为 2 种:基于品牌保护和基于钓鱼网页共性特征[4-7]。其中,基于品牌保护的方法通常依赖与知名品牌相关的信息,通过比较网页中的品牌信息和受保护品牌列表进行相似度计算。这类方法对于检测零日钓鱼攻击效果显著,但在处理未知品牌或信息被遮挡的情况时表

基金项目: 陕西省自然科学基础研究计划 (2023-JC-YB-581)。
作者简介: 李成宁(1997—),男,硕士研究生,主要研究方向:网络安全,人工智能。
通信作者: 孙连山(1977—),男,博士,副教授,硕士生导师,主要研究方向:软件安全工程,数据起源安全。Email:sunlianshan@sina.com。
收稿日期: 2023-12-25

现不佳。最近的研究工作中，Abdelnabi 等学者[8]提出了一种通过三重网络和卷积子网络学习网页截图相似性的方法，计算可疑网页与保护品牌网页的相似度用于基于品牌的检测。但网页截图中最能代表品牌信息的内容往往是 logo 图片，故 Lin 等学者[9]通过先给网页截图中边界框分类的方式，尝试定位网页中的 logo，以增强对品牌信息的识别。Liu 等学者[10]在处理 logo 图片相似度计算时，注意到传统方法仅考虑了图像特征而忽略了文本特征。为此，研究提出一种综合考虑图像和文本特征的方法，结合了 logo 图片中的图像和文本信息。此外，又设计了一个网页布局摘要检测器，直接从网页中提取 logo，为品牌信息的准确识别提供了更全面的支持。除了网页 logo，网页的 title 同样可能包含品牌信息，故 Bram 等学者[11]通过借助第三方搜索引擎的方式，同时把网页的 title 文本和 logo 区域截图分别作为文本搜索引擎和反向图片搜索引擎的搜索语料，通过相似度计算将当前网页与返回结果网页进行比对。这一综合考虑不同信息源的方法有助于更全面地了解网页的品牌信息，提高对钓鱼网页的检测准确性。

钓鱼攻击者为了节省攻击成本或为了躲避现有方法的检测，会设计出与合法网页有一些差异特征的钓鱼网页。比如，Guo 等学者[12]发现钓鱼网站上会布置更多的外来链接或空链接，故在检测时将从网页中提取各种链接信息，并使用域名和资源对象构建异构信息网络来检测钓鱼网页。此外，钓鱼 url 的设计风格往往与合法 url 不同，通过 url 的字符特征可以快速检测钓鱼 url。但钓鱼攻击者可以通过设计合法风格的 url 来躲避检测，故 Kim 等学者[13]将每个 url 看作一个句子，通过语法和标点符号将其分割为子字符串(单词)，构建了一个异构网络。应用定制的信念传播算法，实现了对钓鱼 url 的检测。

在 2 类方法中，前者虽然能够有效检测零日钓鱼攻击，但对于未知品牌或品牌信息被遮挡的情况有所不足。后者尽管能够检测各类品牌的钓鱼网页，但仍受制于钓鱼攻击者通过微调网页内容或基础设施来规避检测的问题。可见，当前网络钓鱼检测方法的局限性主要体现在钓鱼攻击者躲避检测的成本较低。

针对当前检测方法的局限性，本文关注网络钓鱼攻击者在部署钓鱼网站时所面临的有限资源问题，即攻击者需要设计一个与目标品牌内容相似的网页，并选择一个 url 作为部署的基础设施。因此，本文提出了一种基于注册信息与网页内容特征的钓鱼网页检测模型 GRIWC (GAT - based with Registration Information and Webpage Content phish detection)。实验结果表明，GRIWC 模型能通过关联已知钓鱼 url 之间有限的注册信息和网页内容资源，检测出新的钓鱼 url。

1　模型设计与实现

1.1　模型框架

网络钓鱼组织发动的攻击活动通常具有一定的持续时间，故本模型首先采集了在一个连续时间段内的钓鱼 url 作为训练集，并将最新发现的钓鱼 url 用作测试集。整体框架如图 1 所示，在数据预处理阶段，获取这些 url 的域名注册信息、网页源码和截图，并从中提取特征。这些特征用于在 url 关系图构建阶段生成节点的特征向量，并建立边关联。接着，将构建好的 url 关系图输入 GAT 图注意力网络，通过有监督训练，让模型学习良性网页和钓鱼网页在注册信息和网页内容层面的特征差异。最后，实现分类功能得到疑似钓鱼的 url。

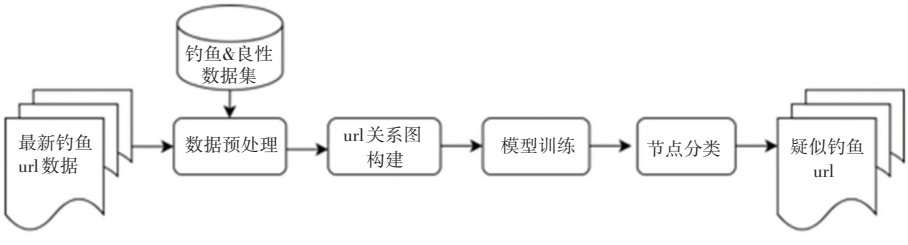


图 1　模型架构

Fig. 1　Architecture of the model

1.2　数据预处理与特征选取

url 本身仅仅包含其文本特征，故构建 url 关系图需要获取 url 的注册信息、网页源码和网页截图。钓鱼 url 中的域名通常在注册信息上表现为相同注册

商、相近注册时间、相同或匿名注册人、存活时间短、字符相似、解析 NS 服务器和 IP 存在重合等特征。例如,批量注册的钓鱼域名通常会存在多个类似的域名,以欺骗用户访问假冒网站。在新的注册隐私政策下,注册人等信息无法再被获取,故通过 whois 查询获取域名的注册商、注册时间、最大存活时间、解析 NS 服务器内容作为节点在注册信息层面的特征。

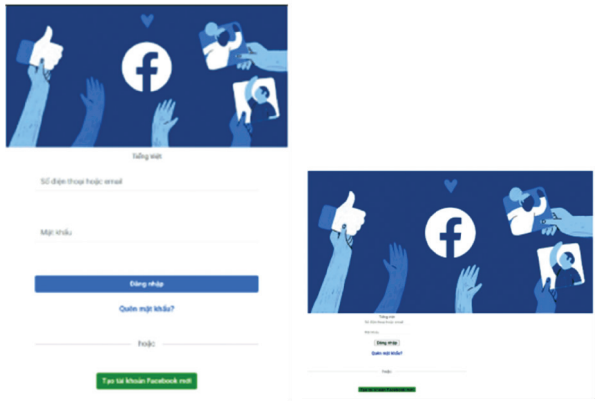
网页中可点击链接通常可跳转至其他页面,但钓鱼攻击者为降低成本,会将这些链接指向原网站的某些链接,甚至置为无效链接。而登录界面往往包含较少的可点击链接,可能影响链接的数量关系在节点特征中的重要性。登录界面的输入框至少成对出现,分别用于输入账号和密码。因此,获取当前网页中链接的总数量、链接种类数量、外部链接数量、无效链接数量和输入框数量共同作为节点在源码层面的特征。通过双向 LSTM 自编码器提取文本类型特征的字符特征,每个节点生成一个 199 维的特征向量。具体节点特征选取结果,见表 1。

表 1 节点特征选取结果表

Table 1 Comparison table of node feature selection

特征	说明
域名	文本类型,生成 64 维的特征向量
注册商	文本类型,生成 64 维的特征向量
解析 NS 服务器	文本类型,生成 64 维的特征向量
域名注册时间	整型类型,注册时间的时间戳,处理为连续型 1 维特征
存活时间	整型类型,到期时间与注册时间的时间戳之差,处理为连续型 1 维特征
链接总数量	整型类型,处理为连续型 1 维特征
链接种类数	整型类型,处理为连续型 1 维特征
外部链接数量	整型类型,处理为连续型 1 维特征
无效链接数量	整型类型,处理为连续型 1 维特征
输入框数量	整型类型,处理为连续型 1 维特征

为了避免用户怀疑,钓鱼网页在制作时不得不在最引人注目的网页标题(title)和内容上体现网页的品牌信息,这导致钓鱼攻击者用以制作 title 的文本资源和网页内容的图像资源是有限的。通过长时间对最新钓鱼网页截图的观察,钓鱼攻击者通常会微调原网页内容,以躲避品牌保护等基于图像相似度的钓鱼检测。这种微调包括对原网页内容进行局部放缩、移动、拆分以及修改部分文字使得与原网页相似度降低的同时,又让用户难以察觉。2 个经过微调的 Facebook 钓鱼网页如图 2 所示。钓鱼攻击者通过微调网页内容使得 Facebook 的 logo 图标不易被检测的同时,不影响用户识别。即使如此,仍然可以通过网页的主题颜色迅速区分属于同一品牌的网页。因此,本文设计了一个定制的主题颜色种类衡量值(customized metric for theme color categories, *cmtcc*),宏观地给每一个网页截图计算出一个 *cmtcc* 值。从而忽略网页截图中的细节特征,快速用一个数字来粗粒度地标记一个网页截图在主题颜色上的特征。*cmtcc* 值可以把所有网页截图分成 2^64 种不同的主题色组合,在构建 url 关系图时可以把 *cmtcc* 值相同的节点关联起来。具体而言, *cmtcc* 值的计算方式如下:



(a) 钓鱼网页 a (b) 钓鱼网页 b

图 2 2 个经过微调的 Facebook 钓鱼网页

Fig. 2 Two fine-tuned Facebook phishing webpages

(1)将每个 *RGB* 颜色表示为(R, G, B),其中 R、G、B 的大小分别代表红色、绿色和蓝色通道的强度值,范围从 0 到 255。将每个通道的大小分为 4 个大小均为 64 的连续空间,将一个三通道的 *RGB* 值映射到这个 4×4×4 的空间里,以此把所有颜色分为 64 类。

(2)获取网页截图中所有 *RGB* 值并去重,再将这

些去重后的 *RGB* 值分别映射到这个大小为 64 的空间里，得到当前网页截图中包含的颜色种类数量。

(3)用一个 64 位大小的整数作为位图，存储当前网页截图的 *cmtcc* 值，即如果包含相应颜色就把位图中对应比特位置为 1，否则置为 0。

1.3　url 关系图构建

由于部分钓鱼 url 之间在注册信息和网页内容上具有很强的关联性并存在重合关系，如 url 的解析 ip、title 和 *cmtcc* 值。在建立 url 关系图时，把所有解析 ip 相同的 url 节点关联起来的同时，把网页截图中 *cmtcc* 值或 title 相同的 url 节点也关联起来，能解决 url 中域名解析 ip 变化导致边关联无法建立的问题。定义 url 关系无向图 $G=(V, E)$，将 url 集合记为 $U=\{u_1, u_2, u_3, \cdots, u_n\}$，其对应的节点集合 A 为 $V=\{v_1, v_2, v_3, \cdots, v_n\}$，节点 v_i 对应 u_i，Qi 为节点 i 的解析 ip、title 和 ntc 值组成的集合。为了构建 url 关系图，制定以下规则：若存在 $Q_i \cup Q_j \neq \phi$，即 url_1 和 url_2 在注册信息和网页内容上存在重合关系，则 v_i、v_j 之间存在边，边权重 $e_{ij}=|Q_i \cap Q_j|$。通过以上 2 种规则，可构建 url 基于注册信息和网页内容关系的网络图，并在完成构图后对边权重进行归一化。url 关系图构建示意如图 3 所示。

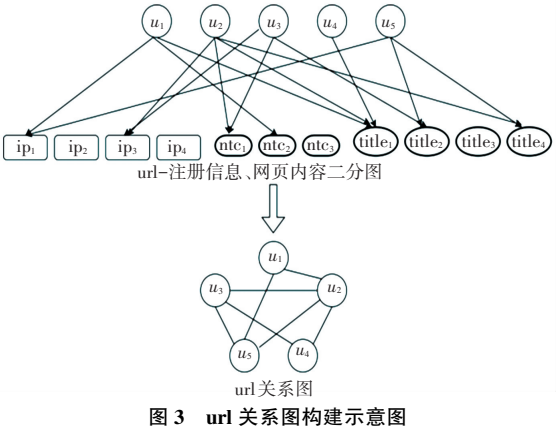


图 3　url 关系图构建示意图

Fig. 3　Diagram of url relationship graph construction

1.4　模型训练与分类

在完成 url 关系图的构建后，可从中获得节点特征矩阵 $\boldsymbol{X}$、稀疏格式的邻接矩阵 $\boldsymbol{A}$、标签向量 $\boldsymbol{Y}$ 作为输入数据，其中图的节点数为 n，边数为 e，特征维度为 m。邻接矩阵 $\boldsymbol{A}$ 的大小为 $2\times 2e$，节点特征矩阵 $\boldsymbol{X}$ 的大小为 $n\times m$，标签向量 $\boldsymbol{Y}$ 的大小为 $1\times n$。

将输入数据分成训练集、验证集和测试集，在训练阶段使用掩码屏蔽测试集的节点，用带标签的钓鱼 url 和良性 url 对图进行训练，此后再取消屏蔽，让 GAT 对测试数据进行节点分类，并获取模型的分类结果。

邻接矩阵 $\boldsymbol{A}$ 和特征矩阵 $\boldsymbol{X}$ 分别被传递给不同的输入层，特征矩阵 $\boldsymbol{X}$ 在经过第 1 个 Dropout 层后和邻接矩阵 $\boldsymbol{A}$ 一起进入第 1 个图注意力层，通过 LeakyReLU 激活函数获得输出，再经过第 2 个 Dropout 层后进入第 2 个图注意力层，最后通过 Softmax 激活函数获得分类结果。

2　实验与分析

实验运行在 NVIDIA TITAN Xp GPU，开发环境为 Python 3.7.11、TensorFlow 2.5.0、Keras 2.4.3、PyTorch 1.13.0、CUDA 10.1、cuDNN 7.6。

2.1　数据集

为了确保本文提出方法能够有效检测最新的网络钓鱼攻击，在 2023 年 5 月至 7 月期间从开源网络钓鱼信息源(如 OpenPhish [14-15]、PhishTank[16] 和 PhishStats [17-18])中获取了最新的钓鱼 url 作为钓鱼 url 数据集。良性 url 数据集来自 DMOZ 数据库[19]，该数据库包含 90 种语言的 3 861 202 个合法网页。与在网络钓鱼研究中常用的其他来源(例如 Alexa 前 100 名网页)不同，DMOZ 数据库包含了并不流行的网页。因为，包含较少人知晓的网站也是重要的，以确保该方法可以推广到真实的场景中。

研究实时爬取了这些 url 对应的注册信息、网页源码和网页截图以作为实验的数据集。移除了那些注册信息不完整、网页无法访问等无效的 url 后，构建了一个包含 10 000 个条目的 url 数据集，其中钓鱼 url 和良性 url 各有 5 000 个。该数据集均经过手动验证，以确保数据集的准确标注以及有效性。使用最新发现的 500 个钓鱼 url 和 500 个随机抽取的良性 url 作为测试集，其余 url 作为训练集；训练集和验证集的比例为 8∶2。实验数据摘要见表 2。

表 2　实验数据摘要表

Table 2　Summary table of experimental data

数据项	数值
节点数量	10 000
边数量	119 526
节点特征维度	199
训练集节点数	7 200
验证集节点数	1 800
测试集节点数	1 000

2.2　评价指标

实验使用准确率(*Accuracy*)、精确率(*Precision*)、召回率(*Recall*)、*F*1 值(*F*1 - *Score*)和

误报率(*False Positive Rate*,*FPR*)来评价检测模型准确性。评价指标计算公式如下:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

$$FPR = \frac{FP}{FP + TN} \tag{5}$$

其中,*TP* 表示实际是钓鱼样本,且预测为恶意样本的数量;*TN* 表示实际是良性样本,且预测为良性样本的数量;*FP* 表示实际是良性样本,但被错误预测为钓鱼样本的数量;*FN* 表示实际是钓鱼样本,但被错误预测为良性样本的数量。*Accuracy* 表示被正确分类的样本数量占总样本数量的百分比;*Precision* 表示预测为良性的样本中,确实为良性的样本所占的百分比;*Recall* 表示实际为良性的样本中,被预测为良性的样本所占的百分比;*F1 - Score* 是对精确率和召回率的调和平均值;*FPR* 表示实际为良性的样本中,被错误预测为钓鱼的样本所占的百分比。

2.3 url 关系图构建效果分析

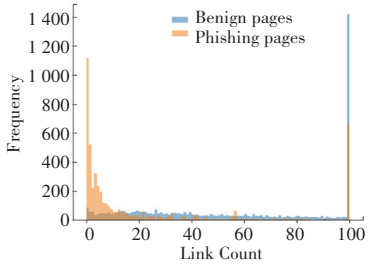
为了将使用相同注册信息、网页源码和网页截图资源的钓鱼网页建立关联,本文提出将 ip、title 和 *cmtcc* 值相同的 url 节点建立边关联,表 3 展示了不同边类型建立边关联的数量效果。结果表明,使用 title 和 *cmtcc* 值能够在 ip 不同的 url 之间有效建立边关联,并将边关联数量扩大 2.3 倍。值得注意的是,在本文最新提出的 *cmtcc* 值建立的 27 912 个边中,有 88%的边包含在通过 ip 和 title 建立边关联里,这表明 *cmtcc* 值建立的边关联是可靠的。此外,剩余的 12%边能够让通过 ip 和 title 无法建立的节点间建立关联。

表 3 不同边类型建立的边关联数量表

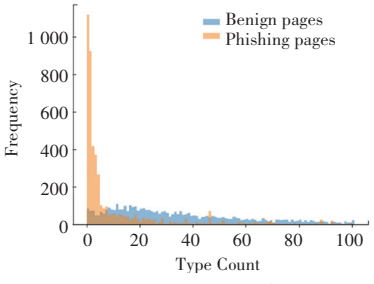
Table 3 Number of edge associations established by different edge types

边关联	边数量
ip	52 396
title	89 100
cmtcc	27 912
ip∪title	116 165
ip∪cmtcc	665 340
ip∪title∪cmtcc	119 526

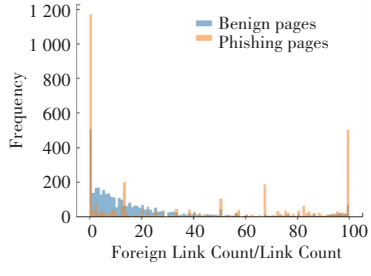
为了验证在最新的钓鱼数据中,使用网页源码中链接的数量特征作为节点特征的可行性,研究中分别对 2.1 节中的 5 000 个钓鱼网页和 5 000 个良性网页源码中的 4 种链接数量进行了统计,并将统计结果以直方图的形式呈现。实验结果如图 4 所示。由图 4(a)、(b)可知,钓鱼网页中的链接总数量和链接种类数量往往更少,甚至有大量钓鱼网页中没有任何链接。因此,在统计外部链接和无效链接占网页源码总链接数量的百分比时,剔除了那些没有任何链接的网页。由图 4(c)、(d)可知,钓鱼网页中的外部链接以及无效链接的占比更高。综上所述,使用链接的 4 个数量特征可以作为区分钓鱼网页与合法网页的重要特征。



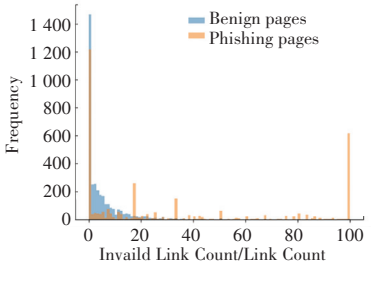
(a) 链接总数量对比直方图



(b) 链接种类数量对比直方图



(c) 外部链接数量占链接总数量对比直方图



(d) 无效链接数量占链接总数量对比直方图

图 4 良性网页和钓鱼网页链接数量对比

Fig. 4 Comparison of the number of links between benign and phishing webpages

2.4　实验细节与结果

经过 100 个 *epoch* 后，GRIWC 模型训练损失与准确率曲线如图 5 所示。训练结束后，模型在训练集上和测试集上的结果见表 4。

表 4　GRIWC 模型的训练结果

Table 4　Training results of GRIWC model

数据集	*Accuracy*	*Precision*	*Recall*	*F*1 − *Score*	*FPR*
训练集	0.950 6	0.960 9	0.939 0	0.950 0	0.382
测试集	0.963 0	0.953 0	0.974 0	0.963 4	0.480

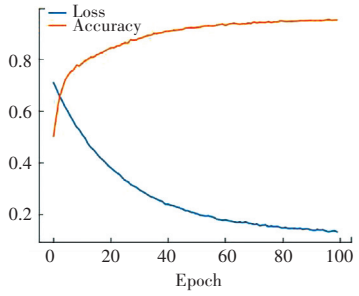


图 5　GRIWC 模型训练损失与准确率曲线

Fig. 5　Training loss and accuracy of GRIWC model

表 5 展示了不同节点特征选取和边关联的组合对模型性能的影响。实验结果表明，仅使用注册信息作为节点特征，解析 ip 作为边关联时的效果最差。当使用注册信息、链接数量和输入框数量作为节点特征，使用 ip、title 和 *cmtcc* 值作为边关联时，模型的效果最好。把网页源码中的链接数量作为节点的组合特征时，能有效提高检测的准确率，并且登录框的数量可以作为链接数量的补充特征来进一步提高检测的准确率。

表 5　不同节点特征选取和边关联对模型性能的影响

Table 5　Effects of varied node feature selection and edge associations on model performance

节点特征			边关联			评价指标				
注册信息	链接数量	输入框数量	解析 ip	title	*cmtcc* 值	*Accuracy*	*Precision*	*Recall*	*F*1 − *Score*	*FPR*
√	×	×	√	×	×	0.934 0	0.912 5	0.960 0	0.935 7	0.92
√	√	√	√	×	×	0.939 0	0.918 1	0.964 0	0.940 5	0.86
√	√	√	√	√	×	0.960 0	0.954 7	0.970 0	0.962 3	0.46
√	√	√	√	×	√	0.943 0	0.930 1	0.958 0	0.943 8	0.72
√	×	×	√	√	√	0.947 0	0.940 8	0.954 0	0.947 4	0.60
√	√	×	√	√	√	0.956 0	0.952 4	0.960 0	0.956 2	0.48
√	√	√	√	√	√	**0.963 0**	**0.953 0**	**0.974 0**	**0.963 4**	**0.48**

在 2.3 节中，表 3 仅证明了 title 和 *cmtcc* 值均能在不同钓鱼网页之间有效建立边关联，且 title 建立的边关联数量更多。本节的表 5 证明了 title 和 *cmtcc* 值分别与 ip 组合作为边关联时，都能提高模型准确率，且建立边关联数量更多的 title 表现优于 *cmtcc* 值。此外，ip、title 和 *cmtcc* 值共同组合作为边关联的效果最好，这表明 *cmtcc* 值能够在 ip 和 title 都不同的网页之间仍然建立边关联。同时，证明了本文新提出的 *cmtcc* 值作为边关联是有效的，即在钓鱼攻击者批量制作钓鱼网页之间，存在主题颜色种类数量重合的现象。

2.5　对比实验

本节使用 2.1 节中介绍的有监督数据集分别对 AI−Alyan 等学者[20]、胡忠义等学者[21]以及 Yang 等学者[22]提出的基于 url 特征的钓鱼网站识别模型进行训练与测试，并采用本文的数据集进行性能评估。实验结果见表 6。

表 6　不同模型的检测结果对比

Table 6　Comparison of detection results of different models

模型	*Accuracy*	*Recall*	*F*1 − *Score*
AI−Alyan 等学者[20]	0.91	0.86	0.88
胡忠义等学者[21]	0.95	0.97	0.96
Yang 等学者[22]	0.87	0.90	0.89
GRIWC	0.96	0.97	0.96

从表 6 可以看出，GRIWC 模型的准确率、召回率和 *F*1 值相较于以往仅针对 url 特征的钓鱼检测模型，有着显著提升。尽管本文模型与胡忠义等学者[21]模型的结果相近，但从钓鱼攻击者躲避检测的成本上来看，钓鱼攻击只需修改 url 就可以躲避胡

忠义等学者[21]模型的检测;而躲避 GRIWC 模型则不仅需要更多的 url 基础设施来部署钓鱼网页,还需要修改网页源码中的各种链接以及网页的图像特征,这将大大提高钓鱼攻击者制作钓鱼网站的成本。

3 结束语

本文把注册信息、网页源码中的 4 种链接数量和输入框数量作为节点特征,解析 ip、title 和新提出的 *cmtcc* 值作为节点之间的边关联建立 url 关系图,通过图方法、神经网络和注意力机制提取有限特征中的隐藏信息,提出了一种基于注册信息与网页内容特征的钓鱼网页检测模型 GRIWC。模型能够通过一段时间内已经出现的钓鱼网页,来检测新出现的钓鱼网页,并在测试集上获得了 96.30%的准确率。但模型依赖于钓鱼攻击者在批量制作钓鱼网页时,持有的基础设施资源和网页内容资源的有限性来在不同网页直接建立关联。当钓鱼攻击者通过其他方式使得这些资源发生变化,就可能会导致检测准确率下降。故考虑在未来设计出一种能够增大钓鱼攻击者躲避钓鱼检测成本的检测模型。

参考文献

[1] APWG. Phishing Activity Trends Report[EB/OL]. (2024-08-21). https://apwg.org/trendsreports/.

[2] LEVI O, HOSSEINI P, DIAB M, et al. Identifying nuances in fake news vs. satire: using seman tic and linguistic cues[J]. arXiv preprint arXiv,1910.01160, 2019.

[3] VELIČKOVIĆ P, CUCURULL G, CASANOVA A, et al. Graph attention networks[J]. arXiv preprint arXiv,1710.10903, 2017.

[4] BULAKH V, GUPTA M. Countering phishing from Brands′ vantage point[C]//Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics (IWSPA′16). New York:ACM, 2016: 17-24.

[5] FENG Jian, ZOU Lianyang, YE Ou, et al. Web2vec: Phishing webpage detection method based on multidimensional features driven by deep learning[J]. IEEE Access, 2020, 8: 221214-221224.

[6] 王琦菲,张大为. 一种基于 GAT 的小样本均衡补偿文本主题分类模型[J]. 智能计算机与应用,2023,13(1):100-103.

[7] 吴松泽. 关于无线网络"钓鱼"问题的研究[J]. 智能计算机与应用,2014,4(6):109-111.

[8] ABDELNABI S, KROMBHOLZ K, FRITZ M. VisualphishNet: Zero-day phishing website detection by visual similarity[J]. arXiv preprint arXiv,1909.00300,2019.

[9] LIN Yun, LIU Ruofan, DIVAKARAN D M, et al. Phishpedia: A hybrid deep learning based approach to visually identify phishing webpages[C]//Proceedings of the 30th USENIX Security Symposium. USENIX, 2021:1-19.

[10] LIU Ruofan, LIN Yun, YANG Xianglin, et al. Inferring phishing intention via Webpage appearance and dynamics: A deep vision based approach[C]//Proceedings of the 31st USENIX Security Symposium. Boston, USA: USENIX, 2022:1-18.

[11] BRAM V D, BURDA P, ALLODI L, et al. Combining text and visual features to improve the identification of cloned webpages for early phishing detection[C]// Proceedings of the 16th International Conference on Availability, Reliability and Security. New York: ACM,2021:1-10.

[12] GUO Bingyang, ZHANG Yunyi, XU Chengxi, et al. HinPhish: An effective phishing detection approach based on heterogeneous information networks[J]. Applied Sciences,2021,11(20): 9733.

[13] KIM T, PARK N, HONG J, et al. Phishing URL detection: A network-based approach robust to evasion[C]//Proceedings of 2022 ACM SIGSAC Conference on Computer and Communications Security. New York:ACM, 2022: 1769-1782.

[14] HUNG L, PHAM Q, SAHOO D, et al. URLNet: Learning a URL representation with deep learning for malicious URL detection[J]. arXiv preprint arXiv,1802.03162 ,2018.

[15] 冯健,邹联扬,乔鱼强,等. 基于主辅特征和深度学习的钓鱼网页检测方法[J]. 计算机工程与设计,2021,42(10):2748-2754.

[16] ALEROUD A, KARABATIS G. Bypassing detection of URLbased phishing attacks using generative adversarial deep neural networks[C]//Proceedings of the Sixth International Workshop on Security and Privacy Analytics. New York:ACM, 2020: 53-60.

[17] NGUYENL A T, TO B L, NGUYEN H K, et al. Detecting phishing web sites: A heuristic URL-based approach[C]// Proceedings of 2013 International Conference on Advanced Technologies for Communications (ATC 2013). Piscataway,NJ: IEEE, 2013:597-602.

[18] VIELKA A V A, GABRIEL A C, REYES E J M D, et al. Do not feed the Phish: Phishing Website detection using URL-based features[C]//Proceedings of the 5th World Symposium on Software Engineering (WSSE′23). New York:ACM,2023:135-141.

[19] SAYAK S R, UNIQUE K, SHIRIN N. Phishing in the free waters: A study of phishing attacks created using free website building services[C]//Proceedings of 2023 ACM on Internet Measurement Conference (IMC′23). New York: ACM, 2023: 268-281.

[20] AI-ALYAN A, AI-AHMADIS. Robust URL phishing detection based on deep learning[J]. KSII Transactions on Internet and Information Systems, 2020, 14(7): 2752-2768.

[21] 胡忠义,张硕果,吴江. 基于 URL 多粒度特征融合的钓鱼网站识别[J]. 数据分析与知识发现,2022,6(11):103-110.

[22] YANG Peng, ZHAO Guangzhen, ZENG Peng. Phishing Website detection based on multidimensional features driven by deep learning[J]. IEEE Access, 2019, 7: 15196-15209.