基于代价敏感学习的恶意 URL 检测研究

蔡勋萌[1], 王 健[1], 李鹏博[2]

[1]北京交通大学计算机与信息技术学院 北京 中国 100044
[2]中电科网络空间安全研究院 北京 中国 100085

摘要 随着大数据时代的到来, 恶意 URL 作为 Web 攻击的媒介渐渐威胁着用户的信息安全。传统的恶意 URL 检测手段如黑名单检测、签名匹配方法正逐步暴露缺陷, 为此本文提出一种基于代价敏感学习策略的恶意 URL 检测模型。为提高卷积神经网络在恶意网页检测领域的性能, 本文提出将 URL 数据结合 HTTP 请求信息作为原始数据样本进行特征提取, 解决了单纯 URL 数据过于简单而造成特征提取困难的问题, 通过实验对比了三种编码处理方式, 根据实验结果选取了最佳字符编码的处理方式, 保证了后续检测模型的效果。同时本文针对 URL 字符输入的特点, 设计了适合 URL 检测的卷积神经网络模型, 为了提取数据深层特征, 使用了两层卷积层进行特征提取, 其次本文在池化层选择使用 BiLSTM 算法提取数据的时序特征, 同时将该网络的最后一个单元输出达到池化效果, 避免了大量的模型计算, 保证了模型的检测效率。同时为解决数据样本不均衡问题, 在迭代过程中为其分配不同惩罚因子, 改进了数据样本初始化权重的分配规则并进行了归一化处理, 增加恶意样本在整体误差函数中的比重。实验结果表明本文模型在准确率、召回率以及检测效率上较优于其他主流检测模型, 并对于不均衡数据集具有较好的抵抗能力。

关键词 深度学习; 恶意网页; URL 检测; 代价敏感学习; 神经网络
中图法分类号 TP391.1 DOI 号 10.19363/J.cnki.cn10-1380/tn.2023.03.05

Research on Malicious URL Detection Based on Cost-sensitive Learning

CAI Qingmeng[1], WANG Jian[1], LI Pengbo[2]

[1] School of Computer and Information Technology, Beijing Jiaotong University, Beijing, 100044, China
[2] CETC Cyberspace Security Research Institute, Beijing, 100085, China

Abstract In the wake of the advent of big data era, whereas the malicious URL, as the medium for Web attacking, progressively threatens the security of users' information. Traditional detection methods in terms of malicious URL, such as blacklist detection and signature matching, are exposing their intrinsic defects, to this end, this paper proposes a malicious URL detection model based on a cost-sensitive learning strategy. In this thesis, HTTP request parameters together with URL information are employed as the original data samples to extract features; and the corresponding data processing is carried out to resolve the problem of difficult feature extraction incurred by simple URL data. In addition, by comparing three encoding processing methods through tests, this research has chosen the best processing approach in term of character encoding. By doing so, it has ensured the effectiveness of the subsequent detection model. Regarding the model of neural network, the Convolutional Neural Network model suitable for URL detection is specialized designed for the characteristics of URL character input. In this model, in order to extract the deep features of the data, two convolutional layers are broadly used. Secondly, this research utilizes a Bidirectional Long Short-Time Memory to extract the temporal features of the data from the pooling layer, while in the last unit of this network outputs the temporal features to achieve the pooling effect, this research method not only effectively extracts the contextual information regarding the data, also avoids an abundant model calculations and thus, ensures the efficiency of model detection. At the same time, in order to solve the problem of unbalanced data samples, it assigns different penalty factors to data samples during the iterative process, improves the rules for assigning initialization weights to data samples and normalizes them, increases the weight of malicious samples in the overall error function. Experimental results show that this model is better than other mainstream detection models in accuracy, recall and detection efficiency, and has better resistance to imbalanced data sets.

Key words deep Learning; malicious web page; URL detection; cost-sensitive learning; neural networks

通讯作者: 王健, 博士, 副教授, 博导, Email: wangjian@bjtu.edu.cn。
本课题得到中国国家铁路集团有限公司科技研究开发计划重点课题(No. N2020W005), 以及国家保密技术测评中心项目(No. K20GY500010)的支持资助。
收稿日期: 2021-07-21; 修改日期: 2021-11-15; 定稿日期: 2023-01-03

1 引言

由于近些年来恶意网页数量的飞速增长, 其对于网络安全的威胁也越来越大, 因此如何高效准确的识别恶意 URL(Uniform Resource Locator)已成为 Web 安全领域的重点研究内容。随着人工智能领域的飞速发展, 深度学习已逐步成为核心的研究课题, 并且随着其在各个领域的应用逐步成熟, 也为 Web 安全领域的智能检测提供了坚实的理论基础与创新的研究思路。在恶意 URL 检测方面, 传统检测依赖安全专家、黑名单等手段具有滞后性且效率低, 只能针对现有的 URL 进行检测, 无法及时更新[1]。在特征匹配与机器学习相结合的检测方法中, 以往的工作主要集中在分类器的改进上, 部分研究涉及特征选择和特征提取, 但不专门用于特征变换, 即在现有特征的基础上产生新的特征, 以方便下游算法。因此针对以上问题, 引入深度学习进行恶意检测的方法, 可以通过计算大量数据进行自动提取特征, 省去了人工提取特征工程所耗费的大量资源与成本, 且基于深度学习的方法可以有效的检测未知类型的恶意 URL, 避免了传统恶意检测的局限性与滞后性。

当下的多数研究均基于单一的 URL 检测, 其存在以下难点: 对于长度较短的恶意 URL 无法捕捉到足够的信息, 且有些恶意 URL 并未直接在 URL 上体现出恶意性。本文提出将 URL 信息与 HTTP (HyperText Transfer Protocol)请求信息相结合的方法, 恶意 HTTP 请求参数之间必然存在一定的关联性, 结合 HTTP 请求参数既可以使检测结果更加准确全面, 又在一定程度上节约了资源与成本的消耗。此外, 现有工作都是在假设有大量的训练数据可用的前提下进行的, 这显然是不切实际的, 因为人工标记成本可能非常昂贵。并且在实际的恶意 URL 检测任务中, 正常 URL 的样本数量与恶意 URL 的样本数量一般非常不均衡, 传统的检测方式及研究大多数均着眼于提高检测的准确率, 但模型在实际应用中单单评估其检测准确率往往不够客观, 因此需要从多角度评价模型的性能, 使模型具有更高的泛化性和可扩展性。

本篇论文针对当下 Web 安全领域中的恶意 URL 检测所面临的相关问题进行深入研究, 目前恶意 URL 的检测主要面临的问题有: 单纯检测恶意 URL 很难准确全面检测出其恶意性; 其次在检测 URL 恶意性分类的过程中, 极易丢失其上下文信息和位置相关信息; 并且在真实的检测中原始数据样本中恶意样本和正常样本的分布极度不均衡。因此针对以上问题, 本文提出将 URL 样本数据结合 HTTP 请求相关信息, 扩充原始数据集的可提取特征, 构建基于 CNN-BiLSTM 算法的恶意网页模型, 且在此基础上结合代价敏感学习策略。具体的论文主要工作以及创新点如下:

1) 由于简单的 URL 很难直接从其本身特征检测出隐藏的恶意性, 因此将恶意 URL 和 HTTP 请求中的其他参数进行结合, 扩充检测分类的可提取特征; 通过实验对比了 One-Hot、Word2Vec 以及 BoW(Bag of Words)等字符向量化方法并选取最佳编码方式, 从一定程度上保证了后续分类检测模型的准确性。

2) 使用卷积神经网络对得到的向量做特征提取, 由于特征数据量较大且 URL 为序列数据, 数据之间存在依赖关系, 本文使用双向长短期记忆网络算法作为池化层对 URL 时序特征进行提取, 并将双向长短期记忆网络的最后一个神经单元作为输出达到池化效果, 在一定程度上既提高了检测模型的准确率且同时保证了检测模型的效率。

3) 构建基于代价敏感学习的恶意 URL 检测模型, 重新设置了初始权重的分配规则, 且优化了整体误差函数并进行了归一化处理, 此改进方法使模型更加关注困难学习样本, 并对不均衡数据集具有较好的抵抗能力, 使模型具有更好的可扩展性和适用性。

2 研究现状

2.1 国内研究现状

由于人工智能在近几年的研究热潮以及在各领域取得的巨大成功, 机器学习和深度学习成为恶意 URL 检测的主要研究方向[2]。首先基于特征匹配和机器学习相结合的检测方法, 这类方法需要人工对特征进行提取, 针对不同检测场景需要选择不同类型的特征从而进行相应的特征匹配检测, 在特征提取的工作上, 需要安全专家具有相关经验知识或者通过大量正常网页和恶意网页的真实样例进行数据分析, 从而区分二者的特征进行相应提取, 再结合机器学习算法进行分类检测。

莫玉力[3]等人提出了基于混合分类器的恶意 URL 划分机制。此机制引入了 SVM(Support Vector Machine)和神经网络算法, 通过设置划分机制先过滤掉黑名单和白名单中的 URL, 对未知的 URL 使用多个分类器再次进行安全判决, 其最终对于恶意 URL 的检测准确率达到 95%左右且模型较为稳定。

陈康[4]等人提出一个完全基于词法特征的检测

方法, 使用语料训练神经网络得到 URL 中字符分布的嵌入式表示, 并训练神经网络模型对生成的 URL 特征图像进行分类, 提升了 URL 检测的精确度和 F1 值。

李苒[5]等人提出了从 DPI 数据中检测出恶意 URL 并加以防范的方法, 融合基于规则的过滤器、基于 XGBoost(eXtreme Gradient Boosting)模型的过滤器和基于深度学习算法的过滤器, 通过网络抓取获得 URL 的相关特征, 对 URL 的安全性进行判别, 实现了一个高性能的恶意 URL 判别模型。

易磊[6]等人使用传统机器学习方法与深度学习方法进行结合, 以 URL 及网页前端代码作为特征数据, 对此将人工制定规则提取到的特征与卷积神经网络自动提取到的特征进行融合, 再使用传统的机器学习算法对融合后的特征进行检测分类。此类方法对于恶意网页的检测较为全面, 但对于特征提取工作上需要大量的人工成本, 且结合网页代码的检测工作量较为庞大, 不利于整体的检测效率。

2.2 国外研究现状

在恶意网页检测的研究领域中, 许多国外学者也进行了大量研究。从最初的黑名单检测方法的优化, 逐步到基于机器学习与深度学习模型的检测方法优化, 研究人员提出了很多检测方法。

在早期的黑名单检测工作中, Prakash[7]等人使用五种启发式方法对已知的钓鱼网站进行简单组合, 再通过近似匹配算法将黑名单中的条目进行匹配对照, 以此来识别新的钓鱼网站。Bo[8]等人为了确保黑名单的持续有效性, 提出了一个自动黑名单生成器(AutoBLG)的框架, 该框架可以使用给定的现有 URL 黑名单自动识别新的恶意 URL。

此类研究方法均在传统的黑名单检测方法基础上进行了的改进, 一定程度改善了黑名单的滞后性问题。近些年来, 国外的研究学者同样使用机器学习和深度学习研究方法进行恶意网页的检测研究。

Vanhoenshoven[9]等人提出采用三种方法对特征进行抽取, 并验证决策树、K-近邻、贝叶斯、随机森林、支持向量机等多种分类器的性能, 其中随机森林和多层感知器达到了最高的精度。Jayakanthan[10]等人提出一种基于 EPCMU(Enhanced Probing Classification algorithm to detect Malicious URL)与朴素贝叶斯算法相结合的恶意 URL 检测方法, 该方法选用特殊字符数量、字符‘/’的数量、字符‘@’的数量、是否被黑名单标记等多种 URL 特征组成特征向量, 并利用朴素贝叶斯分类器进行恶意 URL 分类检测。

Hyrum[11]等人利用生成对抗网络构造一个基于深度学习的分布式遗传算法, 此方法通过生成对抗网络生成的域名集可以避开基于深度学习的特征检测器, 最后达到扩充恶意域名集的目的, 且扩充后的域名集可以有效的提高对未知数据集的检测。

Sungjin[12]等人提出基于攻击者习惯行为分析的恶意 URL 防护。学者通过深入分析攻击者在 URL 方面的战术行为, 提取了常见的行为特征并将其划分为不同的功能池, 并且采用相似性匹配技术, 通过将新网址与攻击者的习惯性网址操作行为进行相似匹配, 以此来识别其恶意性。此类方法可以通过使用较小的功能集覆盖大量的恶意 URL。

3 恶意 URL 检测模型

3.1 模型概述

本文构建了基于CNN-BiLSTM算法的恶意URL检测模型, 主要涉及到的神经网络有: 卷积神经网络、全连接神经网络和双向长短期记忆神经网络。卷积神经网络主要通过卷积核等计算操作对输入数据进行特征提取, 输入数据为由输入字符经过数字向量化处理转换的嵌入式矩阵[13]。全连接神经网络主要负责将获取到的特征进行整合与分类工作, 相当于检测模型中的“分类器”。而双向长短期记忆神经网络可以处理长期依赖问题, 更进一步提取到具有时序特征的高阶特征。最后在此基础上将代价敏感策略引入到神经网络模型中。本文考虑到目标检测类型为文本数据, 且结合相应的检测需求, 构造如图 1 所示的整体检测模型。

3.2 字符级向量编码处理方式

由于深度学习检测模型仅能对数值型的数字向量进行处理, 因此需要对数据特征进行相应的预处理工作[14]。通过调查各研究对原始数据的数字化表示方法不同, 这会造成检测效率和检测准确率较低。大多数的研究中均根据分析选择较为适合的词向量处理方式, 但是由于本文的研究目标是建立一个较优的检测模型, 因此本文分别对多种词向量处理方式进行了对比实验, 分别对比了 One-Hot[15]、Word2Vec[16]、BoW[17]等编码方式, 并选择标准的 CNN 算法作为基准模型, 分别将处理后的词向量作为输入, 且选择相同的测试数据集进行模型验证对比。

(1) One-Hot: 首先根据原始数据, 剔除不符合 URL 及 HTTP 请求参数命名规则的字符, 最终选定 97 个字符设置为适用于编码处理 ASCII 字符集。One-Hot 每个单个字符使用长为 97 个字符的向量来表示, 即每个字符表示的向量的元素个数为 97, 即

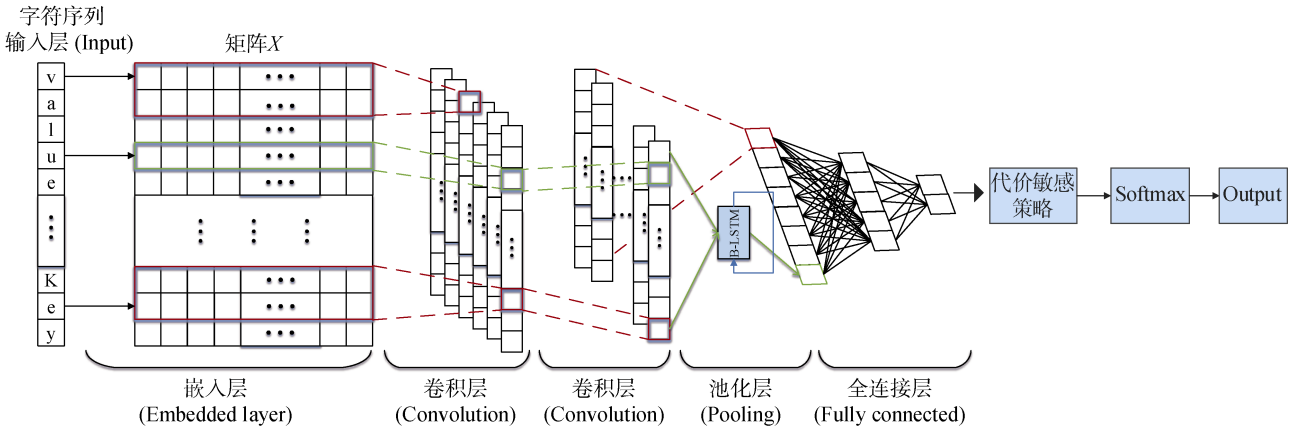

图 1　基于代价敏感学习的 CNN-BiLSTM 恶意网页检测模型结构示意图

Figure 1　Schematic diagram of the structure of the CNN-BiLSTM malicious webpage detection model based on cost-sensitive learning

$|\vec{c}|=97$，通过遍历数据集生成字符映射表，表中每个字符都是唯一的。以简单样例数据为例进行 One-Hot 编码后得到的输出如图 2 所示，其特点是矩阵非常稀疏，$\vec{c}$ 中只有一位为 1，其余均为 0。

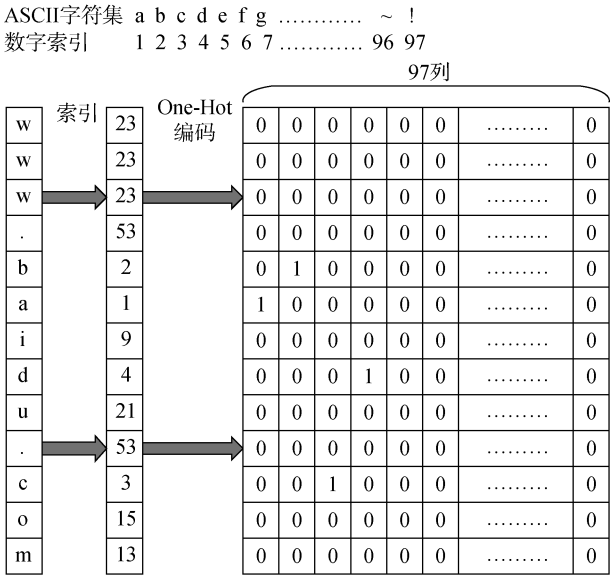

图 2　一位有效编码字符处理示意图

Figure 2　One-Hot Processing diagram

(2) Word2Vec: 由于 Skip-Gram 在大型语料中表现更好，因此本文选用 Skip-Gram 模型进行字符串编码。根据模型需求，将窗口大小设置为 5，根据 One-Hot 编码的向量维度设置神经元的个数为 97，选取训练样本数量 *Batch_size*=200，以此构造 Skip-Gram 三层神经网络模型。以下为以简单 URL——“www.baidu.com”为例，进行 Skip-Gram 编码后的结果示意图如图 3 所示。

(3) BoW: 由于本文创建的字典中包含 97 个字符，因此对于预测文本可以使用一个 97 维向量进行表示。同样以简单 URL——“www.baidu.com”为例，词袋模型编码处理结果如图 4 所示。

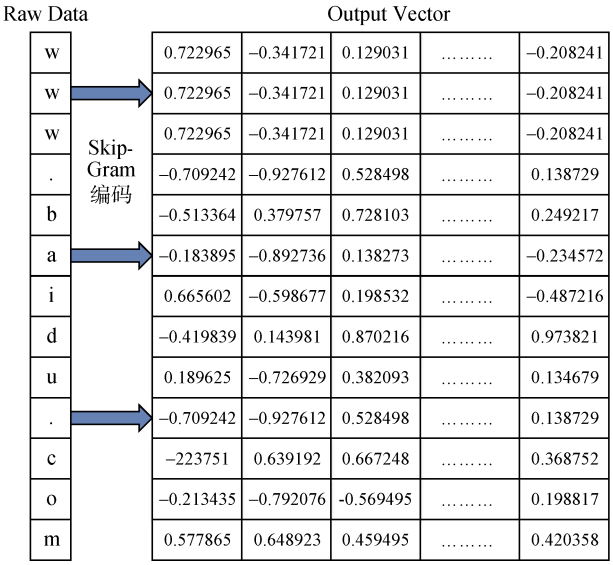

图 3　Skip-Gram 编码模型处理结果示意图

Figure 3　Schematic diagram of processing results of Skip-Gram model

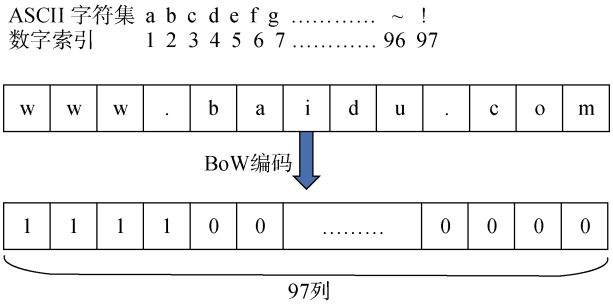

图 4　BoW 编码模型处理过程示意图

Figure 4　Schematic diagram of BoW model processing

3.3 基于 CNN-BiLSTM 算法的检测模型

根据检测算法的模型整体结构, 本节分别详细介绍不同模块的设计细节, 从嵌入层后的向量输入到生成检测结果, 模型可分为特征提取模块和特征分类模块。

3.3.1 特征提取模块构造

(1) 卷积层设计与分析

检测模型中最重要的部分可以说是特征提取模块了, 特征提取是否全面准确关系到后续检测模型的训练效果。特征提取模块主要是将字符编码处理后的矩阵 X 通过卷积运算提取局部特征, 并将提取到的局部特征运输到后续模块中。根据查阅相关研究资料, 随着卷积层数的递增, 特征提取的复杂程度也呈现递增, 即逐步从“简单特征”提取到“复杂特征”[18], 因此本文设计选择连续使用两个卷积层进行特征提取。

特征提取模块的具体过程大致如下, 我们设定模型的输入向量为 X 且维度为 $m \times n$。由于本文设计的两层卷积层分别是从“简单向量”向“复杂向量”的提取过程, 因此在设计卷积窗口大小时作出以下判断, 第二层卷积主要对第一层提取到特征作精细处理, 因此我们将第二层卷积窗口 k_2 的大小设置为 1, 因此卷积核 F 的大小为 $1 \times c_2$; 对于第一层卷积窗口 k_1 的大小设置, 我们选择组合窗口为(2, 4)尺寸的卷积窗口大小。卷积步长均为 $step_1$, 卷积深度分别为 dp_1 和 dp_2。输入向量 X 通过卷积计算后分别得到卷积向量 C 和 C', 具体卷积计算如下式所示:

$$C_i = f\left(W_1 \cdot C_{i \times step_1 + c_1 - 1} + b_1\right)$$

$$C_i' = f\left(W_2 \cdot C_{i \times step_1 : i \times step_1 + c_2 - 1} + b_2\right)$$

$$O_t \in W, W = \left[O_1, O_2, \cdots, O_n\right]$$

该式中, f 为卷积层的激活函数; W_1 和 W_2 分别为卷积核的权重矩阵且 $W_1 \in R^{k_1 \times c_1}$, $W_2 \in R^{1 \times c_2}$; b_1 和 b_2 为卷积的偏置项且 $b_1 \in R$, $b_2 \in R$

(2) 池化层设计与分析

由于恶意网页检测工作面临的一大挑战即是在特征提取的过程中忽视了上下文信息和位置信息, 因此一些较长 URL 或者参数信息在特征提取的效果上不尽如人意。检测模型在通过大量的卷积计算提取到特征信息后, 一方面往往需要池化层来筛选去除冗余特征[19], 另一方面需要提取到检测目标的时序特征, 因此结合了以上两种需求, 本文设计使用 BiLSTM 算法替换池化层。

通常对于较长恶意 URL 的特征提取工作中, 字符前后之间存在一定相关性, 往往需要关联其上下文信息, 从而提高对分类的细粒程度。因此对于字符处于不同位置产生的前后顺序信息进行提取, 即提取高阶特征中的时序特征。

BiLSTM 其中一个单元的结构原理如图 5 所示。根据双向长短期记忆神经网络的模型结构可知, 前向 LSTM 层的输出和反向 LSTM 的输出共同构成了神经元的输出[20], 其计算公式如下式所示:

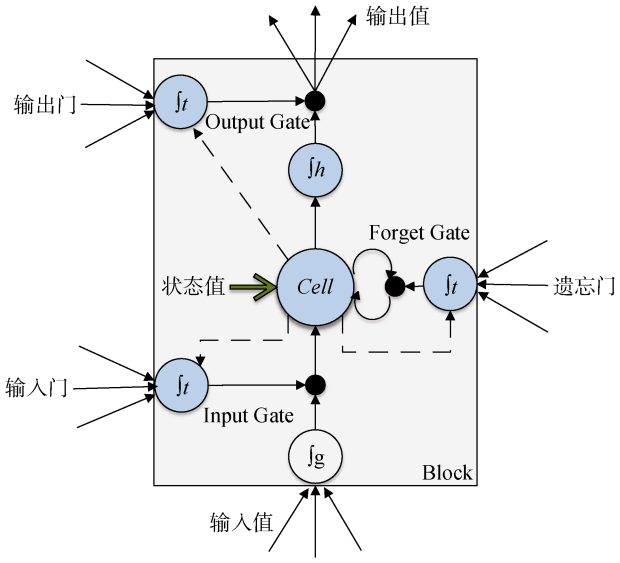

图 5 BiLSTM 模型中任一单元的结构原理图

Figure 5 Schematic diagram of the structure of any unit in the Bidirectional LSTM model

$$h_t = f\left(w_1 x_t + w_2 h_{t-1}\right)$$

$$h_t' = f\left(w_3 x_t + w_5 h_{t-1}'\right)$$

$$O_t = g\left(w_4 h_t + w_6 h_t'\right)$$

其中 w_i 为不同层对应的权值, 输入到向前和向后隐含层为(w_1 , w_2), 隐含层到隐含层自己为(w_3 , w_5), 向前和向后隐含层到输出层为(w_4 , w_6)。其中 $O_t \in W, W = \left[O_1, O_2, \cdots, O_n\right]$ 为 BiLSTM 网络提取到的时序特征。为了捕获原始数据的上下文关系并同时达到池化效果, 我们从提取的高阶特征选择 BiLSTM 网络的最后一个单元作为结果输出给后续神经网络模块进行分类。

3.3.2 特征分类模块构造

在特征分类模块中, 本模型在对于全连接层采用两层神经网络进行分类。其中以特征提取模块提取出的特征值作为全连接层神经元的输入, 通过增加一个隐藏层, 再与后续的 Softmax 激活层进行全连

接。本文之所以选择在全连接层增加隐藏层, 是由于理论证明, 两层神经网络可以无限逼近任何连续函数。全连接层中每个神经元的输入与前一层的神经元的输出都是相互连接的, 这样就形成了输入和输出之间线性关系, 进而可以得到特征映射的结果。特征分类模块的具体原理示意图如图 6 所示。

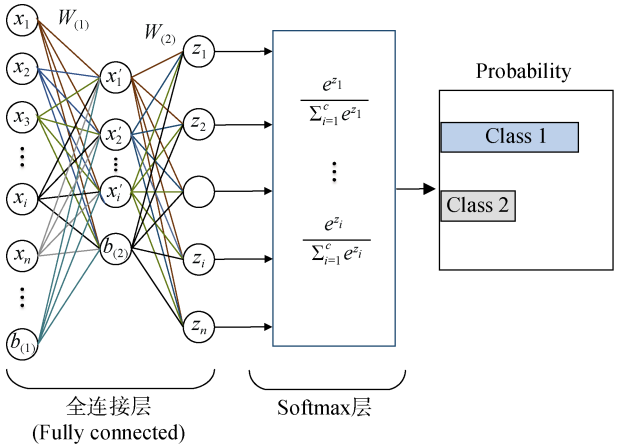

图 6　特征分类模块的原理示意图

Figure 6　Schematic diagram of the feature classification module

本模块的输入数据为特征提取模块提取出的特征数值, 此特征数值为 x_i, 特征数值计算后进入第二层隐藏层后为 $x_i^{'}$, 再次经过运算后最终输出为 z_i。如图 6 所示, 每层的最后一个神经元只起到存储作用, 通过此节点进行偏置参数为 $b_{(1)}$ 和 $b_{(2)}$ 的计算, 其本身不参加神经网络的计算, 被称为偏置节点。在正向传播的过程中, 第一层的权重矩阵为 $W_{(1)}$, 第二层的权重矩阵为 $W_{(2)}$。

第一层的计算过程如下式所示:

$$x_i^{'} = W_{(1)}x_i + b_{(1)}$$

第二层的计算过程如下式所示:

$$z_i = W_{(2)}x_i^{'} + b_{(2)}$$

通过以上两层神经网络的计算最终得到特征的映射结果, 此时模型已经完成由数字特征向输出类别的拟合过程, 这个拟合过程就是通过梯度下降法逐步在大量的样本中找到最佳拟合方式, 最佳方式将确定模型神经元之间的权重, 然后完成映射结果的输出, 该映射结果为每个神经元的激活值。需要再进行后续激活层的操作实现最终分类归一化。

3.4　基于代价敏感学习策略的检测模型

通常卷积神经网络对于训练模型最小化整体误差的方法为利用反向传播算法[21], 反向传播算法是一种建立在梯度下降基础之上的算法, 其实质是输入到输出之间的由 n 维欧式空间向 m 维欧式空间的映射关系, 最终使模型达到最优训练效果。我们假设原始数据集包含数量为 N 个的样本, 其中样本被划分为 K 种类别, 其具体运算过程下式所示:

$$X = \underset{j=1}{\overset{K}{U}} X_{cj}$$

$$N = \sum_{j=1}^{K} N_j$$

此训练集在学习模型中的整体误差 E 为以下计算过程如下式所示:

$$E = \frac{1}{N}\sum_{i=1}^{N} E(X_i)$$

通常当样本数据分布均衡时, 此类误差计算方法可以发挥好的效果, 实现优化的效果。但对于像恶意检测这类样本数据, 实际应用中恶意样本数量远远小于正常样本, 很容易造成训练模型“偏向”正常样本, 对其产生依赖性, 造成最终实验效果的假阳现象[22]。因此针对此问题, 本文对不同类别的样本数据分配不同的权重, 即为其赋予不同的惩罚代价, 这样可以使整体模型更加“偏向”于少数样本。

本文在构建代价敏感策略时给每个样本均分配不同的权重, 并根据神经网络模型学习过程中的更新预判错误率对权重不断进行调整, 并且本文对更新后的样本数据权重进行了相应的归一化处理。

因此, 训练集中的整体误差函数即变为下式所示:

$$E = \frac{1}{N}\sum_{i=1}^{N}\left(E(X_i)\cdot W_l(X_i)\right)$$

当模型迭代训练到第 l 次时, 其对于样本 X_i 的权重为 $W_l(X_i)$, 初始权重即为迭代次数为 $l=0$ 时的值, 即 $W_0(X_i)$。

以传统的代价敏感学习算法 Adaboost 算法为例, 其对于同一类别之间的每个样本所设置的初始权重均为 $\frac{1}{N}$, 而本文根据需求要对类内的样本设置不同初始权重, 该初始权重 $W_0(X_i)$ 为下式所示:

$$W_0(X_i) = \frac{1}{KN_j}$$

其中 $X_i \in X_{cj}$, 当该计算完成第一次迭代训练后, 算法根据当前已经训练好的神经网络模型对训练集进行测试, 并计算错误率 ε 为下式所示:

$$\varepsilon = \frac{N_{error}}{N}$$

在错误率小于一定阈值时，权重将重新进行计算。由于本文为针对 URL 进行的二分类检测，因此此处设置阈值为 0.5，即 $\varepsilon \leqslant 0.5$。此时我们判定该测试结果为困难学习样本，需为此提高权重，更新参数为下式所示:

$$\alpha = 0.5 \ln \frac{1-\varepsilon}{\varepsilon}$$

随着样本迭代更新，权重更新规则如下式所示:

$$W_l(X_i) = W_{l-1}(X_i) e^{-\alpha T(d_i, y_i)}$$

$$T(d_i, y_i) = \begin{cases} 1 & d_i = y_i \\ 0 & d_i \neq y_i \end{cases}$$

其中 X_i 代表原始数据中的某一个样本，$W_l(X_i)$ 表示第 l 次迭代的权重值，d_i 表示该样本的真实分类类别，而 y_i 则表示为该样本的预测结果类别。

根据整体的误差函数公式可以得出，网络模型的整体误差为所有样本误差的加权和。为了加强神经网络模型对于困难样本的学习，提高困难样本在整体误差函数中的比重，我们需要在更新权重后再次进行归一化处理，此步骤可以使每个类别的样本的权重和为 $\frac{1}{K}$，即如下式所示:

$$\sum_{X_i \in X_{cj}} W_l(X_i) = \frac{1}{K}$$

本文根据以上规则细节优化本文的代价敏感策略，选择更适合本文模型的代价敏感策略，在模型训练过程中增加对误分类样本的“注意力”，以此优化整体模型效果，并且为了保证模型效率，本文最终区别于 Adaboost 算法对于弱分类器进行集成分类的方法，仅选择最优的训练模型作为算法的输出模型[23]。这样既满足了模型的需求，同时保证了检测效率。

4 实验

4.1 数据集概述与预处理

本文采用了公开数据集 HTTP Dataset CSIC 2010 数据集(http://www.isi.csic.es/dataset/)和 Adam 利用模拟 API(Application Programming Interface)构造的访问日志数据集。HTTP Dataset CSIC 2010 数据集中总计 61000 条数据，其中包含了正常请求 36000 个，恶意请求为 25000 个，它是由西班牙研究委员会(CSIC)信息安全研究所制作的。Adam 构造的模拟数据集是通过 API 来模拟一个简单的电子商务应用程序，从而进行请求访问生成相应的访问日志，该数据集中包含 23000 条日志数据，其中恶意样本 11330 条，正常样本 11670 条。

由于 HTTP DATASET CSIC 2010 数据集和 Adam 数据集中包含了多个 Web 参数，大量复杂且冗余的信息一定程度上会影响后续检测效率，因此需要对数据进行适当的清理与预处理工作，将数据集转换为标准格式，以方便后续模型检测的数据处理工作。本文将该数据集中与恶意检测关联不大的参数进行剔除，保留与恶意检测关联较大的参数作为模型的输入数据，并将原始数据处理为标准化格式的 TXT 文件。

4.2 特征数据实验对比

根据研究现状调研显示，已有针对恶意网页的检测大多均针对 URL 进行检测，近些年的研究多结合 HTML 代码以及 JS 代码进行特征检测，而大部分恶意网页中 HTTP 请求参数中同样含有恶意载荷，因此对于 HTTP 请求参数的特征同样无法忽视。为了验证 HTTP 请求参数之间的关联性且对于恶意网页检测的影响，证明 HTTP 请求参数作为特征输入同样能丰富扩充检测特征，对检测效果带来一定提升，本文选择仅包含 URL 信息的数据集进行对比验证。

从以上实验结果图 7 可以看出，结合 HTTP 请求参数的特征数据的确可以在一定程度上提高检测准确率，单纯 URL 数据的表现结果明显次于结合 HTTP 请求参数的数据。当然不可否认的是单纯 URL 数据在收敛速度上高于结合 HTTP 参数的数据，但

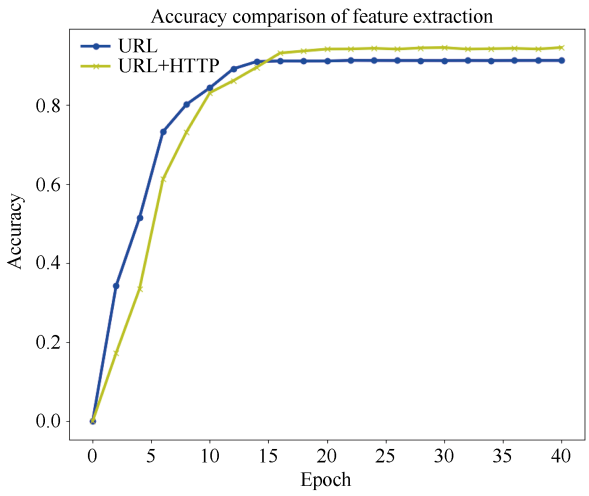

图 7 不同特征数据集准确率对比

Figure 7 Comparison of accuracy of different feature data sets

是二者在最终都可以达到较好的收敛效果，因此收敛速度对整体模型影响不大。综合对比来看，结合 HTTP 参数的特征数据不仅能提高相应的准确率，同时又能保证整体模型检测的效率，很好的证明了 HTTP 请求参数在恶意载荷上的关联。

4.3 字符化向量方法实验对比

由于本文需要训练出整体检测效果最佳的检测模型，因此需要对数据特征的字符化向量处理方法进行实验对比，实验结果如图 8 所示。

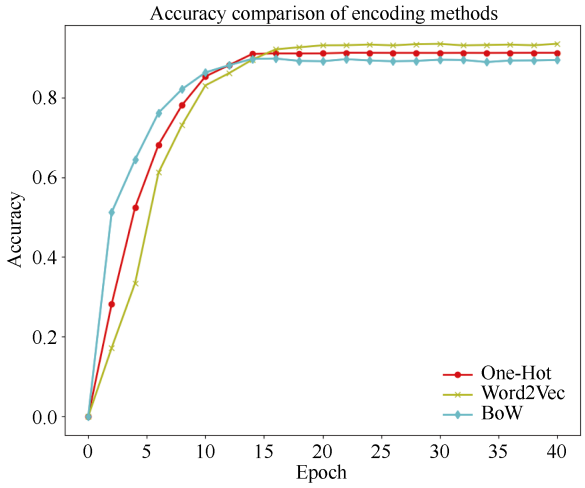

图 8 编码处理方式的准确率对比

Figure 8 Accuracy comparison of encoding processing methods

从上图可以看出三种字符编码方式在迭代 10 次左右基本实现收敛，在收敛速度上可以看出 BoW 编码方式的收敛速度最快，Word2Vec 收敛速度稍慢，其中 Word2Vec 取得的准确率最高。由于其收敛速度上差别较小，对于模型效率影响不大，因此准确率作为第一评估标准。对于三种编码方式，其效果最优的为 Word2Vec 的 Skip-Gram 模型，本文选用此编码方式实现后续深度模型构造。

4.4 检测模型效果实验对比

(1) 与其他神经网络模型对比分析

通过调研现今检测分类的工作，大多使用单个神经网络模型或多个神经网络混合模型[24]。现今单个神经网络模型最常用的为 CNN 模型与 LSTM 模型，多个神经网络混合模型最常使用的为 CNN-LSTM 模型，此混合模型通常在卷积神经网络模型后通过在全连接层后再次接入长短期记忆神经网络模型，由此组合成混合神经网络模型。神经网络模型均采用一个卷积层，一个池化层，一个全连接层，激活函数使用 ReLU(Rectified Linear Unit)函数，全连接层使用 Softmax 分离器。其中卷积层设置卷积窗口 *filter* 的长度 $k=4$，卷积核大小为 128，初始的学习率大小为 0.001，*dropout* 大小为 0.5，经过 40 轮次的迭代训练。针对以上三种神经网络模型与本文的设计模型进行对比，其具体实验结果数据如下图 9 所示。

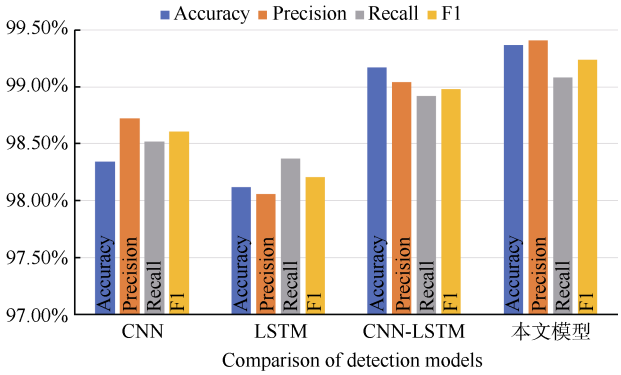

图 9 三种神经网络模型与本文模型实验结果对比柱状图

Figure 9 Comparison of the experimental results of the three neural network models and my model

为了更加全面且立体的对模型进行评价，我们引入了四种评价指标，分别为准确率、精确率、召回率以及 F1，由图 9 的实验结果综合来看本文设计的模型在最终检测效果上均取得了较优的表现。

(2) 与其他论文方法对比分析

为验证本章构建的基于 CNN-BiLSTM 的恶意网页检测模型方法的效果。本文同其他文献的恶意 URL 检测方法进行了相应的对比分析，以验证本章构建的检测模型在经过不同角度的设计及优化后的良好效果。

文献[25]将注意力机制引入到 LSTM 算法模型中，并提出了 URL2Vec 的向量化方法。文献[6]提出的集成学习方法的检测模型，采用特征融合并使用多种检测手段相结合的方法进行恶意 URL 检测。文献[26]采用本文相同的数据集，构建了基于 CNN 的多层复杂网络模型 Multilayer-CNN，其中包含 3 层卷积层，2 层最大池化层、2 层 Dropout 层以及 3 层全连接层。文献[27]提出一种基于卷积神经网络和独立递归神经网络的双向 LSTM 算法(CBIR)，提取了表示 URL 内容相似性的 Btexture 指纹特征进行恶意网页检测。本文针对以上恶意 URL 检测方法进行了准确率对比，对比结果如表 1 所示。

由表 1 中的实验结果准确率分析可知，相比其他恶意 URL 检测分类的研究方法，本文模型在对于 URL 检测的准确率上高于多数检测模型，由此说明本文构建的基于 CNN-BiLSTM 的恶意网页检测模型在检测效果上取得了良好的表现。我们发现文献[26]

表 1　与其他论文模型对比结果

Table 1　Comparison result with models of other papers

恶意 URL 检测方法	Accuracy
LSTM-Attention	99.14%
集成学习方法	95.7%
Multilayer-CNN 模型	99.87%
CBIR 模型	98.45%
本文模型	99.37%

构建的 Multilayer-CNN 网络模型在准确率上高于本文模型, 但由于其设计了较多层的神经网络模型, 模型训练过程中必然需要大量的卷积运算, 其检测效率有待验证, 且单从准确率维度评价模型并不客观, 因此本文在下节内容验证了该模型的检测效率问题。

本文与文献[26]均采用了相同的恶意 URL 检测数据集, 本文根据其模型具体设置, 进行检测时间性能的实验对比。采用相同的训练集与测试集, 根据每千条数据的模型检测时间耗费来对比模型的检测效率, 以此得到的实验结果如图 10 所示。

由实验结果图 10 可以看出, 本文模型对每千条数据检测的运行时间平均约为 40s 左右, 而文献[26]构建的 Multilayer-CNN 模型对每千条数据检测的运行时间平均为 70s。可以看出本文模型与 Multilayer-CNN 模型在检测准确率上效果差别并不大, 但在检测效率上明显优于其模型。因此也可以验证本文模型在提高准确率的同时保证了模型的检测效率, 这也符合本文模型优化的整体目标。

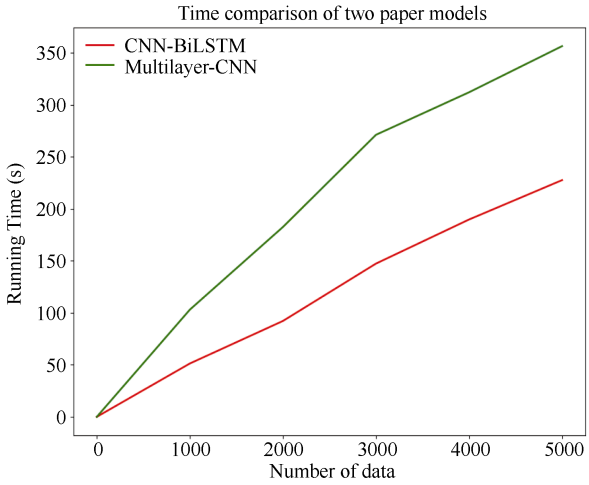

图 10　两个论文模型的每千条数据运行时间对比

Figure 10　Comparison of running time per thousand data of two paper models

4.5　不均衡数据集实验对比

为了验证本文构建的神经网络模型在结合代价敏感策略后的检测效果, 我们将构造不同失衡比例的数据集进行验证, 以此来判别本文神经网络模型在引入代价敏感策略后抵抗不同失衡比例数据的能力。对此我们分别构造了 6 种比例的数据集进行验证, 数据集具体数据比例如下表 2 所示, 其中大类数量为正常样本数量, 小类数量为恶意样本数量。

表 2　不同失衡比例数据集详情

Table 2　Details of data sets with different imbalance ratios

数据集编号	样本数量	大类数量	小类数量	失衡比例
数据集 1	46000	36000	10000	3.6
数据集 2	42000	36000	7000	5.14
数据集 3	41000	36000	5000	7.2
数据集 4	30800	27700	3100	8.94
数据集 5	51330	47000	4330	10.85
数据集 6	64200	60000	4200	14.29

对于神经网络模型的训练参数选择前文工作基础的参数, 为此我们设置迭代次数 Epoch=100, 学习率设置为 0.001。分别将六种数据集的数据作为模型训练的输入进行学习, 我们构建测试集选择 15000 条数据进行测试, 其中正常样本 12000 条, 恶意样本 3000 条, 以此进行性能测试, 其测试结果如下表 3 所示。

由表 3 的评价数据可得出, 本文的神经网络模型在检测分类时对于失衡比例达到 10 的数据集依然有较优秀的整体表现效果, 在失衡比例上升到 15 左右时, 模型的检测率以及召回率等指标均呈现一定程度的退化, 但此退化仍然在可承受范围内。同时由数据集 1 的检测效果可以看出, 本文的代价敏感策略在一定程度上提高了检测模型的准确率、精确率以及召回率等指标效果, 并且对于小类样本的检测准确率也达到很好的效果。以此可以说明本文的检测模型可以抵御较多数据不均衡的情况, 并且随着对恶意样本及困难学习样本施加惩罚代价(权重), 模

表 3 不同数据集在本章模型上的验证效果

Table 3 The verification effect of different data sets on the model of this chapter

数据集编号	Accuracy	Precision	Recall	F1	SE	SP
数据集 1	99.61%	99.42%	99.51%	99.46%	99.36%	99.51%
数据集 2	99.53%	99.37%	99.45%	99.41%	99.29%	99.45%
数据集 3	99.44%	99.31%	99.37%	99.34%	99.25%	99.37%
数据集 4	99.38%	99.13%	99.25%	99.19%	99.04%	99.25%
数据集 5	99.11%	98.92%	99.01%	98.96%	98.46%	99.01%
数据集 6	97.89%	96.25%	97.83%	97.03%	96.03%	97.83%

型对于困难样本的学习更加关注, 这使得检测工作整体效果实现优化。

为了验证代价敏感策略对于在处理数据不均衡问题的良好表现, 本文对于前文构建的 CNN-BiLSTM 检测模型与结合代价敏感策略的模型进行对比验证。对此我们首先构建不均衡数据集 A, 训练集 A1 包含 41000 条数据, 其中正常样本 36000 条, 恶意样本 7000 条, 其大致比例为 5∶1; 测试集 A2 选择 15000 条数据, 其中正常样本 12000 条, 恶意样本 3000 条。根据此数据集分别对模型进行训练。设置训练迭代轮次 Epoch=100, 训练结果在同样的评价指标下呈现的效果如图 11 所示。

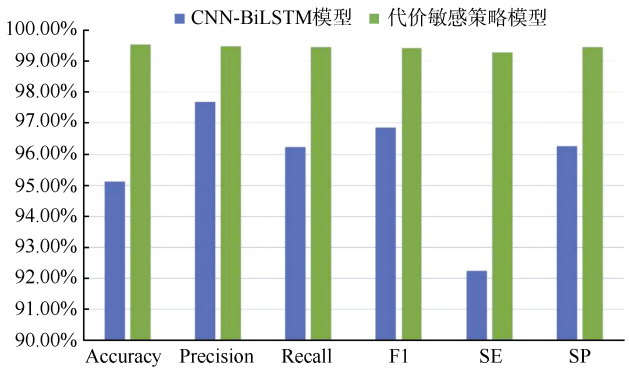

图 11 代价敏感策略引入神经网络模型的影响对比

Figure 11 Comparison of the impact of cost-sensitive strategies introduced into network models

由以上结果图 11 可以看到, 当使用不平衡数据集对 CNN-BiLSTM 检测模型进行训练时, 随着小类样本的数据减少, 对于恶意样本的检测力降低, 其模型能力明显发生了退化。

由损失函数对比结果图 12 所示, 引入代价敏感策略明显降低了模型的整体损失函数。对比而言, 引入了代价敏感策略的模型在检测水平上相较于 CNN-BiLSTM 模型有了明显的进步, 由于我们加强对类内样本之间差异的关注, 这使得模型不仅仅关注少数恶意类别, 更加关注了样本中的困难学习样本, 这使得检测模型整体检测效果的提升。

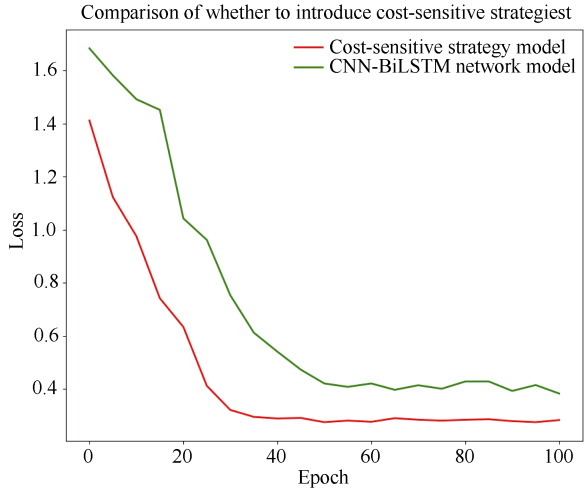

图 12 代价敏感策略引入神经网络模型的损失函数对比

Figure 12 Comparison of the loss function of the two network models

本文使用上文构建的不均衡数据集 A 以及相同的训练集 A1 和测试集 A2, 对文献[26]构建的模型 Multilayer-CNN 进行模型训练与测试, 并与本文引入代价敏感学习的检测模型进行实验对比分析。

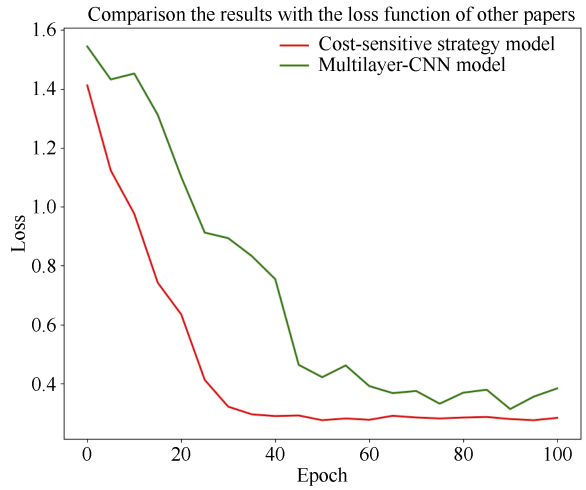

图 13 与其他论文模型的损失函数对比

Figure 13 Comparison of the loss function of the two network models

实验结果显示 Multilayer-CNN 模型的准确率仅达到了 96.12%, 模型整体检测效果明显发生的退化, 因此该检测模型仅适合理想的检测场景, 而基于代价敏感策略的检测模型在准确率效果上并未受到影响。且本文对比了二者的损失函数如图 13 所示, 引入代价敏感策略的检测模型明显优于 Multilayer-CNN 模型的整体损失函数, 由此验证了代价敏感策略在实际应用环境中的适用性。

5 结论

本文的研究目标是对恶意网页进行检测并准确分类, 同时保证模型的检测效率, 基于此研究目标, 本文设计了基于代价敏感学习策略的 CNN-BiLSTM 检测模型。本文将 HTTP 请求参数与恶意 URL 相结合作为原始数据进行特征提取, 通过与单纯 URL 进行特征提取检测的实验对比, 验证了 HTTP 请求参数的结合对于恶意网页检测的可行性。设计并构造了基于 CNN-BiLSTM 的神经网络模型, 优化了特征提取模块与特征分类模块的相应工作, 提高了模型的检测准确率且保证了模型检测效率。并在此基础上引入了代价敏感策略, 使模型更加适用于实际应用的数据环境, 增强了整体检测模型的适用性与可扩展性。

在本文的研究过程中, 仍然有很多想法有待未来考究, 为此仍有很多不足之处以及需要改进的地方, 如本文对于恶意网页的检测仍然停留在二分类问题上, 只能判别网页的恶意性, 但无法具体检测划分恶意类别。且在代价敏感策略的引用上仅仅解决了数据分布不均衡的问题, 可以使模型进行进一步的主动学习, 适应数据的不断更新迭代, 进一步增强模型的可扩展性。

致 谢 这项工作得到了中国国家铁路集团有限公司科技研究开发计划重点课题(No. N2020W005), 以及保密测评中心红果园项目(No. K20GY500010)的支持。此篇文章从选题、构思、撰写到最终的定稿, 得到了老师们很多宝贵意见。人生里遇到的每一位良师都将影响自己的终生, 在此对每一位老师表示深深的感谢。

参考文献

[1] Jia Y, Liu S C, Jiang S Y. Analysis of the Development Status of Artificial Intelligence Technology at Home and Abroad[C]. *2019 International Conference on Virtual Reality and Intelligent Systems*, 2019: 195-198.

[2] Roldán J C, Jiménez P, Corchuelo R. On Extracting Data from Tables that are Encoded Using HTML[J]. *Knowledge-Based Systems*, 2020, 190: 105157.

[3] Mo Y L. *SVM and neural network based URL safety detection*[D]. Beijing: Beijing University of Posts and Telecommunications, 2016.
(莫玉力. 基于 SVM 和神经网络的 URL 安全检测[D]. 北京: 北京邮电大学, 2016.)

[4] Chen K, Fu H Z, Xiang Y. Malicious URL Detection Based on Deep Learning[J]. *Computer Systems & Applications*, 2018, 27(6): 27-33.
(陈康, 付华峥, 向勇. 基于深度学习的恶意 URL 识别[J]. *计算机系统应用*, 2018, 27(6): 27-33.)

[5] Li R. *Malicious URL detection based on DPI data*[D]. Beijing: Beijing University of Posts and Telecommunications, 2019.
(李苒. 基于 DPI 数据的恶意 URL 检测方法研究[D]. 北京: 北京邮电大学, 2019.)

[6] Yi L. *Design and implementation of malicious web detection system based on integrated*[D]. Chengdu: University of Electronic Science and Technology of China, 2020.
(易磊. 集成学习的恶意网页检测系统的设计与实现[D]. 成都: 电子科技大学, 2020.)

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

[8] Sun B, Akiyama M, Yagi T, et al. AutoBLG: Automatic URL Blacklist Generator Using Search Space Expansion and Filters[C]. *2015 IEEE Symposium on Computers and Communication*, 2016: 625-631.

[9] Vanhoenshoven F, Nápoles G, Falcon R, et al. Detecting Malicious URLs Using Machine Learning Techniques[C]. *2016 IEEE Symposium Series on Computational Intelligence*, 2017: 1-8.

[10] Jayakanthan N, Ramani A V, Ravichandran M. Two phase classification model to detect malicious URLs[J]. *International Journal of Applied Engineering Research*, 2017, 12(9): 1893-1898.

[11] Anderson H S, Woodbridge J, Filar B. DeepDGA: Adversarially-Tuned Domain Generation and Detection[C]. *The 2016 ACM Workshop on Artificial Intelligence and Security*, 2016: 13-21.

[12] Kim S, Kim J, Kang B B. Malicious URL Protection Based on Attackers' Habitual Behavioral Analysis[J]. *Computers & Security*, 2018, 77: 790-806.

[13] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[C]. *Curran Associates Inc. Proceedings of the 25th International Conference on Neural Information Processing Systems*, 2012: 1097-1105.

[14] Rajalakshmi R, Raymann J, Prabu A, et al. Deep URL: Design of Adult URL Classifier Using Deep Neural Network[C]. *The International Conference on Advanced Information Science and System*, 2019: 1-5.

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

[16] Al-Matham R N, Al-Khalifa H S. SynoExtractor: A Novel Pipeline for Arabic Synonym Extraction Using Word2Vec Word Embeddings[J]. *Complexity*, 2021, 2021: 1-13.

[17] Li W S, Dong P, Xiao B, et al. Object Recognition Based on the Region of Interest and Optimal Bag of Words Model[J]. *Neurocomputing*, 2016, 172: 271-280.

[18] Young T, Hazarika D, Poria S, et al. Recent Trends in Deep Learning Based Natural Language Processing [Review Article[J]. *IEEE Computational Intelligence Magazine*, 2018, 13(3): 55-75.

[19] Murray N, Perronnin F. Generalized Max Pooling[C]. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014: 2473-2480.

[20] Shengwei T, Xingfa Z, Long P, et al. Causal relationship extraction based on bidirectional LSTM in Uighur language [J]. *J Electron Inf Technol*, 2018, 40(1): 200-208.

[21] Sornborger A, Tao L, Snyder J, et al. A Pulse-Gated, Neural Implementation of the Backpropagation Algorithm[C]. *The 7th Annual Neuro-inspired Computational Elements Workshop*, 2019: 1-9.

[22] Pei W B, Xue B, Shang L, et al. Genetic Programming for Development of Cost-Sensitive Classifiers for Binary High-Dimensional Unbalanced Classification[J]. *Applied Soft Computing*, 2021, 101: 106989.

[23] Khan S H, Hayat M, Bennamoun M, et al. Cost-Sensitive Learning of Deep Feature Representations from Imbalanced Data[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2018, 29(8): 3573-3587.

[24] He K M, Zhang X Y, Ren S Q, et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[C]. *2015 IEEE International Conference on Computer Vision*, 2016: 1026-1034.

[25] Liu Y. *Research on malicious URL detection based on LSTM*[D]. Wuhan: Central China Normal University, 2019.
(刘煜. 基于 LSTM 的恶意 URL 检测研究[D]. 武汉: 华中师范大学, 2019.)

[26] Cui Y P, Liu M, Hu J W. Malicious Web Request Detection Technology Based on CNN[J]. *Computer Science*, 2020, 47(2): 281-286.
(崔艳鹏, 刘咪, 胡建伟. 基于 CNN 的恶意 Web 请求检测技术[J]. *计算机科学*, 2020, 47(2): 281-286.)

[27] Wang H H, Yu L, Tian S W, et al. Bidirectional LSTM Malicious Webpages Detection Algorithm Based on Convolutional Neural Network and Independent Recurrent Neural Network[J]. *Applied Intelligence*, 2019, 49(8): 3016-3026.

蔡勋萌 于 2021 年在北京交通大学计算机与信息技术学院软件工程专业取得硕士学位。研究方向主要为网路安全、Web 检测技术等。Email: 19140044@bjtu.edu.cn.

王健 (1975−), 男, 山东烟台人, 博士, 北京交通大学副教授、博士生导师。主要研究方向为密码应用及区块链、网络安全。Email: wangjian@bjtu.edu.cn.

李鹏博 于 2019 年在上海大学电子与通信工程专业取得硕士学位。2020 年就职于中电科网络空间安全研究院。主要研究方向为网络安全、网络靶场等。Email: 996751780@qq.com