基于深度学习与特征融合的恶意网页识别方法研究

杨胜杰[1], 陈朝阳[1], 徐 逸[1], 刘建刚[2]

[1]湖南工商大学 计算机学院 长沙 中国 410205

[2]湖南工商大学 理学院 长沙 中国 410205

摘要 互联网环境的高度开放性和无序性导致了网络安全问题的普遍性和不可预知性, 网络安全问题已成为当前国际社会关注的热点问题。基于机器学习的恶意网页识别方法虽然卓有成就, 但随着对恶意网页识别需求的不断提高, 在识别效率上仍然表现出较大的局限性。本文提出一种基于深度学习与特征融合的识别方法, 将图卷积神经网络(Generalized connection network, GCN)与一维卷积神经网络(Convolution neural network, CNN)、支持向量机(Support vector machine, SVM)相结合。首先, 考虑到传统神经网络只适用于处理结构化数据以及无法很好的捕获单词间非连续和长距离依赖关系, 从而影响网页识别准确率的缺点, 通过 GCN 丰富的关系结构有效捕获并保持网页文本的全局信息; 其次, CNN 可以弥补 GCN 在局部特征信息提取方面的不足, 通过一维 CNN 对网页 URL(Uniform resource locator, URL)进行局部信息提取, 并进一步将捕获到的 URL 局部特征与网页文本全局特征进行融合, 从而选择出兼顾 CNN 模型和 GCN 模型特点的更具代表性的网页特征; 最终, 将融合后的特征输入到 SVM 分类器中进行网页判别。本文首次将 GCN 应用于恶意网页识别领域, 通过组合模型有效兼顾了深度学习与机器学习的优点, 将深度学习网络模型作为特征提取器, 而将机器学习分类算法作为分类器, 通过实验证明, 测试准确率达到 92.5%, 高于已有的浅层的机器学习检测方法以及单一的神经网络模型。本文提出的方法具有更高的稳定性, 以及在精确率、召回率、F1 值等多项检测指标上展现出更加优越的性能。

关键词 恶意网页; 机器学习; 深度学习; 特征融合

中图法分类号 TP309.5 DOI 号 10.19363/J.cnki.cn10-1380/tn.2024.05.12

Research on Malicious Web Page Identification Method Based on Deep Learning and Feature Fusion

YANG Shengjie[1], CHEN Zhaoyang[1], XU Yi[1], LIU Jiangang[2]

[1] School of Computer Science, Hunan University of Technology and Business, Changsha 410205, China

[2] School of Science, Hunan University of Technology and Business, Changsha 410205, China

Abstract The high degree of openness and disorder of the Internet environment has led to the universality and unpredictability of network security issues. Network security issues have become a hot issue that the international community is currently concerned about. Although the method of identifying malicious web pages based on machine learning has made great achievements, with the continuous improvement of the demand for identifying malicious web pages, it still shows great limitations in the identification efficiency. In this paper, a recognition method based on deep learning and feature fusion is proposed, which combines graph convolutional neural network (GCN) with one-dimensional convolutional neural network (CNN), support vector machine (SVM) combined. First, considering the shortcomings of traditional neural networks that are only suitable for processing structured data and cannot extract the discontinuity and long distance dependence between words well, which affects the accuracy of web page recognition, the rich relational structure of GCN effectively captures and maintains the global context of web page texts. Secondly, CNN can make up for the deficiency of GCN in extracting local feature information, the local information of the URL of the web page is extracted by one-dimensional CNN, and the local features of the captured URL are further fused with the global features of the web page text, so as to select More representative webpage features that take into account the characteristics of the CNN model and the GCN model; finally, input the fused features into the SVM classifier for webpage discrimination. In this paper, GCN is applied to the field of malicious web page identification for the first time, and the advantages of deep learning and machine learning are effectively taken into account through the combined model. The deep learning network model is used as the feature extractor, and the machine learning classification algorithm is used as the classifier. The accuracy rate reaches 92.5%, which is higher than the existing shallow machine learning detection methods and a single neural network model. The method proposed in this paper has higher stability, and shows more superior performance in multiple detection indicators such as precision rate, recall rate, and F1 value.

通讯作者: 陈朝阳, Email: czy_0811@163. com

本课题得到湖南教育厅科学研究项目(No. 21A0385, No. 22B0612)和湖南省自然科学基金面上项目(No. 2022JJ30214)资助。

收稿日期: 2022-10-07; 修改日期: 2023-01-06; 定稿日期: 2024-01-31

Key words malicious web page; machine learning; deep learning; feature fusion

1 引言

随着互联网的蓬勃发展, 互联网技术已经深度融入人们的日常生活, 人们越来越多地通过互联网完成社交、购物、信息查询等行为。中国互联网中心发布了第 50 次《中国互联网络发展状况统计报告》。报告中显示截至 2022 年 6 月, 我国网民规模达 10.51 亿, 使用手机上网的比例达 99.6%, 较 2021 年 12 月增长 1900 万, 互联网普及率达 74.4%[1]。随着网络技术的迅速发展, 恶意网页的数量也大量增加, 涌现出大量的网络犯罪行为。制作恶意网页的不法分子通过钓鱼网站、垃圾广告和恶意软件推广等方法, 在用户不知情的情况下, 或者诱导使用者进行点击、浏览等行为, 获取用户的敏感信息, 包括身份信息、银行卡信息和其他个人隐私数据来获取利益[2]。2021 年 4 月, 来自 106 个国家和地区的超过 5.33 亿 Facebook 用户的个人信息被泄露[3]; 6 月领英有 7.56 亿用户, 也就是约有 92%的领英用户的个人信息被在暗网平台出售, 此次事成为领英历史以来最大规模的数据泄漏[4]。恶意网页已经成为犯罪分子使用的最常见的网络犯罪手段, 给广大网民以及各大商家企业都造成极大的安全隐患。因此, 网络安全的维护显得尤为重要, 而如何对恶意网页进行及时、高效的识别也是一个迫切需要解决的问题。

针对恶意网页的识别问题, 研究人员提出了多种不同的识别技术和方法。Sumit Sahu 和 Bharti Dongre 等人[5]研究了一种利用 MATCH SCORE 特征来对非法页面进行分类的算法。该方法主要是对页面的标题和 URL 进行分析, MATCH SCORE 是网页标题和 URL 通过 N-gram 分词方法进行匹配的匹配分值, 并根据其大小来判别。Ma 等人[6-8]根据 DNS 信息、WHOIS 信息和 URL 的语法特征, 采用机器学习算法对 URL 进行识别。Canali 等人[9]在此基础上加入了 JavaScript 和 HTML 特征, 提升了恶意网页识别准确率。Wang 等[10]提取了 URL 特征、HTML 特征和 JavaScript 代码特征采用决策树算法进行分类, 比 Avira 和 360 安全分类真阳性要高.虽然目前基于机器学习的网页特征提取与分类颇有成效, 但其分类效果仍有很大的提升空间。

恶意网页识别技术已经成为热门的研究领域, 各种新的研究思路和方法不断被提出。本文正是以此为背景, 提出一种新的基于深度学习与特征融合的恶意网页识别方法。本文的主要贡献如下:

(1) 本文首次将 GCN 应用于恶意网页识别领域, 采用 GCN 建模可以处理复杂的关系结构以及捕获全局语义信息, 同时利用 CNN 模型对局部特征信息提取的优势对网页 URL 信息进行提取。不仅弥补了 GCN 捕获局部信息的不足, 还解决了传统神经网络只适合处理结构化数据以及仅能捕获文本内短距离连续的语义关系等问题。

(2) 本文提出新的组合模型, 有效结合深度学习网络模型与机器学习分类算法的优点, 经实验测试准确率达到 92.5%, 较浅层机器学习算法和单一神经网络模型都好。实验结果证明了本文提出的组合模型的合理性和准确性。

本文章节的组织结构如下: 第 1 节介绍了恶意网页的研究背景; 第 2 节介绍了恶意网页的相关知识和研究方法; 第 3 节介绍了本文设计的恶意网页识别模型; 第 4 节归纳了当前深度学习模型采用的数据集、评价指标并对本文设计的识别模型进行仿真实验, 对仿真结果进行实验分析; 第 5 节对全文进行总结, 并就恶意网页识别技术今后的研究方向进行了展望。

2 相关工作

2.1 相关知识

目前的恶意网页类型主要有: 钓鱼网页[11-12]、木马网页、垃圾广告网页等。所谓的“钓鱼”, 就是用一种类似于普通网页的方法, 包括 URL、界面布局等等[13]。其目的是让使用者认为他们在正常的页面上进行操作, 从而窃取用户的敏感信息(如银行卡, 身份证信息等); 木马网页是一种外表看起来很普通的网页, 一旦使用者打开, 它就会自动下载服务器端的木马程序, 而这一过程并不会被使用者察觉[14-15], 这类网页的执行方式是将 Web 脚本(JavaScript)注入到客户机; 恶意广告网页是指在没有经过使用者许可的情况下, 持续地发送大量的恶意广告, 诱使用户点击而受骗, 而常用的恶意广告网页面代码中会包含大量的定时、弹出窗口等函数。由于不同的恶意网页, 其触发机制也不尽相同, 所以, 寻找高辨识度的网页特征, 是提升恶意网页识别准确率的关键。

根据恶意网页的类型, 可以将其识别特征分为静态和动态两种类型。

(1) 静态特征是指网页的静态信息包含 URL 词汇特性(URL 长度、特定字符个数、词汇信息等), 网页主机信息(WHOIS 信息、服务器信息等), 网页内容

信息(网页文本、网页主题、网页 HTML 特殊标签数量等), 以及网页源代码(主要包含恶意脚本语言、链接关系、含有被认定为恶意函数的数量等)[16]。

(2) 动态特征来源于网页加载过程中的动态行为, 种类虽然少, 但是提取难度较大, 主要包括浏览器的行为、注册表和文件的变化情况、生成的 HTTP 信息[17-18]等。这些特征的获取, 研究者们必须在实际环境中进行网页的访问, 并对所发生的系统变化进行深入分析。在采集动态特征时, 往往需要将蜜罐技术和虚拟技术相结合来提取, 蜜罐技术模拟的真实环境会被有些恶意网页归类为虚拟环境而不执行全部的恶意行为[19]。

其中, 动态特征中浏览器的行为主要是基于 JavaScript 脚本语言实现的动态效果, 是实现人机交互的主要途径; 而静态特征中文本是传递信息的重要途径和载体。本文采集的公开数据集中, 包含整个网页的源码信息, 即网页文本信息和 JavaScript 源码信息等。因此, 融合网页 URL 信息和网页源码信息来进行恶意网页识别, 一定程度上能够提高识别模型的准确率。

2.2 早期特征标记识别方法

在早期恶意网页识方法中, 主要采用了黑名单和基于启发式的特征标记识别方法。黑名单是一个包含了恶意网页的 URL、IP 或者关键字信息的列表。这种方法是把已经发现的恶意网页信息存储到一个资料数据库中, 然后访问一个网页时, 再对该资料数据库进行查询, 看资料数据库中是否包含该网址, 存在则视为恶意网页。由于其技术简单、操作简便、快速查询等特点, 在谷歌浏览器、Malware 及 PhishTank[20]等实际项目和系统中得到广泛的应用。不难发现, 由于黑名单列表是持续更新的, 因而它仅能识别已经被标记过的恶意网页。黑名单法在用户访问一个未曾记录的恶意网页时, 会引起漏判。Prakash 等人[21]提出一种用于黑名单技术的改良方法 PhishNet。PhishNet 利用已知的网络钓鱼 URL 作为先验知识, 通过 URL 分解与相似度运算, 来识别和发现新的钓鱼网页。这样 PhishNet 可以扩展黑名单的适用范围, 帮助用户识别某些尚未被发现的恶意网页, 但是其识别能力依赖于原有黑名单集合的规模, 并且随着黑名单规模的增大, 查询代价也会呈线性增加。除此之外, 当发现一个恶意网页并将其上传到黑名单数据库中, 同样需要大量的时间来完成, 从而导致其时效性的下降。

针对黑名单方法存在的问题, 文献[22-23]中相关研究人员提出了一种基于启发式的恶意网页识别方法。其工作原理是依据恶意网页之间存在的相似性来设计, 从而实现启发式规则, 进而发现和识别恶意网页。它无需事先知道恶意网页的网址信息, 便能根据已有的规则, 对一些尚未被发现的恶意网页进行识别。所以它在主流浏览器上得到广泛的应用, 而且通常是作为一个浏览器的安全插件。然而, 传统的特征统计和启发式规则方法在大规模网页识别中已经不能很好的解决问题。第一, 由于启发式规则是基于模糊匹配, 这种方法会极大地提高网页的误判率。第二, 规则的更新比较困难, 因为启发式规则是根据现有的恶意网页页面特征进行统计和人工归纳而来, 所以规则的更新取决于相应的领域知识。

2.3 传统机器学习识别方法

由于早期特征标记识别方法存在漏判率高、误报率高和规则更新困难等缺点, 研究者们提出了一种更为系统化的基于传统机器学习的识别方法。这类方法首先将恶意网页的识别看作是一个文本分类或者聚类问题。相关的传统机器学习模型可以归纳为以下 3 大类: 基于概率统计、基于统计和基于几何。常见学习方法如贝叶斯模型, K 最近邻法(K-Nearest Neighbor, KNN)、决策树(Decision Tree, DT)、支持向量机(SVM)等。

目前, 使用最多的是基于概率模型的朴素贝叶斯分类器[24]。朴素贝叶斯法是最早用于文本分类的分类器算法, 其基本思想是对于指定的待分类项, 求解出该项出现的条件下, 类别出现的概率, 概率最大的就视为该待分类项的类别。基于贝叶斯决策论并且基于此项独立的假设, 几种不同属性对分类结果的影响是独立的。假定 d 为待分类文档的表示向量, 它属于文档类别集合 $W=\{w_1, w_2, w_3, ..., w_n\}$中某一类, 按照贝叶斯公式有:

$$P(d)=\sum_{k=1}^{N}P(w_j)P(d|w_j) \tag{1}$$

$$P(w_j|d)=\frac{P(w_j)P(d|w_j)}{P(d)},\ j=1,2,\cdots,n \tag{2}$$

其中 $P(w_j)$表示类别 w_j在样本集中的比例, 通过概率密度函数求出 $P(d|w_j)$。在进行分类时, 最大值 $P(w_j|d)$对应的类别 $C_{\max}$ 就为该分类文档类别。该方法具有计算简单、实用性强、分类效果良好等特点, 但其关于词项独立的假设受到了质疑。

基于统计模型的方法中最具代表性的就是 K 最近邻分类算法 (KNN)[25-26]。该算法的主要思想是对于一篇待分类文档, 在训练集中寻找 K 个最相近的邻居。将这 K 个邻居的类别作为该文档的候选类别, 该文档与 K 个邻居之间的相似度为候选类别的权重,

再利用设定的相似度阈值来获得这个文件的最终分类。KNN 算法的过程描述:

(1) 将文本内容变换成文本特征向量;

(2) 采用欧式距离公式计算测试用例与每个训练用例之间的距离, 即直线距离;

2 维空间中的欧几里得距离计算公式为:

$$d(x,y)=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2} \tag{3}$$

n 维空间中的欧几里得距离计算公式为:

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{4}$$

(3) 对上述计算得出的距离进行排序, 选择距离最小的 K 个用例;

(4) 分别计算所选 K 个用例的类别频率;

(5) 选择频率最高类别作为该测试用例的类别。

KNN 可以较好地避免样本的不平衡问题, 有较好的分类准确性和稳定性[27]。但对于数据集较大的分类任务时会导致空间复杂度高、计算效率降低, 甚至造成维度灾难。

基于几何的模型方法是以向量空间为模型, 把文本内容表示为多维空间向量。其目标是找到一个能够分割训练集各种数据的超平面。Vapnik 等人[28]提出该超平面能够在边界区域的中界上(也就是类别边界沿着与其垂直的方向的距离最大), 以确保预测时的稳定性和鲁棒性。其中, 最典型的分类器就是支持向量机(SVM), SVM 本质上就是一种能够区分普通正反两种数据的二分类器。该算法对二分类问题处理效率很高, 它采用了结构风险最小原则, 在多维空间里确定一个可以将样本分为两类、边缘值最大、分类准确率最高的超平面, 这样的超平面为最优超平面。

2.4 深度学习识别方法

现有传统机器学习识别方法还存在一定的局限性。例如: 传统的机器学习方法或规则匹配方法对特征选取的依赖程度较高, 其识别效果较大程度上取决于人工构造的特征的好坏, 往往由于特征选取不全面、对特征集合主动降维而特征辨识度降低等情况, 导致传统方法表现不稳定。

深度学习具有规模较大的参数模型, 能够自动完成复杂的特征提取工作, 为恶意网页识别提供了新思路。其中, 卷积神经网络采用权值共享的网络结构, 可以降低网络模型的复杂程度, 减少权值数量, 具有良好的鲁棒性。目前, 深度学习在自然语言处理中常用网络模型包括:

多层感知机 (Multi-layer perceptron, MLP) [29]: 至少含有一个隐藏层的由全连接层组成的神经网络, 且每个隐藏层的输出通过激活函数进行变换, 能处理非线性可分离的数据。

卷积神经网络 (CNN) [30]: 包含卷积计算, 是具有深层结构的前馈神经网络。它具有特征学习能力, 可以对每个层次结构的输入信息进行平移不变分类, 进行监督学习和非监督学习。由于其隐藏层中的卷积内核参数共享和层连接稀疏性, 卷积神经网络可以在较少的计算量下学习网格点化特征。

循环神经网络[31](Recurrent neural network, RNN): 是指在全连接神经网络的基础上增加了前后时序上的关系。RNN 的目的就是用来处理序列数据的, 网络的输出不仅仅依靠当前的输入, 而且还有前一步的神经元状态。

长短期记忆网络(Long short-term memory, LSTM): 是一种特殊的 RNN 网络, 它是为了解决传统 RNN 存在的长期依赖、梯度消失问题, 由 Hochreiter 和 Schmidhuber[32]提出来的。LSTM 的主要思想是引入了自适应的门控制机制, 来决定 LSTM 单元保存先前状态的程度和记忆当前输入单元的抽取特征程度。

3 恶意网页识别模型

3.1 模型结构

本文提出的基于深度学习与特征融合的恶意网页识别方法的模型结构如图 1 所示。总体流程包括:

(1) 获取 URL 和网页文本等网页特征数据, 构造数据集。

(2) 构造训练集和测试集, 并对数据进行预处理, 以适应深度学习模型。

(3) 设计神经网络模型, 即针对恶意网页识别任务, 设计能提取 URL 和网页文本特征的神经网络, 并利用训练数据对该部分两个神经网络模型进行训练, 逐步调优参数; 将训练好的神经网络模型用于处理好的数据集上, 进行特征提取; 再将提取到的 URL 特征与网页文本内容特征进行特征融合。

(4) 特征融合后采用机器学习算法对网页的恶意与否进行判别。

3.2 网页特征的选取和预处理

网页特征的选取是恶意网页识别过程中的重要步骤之一, 它直接影响到对网页进行分析和处理。找到具有较高识别特性的网页特征, 能够进一步提高恶意网页识别准确率。本文选取网页 URL 和网页文本作为恶意网页识别的研究对象。基于这两种网页特征研究的出发点是 URL 作为网页的入口, 攻击者多以 URL 为途径进行各种网络攻击, 且整个攻击流程不需要用户参与交互, 攻击方式非常隐蔽; 而恶

意网页的恶意行为, 绝大多数都会体现在网页内容上, 根据网页的内容可以判断网页的恶意与否。因此, 促成了结合网页 URL 和网页文本特征进行恶意网页识别的研究方向。

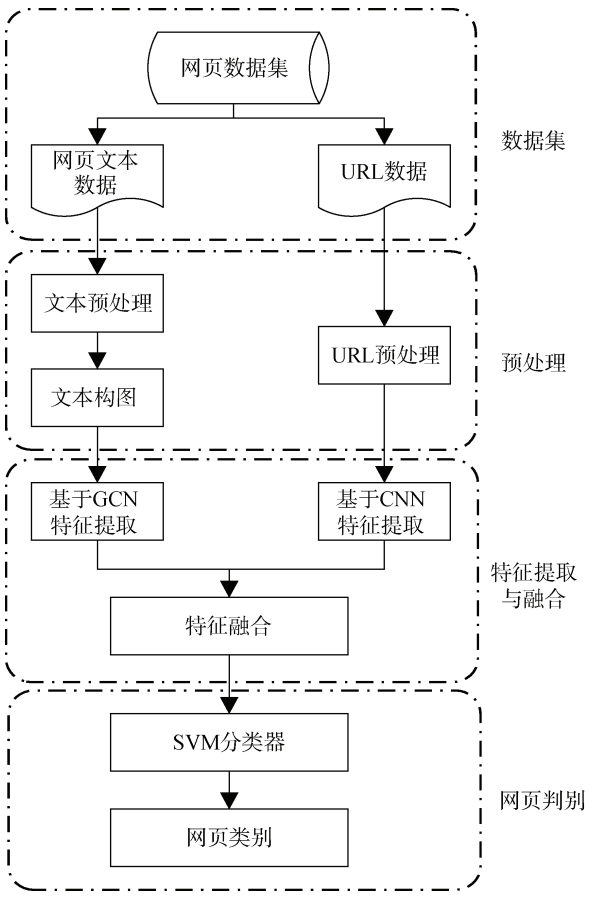

图 1 基于深度学习与特征融合识别方法的模型结构

Figure 1 Model structure of recognition method based on deep learning and feature fusion

3.2.1 URL 特征

URL 指的是各种资源的互联网地址, 结构如图 2 所示, 一般包括通信协议、主机名和域名、请求资源目录或文件地址三个重要部分。

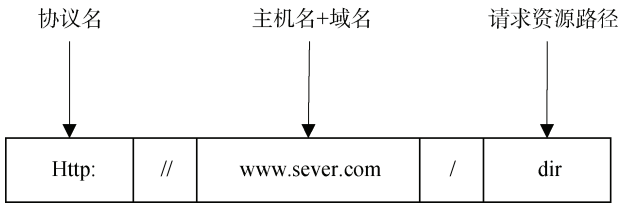

图 2 URL 结构图

Figure 2 URL structure diagram

通过对网页样本中 URL 进行分析, 发现恶意网页的 URL 通常会包含大量随机字符串、特殊字符等, 其长度远远长于正常网页的 URL。因此, 恶意 URL 在文本上就与正常 URL 有一定的区别。本文选择的 URL 特征有:

特征 1: URL 中是否含有互联网协议(Internet Protocol, IP)地址

从协议上面来看, 恶意网页的 URL 往往会采用 IP 地址代替域名来进行伪装。如使用“http://www.microsoft.com.cn@65.54.153.254/members/zjr72/”来代替真实网站“http://www.microsoft.com.cn”的域名, 一般多用于钓鱼网页中。

特征 2: URL 中的子域名

正常网页的子域名一般为“www”这样的格式深度为 1, 而恶意网页子域名深度则较大, 例如部分虚假钓鱼网页往往使用正规网站的域名作为其子域名用于欺骗用户, 如“qq.asld.com”由此更增加了子域名深度。

特征 3: URL 中的域名

根据数学统计发现, 一些敏感词如“login”、“alibaba”等在恶意网页样本 URL 中的域名中出现的频率较高。

特征 4: URL 中的域后缀

正常网页的域名后缀一般是“.com”、“.net”、“.org”、“.cn”等, 而恶意网页常采用的域名后缀有“.xyz”、“.ml”、“.info”等罕见字符, 如恶意网页“http://www.sesso-gratis.info”中采用的“.info”后缀。

综上所述, 本文选择提取的网页 URL 特征如表 1 所示。

表 1 网页 URL 特征

Table 1 Webpage URL characteristics

特征	内容
URL 中是否包含 IP 地址	这种特征是恶意网页 URL 经常使用的方法, 使用 IP 地址代替域名信息对恶意网页进行伪装
URL 中的子域名	正常网页 URL 的子域名一般较短, 一般为 1
URL 中的域名	恶意网页的域名经常使用一些敏感词汇如“account”、“login”、“alibaba”
URL 中的域名后缀	正常网页的域名后缀一般以“.com”、“.net”、“.org”、“.cn”结尾

在进行特征提取之前, 需要把计算机不能直接识别的词汇、句子等信息转化成数值形式。最常用的方法是基于传统机器学习的 one-hot 编码方式和基于神经网络的词嵌入技术。前者 one-hot 编码, 它是一种极易导致“维数灾难”的稀疏编码方式, 而且这种转换方式无法表现出词汇和词汇之间的相似性, 导致泛化能力不强。词嵌入是自然语言处理中语言模型与表征学习技术的统称。词嵌入技术的出现, 使其可以将一个维数为所有词数量的高维空间嵌入到

一个维数较低的连续向量空间, 并将每个单词或词组映射为实数域上的词向量。解决了 one-hot 编码维度过高的问题, 且该词向量的相似度在一定程度上反应了单词语义上的相似度。

作为词嵌入技术代表的 Word2Vwec(Word to Vector)方法是由 Tomas Mikolov 等人[33]提出的一种用于学习从文本语料库嵌入独立词语的统计方法, 核心思想是基于上下文, 先用向量代表各个词, 然后通过一个预测目标函数学习这些向量的参数, 网络的主体是一种单隐层前馈神经网络, 网络的输入和输出都是词向量。

本文参照词嵌入技术的思想, 针对字符而不是词进行建模。通过预处理将字符映射到 N 维向量空间(N等于URL的最大长度), 将其转为连续值的向量表示, 以适应深度学习模型的特征提取工作。本文对 URL 特征进行预处理的步骤如下:

(1) 对数据集中的 URL 进行解析, 提取出所有 URL 的子域名、域名以及域名后缀, 分别用 sub_domain、domain、domain_suffix 三项来表示, 三者种类数量统计情况如图 3 所示。若三项中存在某项为空值, 则采用“<empty>”填充。对解析后的各类域名、子域名、域名后缀进行统计编号处理, 使得各自对应唯一编号。

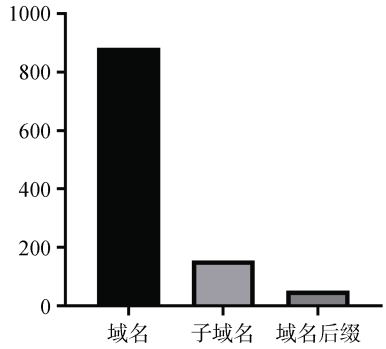

图 3 种类信息统计图

Figure 3 Statistical chart of type information

(2) 将数据集中的每个 URL 属性按字符进行分割。分割后的字符集统一编号作为 URL 词表, 然后对应 URL 词表中每个字符编码将数据集中的 URL 字符串向量化, 转换成对应的向量编码。

(3) 为了特征提取阶段能够提取充足的 URL 特征, 将(1)中提取到的域名编码与(2)中提取到的 URL 文本属性向量进行融合, 由此初步得到能够表示 URL 的特征向量。

(4) 由于初步得到的 URL 特征向量长度不一致, 为此获取数据集 URL 中长度最大值, 对所有长度小于最大长度的URL进行padding操作, 用0作为填充字符。故而将所有 URL 文本编码向量化为最大长度 131 维的特征向量。

如图 4～6 所示, 分别统计了按标签分组的数量前十的域名、子域名、域名后缀。图中纵坐标表示相应数值取对数之后的值, 横坐标则代表相应的域标签名。从图中可以得出, 恶意网页中最常用的域名为“geocities”, 子域名为“www”, 域名后缀为“com”。相比于域名、子域名而言, 恶意网页在域名后缀中的种类较多。

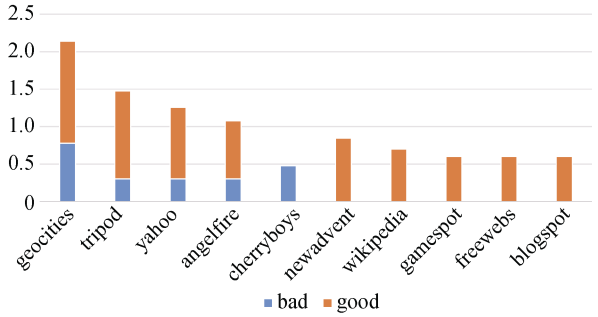

图 4 按标签分组的 TOP-10 Domains

Figure 4 TOP 10 domains grouped by label

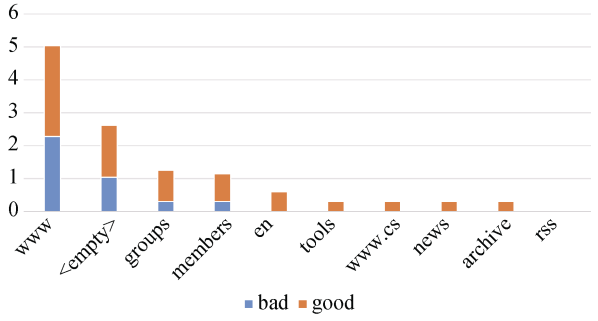

图 5 按标签分组的 TOP-10 Sub_domains

Figure 5 TOP 10 sub_domains grouped by label

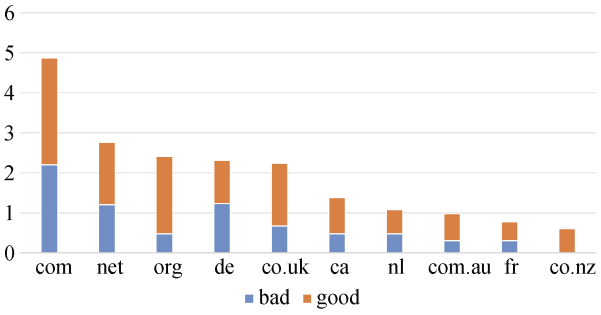

图 6 按标签分组的 TOP-10 domains_suffix

Figure 6 TOP 10 domains_suffix grouped by label

3.2.2 网页文本特征

通过对数据集的恶意网页进行分析, 可以发现在网页的文本属性中, 存在着大量不安全登录、虚假

支付等敏感信息。例如, 在网页源代码中的<a>标签中的文本内容也会包含一些与网络钓鱼、垃圾邮件等明显相关的关键词。所以有必要从网页中抽取文本的特征, 本文对网页文本特征进行预处理如下:

(1) 将文本内容按段落进行切割后, 再对每段文本内容中的无用字符, 如空白字符、“?”、“\”等进行清洗。

(2) 采用字典的形式存储每个单词以及词频, 定义词频数小于 5 的为无用词, 结合自然语言处理工具包 nltk(Natural language toolkit, nltk)中的英文停用词模块, 它包括常见停用词如“and”、“however”、“neither”等, 去掉所有网页文本段落中的无用词。

(3) 遍历所有清洗后的文本数据构建单词库, 并为每个单词映射到唯一 ID 编号。为了充分提取网页文本特征, 利用 nltk 工具包中的“wordnet”模块来查找单词库中所有单词各自的同义词集。由于每个单词不同词性对应不同的定义, 因此需获取每个单词及其同义词的词汇定义, 再将所有词汇定义用空格拼接起来.若定义为空, 就采用“<empty>”填充,采用字典的形式存储每个单词及其对应单词定义。

(4) 将(3)中处理得到的每个单词的词汇定义作为输入, 使用 python 中的向量转换工具 TfidfVectorizer 将文本向量化, 进一步将向量化的文本数据进行标准化, 保证每个维度的特征数据方差为 1, 均值为 0, 使得判别结果不会被某些维度过大的特征值主导。经预处理后将单词库中的所有单词数据均表示为一个 200 维的特征向量。

3.3 特征提取

3.3.1 CNN 提取 URL 特征

Yoon Kim 在 2014 年将卷积神经网络应用到文本分类任务, 该算法通过利用多个大小不同的卷积核来对句子进行特征提取, 从而能够捕获较好的局部信息。卷积神经网络关键在于捕捉局部特征, 对于本文提取的 URL 来说, 局部特征就是若干字符组成的滑动窗口, 提取和合并各种抽象层面的语义信息, 然后充分利用池化层对关键特征进行提取和优化。

本文采用一维卷积神经网络来提取网页的 URL 特征, 按照卷积方式的不同可以将卷积分为: Full 卷积、Same 卷积、Valid 卷积。与二维卷积层一样, 一维卷积层使用一维的互相关运算。一维卷积的互相关计算是施加于两个数组(一个输入 I, 一个卷积核 K)的一种计算。三种不同的卷积计算过程各不相同: Full 卷积计算是将 K 沿着 I 顺序移动, 每移动到一个固定位置, 对应位置的值相乘再求和; Same 卷积计算的卷积核 K 都有一个锚点, 将该锚点顺序移动到张量 I 的每一个位置处, 对应位置值相乘再求和; Valid 卷积计算则只考虑 I 能完全覆盖 K 的情况, 即 K 在 I 内部移动。为了获取更多有效特征, 本文一维卷积模型采用的是 Valid 卷积计算方式来对网页 URL 特征进行提取, 计算过程如图 7 所示。

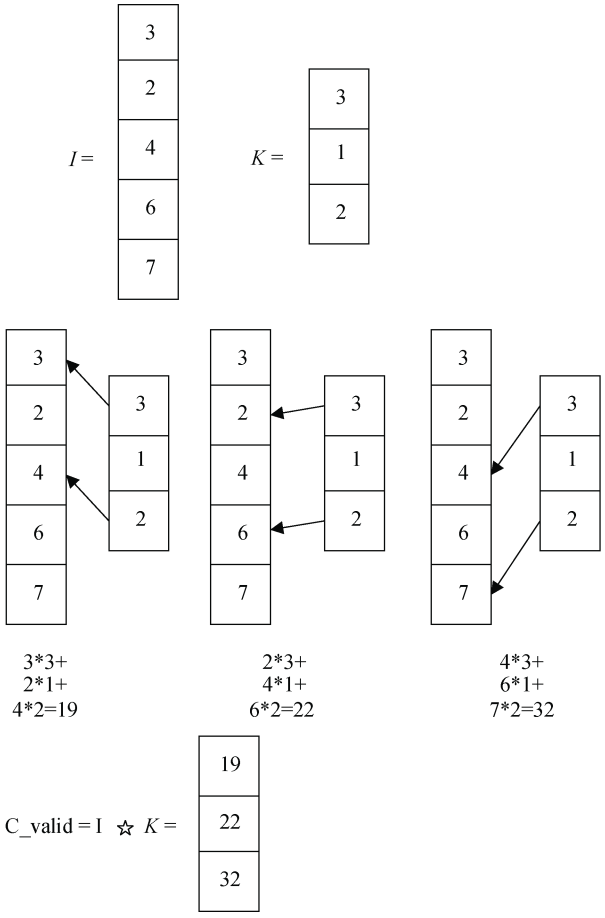

图 7 Valid 卷积计算过程图

Figure 7 Valid convolution calculation process diagram

(注: 图中星形符号代表卷积)

本文设计的一维卷积神经网络模型包括三层卷积层、三层池化层、一层 Dropout 层、两层全连接层。经实验验证, 使得卷积模型提取特征效果达到最佳的设置如下: 三层卷积层分别选取 7, 5, 3 不同窗口大小作为卷积核, 且步长设置为 1; 三层池化层为了获得最优解, 前两层采用 MaxPooling 技术进行下采样, 最后一层则采用 GlobalAveragePooling 技术; Dropout 层则是通过随机移除神经网络中的一些神经元, 来缓解 CNN 模型训练过程中发生过拟合现象, dropout 率设置为 0.5。本文设计的一维卷积神经网络模型结构如图 8 所示。

3.3.2 GCN 提取网页文本特征

图卷积神经网络(GCN)在 2017 年由 Kipf 和

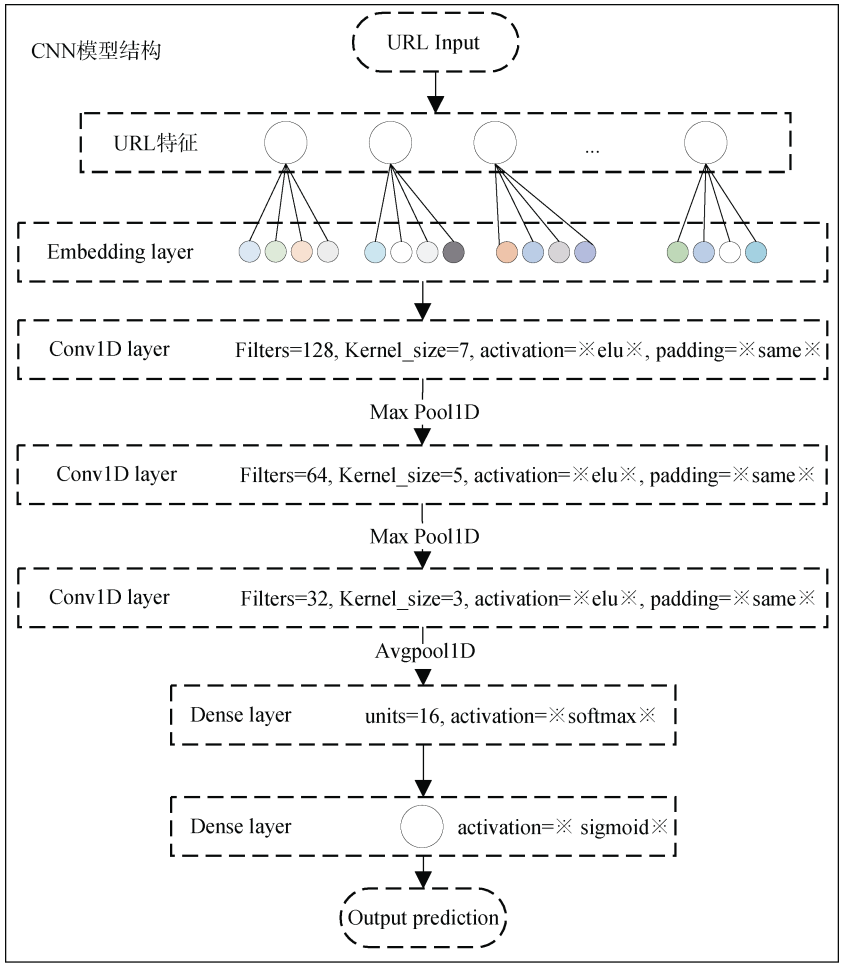

图 8　CNN 模型结构图

Figure 8　CNN model structure diagram

Welling 首次提出, 是一种多层神经网络。实际上与 CNN 作用一样, 是一个特征提取器。GCN 巧妙的设计了一种从图数据中提取特征的方法, 该模型通过卷积运算将节点的邻域信息聚合到自身节点, 经过多次聚合能够提取到高阶的邻域信息和节点间非连续的依赖关系, 从而可以使用这些特征对图数据进行节点分类、图分类、边预测等。本文采用 GCN 对网页文本内容进行特征提取, 将文本分类任务转变为节点分类任务。

若将图定义为: $G=(V,E)$, 其中 $V(|V|=n)$和 E 分别表示节点和边的集合。假设每个结点连接到自身, 则 $(v,v)\in E$ 对与任何节点 v 都成立。让 $X\in R^{n\times m}$ 代表包含 n 个节点及其特征的矩阵, 其中 m 是每个节点的特征向量的维度。引入 G 的邻接矩阵 $\boldsymbol{A}$ 及其度矩阵 $\boldsymbol{D}$, 其中 $D_{ii}=\sum_j A_{ij}$ 。在 $\boldsymbol{A}$ 矩阵中所有对角元素都置为 1, 是因为每个节点自循环的原因。一层 GCN 只能通过一层卷积来捕获关于近邻的信息。单层 GCN 捕获新的 k 维节点特征矩阵 $\boldsymbol{L}^{(1)}\in R^{n\times k}$ 计算为:

$$\boldsymbol{L}^{(1)}=\rho\left(\tilde{\boldsymbol{A}}X W_0\right) \tag{5}$$

其中, $\tilde{\boldsymbol{A}}=D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ 是归一化对称矩阵, $\boldsymbol{W}_0$ 是一个权重矩阵。$\rho(\cdot)$表示的是非线性激活函数, 例如, ReLU $\rho(x)=\max(0,x)$ 。当堆叠多个 GCN 层时, 就可以集成更多邻域信息。表达式为:

$$L^{(j+1)}=\rho\left(\tilde{A}L^{(j)}\boldsymbol{W}_j\right) \tag{6}$$

其中, $L^{(0)}=X$, $\boldsymbol{W}_j$ 代表第 j 层单词与单词节点之间边的权重矩阵, 该权重矩阵中单词结点之间边的权重 W, 采用逐点互信息(Pointwise Mutual Information, PMI)来衡量, 如$\left(v_1, v_2\right)$之间边的权重即为 $W=PMI\left(v_1, v_2\right)$。

图卷积神经网络通过其文本图中节点之间的边可以迅速增加模型的感受野, 从而捕获文本长远距离的特征, 弥补了卷积神经网络的不足。如图 9 所示, 构建了一个包含单词节点和 URL 序号节点的大型文本异构图。图中深色节点表示的是恶意网页 URL, 浅

色则为良性 URL。

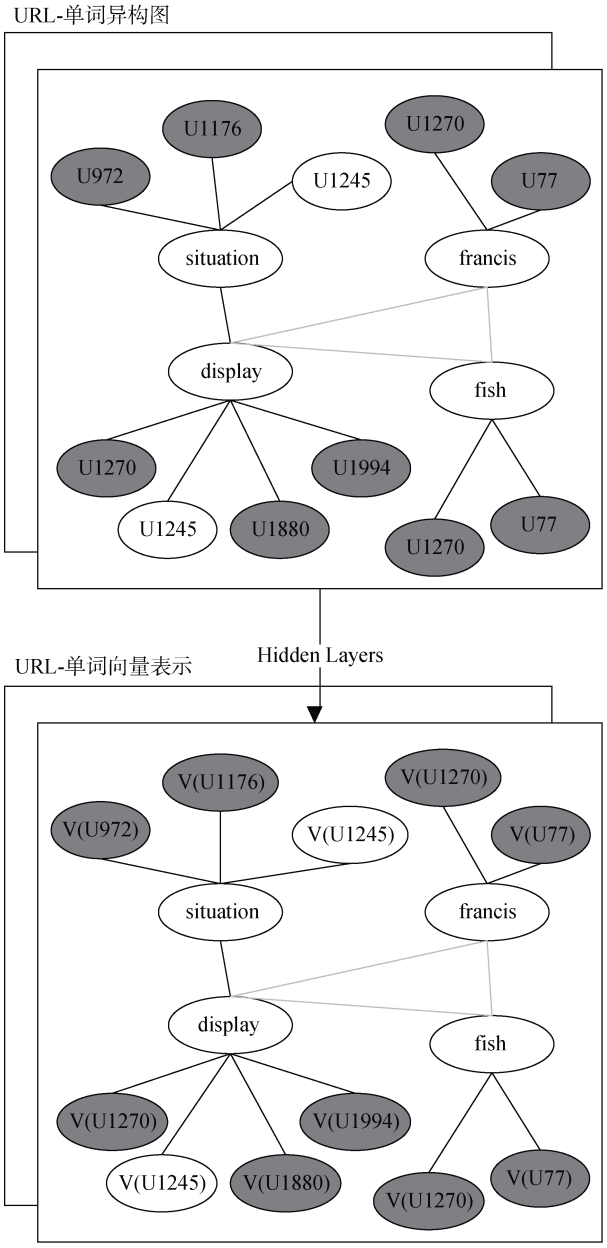

图 9　URL-Word 异构图

Figure 9　URL-Word heterogeneous graph

(注: 以"U"开头的节点代表的是 URL 的序号节点, 其他的都是单词节点。黑色的边表示 URL 与 Word 之间的边, 灰色的边表示 Word 与 Word 之间的边。*V*(*x*)表示对 *x* 进行 Embedding.)

本文设计的图卷积神经网络模型结构如图 10 所示。模型包括: 两层图卷积层、两层全连接层、一层 SoftMax 层。

在图中, 节点|*V*|的数目是词汇表的单词数和 URL 的数目, 总共 4437 个。首先, 设定特征矩阵 ***X*** 设置为单位矩阵, 即 ***X***=*I*。这意味着每一个单词或者 URL 都可以被视为用于 GCN 的 one-hot 编码的向量。上图结构中, 有两类节点: 单词节点和 URL 序号节点, 基于两种节点之间构建图结构的边, 分别是单词与单词节点之间的边和单词与 URL 序号节点之间的边。单词与单词节点之间的边取决于单词在滑动窗口中是否共现; 单词与 URL 序号节点之间的边则是由单词是否在该 URL 的网页文本中出现决定的。将每一个 URL 的文本内容看作一份文档, URL 序号节点和单词节点之间的边的权重是由该 URL 文本内容中单词的词频-逆文档频率(Term frequency-Inverse document frequency, TF-IDF)来表示, 它可以用来评估一个词对语料库中一份文档的重要程度。主要思想是: 如果某个单词在一篇文章中出现的频率高且在其他文章中很少出现, 则认为该词具有很好的类别区分能力, 适合用来分类。

词频(Term Frequency,TF)是单词在该文档中出现的次数, 若将每个 URL 定义为 U_j ,则该 URL 中某个词 *n* 的词频计算为:

$$TF_{ij}=\frac{n_{i,j}}{\sum_k n_{k,j}} \tag{7}$$

其中, $n_{i,j}$ 是该词在 U_j 中出现的次数, 分母则是 U_j 中所有词汇出现的次数总和。

逆文档频率(Inverse Document Frequent, IDF)是所有文档总数与包含该单词的文档数比值的对数。如果包含该词 *x* 的文档越少, IDF 值越大, 则说明该词具有很好的类别区分能力。

$$IDF_i=\log\frac{|D|}{1+\left|\left\{j:x_i\in d_j\right\}\right|} \tag{8}$$

其中, $|D|$ 是语料库中的文件总数, $\left|\left\{j:x_i\in d_j\right\}\right|$ 表示包含词语 x_i 的 URL 数目, 如果该词不再语料库当中就会导致分母为零, 因此一般情况下使用 1+$\left|\left\{j:x_i\in d_j\right\}\right|$ 来做分母。某一特定网页 URL 内文本的高词语频率, 以及该词语在整个语料库中的低文件频率, 可以得出较高权重的 TF-IDF。因此, TF-IDF 倾向于过滤掉常见的词语, 保留重要的词语, 计算公式为:

$$TF-IDF=TF\times IDF \tag{9}$$

为了利用全局单词共现信息, 在语料库中的所有文档上使用固定大小的滑动窗口来收集共现统计信息, 本实验中设置窗口大小为 20。本文用来衡量单词结点间边权重的逐点互信息方法是一种词关联度量, 表示两个变量 $word_1$ 和 $word_2$ 是否有关系, 以及关系的强弱。因此, 本文用来计算两个词节点之间的权重, 具体计算公式如下:

$$PMI(word_1, word_2)=\log\frac{P(word_1, word_2)}{P(word_1)P(word_2)} \quad (10)$$

$$P(word_1, word_2)=\frac{W(word_1, word_2)}{W} \quad (11)$$

$$P(word_1)=\frac{W(word_1)}{W} \quad (12)$$

其中, $W(word_1)$是指包含单词 $word_1$ 的滑动窗口数量, $W(word_1, word_2)$是指同时包含单词 $word_1$ 和单词 $word_2$ 的滑动窗口数量, W 则是语料库中滑动窗口的总数。PMI 为正值表示语料库中单词的语义相关性较高, 若为负值则意味着语料很少或者没有语义相关性。

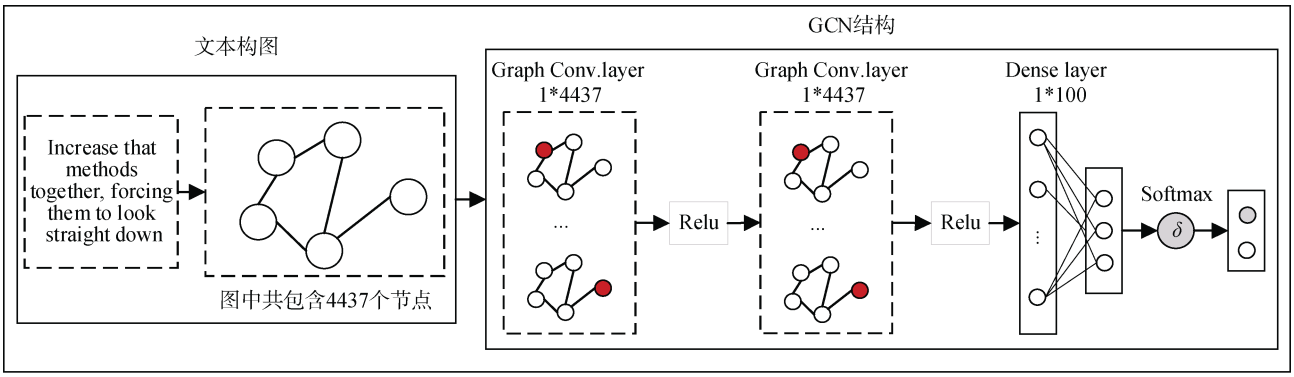

图 10 GCN 模型结构图

Figure 10 GCN model structure diagram

综上所述, 本文设计的文本异构图结构信息如表 2 所示。

表 2 异图构结构信息

Table 2 Heterogeneous graph information

图结构	内容
单词节点	每个 URL 当中文本内容的所有单词作为一类节点
URL 序号节点	每个 URL 序号作为一类节点
单词-单词节点边	在统一窗口共现的单词之间的边
URL 序号节点-单词节点边	在该 URL 文本内容中出现的单词与该 URL 的序号之间的边
单词-单词边权重	单词节点之间边的权重采用 PMI 来衡量
单词-URL 边权重	单词节点与 URL 之间边的权重采用 TF-IDF 来衡量

3.4 网页分类

支持向量机(SVM)的基本思想是: 通过非线性变换将输入空间变换到一个高维度的特征空间, 并在该空间中寻找最优的线性分界面。无论是基于支持向量机的二分类器还是多分类器, 都只能进行监督学习, 主要适用于非线性、高维度的分类问题。SVM 存在硬间隔和软间隔两种方法。前者一般是在理想的线性分类方式下存在, 由于噪音的存在可能会使一部分的点不满足约束条件, 进而会造成最终的分类结果较差。因此采用软间隔来进行分类, 将原有的硬划分变为一种软分类的思想。在原有目标的基础上添加一个 loss 函数, 代表 SVM 允许超平面两侧存在少量异常分类点, 从而降低超平面分割的经验风险, 避免由于少量特殊伪装的恶意网页影响全局分类效果。

在使用支持向量机解决实际问题时, 核函数的选择相当关键, 需要根据具体的问题来构造相应的核函数。但是, 在众多核函数中, 高斯核函数(Gaussian kernel)通常能非常准确地描绘数据的分布结构。因此本文在完成分类工作时采用的也是该核函数, 其表达式如公式 6 所示。

$$K(x_1, x_2)=e^{-q(\|x_1-x_2\|)^2} \quad (13)$$

其中 q 为核函数的宽度。

本文采用 SVM 分类器取代了传统神经网络中的 softmax 层来进行分类任务。softmax 需要计算每个类别的得分, 归一化为概率, 对于词汇量较大的语料库来说计算量超大, 且 softmax 对输入数据十分敏感, 对于输入数据无论大小的改变, 结果都会相应发生变化, 而 SVM 可以有效解决高维度数据的分类问题以及避免因为少量数据的改变而影响全局的分类效果。利用 SVM 分类性能的优点可以有效将 URL 局部特征信息和网页文本全局特征信息融合后的高维度特征向量作为分类器的输入, 经 SVM 分类器得出网页是否为恶意网页。经实验验证, 本文提出的方法在一定程度上能够提高网页识别准确率。

4 实验与分析

4.1 数据集介绍

模型的训练、评估和验证都需要依赖有代表性的数据集。本文从 *Malicious and Benign Webpages*

Dataset 论文[34]中提供的公开样本数据集共获取到约 150 万个网页数据, 该数据集由名为 MalCrawler[35] 的网络爬虫进行收集, 包含了关于网页 URL、IP 地址、JavaScript 等方面的丰富属性以及恶意或者良性分类标签, 常用于恶意网页监督及非监督学习相关研究。该数据集的具体属性信息如表 3 所示。

表 3 数据集属性信息

Table 3 Attributes of dataset

属性名	描述
URL	每个网页的 URL
URL_len	每个网页的 URL 的长度
IP_add	承载每个网页的 web 服务器的 IP 地址
Geo_loc	每个 IP 地址所属的国家(该数据集表示来自全球服务器的网页)
Js_len	网页中 JavaScript 代码的长度
Tld	网页的顶级域名
Who_is	给出注册域名的 WHOIS 信息是否完整
Https	给出网站是否使用 HTTPS 或 HTTP 协议
Content	原始网站的内容, 包括过滤和处理后的文本以及 JavaScript 代码
Label	网页为良性或者恶意的分类标签

本文采集了该数据集中 1000 条有效数据, 所用均为筛选后的真实 URL 数据及网页源码。其中良性网页与恶意网页占比如图 11 所示。将原始数据集按 8∶2 的比例分割成训练集和验证集, 再将训练集按 8∶2 的比例分割成正式训练集和预测集用于模型训练。

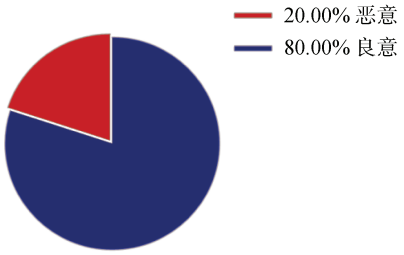

图 11 良性和恶意网页占比图

Figure 11 Proportion of benign and malicious web pages

4.2 评估标准

为评估模型的性能, 本文实验选择将混淆矩阵(Confusion matrix)作为评估标准。在机器学习中, 对于二分类问题常采用的是正确分类为良性的样本数 TP(True Positives)、错误分类为恶意的良性样本数 FP(False Positives)、错误分类为良性的恶意样本数 FN(False Negatives)、正确分类为良性的良性样本数 TN(True Negatives)四个指标, 混淆矩阵的表示如表 4 所示。

表 4 混淆矩阵表示

Table 4 Heterogeneous graph information

混淆矩阵		真实值	
		Positive	Negative
预测值	Positive	TP	FP
	Negative	FN	TN

混淆矩阵里面统计的是预测的数量值, 为了进行标准化的衡量, 在基本统计结果上计算准确率、精确率、召回率、F1 值四个参数指标。将混淆矩阵中数量值转化为取值在 0～1 之间的比率。

准确率(Accuracy, *Acc*)表示正确分类的测试实例的个数占测试实例总数的比例, 计算公式为:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

精确率(Precision, *P*)也叫查准率, 表示正确分类的正例个数占分类为正例的实例个数的比例, 计算公式为:

$$P = \frac{TP}{TP + FP} \tag{15}$$

召回率(Recall, *R*)也叫查全率, 表示正确分类的正例个数占实际正例个数的比例, 计算公式为:

$$R = \frac{TP}{TP + TN} \tag{16}$$

F1-score 是基于召回率与精确率的调和平均, 即将召回率与精确率综合起来评价, 计算公式为:

$$F_1 = \frac{2TP}{2TP + FN + FP} \tag{17}$$

除了上述的评估指标外, 采用 ROC(Receiver Operating Characteristic)曲线, 以减少样本库中由于数据不均衡造成的影响。ROC 曲线是建立在信息检出理论的基础上, 广泛用于数理统计方法当中[36], 其基本原理是根据一系列不同的阈值或者分界值, 以 TPR(True Positive Rate)为纵坐标, FPR(False Positive Rate)为横坐标绘制的曲线, 曲线下面积 AUC (Area Under the Curve)越大, 代表算法精确度越高。

TPR 又称敏感度, 表示在所有实际为阳性的样本中, 被正确地判断为正的比率, 计算公式为:

$$TPR = \frac{TP}{TP + FN} \tag{18}$$

FPR 又称特异度, 表示在所有实际为阴性的样本中, 被错误地判断为阳性的比率, 计算公式为:

$$FPR = \frac{FP}{FP + TN} \tag{19}$$

AUC 被定义为 ROC 曲线下与坐标轴围成的面积, 表示的是分类器的平均性能, 使用该值可以评

估二分类问题效果的优劣。AUC 越接近 1.0, 对应分类器的效果越好[37], 计算公式为:

$$AUC = \frac{1 + TPR - FPR}{2} \tag{20}$$

4.3 结果与分析

4.3.1 参数敏感性分析

本文选择对模型效果影响较大的参数包括学习率的大小、Epochs(使用训练集的全部数据对模型进行完整训练的次数)的大小来进行分析。采用控制变量法每次只改变其中的某一个参数而控制其他参数大小不变, 从而研究改变的参数对实验结果的影响。经实验验证, 学习率大小以及 Epochs 大小对本文模型实验结果的影响如表 5 和表 6 所示。

表 5 不同学习率大小测试(Epochs=200)

Table 5 Test with different learning rate sizes

学习率	精确率	召回率	F1-score	Accuracy
0.1	0.9219	0.8600	0.8853	0.9200
0.01	0.9257	0.8700	0.8933	0.9250
0.001	0.9181	0.8500	0.8772	0.9150

表 6 不同 Epochs 大小测试(学习率 0.01)

Table 6 Test with different epoch size

Epochs	精确率	召回率	F1-score	Accuracy
100	0.9144	0.8667	0.8871	0.9200
200	0.9257	0.8700	0.8933	0.9250
300	0.9100	0.8556	0.8773	0.9010

从表 5 中可以得出, 在其他参数保持不变的情况下, 将学习率设置为 0.01 时准确率是最高的。同理, 从表 6 中得知在取得使准确率达到最高的学习率情况下, Epochs 值设置为 200 是最佳的。取得使模型取得最佳值情况下的参数值, 为下一步对比实验的进行提供依据。

4.3.2 实验结果分析

在参数敏感度分析的基础上, 对实验中的各个参数进行设置, 学习率置为 0.01, Epochs 置为 200。为了分析本文所提出模型分类准确性能的高低, 实验中设置了 KNN、朴素贝叶斯、逻辑回归、决策树 4 种浅层机器学习方法作为实验对照组。实验选取了数据集中 80%的数据作为训练集, 再将训练集的数据按 8∶2 的比例拆分成正式训练集和测试集来测试各个分类算法, 取多次实验的均值作为结果, 实验结果如表 7 所示。

从实验结果可以看出, 基于本文提出的组合模型对网页进行识别的准确率达到 92.5%, 与其他几种不同的浅层机器学习算法相比, 分类效果是最好的, 准确率分别提高了 5%、3.5%、3.45%、4%, 证明了本文所提出模型的有效性。

表 7 实验结果

Table 7 Experimental results

实验算法	精确率	召回率	F1-score	Accuracy
KNN	0.8606	0.7900	0.8164	0.8750
朴素贝叶斯	0.8692	0.8267	0.8447	0.8900
逻辑回归	0.8987	0.8000	0.8342	0.8905
决策树	0.9184	0.7767	0.8186	0.8850
本文模型	**0.9310**	**0.8533**	**0.8556**	**0.9250**

为了更直观的从精确率、召回率、F1 值方面比较分析上述 5 种模型, 实验结果如图 12 所示。纵坐标表示模型得分, 值越大代表模型性能越好。在采集的数实验据集上, 本文模型的精确率达到了 93.1%, 召回率达到了 85.33%, F1 值达到了 85.56%, 本文模型整体达到了不错的效果。与前四个模型相比, 本文模型在精确率上分别提高了 7.04%、6.18%、3.23%、1.26%; 召回率分别提高了 6.33%、2.66%、5.33%、7.66%; F1 值分别提高了 3.92%、1.09%、2.14%、3.7%。

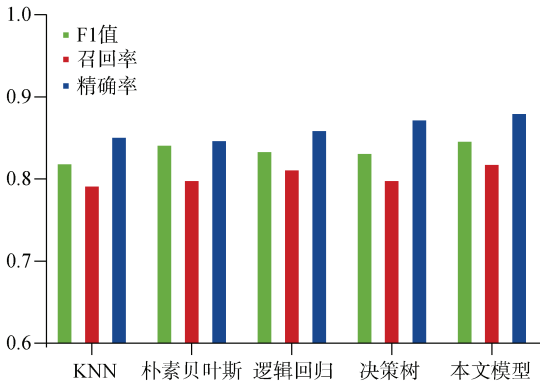

图 12 5 种模型的精确率、召回率、F1 值

Figure 12 Precision, recall, and F1 values for 5 models

为了进一步验证评估结果的客观性与真实性, 绘制了本文模型及四个对比模型的 ROC 曲线图, 如图 13 所示。通过计算得出, 本文模型的 AUC 值为 0.8324, 4 个对比模型: KNN 为 0.8009、逻辑回归为 0.7943、决策树为 0.7566、朴素贝叶斯为 0.7473。根据 AUC 值的大小可以判断, 本文提出模型的 AUC 值最大, 因此分类效果是最佳的。

对于深度学习算法, 本文选取 TextCNN[38], TextRCNN[39], BiLSTM-Attention[40], TextRNN[41] 4 种

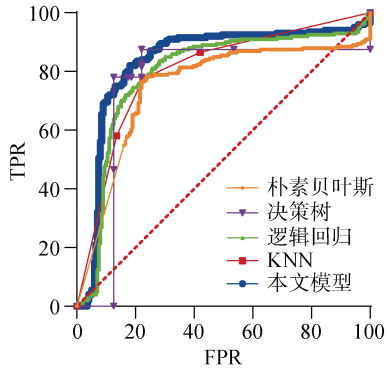

图 13　5 种模型的 ROC 曲线图

Figure 13　ROC curve of 5 model

常用经典的深度学习模型作为对比, 并采用本文预处理后的同一数据集用于上述 4 种模型进行实验, 实验结果如表 8 所示。

表 8　深度学习算法对比实验结果

Table 8　Deep learning algorithms compare experimental results

深度学习算法	精确率	召回率	F1-score	Accuracy
TextCNN	0.7658	0.7826	0.7727	0.7758
TextRNN	0.8102	0.7374	0.8482	0.7907
BiLSTM-Attention	0.8472	0.7658	0.8658	0.8206
TextRCNN	0.8508	0.7784	0.8674	0.8266
本文模型	**0.9310**	**0.8533**	**0.8556**	**0.9250**

由上表可知, 在采用与本文同一数据集的情况下, 本文提出模型的实验效果对比其他 4 种经典深度学习模型, 无论是在准确率、精确率还是召回率上都具有一定的优势。其中, 准确率分别提升了 14.92%、13.43%、10.44%、9.84%; 在精确率上分别提升了 16.52%、12.08%、8.38%、8.02%; 在召回率上分别提升了 7.07%、11.59%、8.75%、7.49%, 从而体现出本文模型的有效性。

4.4　消融实验

为了从多个维度衡量本文提出模型有效性, 对本文提出的模型进行消融实验, 实验结果如表 9 所示。从表中可以观察到, 删除一部分模块的模型相比较于总模型在性能上均存在一定程度的下降。例如, 本文提出模型分别与 CNN+SVM 模型和 GCN+SVM 模型相比, 准确率均有所提高, 尤其对于GCN+SVM结合的模型准确率提升将近 15%左右。一方面, 说明了相对于单一网络模型而言, 本文提出的组合网络模型能够提升恶意网页识别的准确率; 另一方面, 说明了 CNN 模型在一定程度上确实丰富了 GCN 的局部特征信息, 使得本文模型能够提取较为全面的网页特征信息用于分类器进行分类, 进一步提高分类准确度。

表 9　消融实验结果

Table 9　Ablation experimental results

实验算法	精确率	召回率	F1-score	Accuracy
GCN	0.7740	0.5001	0.8726	0.7740
CNN	0.8225	0.6254	0.6496	0.8225
CNN+SVM	0.9305	0.8167	0.8834	0.9200
GCN+SVM	0.8769	0.5100	0.4494	0.7550
本文模型	**0.9310**	**0.8533**	**0.8556**	**0.9250**

4.5　模型讨论

恶意网页识别一直是网络安全领域中一个比较棘手的问题。在对国内外相关文献进行大量查阅、研究的基础上, 提出了文本的识别模型并进行了相关实验工作。虽然本文提出模型在一定程度上弥补了传统机器学习以及单一网络模型的不足, 但是本文模型仍然存在一定的局限性。

(1) 网页数据获取方面: 网页数据每时每刻都在更新, 面对实时的新增数据以及网页文本的多样性、网页内容因被加密而无法直接获取、网页内容被恶意作者故意替换以逃避特征检测等诸多问题。

(2) 模型训练方面: 深度神经网络的训练过程中存在着许多超参数, 例如激活函数、损失函数、迭代次数的等。参数的调节是训练过程中一个非常重要的步骤, 对最后的实验结果有较大的影响。目前还没有一个很好的系统性的指导方法, 仅仅是在其他相关研究工作的基础上的经验总结。

5　结语

本文从提高恶意网页识别的准确率和效率等角度出发, 提出了基于深度学习与特征融合的恶意网页识别方法并进行了设计与实现。成功将图卷积神经网络应用在恶意网页检测识别领域, 突破了传统恶意网识别方法的局限性。本文方法有效结合深度学习与机器学习, 不仅解决了深度学习中依靠单一网络模型效率不高的问题, 同时解决了传统机器学习分类算法需要大量人工操作耗时的问题和因为数据量庞大容易造成维度灾难的问题。本文实验收集公开数据集中良性及恶意网页样本, 有效提取了网页特征数据进行分析, 与已有的多种方法进行对比, 实验结果证明了本文方法的有效性。

在后续的研究中, 针对本模型的局限性采取相应措施: 1) 本文提出的识别模型还需要做进一步的扩展, 可以考虑增加特征维度, 使其能够通过增量

学习的方式不断对实时的新特征提取; 2) 不断地进行模型调优实验, 持续优化现有的分类算法来进一步提高模型的识别效率, 并能够在现有的基础上对恶意网页类型进行更细化的分类。

参考文献

[1] Zhang Xiaona. The 50th "Statistical Report on Internet Development in China" was released[N]. Democracy and Legal Times, 2022-09-02(001).
(张晓娜. 第 50 次《中国互联网络发展状况统计报告》发布[N]. 民主与法制时报, 2022-09-02(001).)

[2] 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.
(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.)

[3] The biggest data breaches, hacks of 2021. https://www.zdnet.com/article/the-biggest-data-breaches-of-2021/. Dec, 2021.

[4] The LinkedIn Breach Exposes Nearly 700 million people. https://www.datalyst.net/blog/the-linkedin-breach-exposes-nearly-700-million-people. July, 2021.

[5] Sumit Sahu, Bharti Dongre, Rajesh Vadhwani. Web Spam Detection Using Different Features[J]. *International Journal of Soft Computing and Engineering,* July 2011, 1(3):70-73.

[6] Ma J, Saul L K, Savage S, et al. Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs[C]. *The 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009: 1245-1253.

[7] Ma J, Saul L K, Savage S, et al. Identifying Suspicious URLs: An Application of Large-Scale Online Learning[C]. *The 26th Annual International Conference on Machine Learning*, 2009: 618-688.

[8] Ma J, Saul L K, Savage S, et al. Learning to Detect Malicious URLs[J]. *ACM Transactions on Intelligent Systems and Technology*, 2(3): 30.

[9] Canali D, Cova M, Vigna G, et al. Prophiler: A Fast Filter for the Large-Scale Detection of Malicious Web Pages[C]. *The 20th international conference on World wide web*, 2011: 197-206.

[10] Wang R, Zhu Y, Tan J F, et al. Detection of Malicious Web Pages Based on Hybrid Analysis[J]. *Journal of Information Security and Applications*, 2017, 35: 68-74.

[11] Khonji M, Iraqi Y, Jones A. Phishing Detection: A Literature Survey[J]. *IEEE Communications Surveys & Tutorials*, 2013, 15(4): 2091-2121.

[12] Jain A K, Gupta B B. Towards Detection of Phishing Websites on Client-Side Using Machine Learning Based Approach[J]. *Telecommunication Systems*, 2018, 68(4): 687-700.

[13] Liang B, Huang J J, Liu F, et al. Malicious Web Pages Detection Based on Abnormal Visibility Recognition[C]. *2009 International Conference on E-Business and Information System Security*, 2009: 1-5.

[14] Zhang H L, Zou W, Han X H. Drive-by-Download Mechanisms and Defenses[J]. *Journal of Software*, 2013, 24(4): 843-858.
(张慧琳, 邹维, 韩心慧. 网页木马机理与防御技术[J]. *软件学报*, 2013, 24(4): 843-858.)

[15] Zhang W F, Liu R C, Xu L. Web Page Trojan Detection Method Based on Dynamic Behavior Analysis[J]. *Journal of Software*, 2018, 29(5): 1410-1421.
(张卫丰, 刘蕊成, 许蕾. 基于动态行为分析的网页木马检测方法[J]. *软件学报*, 2018, 29(5): 1410-1421.)

[16] Sun J C. A Classification Method of Web Page Using Machine Learning[J]. *Netinfo Security*, 2017(9): 45-48.
(孙靖超. 一种基于机器学习的网页分类技术[J]. *信息网络安全*, 2017(9): 45-48.)

[17] Yatagai T, Isohara T, Sasase I. Detection of HTTP-GET Flood Attack Based on Analysis of Page Access Behavior[C]. *2007 IEEE Pacific Rim Conference on Communications*, *Computers and Signal Processing*, 2007: 232-235.

[18] Tripathi N, Hubballi N. Slow Rate Denial of Service Attacks Against HTTP/2 and Detection[J]. *Computers & Security*, 2018, 72: 255-272.

[19] Sahoo D, Liu C H, Hoi S C H. Malicious URL Detection Using Machine Learning: A Survey[EB/OL]. 2017: arXiv: 1701.07179. http://arxiv.org/abs/1701.07179.pdf.

[20] PhishTank[DB/OL]. [2020-12-23]. http://www.phishtank.com. 2019,04.

[21] Prakash P, Kumar M, Kompella R R, et al. PhishNet: Predictive Blacklisting to Detect Phishing Attacks[C]. *2010 Proceedings IEEE INFOCOM*, 2010: 1-5.

[22] Zouina M, Outtaj B. A Novel Lightweight URL Phishing Detection System Using SVM and Similarity Index[J]. *Human-Centric Computing and Information Sciences*, 2017, 7(1): 17.

[23] Jeeva S C, Rajsingh E B. Intelligent Phishing Url Detection Using Association Rule Mining[J]. *Human-Centric Computing and Information Sciences*, 2016, 6(1): 64.

[24] Kang W, Qiu H Z, Jiao D D, et al. Search-Based Short-Text Classification[J]. *Application of Electronic Technique*, 2018, 44(11): 121-123, 128.
(康卫, 邱红哲, 焦冬冬, 等. 基于搜索的短文本分类算法研究[J]. *电子技术应用*, 2018, 44(11): 121-123, 128.)

[25] Lin Y S, Jiang J Y, Lee S J. A Similarity Measure for Text Classification and Clustering[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2014, 26(7): 1575-1590.

[26] Tan S B. An Effective Refinement Strategy for KNN Text Classifier[J]. *Expert Systems with Applications*, 2006, 30(2): 290-298.

[27] Wang C X, Zhang T, Ma C S. Improved SVM-KNN Algorithm for Imbalanced Datasets Classification[J]. *Computer Engineering and Applications*, 2016, 52(4): 51-55, 103.
(王超学, 张涛, 马春森. 改进 SVM-KNN 的不平衡数据分类[J]. *计算机工程与应用*, 2016, 52(4): 51-55, 103.)

[28] Vapnik V, Izmailov R. Reinforced SVM Method and Memorization Mechanisms[J]. *Pattern Recognition*, 2021, 119: 108018.

[29] Introduction to Multilayer Perceptron (MLP). https://blog.csdn.net/fg13821267836/article/details/93405572.

[30] Lu X, Du Y, Xu S J, et al. Research Progress on Semantic Segmentation Based on Convolutional Neural Networks[C]. Proceedings of the 24th Annual Conference on New Network Technologies and Applications in 2020, *China Computer Users Association Network Application Branch,* 2020: 22-26.

(鹿鑫, 杜煜, 徐世杰,等. 基于卷积神经网络的语义分割研究进展[C]. *中国计算机用户协会网络应用分会 2020 年第二十四届网络新技术与应用年会论文集*, 2020: 22-26.)

[31] Goodfellow I, Bengio Y, Courville A. Deep learning (Vol. 1): Cambridge: MIT Press, 2016: 367-415.

[32] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. *Neural Computation*, 1997, 9(8): 1735-1780.

[33] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[EB/OL]. 2013: arXiv: 1301.3781. http://arxiv.org/abs/1301.3781.pdf.

[34] Singh A K. Malicious and Benign Webpages Dataset[J]. *Data in Brief*, 2020, 32: 106304.

[35] Singh A K, Goyal N. MalCrawler: A Crawler for Seeking and Crawling Malicious Websites[C]. *International Conference on Distributed Computing and Internet Technology*, 2017: 210-223.

[36] Shi H S. Comparative Research of the ROC Curve Drawing Based on Case and MATLAB[J]. *Electronic Design Engineering*, 2010, 18(9): 36-39.

(石昊苏. 基于实例与 MATLAB 的 ROC 曲线绘制比较研究[J]. *电子设计工程*, 2010, 18(9): 36-39.)

[37] Li Ziyan. Introduction and Application of ROC Curve under the background of Big Data [J]. *Science and Education Guide*, 2021. (李子言. 大数据背景下 ROC 曲线介绍与应用[J]. *科教导刊*, 2021.)

[38] Yoon K. Convolutional Neural Networks for Sentence Classification[J]. *CoRR*, 2014, abs/1408.5882.

[39] Lai S W, Xu L H, Liu K, et al. Recurrent Convolutional Neural Networks for Text Classification[C]. *The Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015: 2267-2273.

[40] Liu J M, Zhang Y. Attention Modeling for Targeted Sentiment[C]. *The 15th Conference of the European Chapter of the Association for Computational Linguistics*: *Volume 2*, *Short Papers*, 2017: 572-577.

[41] Pengfei Liu, Xipeng Qiu, Xuanjing Huang. Recurrent Neural Network for Text Classification with Multi-Task Learning[J]. *CoRR*,2016, abs/1605.05101.

杨胜杰 于 2014 年在湖南大学电气与信息工程学院获得博士学位博士。现为湖南工商大学副教授, CCF 会员。研究领域为机器学习及其应用。研究兴趣包括: 复杂系统仿真、优化计算以及区块链系统。Email: ysj427@163.com。

刘建刚 于 2015 年在中南大学信息学院获得博士学位。现任湖南工商大学副教授, 研究领域为协同控制理论及其应用。Email: jgangliu@csu.edu.cn。

陈朝阳 于 2020 年在长沙大学计算机工程与应用数学学院专业获得学士学位。现在湖南工商大学电子信息专业攻读硕士学位, 研究领域为深度学习、恶意网页识别。Email: czy_0811@163.com。

徐逸 于 2020 年在南京理工大学泰州科技学院移动互联网学院获得学士学位。现在湖南工商大学电子信息专业攻读硕士学位。研究领域为用户用电行为心底及其负荷预测。Email: 1492974489@qq.com