第50卷第7期 中国科学技术大学学报 Vol. 50, No. 7
2020年7月 JOURNAL OF UNIVERSITY OF SCIENCE AND TECHNOLOGY OF CHINA Jul. 2020
文章编号:0253-2778(2020)07-1019-07

基于CNN的假冒域名识别方法研究

杜淑颖[1,2],杜 鹏[1],丁世飞[1]
(1. 中国矿业大学计算机科学与技术学院,江苏徐州 221116;
2.徐州生物工程职业技术学院信息管理学院,江苏徐州 221000)

摘要:近年来,以僵尸网络为载体的各种网络攻击活动是目前互联网面临的安全威胁之一,各种恶意软件使用域名生成算法(domain generation algorithm, DGA)自动生成大量伪随机域名以连接到命令和控制服务器.为此提出以基于卷积神经网络(CNN)的方法来检测和分类伪随机域名.简要介绍了僵尸网络的危害、基本原理以及假冒域名在僵尸网络中的作用.在分析DGA算法的原理以及传统的DGA域名识别算法的缺陷以后,将重点放在基于卷积神经网络的假冒域名识别方法研究.阐述了关于卷积神经网络的基本概念,模拟了在不同的超参数,不同的激励函数下模型对于解决分类问题效果的差异.分析了数据预处理的原理、模型定义中对于超参数和激励函数、学习速率等选择的合理性.在模型运行结果分析时,给出了卷积神经网络模型识别域名的准确率和损失函数的变化,使用准确率、召回值、F_1值、ROC曲线等评估指标,各项指标均显示模型取得了优秀的分类效果,证明了基于CNN的假冒域名识别是一个可靠的方法.

关键词:域名生成算法;混合词向量;深度学习;卷积神经网络
中图分类号:TP391 **文献标识码**:A doi:10.3969/j.issn.0253-2778.2020.07.020

引用格式:杜淑颖,杜 鹏,丁世飞. 基于CNN的假冒域名识别方法研究[J]. 中国科学技术大学学报,2020,50(7):1019-1025.
DU Shuying, DU Peng, DING Shifei. A malicious domain name detection method based on CNN[J]. Journal of University of Science and Technology of China, 2020,50(7):1019-1025.

A malicious domain name detection method based on CNN

DU Shuying[1,2], DU Peng[1], DING Shifei[1]
(1. *School of Computer Science and Technology, China University of Mining and Technology, Xuzhou* 221116, *China*;
2. *School of information management, Xuzhou Vocational College of Bioengineering, Xuzhou* 221000, *China*)

Abstract: In recent years, various cyber attacks based on botnets have been one of the cyber security threats. Various malwares use the Domain Generation Algorithm (DGA) to automatically generate a large number of pseudo-random domain names to connect to commands and control servers. The detection and classification of pseudo-random domain names based on the convolutional neural network (CNN) method is focused on. A brief introduction is given to the hazards, basic principles of botnets, and the role of fake domain names in botnets. After analyzing the principle of DGA algorithm and the defects of traditional DGA domain name recognition algorithm, emphasis is laid on the research of fake domain name recognition method based on convolutional neural network. The basic concept of convolutional neural network is expounded by simple neural network training experiments. The differences of the model's effect on solving classification problems under different hyperparameters and different excitation functions are simulated. In the analysis of the model operation results, the accuracy and loss function of the domain name identification by the convolutional neural network model are given, and the evaluation indexes of the accuracy, recall, F1 and ROC curves are printed out. All indicators show that the classification of the model is good. It is concluded that counterfeit domain name recognition based on CNN is a reliable method.

Key words: domain generation algorithm (DGA); word embedding; deep learning; convolutional neural network (CNN)

收稿日期:2020-06-03;**修回日期**:2020-06-21
基金项目:国家自然科学基金(61976216, 61672522)资助.
作者简介:杜淑颖,女,1981年生,博士生/副教授. 研究方向:机器学习、信息安全. E-mail: du3477@139.com
通讯作者:丁世飞,博士/教授. E-mail: dingsf@cumt.edu.cn

0 引言

僵尸网络(botnet)是一种新型攻击方式,由传统恶意代码形式进化而来,是指使用一种或者多种传播手段,将大量主机感染僵尸程序病毒,攻击者从而在被感染主机网络中成为控制者[1].僵尸网络为攻击者提供了一种隐秘性高、方法灵活且效率高的一对多命令与控制机制,控制者会通过一个信道给被感染的主机发送命令从而控制主机,攻击者可以通过僵尸网络控制大量的主机来进行其他类型的网络攻击,诸如对大量主机进行信息窃取以影响主机用户的隐私;借助于客户端/服务器技术,联合已控制的大量主机来进行分布式拒绝服务攻击(DDoS),耗尽其他网络或系统的资源;或者在僵尸网络内发送大量的垃圾邮件,占用大量网络带宽,影响网络的传输与运算速度,传播不良内容,攻击者可以借此来牟利.由此可见,僵尸网络的破坏力极大.

僵尸网络中有一个中心机器——C&C(control & command)服务器,负责控制主机,处理信息并给被控制主机下发任务命令.要对抗僵尸网络,如果能够消灭掉其C&C服务器,就能很轻松地毁灭掉此僵尸网络[2].于是僵尸网络缔造者都会想尽办法来保护自己的C&C服务器域名以避免被安全人员查杀,因此诞生了基于域名生成算法(domain generation algorithm, DGA)方法的僵尸网络,使用DGA可以自动生成大量的假冒域名,DGA的基本原理是主控端与被控端协商好的一种基于随机算法的域名生成协议,即使用一个随机字符串作为域名进行注册,并将这个注册的域名作为C&C服务器的域名经常进行更换.由于DGA生成的域名具有很强的随机性和短时效性,增加了安全人员查杀僵尸网络的难度.

作为一项日益严重的互联网安全问题,僵尸网络已经成为网络安全领域研究者共同关注的热点话题,鉴于僵尸网络对互联网用户造成了严重的威胁,研究能够识别DGA生成的假冒域名的方法意义深远.DGA域名检测技术从依靠手工提取特征发展到自动提取特征的基于深度学习模型的方法,在DGA域名检测任务中取得了较大的进展[3].一种仅基于域名字符串进行域名是否为DGA域名预测的模型因其通用性而受到特别关注,因为获取除域名字符串之外信息的方法往往不可用或者获取信息的成本过于昂贵.

本文构建并训练了一个能够仅凭域名字符串就可判别域名是否为正常域名或者DGA域名做出预测的深度学习架构.本文首先描述DGA域名被大量应用于僵尸网络的问题背景,并简要介绍DGA算法和过去识别DGA域名的一些对策,并阐述本文使用的卷积神经网络架构,包括输入层、卷积层、池化层、全连接层、Softmax层等.然后在公开数据集上进行实验,涉及的步骤包括域名编码、预处理输入嵌入层、通过卷积层和池化层来采样和提取目标的特征、进行扁平处理后经过全连接层.最后全连接层输出模型的结果.同时本文还将讨论模型的训练方法,通过调整模型的一些参数如批尺寸、迭代轮数以及学习速率等,给出了预测效果最好的参数.

1 相关工作

近几年黑客攻击者为了防止恶意域名被发现,会使用Domain Flux或者IP Flux来快速生成大量恶意域名[4].Domain Flux通过不断地变换域名,指向同一个IP,而IP Flux是只有一个域名不断变换IP,多个IP可以被一个域名使用.DGA是一种利用随机生成的字符串作为C&C域名,从而逃避域名黑名单检测的技术手段[5].

DGA使用的种子有许多种类,包括日期、社交网络搜索热词、随机数、字典等生成一串字符前缀,再添加TLD如.com、.org等得到最终的生成域名(algorithmically generated domain, AGD).这就是DGA使用种子生成域名的基本原理.根据生成算法分类,DGA主要有以下几种常见的方法:①基于算术(arithmetic-based)的DGA,这种方法通过计算一系列的数值,这些值能根据ASCII码直接表示成域名,或者将这些值作为偏移值指向在DGA硬编码字符表中的一个字符.②基于哈希(Hash-based)的DGA.使用基于十六进制表示的哈希值生成最终生成域名AGD.这种DGA经常使用SHA-256或MD5两种哈希值.③基于单词表(wordlist-based)的DGA.从一个或多个单词表中选择单词并连接成一个域名,这样的域名由于直接出自单词,因此更具有迷惑性.④基于置换(permutation-based)的DGA.这种DGA通过对一个域名进行置换操作直接生成多个新的域名.

捕获动态生成的恶意域名的挑战引起了国内外学者的广泛研究,由此衍生出多种检测和识别DGA域名的算法思想和方法,以打击使用DGA算法的僵尸网络来进行互联网恶意活动,主要的研究算法包括:①一个简单的解决方案是分析所有客户端的整个相关活动的网络流量[6].②对恶意软件及其相应的DGA用逆向工程来理解DGA算法的隐藏模式[7].使用种子值,将注册域名列表和自己的服务器配置为C2C结构,称为SinkHoling[8].此方法将有助于劫持僵尸网络,一旦被这种方法劫持,攻击者就必须使用更新了的种子值来重新传播所有的bot程序[9].③另一种对抗DGA的静态方法是黑名单(blacklisting),黑名单包含DGA生成的域名,网络管理员使用此黑名单来阻止与C2C服务器的连接.这种方法耗时且需要大量的资源,而且依赖于获得种子值,一旦种子值未知,这种方法也就失效了.④最重要的方法是基于机器学习创建DGA分类器[10-11].DGA分类器驻留在网络中,并在发现DGA生成的域名进行DNS请求时向网络管理员发出警报.DGA分类器是域名信誉系统(domain reputation system, DRS)的重要组成部分[12].DRS将恶意软件的可信评级得分标记为1,将良性软件标记为0.DGA类型分为两类:一是回顾

(retrospective). 聚集大量的域名并在不同的聚类(cluster)上计算统计属性进行分类[13]. 二是实时监测和预防. 在不使用上下文属性的情况下对每个域名进行分类[14]. 与回顾性方法相比,通常认为这种方法较为困难,并且有可能表现出较低的性能.

DGA 检测和分析一直是安全研究人员研究的重点,传统的机器学习方法找到了许多解决方案. 尽管在某些情况下,它们在 DGA 检测中已经取得了显著的性能[15],但这些方案并不一定是有效的,并且不能在实时系统中采用. 主要原因是这种解决方案完全基于特征工程(熵,字符串长度,字母数字字符,元音与辅音的比例等). 有了 DGA 域名生成算法,攻击者就可以利用它来生成用作域名的伪随机字符串,这样就可以有效地避开黑名单列表的检测[16]. 伪随机意味着字符串序列似乎是随机的,但由于其结构可以预先确定,因此可以重复产生和复制. 该算法常被运用于恶意软件以及远程控制软件上. 首先攻击者运行算法并随机选择少量的域(可能只有一个),然后攻击者将该域注册并指向其 C2 服务器. 在受害者端恶意软件运行 DGA 并检查输出的域是否存在,如果检测为该域已注册,那么恶意软件将选择使用该域作为其命令和控制(C2)服务器. 如果当前域检测为未注册,那么程序将继续检查其他域. 安全人员可以通过收集样本以及对 DGA 进行逆向,来预测哪些域将来会被生成和预注册并将它们列入黑名单中. 由于 DGA 可以在一天内生成成千上万的域,我们不可能每天都重复收集和更新列表,因此基于特征工程以机器学习为基础的解决方案可能不适用于现有 DGA 或新生成的 DGA 检测和分析. 一旦出现新的 DGA 域名,则将计算相应的特征集,并且必须对新获得的特征进行训练. 这种方法非常的耗时. 此外,一旦采用了已知的特征集,攻击者就可以轻松地避开采用基于已知特征集的机器学习解决方案.

2　域名预处理和卷积神经网络

DGA 域名检测工作中,针对每一个域名 X,判断其属于 DGA 域名还是正常域名,这是二分类问题;判断其属于正常域名还是 DGA 域名的具体家族,这是多分类问题.

非 DGA 的正常域名,就是我们日常访问互联网时提供正常各项互联网服务的域名,如 baidu.com,google.cn 等. 正常域名通常是由具有意义的单词或单词片段组成,而 DGA 域名通常是由杂乱无意义的字符组成,例如 chaqsxphhmr.org. 直观上看,DGA 域名与正常域名在字符分布上应存在差异,考虑到自然语言中的英文单词或中文拼音,其元音与辅音字母的分布存在一定的规律性,存在着多种元音字母与元音字母,辅音字母与辅音字母的固定组合方式. 而在使用 DGA 算法生成的域名字符串中,这样的字母分布特点并不明显. 于是考虑使用深度学习模型进行 DGA 域名的检测与分类,深度学习模型可以发现字符之间隐含的分布特征.

2.1　域名预处理-以字符级别编码的域名

域名实际上为文本字符串,域名表示(domain representation)通常称为域名编码(domain encoding). 域名编码包含两个步骤. 第一步,对原始域名进行预处理并将其标记为字符. 在预处理中,删除顶级域名,并将所有字符转换为小写. 第二步涉及仅使用训练数据作为初始步骤的词汇创建. 单词大小参数值依赖于每个类的训练向量与给定学习任务的参数数量之间的对称性,这里为了限制单词大小,只选择符合最小频率的字符.

首先对字符进行编码,使用 one-hot 编码. 输入字符嵌入权重的维度 $x=$(nb-字符,字符嵌入维度). 其中输入维度=(nb-字符,单词长度). nb-字符表示顶部字符(top characters)的数量,单词长度表示唯一字符的数量,每个字符以 one-hot 编码格式表示. one-hot 编码又称为一位有效编码,主要是采用 N 位状态寄存器来对 N 个状态进行编码,每个状态都有独立的寄存器位,并在任意时候只有一位有效. 在实际的机器学习任务中,特征有时候并不总是连续值的,有可能是一些分类值[17],如性别可分为“male”和“female”. 在机器学习任务中,对于这样的特征,通常我们需要对其进行特征数字化. 比如有 3 个特征属性:性别:[“male”, “female”],地区:[“Asia”, “US”, “Europe”], 浏览器:[“Firefox”, “Chrome”, “Safari”, “Internet Explorer”]. 对于某一个样本,如[“male”, “Asia”, “Chrome”],我们需要将这个分类值的特征数字化,最直接的方法,就是采用序列化的方式如[0, 0, 1],但是这样的特征处理并不能直接放入机器学习中. one-hot 编码的处理方式为:对于上述问题,性别的属性是二维的,地区是 3 维的,浏览器则是 4 维的,这样,采用 one-hot 编码的方式对于上述样本[“male”, “Asia”, “Chrome”]编码,“male”对应着[1, 0],“Asia”对应着[1, 0, 0],同理“Chrome”对应着[0, 1, 0, 0],则完整的特征数字化的结果为:[1, 0, 1, 0, 0, 0, 1, 0, 0]. 这就是 one-hot 编码.

接下来将稀疏的 one-hot 编码映射为词向量表示. 由于 one-hot 编码过于稀疏,包含的特征仅仅是某个字符在词汇表 D 中的序号,难以被深度学习模型利用,因此引入稠密的词向量表示方法,使得深度学习模型可以学习到更多维度的特征表示. 使用查找表(lookup table)操作将字符 ID 特征转换为特征向量. 此外,最常出现的字符以升序编号. 该特征向量变换可以用数学公式表示为:查找表层(lookup table layer,LUT)将每个字符 $c \in D$ 表示为内维特征向量 d_{wvd},$\mathrm{LUT}_c C=\langle C\rangle_c^l$,$C \in R^{d_{cvd}\times|D|}$ 表示权重矩阵的学习参数,表示通过超参数调整选择的字符矢量大小. 通过上述步骤应用于字典 D 中的每个字符,得到每个字符的特征向量,其形式为

$$\mathrm{LUT}_c([C]_l^s)=(\langle C\rangle_{[c]_1}^l\langle C\rangle_{[c]_2}^l\langle C\rangle_{[c]_3}^l\ldots\langle C\rangle_{[c]_s}^l) \tag{1}$$

式中,$[C]_l^s$ 表示从词汇表 D 中选择的字符数. 作为下一步,为不同的唯一词汇 ID 序列设置固定的最大长度并馈送到字符嵌入层(embedding layer). 字符嵌入层通过执行如下所示的简单数学运算将字符转换为其字符嵌入.

字符嵌入权重=(单词长度,字符嵌入维度),字符嵌入维度表示字符嵌入向量的大小.嵌入权重矩阵中的第 j 行表示为整数 j .

字符嵌入的维度可以被认为是深度学习算法的超参数之一[18].此操作将离散字符映射为连续数字的向量[19].字符嵌入通过将它们映射到高维几何空间得到给定域名序列的语义含义.这种高维几何空间称为嵌入空间.如果将编码作为实数值向量来正确地学习嵌入的域名的语义,则相似的字符出现在高维几何空间中彼此接近的相同簇(cluster)[20],得到的嵌入输出向量被传递到任何其他层.

2.2 卷积神经网络

神经网络经常使用监督学习的方式来合理地设置参数取值,设置神经网络参数的过程就是神经网络的训练过程,只有经过有效训练的神经网络模型才能真正地解决分类或者回归问题[21].监督学习是指利用一组已知类别的样本调整分类器的参数使分类器达到所要求性能的过程.监督学习最重要的思想是:在已知答案的标注数据集上,模型给出的预测结果要尽量接近真实的答案.通过调整神经网络中的参数对训练数据进行拟合,可以使得模型对未知的样本提供预测的能力.比如在用训练集训练模型识别 DGA 域名之后,再用测试集输入神经网络,神经网络就能输出它对域名分类的预测.

使用神经网络解决分类问题主要可以分为 4 个步骤:

Step1 提取问题中实体的特征向量作为神经网络的输入.不同的实体可以提取不同的特征向量.

Step2 定义神经网络的结构,并定义如何从神经网络的输入得到输出.这个过程就是神经网络的前向传播算法.

Step3 通过训练数据来调整神经网络中参数的取值,这就是训练神经网络的过程.

Step4 使用训练好的神经网络来预测未知的数据.

随着数据量的增多,神经网络会不断优化参数,在众多优化算法中,最常用的方法是反向传播算法(back propagation).反向传播算法流程图如图 1 所示.

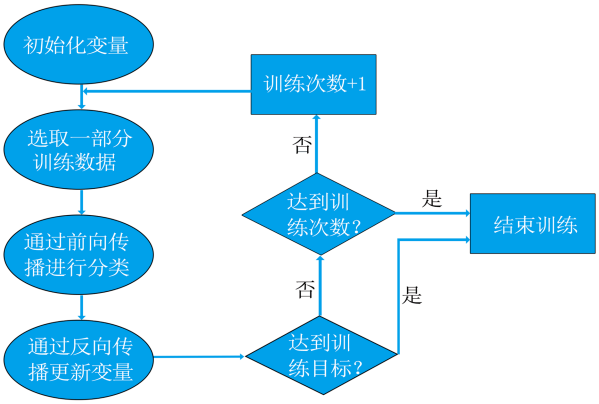

图 1 神经网络反向传播优化流程

Fig. 1 Flowchart of neural network back propagation optimization

深度学习中损失函数(loss function)通过对真实样本标记和预测样本产生的误差反向传播知道神经网络参数学习[22].神经网络的效果以及优化的目标是通过损失函数来定义的.损失函数是用来评估模型的预测值 $f(x)$ 与真实值 Y 的不一致程度,它是一个非负实值函数,损失函数越小,表明模型的健壮性越好.在 DGA 域名识别模型中使用交叉熵(cross entropy)损失函数,交叉熵刻画了两个概率分布之间的距离,它是分类问题中使用较广的一种损失函数.

学习率(learning rate)用来控制每次神经网络参数更新的幅度,直观上理解学习率就是每次参数移动的幅度.

卷积神经网络整体结构上和全连接层神经网络相似,通过一层一层的节点组织起来[23],但卷积神经网络相邻两层之间只有部分节点相连.使用全连接层的问题往往在于全连接层的参数太多,相邻两层节点的连接方式是全连接层神经网络和卷积神经网络之间的唯一区别.

一个卷积神经网络的主要结构有以下 5 个部分:

输入层.输入层作为整个卷积神经网络的输入,比如处理图像的卷积神经网络中,输入代表了图像的像素矩阵.

卷积层.卷积层是卷积神经网络中最重要的一个部分.与全连接层不同,卷积层上每个节点的输入层只是上一层神经网络的一小部分.卷积层尝试对神经网络中的每一小块进行更深入地分析进而提取抽象程度更高的特征.卷积层主要是对网络中前一层的一个或者多个特征图与一个或者多个卷积核进行卷积操作,产生一个或者多个输出.卷积核用 w 表示,$w \in R^{hk}$.其中,h 表示卷积核窗口高度,k 表示词向量的维度大小.每经过一个高度为 h ,宽度为 k 的词序列窗口时就产生一个新的特征值.,$w_{i:i+h}$ 表示一个长度为 h 的单词序列 (W_i, W_{i+1},…W_{i+h}) ,W_i 表示一个单词,每个特征值的计算过程为

$$c_i = f(w \cdot W_{i:i+h} + b) \tag{2}$$

式中,w 为卷积核的权重参数,b 是卷积层的偏置项,$b \in R$,操作符($\cdot$)表示卷积操作,$f(\cdot)$ 是激活函数,通常可使用 sigmoid 和 tanh 等非线性函数.对短文本中每个窗口中的单词序列 ($W_{1:h}$, $W_{2:h}$,…$W_{N-h+1:N}$) 进行卷积操作可以得到一个特征图,具体的计算过程为

$$c = (c_1, c_2, \cdots, c_{N-h+1}) \tag{3}$$

式中,N 表示一个短文本中单词的个数,h 表示卷积核窗口的高度,C 为短文本经过一个卷积核形成的特征图,$c \in R^{N-h+1}$.由于卷积核的高为 h ,宽为词向量的维度,因此经过卷积后形成的特征图是一个高为 $N-h+1$,宽度为 l 的矩阵.不同的卷积核可以从不同的角度提取出短文本中的特征,通过设置卷积核的个数可以得到多个不同的特征图.

池化层(pooling).通过池化层可以进一步缩小最后全连接层中节点的个数,简化网络计算复

杂度，减少整个神经网络中的参数. 比如在 DGA 域名识别模型使用的池化操作 MaxPooling；在图像处理时使用 MaxPooling 压缩特征图，选择步长为 2，则池化操作在每个 2 * 2 的区域中寻找区域最大值，将特征图压缩为原来的 1/4，这样有可能影响神经网络的准确度，因此可以增加特征图的深度来弥补.

全连接层. 在经过多轮卷积层和池化层处理后，卷积神经网络一般会是 1 或 2 个全连接层给出最后的连接结果. 可以将卷积层和池化层的处理看作自动提取特征，特征提取之后，依然需要全连接层来完成分类任务.

Softmax 层. 和全连接层神经网络一样，通过 Softmax 层可以得到样例属于不同种类的概率分布情况.

本文提出的基于 CNN 的仿冒域名识别算法流程图如图 2 所示.

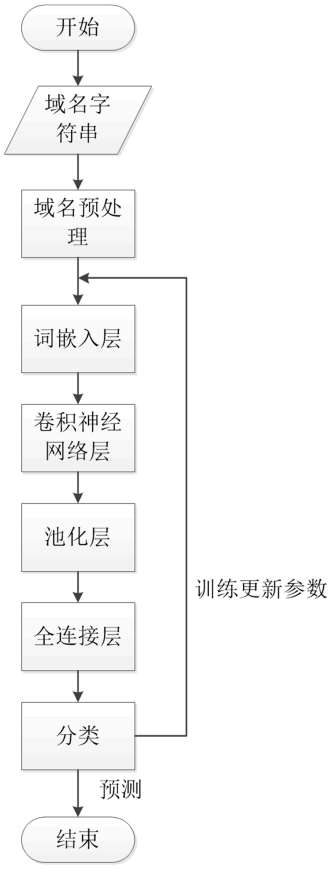

图 2　基于 CNN 的仿冒域名识别算法流程图

Fig. 2　Flowchart of the proposed malicious domain name detection method based on CNN

3　实验与结果

我们对本文提出的方法设计实验并进行验证，实验环境是：操作系统是 Windows10，内存 8GB，使用 GPU 显卡 NVIDIA GeForce GTX 1050 进行加速，使用 Keras 和 TensorFlow-GPU 编写模型代码.

3.1　数据集介绍

在 https://data.netlab.360.com/feeds/dga/dga.txt 网址可以下载 360 netlab 发布的 DGA 域名，作为黑样本，将下载后的 dga.txt 放到一个新建的 PyCharm 文件夹下. 白样本使用 Alexa top 1 million，这是全球访问量前一百万的域名[24]，将白样本 alexa.txt 放在和 dga.txt 一个文件夹下. 可以查看文件是否为所需要的域名.

查看数据集的大小，alexa.txt 包含有一百万个正常域名，而 dga.txt 包含 1114327 个域名. 由此可见，数据集中 DGA 域名略多于正常域名，总数据集大小包含超过两百万个域名，因此有足够多的数据训练卷积神经网络. 我们对这些数据集进行划分，一部分用来训练卷积神经网络模型，优化模型的参数，另一部分用来测试模型对于给定的域名预测的准确性，以评估模型的效果. 对于我们要使用的卷积神经网络模型，数据集大小已经足够.

3.2　实验步骤

首先我们对数据进行预处理，先导入 alexa 域名与 DGA 域名分别存放在列表中. 因为模型使用的学习方法是监督学习，所以再定义一个标签数组 y_train，长度与域名列表 x 相同，并且前一百万个元素为 0，后面的元素全为 0，我们标记 alexa 域名的标签为 0，定义 DGA 的域名为 1. 再定义一个分词器 token，需要保留的最大词数为 256，还设置了将文本全部转换为小写并将每个字符都视为一个标记，之后对域名列表 x 转换为序列列表，列表中的每一个序列对应一个输入文本. 设置序列的最大长度为 50，大于此长度的序列会被截断，小于此长度的序列在序列前端补 0，最后返回数据集的二维张量与其标签的列表. 对数据集进行划分. 将域名列表 X 和标签数组 y 分别按 60%和 40%的比例进行随机划分，训练神经网络时数据是按批(batch)输入神经网络的，这个批的大小就是 batch_size，本文设为 128，window_size 为卷积窗口大小，设置为 3，embedding_Dim 为全连接嵌入维度的大小，设置为 128，input_Len 是卷积神经网络期望输入的长度，预处理的时候已经将域名处理为长度为 50 的整型串.

与传统机器学习方法相比，本文使用随机森林、决策树、逻辑回归等算法对域名进行分类；将字符的双字符组序列作为域名的人工输入特征.

3.3　评价指标

实验采用如下评价指标对本文提出的方法进行评价：精确率(precision)，表示模型的正确率，定义为正确识别的个体总数与识别出的个体总数之比；召回率(recall)，定义为正确识别的个体总数与测试集中存在的个体总数之比；$F1$ 值是 precision 和 recall 的调和平均值，由公式 precision * recall * 2/(precision+recall)计算得到.

3.4　实验结果

模型的各项指标如表 1 所示. 从表 1 结果可以看到，模型对 alexa 域名的正确率为 99.13%，对 DGA 域名的正确率为 99.06%，对所有域名的分类

结果的正确率是 99.10%.

表 1 二分类实验测试结果

Tab. 1 Summary of test results for binary classification

method	precision	recall	F_1	sample number
0	0.991 3	0.992 0	0.991 6	400 643
1	0.990 6	0.989 8	0.990 2	445 088
Avg	0.991 0	0.991 0	0.991 0	845 731

本文提出的基于 CNN 方法与其他传统机器学习方法的对比实验结果如表 2 所示，从表 2 结果可以看出，基于 CNN 的深度学习方法在各项性能指标上皆优于决策树、随机森林、逻辑回归等传统机器学习方法.

表 2 对比实验测试结果

Tab. 2 Summary of test results of models

method	precision	recall	F_1
RF	0.958 3	0.990 0	0.973 9
DT	0.859 1	0.992 5	0.921 0
LR	0.966 7	0.987 0	0.976 7
CNN	0.991 0	0.991 0	0.991 0

本文还设计了实验探究超参数对 CNN 模型的影响，在本文提出的 CNN 里，主要的超参数包括卷积窗口大小 k 和卷积核数 n 这两个对分类结果有较为明显影响的超参数. 超参数的不同取值对分类结果的影响如表 3 所示. 实验结果表明，在 4 种卷积窗口和卷积核数的设置下，当 $k=3$，$n=64$ 时，在 CNN 对假冒域名的识别效果最好.

表 3 不同超参数的 CNN 测试结果

Tab. 3 Summary of test results of different hyperparameters

k	n	precision	recall	F_1
2	32	0.989 6	0.989 6	0.989 6
2	64	0.990 7	0.990 7	0.990 7
3	32	0.990 0	0.990 0	0.990 0
3	64	0.991 0	0. 991 0	0. 991 0

模型精确度曲线如图 3 所示，由图 3 可以看出，随着训练轮数 epoch 的增加，模型准确率有了提高，在前几轮增长速度较快，之后增长速度逐渐放缓，最后趋于收敛.

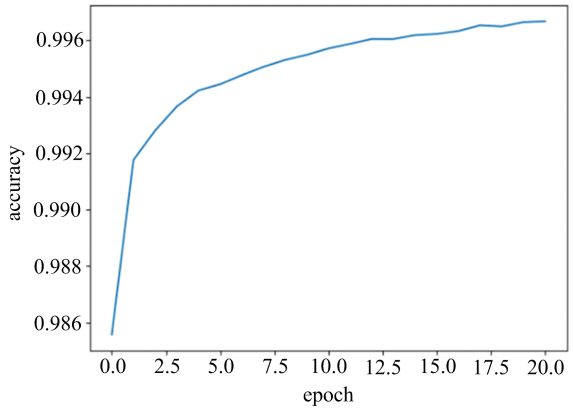

图 3 模型精确度随训练轮数的变化

Fig. 3 Model accuracy changes with the number of training rounds

模型损失函数曲线如图 4 所示，随着训练轮数 epoch 的增加，损失函数逐渐下降，在前几轮下降速度较快，之后下降速度逐渐放缓，最后趋于收敛.

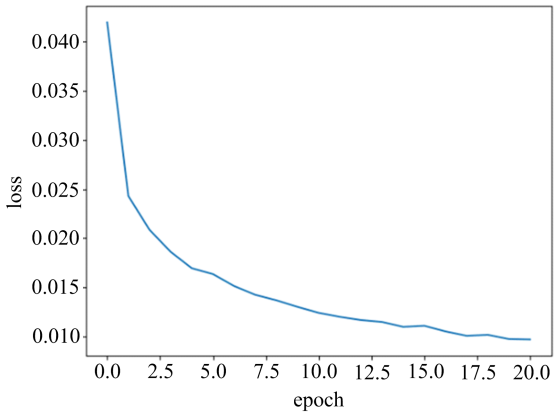

图 4 损失函数随训练轮数的变化

Fig. 4 Variation of the loss function with the number of training rounds

“受试者工作特征”(receiver operating characteristic，ROC)曲线的横轴是“假正例率”，纵轴是“真正例率”. 两者的定义为 TPR=TP(TP+FN)，FPR=FP/(TN+FP). 其中 TP、FN、FP、TN 的定义分别为：①若一个实例是正类且预测为正类，则为真正类(true positive，TP)；②若一个实例是正类且预测为负类，则为假负类(false negative，FN)；③若一个实例是负类且被预测为正类，则为假正类(false positive，FP)；④若一个实例是负类且被预测为负类，则为真负类(true negative，TN). ROC 曲线越靠近(0,1)点越好，曲线越偏离 45 度对角线越好，从图 5 的 ROC 曲线可以看出，模型评估效果很好.

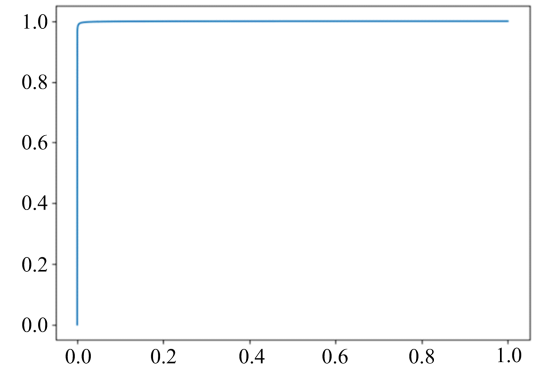

图 5 卷积神经网络模型的 ROC 曲线图

Fig. 5 ROC graph of the convolutional neural network model

4 结论

本文从形成原因、作用特点和危害等方面，详细分析了 DGA 域名对僵尸网络的重要性和对网络安全的影响. 传统的应对 DGA 域名的对策耗费大量的人力且效率低下，因此需要基于卷积神经网络的模型来识别 DGA 域名.

深度学习常用于解决二分类问题，本文阐述了深度学习以及卷积神经网络的模型架构以及各层

结构的作用，并获取了 DGA 域名识别模型所需要的数据集，数据集来源权威可靠，数量够多以保证模型的结果真实可信. 在 Python 代码实现的模型中使用了经典的卷积神经网络架构，通过合理设置超参数，并根据经验选择合适的激励函数、学习速率、优化函数等，根据实验结果不断调整超参数以及模型结构，最终得到一个识别效果较好的基于 CNN 的 DGA 域名识别模型.

模型运行结果表明，基于 CNN 的 DGA 域名识别模型性能十分卓越，准确率、召回值、F_1 值、ROC 曲线等评估指标均显示如此，并且识别速度较快，经过多次测试后发现模型识别域名具有准确率高、速度快、成本低等特点，因此基于 CNN 的 DGA 域名识别是一种可靠的方法.

参考文献(References)

[1] 诸葛建伟，韩心慧，周勇林，等. 僵尸网络研究[J]. 软件学报，2008，19(3)：702-715.

[2] 江健，诸葛建伟，段海新，等. 僵尸网络机理与防御技术[J]. 软件学报，2012，23(1)：82-96.

[3] 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.

[4] ANTONAKAKIS M，PERDISCI R，LEE W，et al. Detecting malware domains at the upper DNS hierarchy [C] // Proceedings of the 20th USENIX Security Symp (Security'11). Berkeley，CA：USENIX Association，2011：1-16.

[5] WOODBRIDGE J，ANDERSON H S，AHUJA A，et al. Predicting domain generation algorithms with long short-term memory networks [J]. 2016，arXiv：1611.00791.

[6] YU B，SMITH L，THREEFOOT M. Semi-supervised time series modeling for real-time flux domain detection on passive DNS traffic [C] // Proceedings of the 10th International Conference on Machine Learning and Data Mining，2014：258-271.

[7] GEFFNER J. End-to-end analysis of a domain generating algorithm malware family [C/OL] //Black Hat USA 2013. [2020-04-17]. https://media.blackhat.com/us-13/US-13-Geffner-End-To-End-Analysis-of-a-Domain-Generating-Algorithm-Malware-Family-WP.pdf.

[8] STONE-GROSS B，COVA M，CAVALLARO L，et al. Your botnet is my botnet：Analysis of a botnet takeover [C] // Proceedings of the 16th ACM Conference on Computer and Communications Security. New York：ACM，2009：635-647.

[9] YU B，SMITH L，THREEFOOT M，et al. Behavior analysis based DNS tunneling detection with big data technologies [C] // Proceedings of the International Conference on Internet of Things and Big Data，2016：284-290.

[10] MAC H，TRAN D，TONG V，et al. DGA Botnet Detection Using Supervised Learning Methods [C] // Proceedings of the Eighth International Symposium on Information and Communication Technology. ACM，2017：211-218.

[11] VINAYAKUMAR R，SOMAN K P，POORNACHANDRAN P，et al. Evaluating deep learning approaches to characterize and classify the DGAs at scale [J]. Journal of Intelligent & Fuzzy Systems，2018，34(3)：1265-1276.

[12] ANTONAKAKIS M，PERDISCI R，DAGON D，et al. Building a dynamic reputation system for dns [C] // Proceedings of the 19th USENIX Security Symp (Security'10). Berkeley，CA：USENIX Association，2010：273-290.

[13] SCHIAVONI S，MAGGI F，CAVALLARO L，et al. Phoenix：DGA based botnet tracking and intelligence，in Detection of Intrusions and Malware，and Vulnerability Assessment [J]. Springer，2014：192-211.

[14] ANTONAKAKIS M，PERDISCI R，NADJI Y，et al. From throw-away traffic to bots：Detecting the rise of DGA-based malware [C] // Proceedings of the 21st USENIX Security Symp (Security'12). Berkeley，CA：USENIX Association，2012：491-506.

[15] KRISHNAN S，TAYLOR T，MONROSE F，et al. Crossing the threshold：Detecting network malfeasance via sequential hypothesis testing [C] //in 2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)，IEEE，2013：1-12.

[16] BARUCH M，DAVID G. Domain Generation Algorithm Detection Using Machine Learning Methods [M]. Cyber Security：Power and Technology. Springer，Cham，2018：133-161.

[17] 丁世飞. 人工智能[M]. 2 版. 北京：清华大学出版社，2015.

[18] GOODFELLOW I，BENGIO Y，COURVILLE A. Deep Learning[M]. MIT Press，2016.

[19] ABADI M，BARHAM P，CHEN J，et al. TensorFlow：A system for large-scale machine learning [C] //Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)，Savannah，Georgia，USA，2016.

[20] WANG Q，ZHANG Y，LI P，et al. Cross-domain sentiment classification based on word2vec [J]. Application Research of Computers，2018，35(10)：2924-2927.

[21] KARPATHY A，JOHNSON J，LI F F. Visualizing and understanding recurrent networks [C] // Proceedings of the International Conference on Learning Representations. San Juan，Puerto Rico：IEEE，2016，arXiv：1506.02078.

[22] LE Q，JAITLY N，HINTON G. A simple way to initialize recurrent networks of rectified linear units [J]. Computer Ence，2015.

[23] KIM Y. Convolutional neural networks for sentence classification [J]. 2014，arXiv：1504.00941.

[24] Does Alexa have a list of its top-ranked websites? [EB/OL]. [2017-04-02]. https:// support.alexa.com/.