

APEA：一种恶意URL新型识别方法

张慧菲¹，杨秀璋²，彭国军¹

(1. 武汉大学国家网络安全学院，湖北 武汉 430072；2. 贵州大学贵州省大数据产业发展应用研究院，贵州 贵阳 550025)

摘要：在数字化时代，随着网络攻击手段不断进化，恶意网页识别已成为网络安全领域的一大挑战。传统的卷积神经网络在恶意网页检测中取得了显著成果，但无法捕获长距离依赖关系且语义捕获能力有限。因此，设计一种新型深度学习模型自适应位置感知嵌入与注意力（adaptive positionally-aware embedding and attention, APEA），为恶意网页识别提出研究新思路。APEA模型采用比Transformer模型更细粒度的字符级信息，以增强模型对恶意统一资源定位符（uniform resource locator, URL）文本中细节特征的感知能力；与Transformer模型类似，将输入的嵌入信息与位置编码相结合，以显式注入顺序信息；新增了全局信息共享机制，有效融合局部和全局特征，以提升模型对复杂恶意URL模式的识别能力。设计的自适应动态权重机制，使模型能够根据不同输入灵活调整多头自注意力机制的各个头的权重，从而捕获更多特征。实验结果表明，相较于现有的基于机器学习和深度学习恶意URL检测方法，APEA模型在检测的准确率、精确率、召回率、F1分数上都表现更好。消融实验结果表明，全局信息共享机制和自适应调整的动态权重更新机制对模型性能的提升均有贡献，相比去掉这两个机制的模型，APEA模型各指标均有2.8%左右的提升。

关键词：APEA；恶意URL识别；全局信息；Transformer

中图分类号：TP393

文献标志码：A

DOI:10.11959/j.issn.2096-109x.2025066

APEA:a novel method for identifying malicious URL

ZHANG Huifei¹, YANG Xiuzhang², PENG Guojun¹

1. School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China

2. Guizhou Big Data Academy, Guizhou University, Guiyang 550025, China

Abstract: In the digital era, with the continuous evolution of cyberattack methods, the identification of malicious web pages has become a major challenge in the field of network security. Traditional convolutional neural networks have achieved remarkable results in the detection of malicious web pages, but they are unable to capture long-distance dependencies and have limited semantic capture capabilities. Therefore, a novel deep learning model named APEA was designed and implemented, offering a new research direction for the identification of malicious web pages. APEA model adopted character-level information with a finer granularity than the Transformer to enhance the model's perception of detailed features in malicious URL texts. Similar to the Transformer, input embedding were combined with positional encoding to explicitly incorporate sequential information. Additionally, a global infor-

收稿日期：2025-02-06；修回日期：2025-08-18

通信作者：彭国军，guojpeng@whu.edu.cn

基金项目：国家自然科学基金（62172308，62562012）；贵州省科技计划项目（黔科合基础-zk [2025] 面上686）；贵州省科技重大专项计划（黔科合重大专项字[2024]014）

Foundation Items: The National Natural Science Foundation of China (62172308, 62562012), Guizhou Provincial Basic Research Program (Natural Science) (zk[2025]686), Major Scientific and Technological Special Project of Guizhou Province ([2024]014)

mation sharing mechanism was incorporated to effectively integrate local and global features, thereby improving the model's ability to recognize complex malicious URL patterns. An adaptive dynamic weight update mechanism was designed, enabling the model to flexibly adjust the weights of each head in the multi-head self-attention mechanism according to different input, thus capturing more features. Experimental results showed that compared with existing machine learning and deep learning-based malicious URL detection methods, APEA achieved better performance in terms of detection accuracy, precision, recall, and $F1$ score. Ablation experiment results indicated that both the global information sharing mechanism and the adaptive dynamic weight update mechanism contribute to the improvement of model performance. Compared with the model with these two mechanisms removed, the APEA model achieves an improvement of approximately 2.8% across all metrics..

Keywords: APEA, malicious URL identification, global information, Transformer

0 引言

随着互联网的普及和数字化技术的飞速发展,网络安全问题日益严峻^[1]。恶意网页作为一种常见的网络威胁,不仅严重危害个人用户的隐私安全与财产安全,也可能危及企业或政府的机密数据、网络服务稳定性,乃至国家网络安全^[2]。恶意网页可能通过诱导用户点击虚假链接、植入恶意代码、钓鱼攻击等方式,损害用户的设备或盗取个人敏感信息,甚至成为大规模网络攻击的起点^[3]。因此,恶意网页的检测与防御已成为当今网络安全研究的一个重要方向,亟须一种轻量级的、安全高效且实时性强的恶意网页检测方法,以应对未知威胁并减少对内容访问和复杂环境的依赖。

通过网页展开的恶意行为可以分为三大类:钓鱼网页、篡改网页、恶意软件网页。钓鱼网页是指攻击者制作与正常网页类似的假冒网页诱骗用户提供敏感信息,通常以邮件、短信等方式传播,用户点击链接或者输入个人信息,信息就会被攻击者窃取;篡改网页是指黑客利用网站漏洞替换原始网页内容,用以张贴非法或虚假信息;恶意软件网页是指被用于对用户或计算机系统造成伤害的网页,其中包含恶意软件下载链接或恶意脚本,后者在用户不知情的情况下执行恶意行为。

按照检测特征分类,恶意网页检测主要分为基于静态特征的检测和基于动态特征的检测。其中,静态特征主要包括主机信息、统一资源定位符(uniform resource locator, URL)信息、网页内容等;动态特征主要包括浏览器行为、网页跳转关系、注册表及文件夹变化等。本文调研了上

述多种特征,最终选取URL信息作为恶意网页检测特征的唯一来源。

从收集难度上来讲,URL信息比网页内容和动态特征更容易获取;基于动态特征的检测往往需要结合蜜网和虚拟化技术识别恶意网页^[4],而蜜罐模拟的环境会被部分恶意网页识别为虚拟环境,致使其不执行全部的恶意行为^[5],从而无法获取真实有效的动态特征;从安全性来讲,基于内容和动态特征的检测需要实际访问网页,甚至下载潜在恶意软件,其安全系数低于纯URL信息;从环境依赖的角度来讲,动态特征常依赖特定操作系统或浏览器版本,限制了检测系统的通用性,而URL信息格式相对统一,不存在此问题。

现有的恶意网页检测技术主要包括黑名单、启发式规则、机器学习、交互式行为及深度学习的恶意网页识别。黑名单更新滞后、误报率高^[6];启发式规则依赖人工总结且更新困难,机器学习依赖人工设计特征且易过拟合;交互式行为的方法部署成本高昂,这些限制促使本文采用深度学习方法,利用自动特征提取和强泛化能力提升检测的准确性和鲁棒性。

深度学习在恶意网页检测领域发展迅速,卷积神经网络(convolutional neural network, CNN)是目前恶意URL检测的基石,例如URLNet^[7]、GramBedding^[8]。然而基于CNN模型的恶意URL检测已难以取得明显性能提升。长短期记忆网络(long short-term memory, LSTM)模型能处理长距离依赖,但仍可能出现梯度衰减;传统深度学习无法捕获长距离依赖且语义捕获能力有限。为此,本文提出了新型深度学习模型自适应位

置感知嵌入与注意力 (adaptive positionally-aware embedding and attention, APEA), 一种基于自适应多头自注意力的细粒度恶意网页检测模型。该模型无须人工设计特征, 本文的贡献如下。

1) Transformer^[9]模型训练数据采用的词级, 本文采用更细粒度的字符级, 通过保留字符顺序信息和微小差异, 增强对恶意URL文本中细节特征的感知能力; 与Transformer模型相同的是, 本文也将输入的嵌入信息与位置编码相结合。

2) 本文采用Transformer模型编码层的部分框架, 保留多头自注意力机制; 不同的是本文对多头自注意力机制进行修改, 设计了自适应动态权重更新机制, 使模型能够根据不同输入特征灵活调整多头自注意力机制的各个头的权重, 从而捕获更多特征。

3) 通过引入全局信息共享机制, 本文能够有效地融合局部和全局特征, 提升模型对复杂恶意网页模式的识别能力, 该部分是Transformer模型所不具备的。

4) 本文提出的模型性能相比于现有机器学习 and 传统深度学习方法均有明显提升; 相比于去掉全局信息共享机制和自适应动态权重机制的模型, APEA有2.8%左右的准确率提升, 消融实验显示, 本文提出的自适应动态权重机制和全局信息共享机制对实验结果均有提升。模型平均预测单个URL的时间约为1.26 ms, 模型检测效率高, 可灵活部署于客户端 (浏览器扩展程序) 或服务端 (邮件服务器、Web服务器)。

1 相关工作

1.1 现有恶意网页识别方法

现有恶意网页识别的方法主要包括5类: 基于黑名单的恶意网页识别方法、基于启发式规则的恶意网页识别方法、基于机器学习的恶意网页识别方法、基于交互式主机行为的恶意网页识别方法、基于深度学习的恶意网页识别方法。

1.1.1 基于黑名单的恶意网页识别

黑名单可以细分为URL黑名单、IP地址黑名单以及DNS黑名单。已确定为恶意网页的URL、IP地址、DNS会进入黑名单, 黑名单简单容易实现, 但难以维护最新的黑名单列表, 存在高误报率的问题^[6], 为此, Prakash等^[10]提出

改进方法PhishNet, 他们将URL拆解为多个组件再与黑名单列表进行匹配, 这有助于识别部分未出现的恶意网页, 但对原始黑名单数据还是存在极强的依赖关系。根据Sheng等^[11]的研究, 约有63%的网络钓鱼行为在最初的2 h就结束, 而47%~83%的钓鱼网址在发现12 h后才能录入黑名单, 由此可见, 黑名单更新时效性低的缺点在很大程度上限制了黑名单的实际使用效果^[4]。

1.1.2 基于启发式规则的恶意网页识别

由于恶意网页之间存在相似性, 根据恶意网页的共有特征设计启发式规则, 可用于识别恶意网页。例如CANTINA (character authority names trust information network assessment)^[10]结合启发式方法检测钓鱼网址, 将网页中TF-IDF (term frequency-inverse document frequency) 值高的术语放入搜索引擎中。如果检索到的网页域名和被测网页域名匹配, 则为合法网页; 否则, 为钓鱼网页。

相较于黑名单方法, 启发式规则虽然能检测到部分未见过的恶意网页, 但是它依赖人为总结规则, 存在误报率高、规则更新困难的缺点。

1.1.3 基于机器学习的恶意网页识别

在过去10年里, 研究人员采用了各种机器学习技术进行恶意网页检测^[12-14]。恶意网页检测主要采用有监督的机器学习算法, 主要包括支持向量机 (support vector machine, SVM)、朴素贝叶斯、决策树等。

支持向量机通过寻找一个超平面进行分割来实现分类, 尤其是对于二分类问题。它能在高维空间中对数据边界准确划分^[12], 因而被广泛用于恶意网页和正常网页的检测^[15-16]。

朴素贝叶斯通过特征概率来预测分类, Aggarwal等^[17]通过基于朴素贝叶斯的方法检测推特上的钓鱼推文取得了较好的效果。

决策树是一种以树形数据结构来展示决策规则和分类结果的模型, 通常选择信息增益、信息增益率、Gini指数作为分裂规则, Cuzzocrea等^[18]通过决策树检测网络钓鱼。

尽管基于机器学习的恶意网页识别在实际应用中取得了一定的成功, 但这些方法依赖于人工设计的特征, 特征的选择和提取过程通常需要专

家知识,这使得该方法的扩展性和灵活性受到限制,还可能导致过拟合等问题。

1.1.4 基于交互式主机行为的恶意网页识别

基于交互式主机行为的恶意网页识别技术通过监测和分析用户与网站之间的交互行为(如创建新进程、下载文件、修改注册表内容等),实现高精度恶意网页检测。核心流程分为4步:构建高仿真虚拟环境,实现蜜罐部署;通过自动化脚本触发网页潜在的恶意行为;通过内核级钩子技术或者行为监控工具捕获系统调用、API调用、文件操作等细粒度行为日志;通过规则匹配或者机器学习模型识别异常行为模式。

尽管该技术具备高精度优势,但其实践仍面临多重挑战。首先,开源沙箱(如Cuckoo)支持自动化网页访问与行为监控,但是虚拟机集群的部署周期和动态分析耗时长,难以满足高并发网页检测需求。其次,行为日志的存储与分析需要高昂的计算成本与存储成本。最后,针对可记录交互行为的沙箱,恶意代码逃逸手段日趋成熟:通过检查进程列表(如vmttoolsd.exe)、磁盘型号(如检测虚拟磁盘固定容量特征)或者执行时间差异^[19]等判断是否为沙箱环境,并中止恶意行为。

1.1.5 基于深度学习的恶意网页识别

深度学习在恶意网页识别领域的应用是网络安全研究中的一个活跃分支,使用基于词级的CNN^[20]和文本级的CNN^[21]的方法在文本分类时表现出很好的性能。URLdeepDetect^[22]通过应用语义向量模型和URL分词来分析语义和词汇特征,并使用LSTM和 k 均值聚类对网址进行分类。URLNet^[7]提出了基于字符信息的高级词嵌入技术,这有助于在测试时对未见过的单词进行嵌入,并且能够识别单词的子词级信息。Yu等^[23]提出的M-BERT模型通过增强语义理解能力来捕捉恶意网址的特征,并通过可解释性分析揭示了模型的决策过程,增强了对其分类结果的理解。

恶意网页识别已逐步转向融合最新和传统深度学习模型,本文基于Transformer模型提出新型的恶意网页识别深度学习模型APEA,为恶意网页识别提出新思路。

2 新型恶意网页识别的深度学习框架

本节将详细介绍APEA模型的实现过程,APEA模型框架如图1所示。该框架包括两个主要部分:字符级恶意网页特征提取模块、带全局信息的自适应多头自注意力分类模块。

本文将带标签的URL数据预处理后,输入字符级恶意网页特征提取模块,该模块添加上首尾信息后进行字符级嵌入以捕获字符间语义关系,位置编码信息嵌入捕获字符的相对位置关系,多个嵌入层信息处理后相加得到带有URL特征序列的张量,该张量即为特征嵌入模块的输出。在带全局信息的自适应多头自注意力分类模块中,计算URL字符信息张量均值并传递给全连接层以生成全局上下文信息,特征嵌入的输出加上全局上下文信息后再输入到编码层,编码层包含多头自注意力部分,并融合自适应权重信息以增强注意力效果,紧接着通过两层线性变换和ReLU激活函数实现卷积操作,最后通过Softmax得到URL的分类结果。

2.1 字符级恶意网页特征提取

字符级恶意网页特征提取模块与BERT^[24]和Transformer模型的特征提取层不同,本文采用更细粒度的字符级嵌入,再结合位置编码,这一方式对字符序列复杂的URL能提取更全面的特征信息。

恶意URL常通过细微字符修改实现伪装(如paypal.com->paypal.com、google.com->g00gle.com),这类变形在词级分词中会被视为不同词汇,导致词级模型难以捕捉语义关联。字符级表征能直接识别字符替换等细节,而恶意URL的许多特点也体现在单个字符上。URL的域名、路径、参数等字段通常由特定字符(如/、.=)分隔。字符级嵌入能保留这些结构信息,而词级模型可能因分词策略丢失关键符号。

首先,本文对URL文本数据进行预处理,为适应序列数据处理需求,加入专用控制字符,表示序列开始的<SOS>和序列结束的<EOS>。本文还构建全面的词汇表,涵盖几乎所有可能出现在URL中的字符,如字母、数字以及常见的特殊符号(含有超出词汇表字符的样本则被剔除),URL文本序列按照字符解析为数字(本文构建的

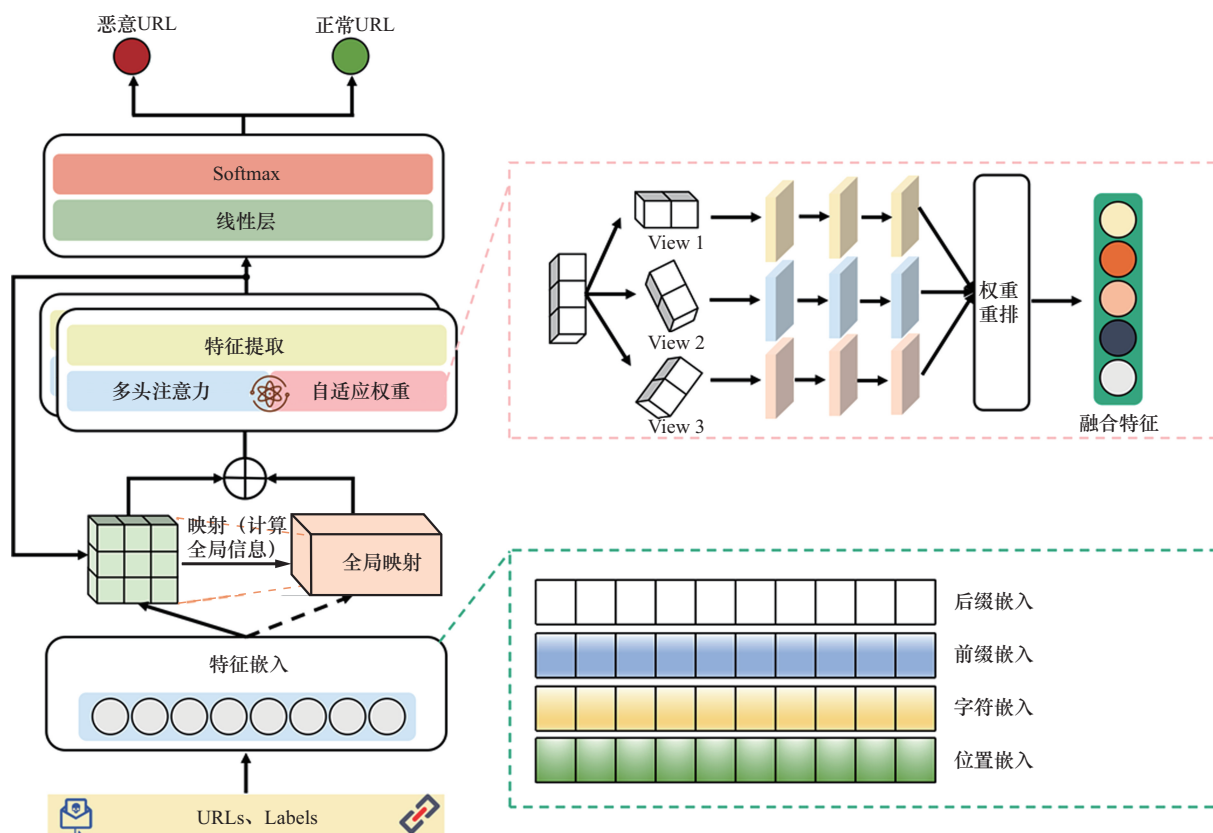


图1 APEA模型框架

词汇表的索引), 由此将URL文本序列转换为整型张量 $\text{index}(c)$ 。

URL的序列以字符索引的形式表示, 形成离散且不包含语义信息的整型张量, 故需要将离散的字符索引张量映射到高维空间, 从而捕获字符间的语义关系。例如, 语义相近的“a”和“A”在嵌入空间会更接近, 而语义不同的字符“a”和“1”则会相距较远。输入包含索引的整型张量 $\text{index}(c)$, 维度为 (B, L) , 其中, B 是批大小 ($B=16$), L 是最长URL序列的长度。嵌入过程的核心是通过查找表为每个字符索引构建嵌入张量, 输出维度为 (B, L, D) 的嵌入张量 e_c , 其中, D 是字符索引嵌入张量维度 ($D=32$), 计算过程见式(1)。

$$e_c = \text{EmbeddingMatrix}[\text{index}(c)] \quad (1)$$

紧接着, 本文采用一种基于正弦和余弦函数的位置编码机制, 此机制为每个字符的嵌入张量增添位置信息, 使模型能够区分相同字符在不同位置的语义, 位置编码部分采用Transformer模型中的位置编码。

$$P(1, \text{pos}, 2i) = \sin\left(\frac{\text{pos}}{10000^{2i/D}}\right) \quad (2)$$

$$P(1, \text{pos}, 2i+1) = \cos\left(\frac{\text{pos}}{10000^{2i/D}}\right)$$

具体地, 创建维度为 $(1, M, D)$ 的位置编码 P , M 默认为 2 500, 对于每个位置索引 pos 和每个维度 i , 位置编码 P 由式(2)计算得到。

P 值随位置周期性变化, 使模型在不同位置识别相似模式。URL的结构信息对于检测恶意URL至关重要, 位置编码可以帮助识别URL中的路径、参数结构等信息。本文将字符级嵌入信息和位置编码信息相加, 字符级嵌入信息维度为 (B, L, D) , 位置编码维度为 $(1, M, D)$, 两者维度并不相同, 因为 $L < M$, 截取前 L 个位置数据, 即 $(1, L, D)$, 再采用广播机制广播到 B 个批量, 即 (B, L, D) , 这时两者维度相同, 两者相加后再进行层标准化, 计算过程见式(3):

$$e'_c = \text{LayerNorm}(e_c + P(\text{pos})) \quad (3)$$

其中, e'_c 为综合嵌入向量, e_c 为字符嵌入张量, $P(\text{pos})$ 为位置编码。

2.2 带全局信息的自适应多头自注意力分类模块

2.2.1 全局信息共享

对特征张量 X 沿序列长度维度计算平均值, 从而提取每个 URL 信息的全局特征表示。具体而言, 输入张量 X 的形状为 (B, L, D) , 对每个样本, 将其所有字符的特征向量进行平均, 得到形状为 $(B, 1, D)$ 的均值张量, 计算过程见式(4), 式中 $b \in [0, B-1]$, $d \in [0, D-1]$ 。

$$\text{avg}(X_{b,0,d}) = \frac{1}{L} \sum_{i=0}^{L-1} X_{b,i,d} \quad (4)$$

本文将计算得到的全局均值 $\text{avg}(X)$ 张量传递给全连接层, 以生成全局上下文信息, 计算过程见式(5)。 W 是全连接层的权重矩阵, 维度是 (D, D) , 即 $(32, 32)$, c 是偏置向量, 计算得到的全局上下文信息张量 $\text{global_context}(X)$ 的维度为 $(B, 1, D)$ 。

$$\text{global_context}(X) = W \cdot \text{avg}(X) + c \quad (5)$$

将 URL 的全局信息加入到每个 URL 字符张量中, 见式(6), 其中, $b \in [0, B-1]$, $d \in [0, D-1]$ 。

$$X_{b,i,d} = X_{b,i,d} + \text{global_context}(X_{b,0,d}) \quad (6)$$

Transformer 的位置编码仅表征字符相对位置, 缺乏对 URL 结构语义的全局感知 (如域名、路径、参数的层级关系)。恶意 URL 的变形常涉及域名仿冒、路径重定向、长参数链、高符号密度等特征, Transformer 因缺乏全局整合, 易出现局部推理正确但整体推理错误的问题。设计全局信息共享机制旨在捕捉输入数据的综合特征, 从而为注意力机制的聚焦和调整提供指导。

2.2.2 自适应多头自注意力机制

本文参考 Transformer 框架的多头自注意力机制, 因为在恶意 URL 检测中可以学习到 URL 不同方面的特征信息, 如 URL 的长度、特定的域名结构等, 同时 Transformer 的编码层能够高效处理长序列, 捕捉其中的长距离依赖关系, 这对于理解长 URL 的结构和语义非常有帮助。

尽管多头自注意力机制在许多任务中表现出色, 但是其结构中每个头的权重都是一样的, 因此本文引入了一种新型自适应注意力机制, 这种机制通过动态权重分配来增强模型的性能和灵活性, 从而更精准地处理和分析序列数据。

APEA 的自适应机制通过引入动态权重矩阵

Head_Weight, 实现对关键特征的自适应聚焦, 全局信息共享为自适应权重提供上下文参考, 形成“全局统计-局部聚焦”的效果。

本文将全局信息共享得到的输出记为 Y , Y 的维度为 (B, L, D) , 类似地, 通过式(7)计算线性投影 Q 、 K 、 V , 其中 W^Q 、 W^K 、 W^V 分别是 Q 、 K 、 V 的线性变换矩阵, 维度均为 (d_model, d_model) 。

$$\begin{aligned} Q &= YW^Q \\ K &= YW^K \\ V &= YW^V \end{aligned} \quad (7)$$

将投影后的 Q 、 K 、 V 分割为 16 个头得到 Q' 、 K' 、 V' , 通过式(8)计算注意力权重, 其中, Softmax 函数用于归一化注意力权重, d_k 是每个头的维度, 通常 $d_k = d_model/h$, $\sqrt{d_k}^{-1}$ 是一个缩放因子, 用于防止内积过大导致的梯度消失问题, 本文的 $d_k = D/H$, H 为自注意力机制头的个数 ($H=16$), 计算得到 $d_k=2$ 。

$$\text{Attention}(Q', K', V') = \text{Softmax}\left(\frac{Q' K'^T}{\sqrt{d_k}}\right) V' \quad (8)$$

与传统多头自注意力机制不同的是, 本文在拼接和线性转换部分做出了修改, 将所有头的输出进行拼接, 并在线性变换的过程中考虑每个头的权重, 其计算过程见式(9)。

$$C_{\text{new}} = \text{Concat}(\text{head}_1, \dots, \text{head}_h) \cdot H_{\text{Weight}} \quad (9)$$

其中, H_{Weight} 为头的权重向量, 是长度为 H 的一维可学习向量, 与注意力头一一对应。 H_{Weight} 通过标准反向传播在每个 batch 同步更新, 受全局共享信息的间接调制。

基于上述自适应多头自注意力机制细节, 本文的算法流程如算法 1 所示。

算法 1 APEA 算法流程

- 1)~4) 特征提取
 - 1) Pre_embedding()#数据预处理
 - 2) Word_embeddings = Embedding()#字符级嵌入
 - 3) Word_embeddings += PositionalEncoding()#位置编码
 - 4) LayerNorm(word_embeddings)#层标准化
- 5) ~7) 全局信息共享
 - 5) Global_x = Global_context()#计算全局信息

6) For layer in enc_layers: # 为每一层的输入添加全局上下文

7) $x = \text{layer}(x + \text{Global_x})$

8)~13) 自适应多头自注意力 (each enc_layer)

8) $Q', K', V' = \text{split_heads}$ #分割头函数

9) $H_cat, A = \text{Scaled_dot_product_attention}(Q', K', V')$

10) #计算了自注意力机制的多头后, 乘上自适应多头权重

11) $H_cat = H_cat \times H_{Weights} \cdot \text{view}(1, -1, 1, 1)$

12) #view(1,-1,1,1)为重塑张量和转置操作

13) $H_cat = \text{Group_heads}()$ #见式(10)

14) FeatureExtraction () #两个全连接层和激活函数

15) ~16) 输出分类结果

15) ReLU()

16) Softmax()

在完成自适应头权重加权后, 通过式(10)将所有头的输出进行拼接, 其中, W^O 为输出投影矩阵并进行线性变换, 得到自适应注意力机制部分的输出, 其维度为 (B, L, D) 。

$$\text{MultiHead}_{\text{new}}(Q', K', V') = \text{Concat}_{\text{new}}(\text{head}_1, \dots, \text{head}_h) W^O \quad (10)$$

为适应恶意URL检测的分类任务需求, 提取URL信息的局部特征, 加入两个全连接层和激活函数 (ReLU), 增加了模型的非线性能力, 有助于模型捕捉更复杂的恶意URL。

最后, 池化操作保留每组的最大值, 再经全连接层和 Softmax 得到分类结果。

2.3 模型可应用与部署的场景

APEA 模型不仅检测性能优秀, 还可直接部署在客户端 (浏览器) 和服务器端^[25] (邮件服务器、部分 Web 服务器), 从而对恶意URL实现实时、高效识别。

在客户端浏览器部署 APEA 模型时, 目标是实时监测用户访问的URL, 防止加载恶意网页, 保护用户隐私与设备安全。模型以浏览器扩展程序形式集成, 在页面加载前即时判定用户输入或点击是否为恶意。由于检测在本地进行, 在URL加密传输前完成, 不受HTTPS等协议加密传输数据的影响, 规避了用户敏感数据泄露风险。相

较于传统浏览器 (如 Chrome、Firefox) 依赖黑名单, 难以应对新型未知威胁的局限, APEA 模型凭借深度学习技术实现对URL字符级特征的实时分析, 显著提升了对新型恶意URL的识别精度与响应速度。

电子邮件是网络攻击的主要传播渠道, 恶意URL检测对邮件安全防护至关重要。在邮件服务器端部署 APEA 模型时, 可获取传输层安全协议 (TLS) 解密后的邮件内容, 通过多用途互联网邮件扩展 (MIME) 解析提取URL, 再调用本地模型实时检测。针对端到端加密邮件, 由于服务器仅处理密文, 需依赖客户端进行恶意URL识别, 保护用户隐私的同时提高邮件访问安全性。

直接面向互联网的 Web 服务器也是恶意URL攻击的主要目标之一, 可在合适位置部署 APEA 模型, 扫描用户提交内容 (评论、互动区等) 中的URL。在高并发场景下, 可分布式部署多个检测节点, 利用消息队列和负载均衡器实现高效处理。

由上述可见, APEA 模型可在客户端和服务端灵活部署, 有效增强整体网络安全防护能力。

3 实验

3.1 实验数据与环境

3.1.1 实验数据

本文从Kaggle平台收集了大量有标签的URL, Kaggle成立于2010年, 现已成为全球最大的数据科学竞赛平台之一。本文整理的URL数据集基于ISCX-URL-2016^[26], 并补充了PhishTank和PhishStorm的钓鱼类URL, 将多源URL收集到单独的数据框架中, 且仅保留URL及其类型。

Kaggle标注了网页的详细类别, 包括正常类、篡改类、恶意软件类和钓鱼类URL, 原始数据集共有651 191个样本。本文APEA模型需要在不同URL类型下做充分的实验, 验证其有效性, Kaggle的URL数据集是APEA模型训练的极佳数据资源。

3.1.2 实验环境及训练配置

硬件环境: 采用配备Intel Ultra 7 155H处理器、Intel ARC Graphics显卡及32 GB内存的计算平台。

软件环境: Windows 11 操作系统, 编程语言为 Python 3.12.5, 深度学习框架选用 PyTorch 1.8。辅助库有 NumPy 2.1.1、Pandas 2.2.3、sklearn 1.5.2 和 Matplotlib 3.9.2。

训练配置: 训练集、验证集与测试集同参数。模型使用 AdamW 优化器, 初始学习率设为 1×10^{-3} , 并采用交叉损失动态调整, 单批次训练样本量为 16。正则化方面, 设置 Dropout 率为 0.1, 防止模型过拟合。模型架构方面隐藏维度为 32, 配备 16 头自注意力机制, 输出类别为 2 类。

3.2 评估指标

本文的恶意网页检测为二分类任务, 选取的评估指标有: 准确率 (accuracy, ACC)、精确率 (precision)、召回率 (recall)、F1 分数 (F1-score)。

准确率是衡量模型预测正确的样本占总样本比例的标准, 计算过程见式(11), 其中, TP 表示预测为正常 URL, 实际也是正常 URL 的用例数量; TN 表示预测为恶意 URL, 实际也为恶意 URL 的用例数量; FP 表示预测为正常 URL, 实际为恶意 URL 的用例数量; FN 表示预测为恶意 URL, 实际为正常 URL 的用例数量。ACC 的值代表了预测 URL 恶意性正确的数量比例。

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \quad (11)$$

精确率指预测为正常 URL 的样本中实际也为正常 URL 的比例, 计算过程见式(12)。

$$precision = \frac{TP}{TP + FP} \quad (12)$$

召回率指衡量模型能够找出多少真正正类样本的标准, 计算过程见式(13), 它是实际为正常 URL 中, 预测也是正常 URL 的比例。

$$recall = \frac{TP}{TP + FN} \quad (13)$$

F1 分数则综合了精确率和召回率, 通过调和平均的方式, 平衡了精确率和召回率的影响, 使模型不能只在一个指标上表现好而在另一个指标上表现差, F1 分数更能反映恶意 URL 检测模型的整体性能, 其计算过程见式(14)。

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (14)$$

3.3 网页恶意性识别能力分析

首先将详细分析和比较现有的不同机器学习

模型和深度学习模型在网页恶意性识别上的表现, 包括决策树、支持向量机、朴素贝叶斯、长短期记忆网络; 然后介绍 APEA 框架的有效性。为详细分析其在篡改网页、恶意软件网页、钓鱼网页上的识别能力, 本文针对性展开了多项实验。

3.3.1 恶意 URL 检测分析

恶意 URL 包括篡改类 URL、恶意软件类 URL、钓鱼类 URL。其中, 篡改类 URL 通常包含明显的不规则域名或重定向; 恶意软件类 URL 通常包含特定的参数以实现病毒、木马等文件的下载; 钓鱼类 URL 通过伪装成合法网站来窃取用户的敏感信息, 它在 URL 结构上和正常 URL 通常只存在细微差别, 甚至没有任何区别, 但是在网页内部通过设计诱导用户输入自己的敏感信息。篡改类 URL 包含正常 URL 几乎不会涉及的重定向等特征, 而恶意软件类 URL 的文件下载等功能在部分正常 URL 也会存在, 所以要精确识别还需要进一步区别下载的参数, 钓鱼类 URL 与正常 URL 的区别是最小的, 故三者模型训练后得到的各项评估指标理应呈现篡改类 URL 的检测实验效果最好, 钓鱼类 URL 的检测实验效果最差。

为验证 APEA 模型中不细分细节标签的恶意 URL 检测效果, 即 URL 数据只有正常和恶意, 本文展开了恶意 URL 检测分析实验, 从 Kaggle 中获取 URL 数据, 数据总量为 12 000 条, 实验采用的 3 类恶意 URL 数量比例为 1:1:1, 正常和恶意数据比例为 1:1, 训练集、验证集、测试集数据比例为 4:3:3。实验结果如图 2 所示。

APEA 模型的各项指标都取得了最高值, 说明 APEA 模型在恶意 URL 检测实验中比其他传统机器学习模型和深度学习模型更优秀。

为进一步验证 APEA 模型的有效性, 本文将其与近几年恶意 URL 检测相关工作进行对比分析。URLdeepDetect^[15]模型使用语义向量模型和 URL 加密等技术分析 URL 的词汇特征, 并使用 LSTM 和 k 均值聚类对 URL 进行分类, 在同类研究中表现出较高基线性能, 其 LSTM 版本在数据集上取得 98.3% 的准确率, k 均值聚类版本准确率达 99.7%。本文采用与 URLdeepDetect

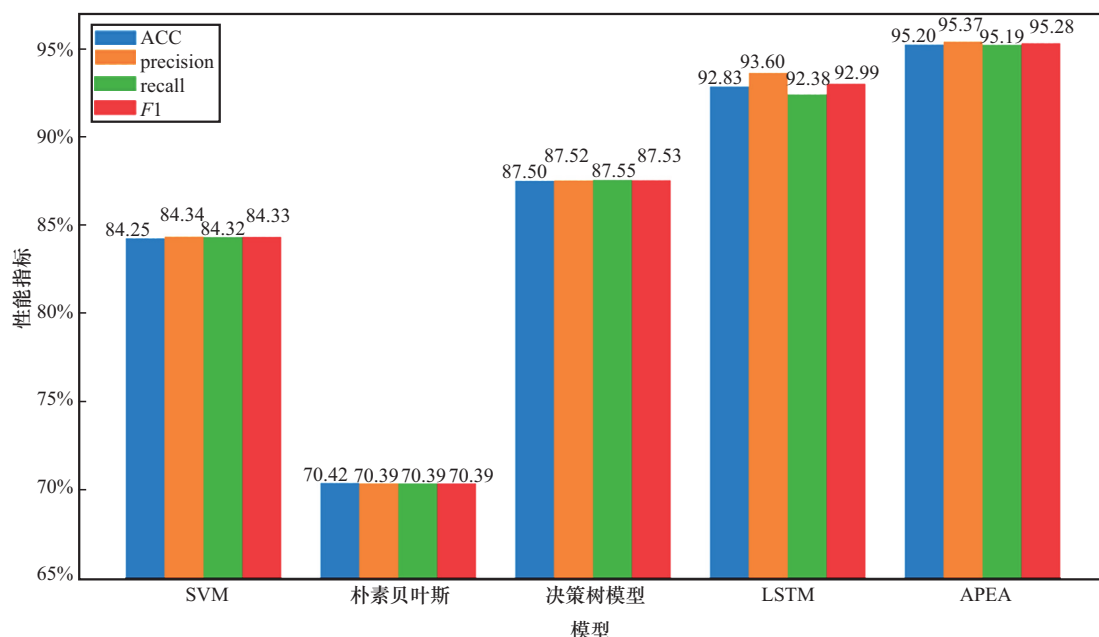


图2 恶意URL检测实验结果

模型同样的数据进行训练和预测，实验结果见表1，本文模型的精确率、召回率、 $F1$ 分数均高于URLdeepDetect。

表1 APEA和URLdeepDetect模型对比实验结果

模型	ACC	precision	recall	$F1$
URLdeepDetect+LSTM	98.3%	99.0%	97.7%	98.3%
URLdeepDetect+ k 均值	99.7%	97.0%	95.7%	96.8%
APEA	99.7%	99.6%	99.7%	99.6%

URLdeepDetect采用词级语义向量建模，难以捕捉字符替换等细节变形；而APEA通过字符级嵌入与自适应多头自注意力机制，可动态聚焦细微特征差异，结合全局信息共享机制对URL整体结构的语义整合，实现了细节识别和全局理解的双重优化，最终在复杂恶意URL检测中展现出更强的泛化能力。

为进一步探索APEA模型在各类URL下的效果，本文针对篡改类URL、恶意软件类URL、钓鱼类URL分别展开实验分析。

3.3.2 篡改类URL检测能力分析

本文从Kaggle数据集中，筛选出96 457条篡改类URL数据和428 103条正常URL数据。本文采用一种随机策略来增强模型的泛化能力，利用Numpy库中的随机函数permutation对数据集

索引进行随机排列，该方法能确保模型在训练过程中接触到多样化的数据样本。本文按照上述随机策略选取数据，数据总量为12 000条，其中，正常URL和篡改类URL样本各占一半，本文将数据集分为训练集（40%）、验证集（30%）和测试集（30%）。本文对APEA模型进行了17个周期的训练，图3和图4分别展示了APEA模型在篡改类URL检测中的准确率、损失值变化趋势。

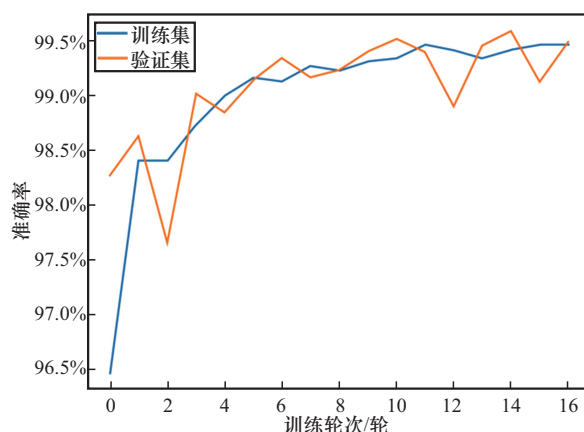


图3 APEA模型在篡改类URL检测中的准确率

由图3可知，APEA模型检测的准确率在训练初期迅速提升并在0.995上波动，由图4可知，APEA模型的训练损失在初始阶段出现显著的下

降趋势,并在第8轮训练左右逐渐趋于稳定,最终稳定在0.32左右,这表明APEA模型在训练集上达到了较高的拟合度,且未出现明显的过拟合现象。训练集和验证集的准确率曲线非常接近,进一步证实了模型的稳定性和泛化性。

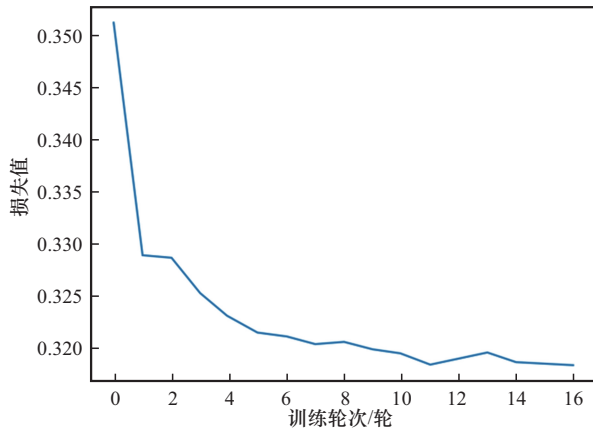


图4 APEA模型在篡改类URL检测中的损失值

为进一步说明APEA模型在篡改类URL检测任务中的性能优势,本文采用和APEA模型同样的数据集和训练数据切分比例,在传统机器学习模型SVM、朴素贝叶斯、决策树、LSTM中训练,结果见表2。

表2 篡改类URL检测实验结果

模型	ACC	precision	recall	F1
SVM	95.41%	95.42%	95.40%	95.41%
朴素贝叶斯	67.17%	68.59%	67.52%	68.0%
决策树	97.41%	97.40%	97.43%	97.41%
LSTM	93.67%	95.62%	92.52%	94.05%
APEA	99.58%	99.58%	99.57%	99.57%

由表2可知,APEA模型在所有评估指标上均达到了最高分,传统机器学习模型中朴素贝叶斯模型表现最差,本文猜想主要原因是朴素贝叶斯过于依赖专家知识,需要对训练的单词级特征做精确的筛选;决策树则表现较好,但仍低于APEA模型。LSTM模型表现一般,各项指标远低于APEA模型,同时APEA模型在精确率和召回率之间取得了良好的平衡。

3.3.3 恶意软件类URL检测能力分析

本文筛选标签为正常和恶意的URL数据,

一共有32 520条恶意URL数据。同样地,采用Numpy中的随机函数对数据集采样,数据总量为12 000条,正常URL和恶意URL各占一半,再细分为训练集(40%)、验证集(30%)和测试集(30%)。图5和图6分别给出APEA模型在恶意URL检测的准确率和损失值变化趋势。

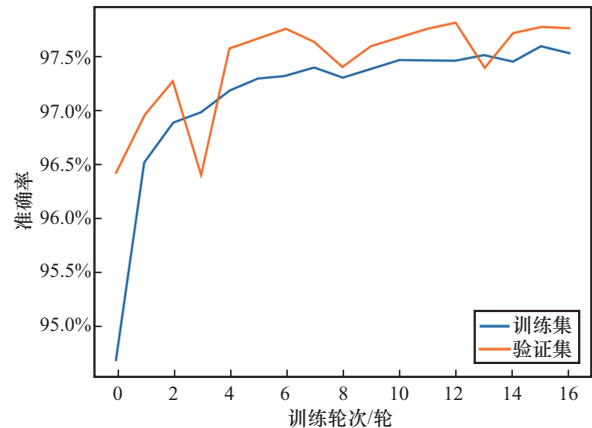


图5 APEA模型在恶意URL检测中的准确率

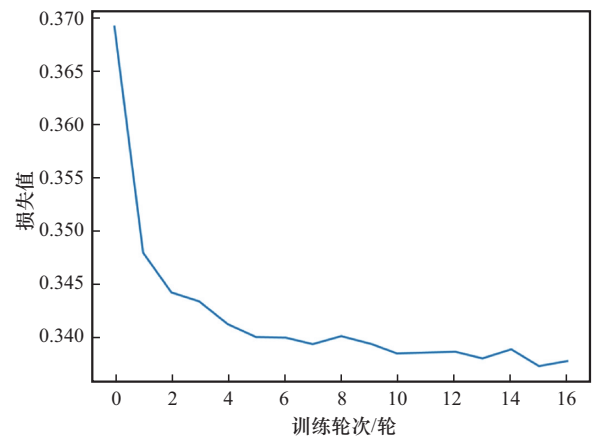


图6 APEA模型在恶意URL检测中的损失值

由图5可知,APEA模型检测的准确率在训练初期迅速提升并随后在0.975周围波动。

由图6可知,APEA模型的训练损失在第6轮训练左右逐渐稳定在0.34左右,说明在恶意URL检测中APEA模型同样具备稳定性和泛化性。

由表3可知,APEA模型在恶意URL检测中各项指标均达到最高分,传统机器学习中朴素贝叶斯得分依旧最低,APEA模型在精确率和召回率之间取得了良好的平衡。

表3 恶意URL检测实验结果

模型	ACC	precision	recall	F1
SVM	94.67%	94.66%	94.66%	94.66%
朴素贝叶斯	61.08%	71.51%	60.06%	65.29%
决策树	93.83%	93.83%	93.82%	93.83%
LSTM	93.17%	95.73%	90.76%	93.18%
APEA	97.65%	97.73%	97.64%	97.68%

3.3.4 钓鱼类URL检测能力分析

本文筛选标签为正常和钓鱼的URL数据, 共有94 111条钓鱼类URL数据, 数据量以及数据的处理和划分与前两个实验一样, 数据总量为12 000条, 正常URL和恶意URL各占一半, 再细分为训练集(40%)、验证集(30%)和测试集(30%)。

由图7可知, 模型的准确率最后稳定在0.94左右, 相比篡改类URL和恶意URL准确率更低, 这一现象符合客观事实。与篡改类URL和恶意URL不同的是, 钓鱼类URL尽可能模仿合法网站的URL来诱骗用户, 它们可能在域名、子域名或路径中包含更多与合法网站相似的元素, 这种刻意模仿行为让钓鱼类URL和正常URL更为接近。由图8可知, 模型的训练损失在第8轮训练后逐渐稳定在0.37左右, 说明在钓鱼类URL检测中APEA模型同样具备稳定性和泛化性。

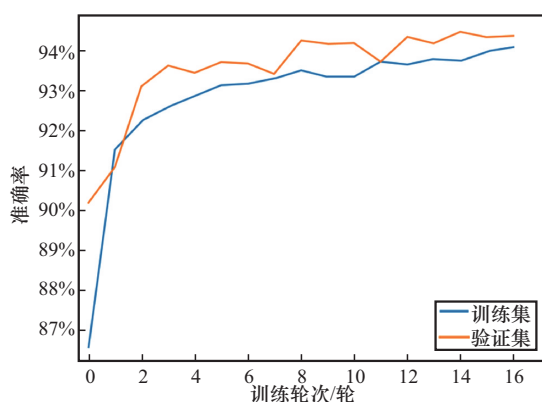


图7 APEA模型在钓鱼类URL检测中的准确率

为进一步验证APEA模型在钓鱼类URL检测中的性能优势, 本文将其与传统机器学习模型和深度学习模型进行对比验证, 实验结果见表4, APEA模型的各项性能指标均优于其他模型。

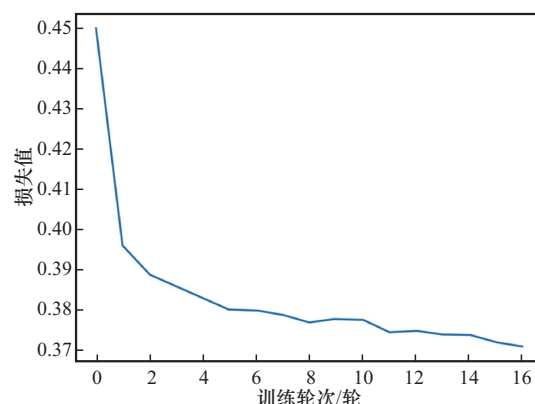


图8 APEA模型在钓鱼类URL检测中的损失值

表4 钓鱼类URL检测实验结果

模型	ACC	precision	recall	F1
SVM	76.00%	76.03%	76.05%	76.03%
朴素贝叶斯	60.58%	62.55%	59.94%	61.21%
决策树	80.17%	80.15%	80.14%	80.15%
LSTM	90.00%	89.42%	90.44%	89.93%
APEA	93.64%	93.70%	93.64%	93.67%

3.4 消融实验

为验证APEA模型各核心模块的有效性, 本文设计了消融实验, 通过移除关键组件对比性能变化, 量化分析全局信息共享机制与自适应多头自注意力机制的独立贡献, 并观察二者的协同效应。消融实验的数据集采用和恶意URL检测分析部分一样的数据集, 其中篡改类URL、恶意软件类URL和钓鱼类URL比例为1:1:1, 实验设置3组对比模型。

将实验的全局信息共享和自注意力机制的自适应功能移除并展开恶意URL检测, 其中数据集和恶意URL检测分析部分采用的数据集一样, 实验结果见表5第1行。相比该对比模型, APEA模型的各项性能提升了2.8%左右。

表5 消融实验实验结果

模型	ACC	precision	recall	F1
APEA-Global_context-Adoptive_mutihead	92.41%	92.42%	92.46%	92.44%
APEA-Global_context	94.43%	95.25%	95.09%	95.17%
APEA-Adoptive_mutihead	93.86%	94.51%	94.49%	94.49%
APEA	95.20%	95.37%	95.19%	95.28%

将实验的全局信息共享功能移除后展开恶意 URL 检测, 其中数据集和恶意 URL 检测分析部分采用的数据集一样, 实验结果见表 5 第 2 行。将实验的自适应多头自注意力机制功能去掉后展开恶意 URL 检测, 训练结果见表 5 第 3 行。

由表 5 可以看出, 移除全局共享机制后, 模型性能有所下降, 移除多头自注意力机制的自适应功能后, 模型性能也有所下降, 两个模块都移除性能最低。这说明两模块存在显著协同作用: 全局信息为自适应权重提供语义导向, 自适应权重将全局语义转化为局部特征的关注度分配, 减少误判和漏判, 形成“全局理解-局部聚焦”的双向增强机制。

3.5 模型检测效率

APEA 模型在保证高检测精度的同时, 展现出优秀的实时性。实验针对篡改类 URL、恶意软件类 URL 和钓鱼类 URL 3 类任务展开, 测试 URL 检测平均时间。结果表明, APEA 在篡改类 URL 检测中平均每组 URL 推理时间为 0.021 s, 单个 URL 平均推理时间为 1.312 5 ms; 平均每组恶意软件类 URL 检测时间 0.016 8 s, 单个 URL 平均推理时间为 1.050 0 ms; 平均每组钓鱼类 URL 检测时间为 0.017 s, 单个 URL 平均推理时间为 1.062 5 ms。

实验针对恶意 URL (混合 URL 样本) 检测任务记录检测时间, APEA 模型平均每组 URL (16 个 URL) 检测平均时间 0.020 1 s, 单个 URL 样本的检测时间约为 1.256 2 ms。

综上所述, APEA 模型在篡改类 URL、恶意软件类 URL 和钓鱼类 URL 检测中, 单样本平均时间分别为 1.31 ms、1.05 ms 和 1.06 ms, 恶意 URL 整体检测单样本时间约 1.26 ms, 满足一般实时检测需求, 在保障精度的同时实现高效推理, 针对高并发的服务器可部署多个检测节点同时检测。

3.6 案例分析

将本文模型和本文涉及的部分 Transformer 模型、朴素贝叶斯模型等进行对比, 其中部分恶意 URL 案例结果见表 6, 部分正常 URL 案例结果见表 7。

由表 6 可知, 针对随机生成的恶意软件类 URL, APEA 模型相比其他模型有更好的结果, 朴素贝叶斯模型效果最差, 几乎不能识别随机生成的 URL 的恶意性, 这可能是因为: 朴素贝叶斯算法假设特征之间是相互独立的, 难以捕获随机生成的 URL 字符信息之间的关联。

钓鱼类 URL 通常使用合法 URL 的变体, 例如, 表 6 中 login.paypal.update.scre-chkot.com 使

表 6 恶意 URL 案例结果

URL (恶意)	类别	APEA	Transformer	朴素贝叶斯
en.wikipedia.org/wiki/feature_driven_development	2	√	√	×
m.facebook.com-----validate-account.bintangjb.com	2	√	×	×
login.paypal.update.scre-chkot.com	2	√	×	×
wellsfargo.com.moclangarden.vn	2	√	√	×
paypal.com.enrichingcreations.com	2	√	√	×
ludo.iwarp.com/gwm/	2	√	×	√
virtual-palace.com/usfreight/	2	√	√	×
subdomain.chase-reg.net16.net	2	√	×	×
nectelephoneengineer.com	2	√	×	×
vk5kqc8a8bqffguw.science	1	√	×	×
w6gvmzycsigrdhbezny3awdf.review	1	√	√	×
x8tobjwvxsvdqyljoc.download	1	√	√	×
alchemywork.com/fan/base/config.jpg	1	√	√	×
x8tobjwvxsvdqyljoc.download	1	√	√	×

注: √表示预测为恶意, ×表示预测为正常。类别 1: 恶意软件类 URL; 类别 2: 钓鱼类 URL。

表7 正常URL案例结果

URL (正常)	APEA	Transformer	朴素贝叶斯
bizofbaseball.com/index.php option=com_content&view=article&id=5441: mariners-athletics-to-open-2012-mlb-season-in-japan&catid=30: mlb-news&itemid=42	√	×	×
http: //tobogo.net/cdsb/board.php board=unfair&bm=view&no=69&category=&auth=&page=1&search=&keyword=&recom=	√	×	×
lewisblack.com/mywork.aspx	√	×	×
dawn08.com/	√	×	×
wtpaz.org/county-ohio-2/	√	×	√
buzzsugar.com/tag/atonement	√	×	√
q.gs/wdoh	√	×	×

注:√表示预测为正常,×表示预测为恶意。

用了 paypal.com 的变体; wells Fargo.com. mo-clangaden.vn 使用了 wells Fargo.com 的变体。en.wikipedia.org/wiki/feature_driven_development 使用了 wikipedia.org 的变体; m.facebook.com-----validate-account. bintangjb.com 使用了 facebook.com 的变体。这些 URL 通过在合法域名后添加额外的子域名或路径, 试图伪装成合法网站。

有的利用品牌域名, 例如 paypal.com.enrichingcreations.com, 试图模仿 paypal 官方域名, 但实际主域名是 enrichingcreations.com, 类似地, ludo.iwarp.com/gwm/还在子域名中包含品牌 LudoKing 关键词, virtual-palace.com/usfreight/在路径中设置 usfreight 模仿货运公司网站结构。Subdomain.chase-reg.net16.net 使用 chase-reg 试图模仿 Chase 银行, 但使用了低信誉的免费域名 net16.net, 诱导用户以为这是 Chase 银行的子域名, 从而输入敏感信息。Nectelephoneengineer.com 是包含知名服务 NEC Telephone Engineer 的主要单词模仿合法服务网站。APEA 模型结合了 Transformer 强大的语义理解能力和全局信息共享机制, 能关注到比 Transformer 更长距离的字符之间的关联, 通过自适应多头自注意力机制能更准确地关注到重点特征, 对于上述几种钓鱼类 URL 有着更好的检测效果。

由表7可知, 在预测长参数链 (http://tobogo.net/cdsb/board.php board=unfair&bm=view&no=69&category=&auth=&page=1&search=&keyword=&recom=、bizofbaseball.com/index.php option=com_content&view=article&id=5441: mariners-

athletics-to-open-2012-mlb-season-in-japan&catid=30:mlb-news&itemid=42) 等需要联系全局信息判断时, APEA 模型能更精准判断其是否为恶意 URL, 而 Transformer 则更加关注局部信息。这里虽然出现了长参数链、高符号密度等可疑因素, 但还需要结合长参数链的参数细节并结合域名等全局信息综合判定, 显然这里的域名对应的网页类型和参数细节是匹配的, 故整体来看是正常 URL。针对域名较短、查询参数少的正常 URL, 不包含全局信息共享和自适应动态权重机制的模型难以学习到其中的正确关系 (例如 dawn08.com/、lewisblack.com/mywork.aspx、q.gs/wdoh 等), 朴素贝叶斯对这两类 URL 的检测效果也没有 APEA 模型效果好。

4 结束语

在数字化时代, 网络空间安全问题日益凸显, 恶意 URL 检测已成为网络安全领域的重要研究课题。本文提出一种新型深度学习框架检测恶意 URL, 该框架采用字符集嵌入和位置编码作为特征提取模块, 结合多头自注意力机制的编码层作为分类模型, 为提升多头自注意力对 URL 信息局部特征和全局特征的捕获能力, 引入全局信息共享机制并动态调整多头自注意力机制的头权重。

实验结果表明, 相较于现有机器学习和深度学习恶意 URL 检测方法, 本文提出的 APEA 模型在检测的准确率、精确率、召回率、F1 分数上都表现更好。本文进一步评估 APEA 模型

对篡改类、恶意软件类、钓鱼类 3 类 URL 的检测效果, 结果显示, 在 3 类 URL 检测中 APEA 模型都能得到最好的效果。与去掉全局信息共享机制和自适应动态权重机制的模型相比, APEA 模型取得约 2.8% 的综合性能提升。模型在效率表现上, 单样本平均检测耗时 1.26 ms, 满足实时需求。该特性便于集成到浏览器扩展、邮件服务器、Web 服务器等实际场景。APEA 模型摒弃传统基于 CNN 的恶意网页识别, 探索 Transformer 模型并取得性能提升, 但研究的深度相比 CNN 仍有限, 基于 Transformer 的恶意网页识别尚待深入, 未来工作将聚焦模型优化与性能提升。

参考文献:

- [1] SALEM A H, AZZAM S M, EMAM O E, et al. Advancing cybersecurity: a comprehensive review of AI-driven detection techniques[J]. *Journal of Big Data*, 2024, 11(1): 105.
- [2] MANKAR N P, SAKUNDE P E, ZURANGE S, et al. Comparative evaluation of machine learning models for malicious URL detection[C]// *Proceedings of the 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSOCiCon)*. Piscataway: IEEE Press, 2024: 1-7.
- [3] REYES-DORTA N, CABALLERO-GIL P, ROSA-REMEDIOS C. Detection of malicious URLs using machine learning[J]. *Wireless Networks*, 2024, 30(9): 7543-7560.
- [4] 沙泓州, 刘庆云, 柳厅文, 等. 恶意网页识别研究综述[J]. *计算机学报*, 2016, 39(3): 529-542.
SHA H Z, LIU Q Y, LIU T W, et al. Survey on malicious webpage detection research[J]. *Chinese Journal of Computers*, 2016, 39(3): 529-542.
- [5] 杨胜杰, 陈朝阳, 徐逸, 等. 基于深度学习与特征融合的恶意网页识别方法研究[J]. *信息安全学报*, 2024, 9(3): 176-190.
YANG S J, CHEN Z Y, XU Y, et al. Research on malicious web page identification method based on deep learning and feature fusion[J]. *Journal of Cyber Security*, 2024, 9(3): 176-190.
- [6] PATIL D R, PATIL J B. Malicious URLs detection using decision tree classifiers and majority voting technique[J]. *Cybernetics and Information Technologies*, 2018, 18(1): 11-29.
- [7] LE H, 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.
- [8] BOZKIR A S, DALGIC F C, AYDOS M. GramBeddings: a new neural network for URL based identification of phishing Web pages through N-gram embeddings[J]. *Computers & Security*, 2023, 124: 102964.
- [9] ASHISH V. Attention is all you need[C]// *Advances in neural information processing systems*, 30 *NeurIPS 2017*. LongBeach, CA, USA: Curran Associates, 2017: 5998-6008.
- [10] PRAKASH P, KUMAR M, KOMPPELLA R R, et al. PhishNet: predictive blacklisting to detect phishing attacks[C]// *2010 Proceedings IEEE INFOCOM*. Piscataway: IEEE Press, 2010: 1-5.
- [11] SHENG S, WARDMAN B, WARNER G, et al. An empirical analysis of phishing blacklists[C]// *Proceedings of the Sixth Conference on Email and Anti-Spam (CEAS 2009)*. Mountain View, CA, USA: CEAS, 2009: 1-9.
- [12] ABAD S, GHOLAMY H, ASLANI M. Classification of malicious URLs using machine learning[J]. *Sensors*, 2023, 23(18): 7760.
- [13] JASWANTHI T, HARSHITHA T, TANYA S, et al. Malicious URL detection: comparative study of machine learning algorithms[C]// *Proceedings of the 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. Piscataway: IEEE Press, 2024: 1-5.
- [14] HAMZA A, HAMMAM F, ABOUZEID M, et al. Malicious URL and intrusion detection using machine learning[C]// *Proceedings of the 2024 International Conference on Information Networking (ICOIN)*. Piscataway: IEEE Press, 2024: 795-800.
- [15] BANNUR S N, SAUL L K, SAVAGE S. Judging a site by its content: learning the textual, structural, and visual features of malicious web pages[C]// *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*. New York: ACM, 2011: 1-10.
- [16] HUANG H J, QIAN L, WANG Y J. A SVM-based technique to detect phishing URLs[J]. *Information Technology Journal*, 2012, 11(7): 921-925.
- [17] AGGARWAL A, RAJADESINGAN A, KUMARAGURU P. PhishAri: Automatic realtime phishing detection on twitter[C]// *Proceedings of the 2012 eCrime Researchers Summit*. Piscataway: IEEE Press, 2012: 1-12.
- [18] CUZZOCREA A, MARTINELLI F, MERCALDO F. A machine-learning framework for supporting intelligent web-phishing detection and analysis[C]// *Proceedings of the 23rd International Database Applications & Engineering Symposium on - IDEAS '19*. New York: ACM, 2019: 1-3.
- [19] 张杰, 彭国军, 杨秀璋. 基于动态 API 调用序列和机器学习的恶意逃避样本检测方法[J]. *山东大学学报(理学版)*, 2022, 57(7): 85-93, 102.
ZHANG J, PENG G J, YANG X Z. Malicious evasion sample detection based on dynamic API call sequence and machine learning[J]. *Journal of Shandong University (Natural Science)*, 2022, 57(7): 85-93, 102.
- [20] DARAGHMI E Y, QADAN S, DARAGHMI Y A, et al. From text to insight: an integrated CNN-BiLSTM-GRU model for Arabic cyberbullying detection[J]. *IEEE Access*, 2024, 12: 103504-103519.

- [21] ZHANG X, ZHAO J, LECUN Y. Character-level convolutional networks for text classification[C]//Advances in Neural Information Processing Systems 28 (NIPS 2015). Montreal, Canada: Curran Associates, 2015: 649-657.
- [22] AFZAL S, ASIM M, JAVED A R, et al. URLdeepDetect: a deep learning approach for detecting malicious URLs using semantic vector models[J]. Journal of Network and Systems Management, 2021, 29(3): 21.
- [23] YU B Y, TANG F, ERGU D J, et al. Efficient classification of malicious URLs: m-BERT: a modified BERT variant for enhanced semantic understanding[J]. IEEE Access, 2024, 12: 13453-13468.
- [24] KOROTEEV M V. BERT: a review of applications in natural language processing and understanding[EB]. arXiv preprint, 2021: arXiv:2103.11943.
- [25] 白天翔, 沈震, 刘雅婷, 等. 平行机器: 一种智能机器的管理与控制框架[J]. 智能科学与技术学报, 2019, 1(2): 181-191.
BAI T X, SHEN Z, LIU Y T, et al. Parallel machine: a framework for the control and management for intelligent machines[J]. Chinese Journal of Intelligent Science and Technology, 2019, 1(2): 181-191.
- [26] MAMUN M S I, AHMAD RATHORE M, LASHKARI A H, et al. Detecting malicious URLs using lexical analysis[M]. Network and System Security. Cham: Springer International Publishing, 2016.

[作者简介]



张慧菲 (2000-), 女, 重庆合川人, 武汉大学硕士生, 主要研究方向为系统安全和恶意代码分析。



杨秀璋 (1991-), 男, 贵州施秉人, 贵州大学副教授, 主要研究方向为系统安全和恶意代码分析。



彭国军 (1979-), 男, 湖北荆州人, 武汉大学教授、博士生导师, 主要研究方向为信息系统安全。