文章编号: 1671-6833(2021)06-0014-07

基于 CNN 和 BiLSTM 的钓鱼 URL 检测技术研究

卜佑军[1], 张 桥[1,2], 陈 博[1], 张稣荣[1], 王方玉[2]

(1.中国人民解放军战略支援部队信息工程大学,河南 郑州 450001; 2.郑州大学 中原网络安全研究院,河南 郑州 450001)

摘 要: 为了解决日益严峻的网络钓鱼问题,提出一种基于卷积神经网络(CNN)和双向长短记忆网络(BiLSTM)的钓鱼 URL 检测方法 CNN-BiLSTM。该方法首先基于敏感词分词的方法对 URL 分词,根据特殊字符和敏感词对 URL 进行单词级别划分,对其中的非敏感词进行字符级别划分,以获取特殊字符和敏感词的有效信息,提升利用 URL 数据信息的程度;然后将分词后的 URL 输入到 CNN 和 BiLSTM 中,通过 CNN 获取 URL 的空间局部特征,通过 BiLSTM 获取 URL 的双向长距离依赖特征,基于自动提取的特征检测钓鱼网页。实验结果表明:基于 CNN 和 BiLSTM 的钓鱼 URL 检测方法能够达到较好的检测效果,其准确率达到了 98.84%,精确率达到了 99.71%,召回率达到了 98.04%,*F*1 值达到了 98.86%。此方法相对于传统的机器学习和黑名单检测方法,无须人工提取特征且能识别新出现的钓鱼网页。

关键词: 钓鱼 URL; URL 分词; 卷积神经网络; 双向长短记忆网络

中图分类号: TP393 **文献标志码**: A **doi**: 10.13705/j.issn.1671-6833.2021.04.022

0 引言

近年来,互联网快速发展,在线购物、电子商务和网络社交等基于互联网的应用给人们的工作和生活带来了极大的便利。据中国互联网络信息中心 CNNIC 统计,截至 2020 年 3 月,中国网民规模达到了 9.04 亿,互联网普及率达到了 64.5%[1]。与此同时,网民信息亦面临着安全威胁,如网络攻击者通过网络钓鱼窃取个人敏感信息进而非法获取经济利益。截至 2020 年 8 月,中国反钓鱼联盟累计认定的钓鱼网站数量达到了 469 252 个[2]。因此,如何及时、有效地检测钓鱼网站已经成为亟待解决的问题。

目前,针对网络钓鱼,黑名单方法只需进行简单的数据库查询操作,是一种较为简单的检测方法。Malware Domain List 和 PhishTank 这 2 种算法使用的都是基于黑名单的检测方法[3-4]。然而目前网址生成算法比较成熟,每天都会出现大量的钓鱼网址,黑名单数据库无法及时包含所有的钓鱼网址。根据 Sheng 等[5]的研究,约 47%~83%的钓鱼网址在钓鱼事件发生 12 h 之后才被列入黑名单中。Aleroud 等[6]指出约有 93%的钓鱼网址没有被主流的黑名单收录。基于黑名单检测钓鱼网页的局限性在于要不断收集钓鱼网址样本并及时更新黑名单数据库。

针对黑名单方法存在的局限性,有研究人员使用机器学习方法来检测钓鱼网页。Liu 等[7]提取网页内链接关系、敏感词排序等特征,利用机器学习识别钓鱼网页,取得了较高的准确率及较低的误报率,实验结果表明,该方法可以识别 91.44%钓鱼网页。Ma 等[8]利用机器学习在多个公开数据集上测试,实验结果表明,该方法的检测准确率达到了 94%。该类方法使用机器学习技术达到了较高的检测准确率且能够识别未知的钓鱼网页,但也存在较大的局限性:①需要大量的手动特征工程,其中许多特征需要相关专家来确认;②需要获取网页内容,增加了客户端开销和风险且检测算法的时间复杂度高;③有些钓鱼网站能够隐藏其网页内容,即向不同的客户端提供不同的内容[9],比如,钓鱼网站可能会将合法页面发送给蜜罐客户端,但将钓鱼网页发送给其他人工访问客户端。

收稿日期: 2021-03-09; **修订日期**: 2021-05-15

基金项目: 国家重点研发计划项目(2017YFB0803201);国家自然科学基金资助项目(61572519)

作者简介: 卜佑军(1978—),男,河南焦作人,中国人民解放军战略支援部队信息工程大学副研究员,博士,主要从事网络安全、深度学习等研究,E-mail: 13140186091@126.com。

为了克服上述 2 种检测方法的弊端,已有研究者使用了深度学习技术,通过自动提取 URL 特征来判别其所属类别,以检测其对应网页是否为钓鱼网页。Kim[10] 于 2014 年利用 CNN 对文本进行分类,实验结果表明,CNN 在文本上具有较强的分类能力。此后有一些研究人员尝试使用 CNN 对钓鱼 URL 进行检测。Zhang 等[11] 利用单词级别的卷积神经网络对 URL 进行分类,即根据特殊字符对数据集中的 URL 进行单词级别的划分并形成一个语料库。训练语料库中的每个单词表示为一个向量,然后将待测 URL 分词,获取单词的向量表示并组合形成一个向量矩阵输入到卷积神经网络中来判断相应的 URL 所属类型。Cui 等[12] 利用字符级别的卷积神经网络检测恶意 URL,即将 URL 按字符划分,获取每个字符的向量且组合形成一个向量矩阵,然后将此矩阵输入到卷积神经网络中来判断相应的 URL 所属类型。Yu 等[13] 在对恶意域名的检测实验中对比了多种深度学习模型,如 CNN、RNN,在这些实验中,基于深度学习的检测方法均优于基于手工特征的传统机器学习方法。

虽然上述工作已经取得了较好的表现,但仍然存在以下 3 个问题: ①基于单词划分 URL 在测试时无法获得新出现的单词的嵌入向量,基于字符划分 URL 会导致 URL 中一些特有的敏感词丢失有效信息; ②无法获取特殊字符的分布与类型及与周围词的前后关系; ③URL 是一种序列数据,数据之间存在着长距离依赖关系, CNN 无法获取 URL 数据的长距离依赖关系。

针对以上问题,本文提出了一种基于卷积神经网络(convolution neural network, CNN) 和双向长短记忆网络(bi-directional long short-term memory, BiLSTM) 的钓鱼 URL 检测方法 CNN-BiLSTM。该方法通过 CNN 来获取 URL 的空间局部特征,通过 BiLSTM 获取 URL 的长距离依赖特征。此外,对 URL 的分词方法做了改进,提出了一种基于敏感词分词的方法,有效提升了 URL 数据信息的利用程度。实验中通过与传统机器学习方法和单一模型的比较表明了所提方法的有效性。

1　字符词向量

深度学习模型只能处理经过数值化的向量,因此在对 URL 数据提取特征时需要先将其分词、编码并转化为 d 维词向量,用不同词在 d 维空间的距离来表示它们之间的语义相似度。当前使用深度学习方法检测 URL 常用的分词方法有基于单词划分 URL 和基于字符划分 URL 2 种。

基于单词划分 URL 使其转化为单词级词向量,利用特殊字符分割 URL 可能会使单词的数量相当大,造成该数据集的特征也按比例增大,通常会大于相应训练数据集中 URL 的数量,导致在进行特征向量的转换时内存受到限制,在测试集上无法获得新出现单词的嵌入向量。

相比于按单词划分 URL,基于字符划分 URL 使 URL 转化为字符级词向量能够在测试集上获得新的 URL 的嵌入向量,避免了无法从不可见的单词中提取特征的问题。另外由于字符总数是固定的,在进行特征向量的转换时不会受到内存的限制且字符级分类器的大小保持固定。但是将 URL 划分为单个的字符会导致一些敏感词如 login、password、registed 等丢失部分有效信息,因此,根据字符划分 URL 不足以使神经网络分类器从 URL 字符串中获取较为全面的信息。

针对上述分词方法存在的问题,本文提出了一种基于敏感词分词的方法,如表 1 中以网址 www.ccd.cn.bank.com 为例。首先根据特殊字符和敏感词对 URL 进行单词级别划分,并将特殊字符看作单词处理以获得特殊字符的有效信息。然后对其中的非敏感词进行字符级别划分,而将其中的敏感词作为一个整体与其余字符进行区分,这样能够明显标记 URL 中的重点信息,有利于神经网络分类器提取更具有代表性的特征。

表 1　URL 的 3 种分词方法

Table 1　Three methods of URL segmentation

URL 分词方法	分词结果
基于单词划分	www　ccd　cn　bank　com
基于字符划分	w w w . c c d . c n . b a n k . c o m
基于敏感词划分	w w w . c c d . c n . bank . c o m

2 模型结构

基于CNN-BiLSTM检测URL类别的模型框架包括4个部分。URL输入依次经过词嵌入层、卷积神经网络层、循环神经网络层和全连接层,最终输出URL的分类结果。其中循环神经网络层采用长短期记忆网络,各层网络的细节如下所述。

2.1 词嵌入层

URL本质上是由一系列字符或由特殊字符分隔的单词组成。词嵌入层将 $\boldsymbol{U}$ 转换为神经网络能够识别的特征向量,即得到它的嵌入矩阵表示 $\boldsymbol{U}\rightarrow\boldsymbol{X}\in\mathbf{R}^{L\times K}$,使得矩阵 $\boldsymbol{X}$ 包含一组相邻分量 $\boldsymbol{x}_i(i=1,2,\cdots,L)$,其中 $\boldsymbol{x}_i$ 为URL中的字符或单词的向量表示,$\boldsymbol{x}_i\in\mathbf{R}^K$ 为 K 维向量。本文根据URL数据集和敏感词汇表(account,admin,administrator,auth,bank,client,confirm,cmd,email host,login,password,pay,private,registed,safe,secure,security,sign,service,signin,submit,user,update,validation,verification,webscr)确定每条URL中字符及关键字的总长度 L 为300。若 L 超过300,则在URL末尾将多余的字符截断;若 L 小于300,则在其末尾用<pad>标记作为附加词填充。若URL中出现未知字符,则用未知字符标记<unk>表示。根据映射表为字符和敏感词赋予唯一编码构建URL的编码矩阵,如式(1)所示:

$$\boldsymbol{U}'=(u_1',u_2',\cdots,u_{300}')。\tag{1}$$

式中:u_i' 为URL中字符或单词的编码。

随后将矩阵 $\boldsymbol{U}'$ 经词嵌入层转换为300×64的包含语义信息的二维稠密矩阵 $\boldsymbol{X}$,作为卷积层的输入,如式(2)所示:

$$\boldsymbol{X}=(\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_{300})。\tag{2}$$

式中:$\boldsymbol{x}_i$ 是64维列向量。

2.2 卷积网络层

如图1所示,将词向量矩阵输入到卷积神经网络中,通过卷积核从特征矩阵中自动提取局部特征,卷积核高度 h 设置为2,宽度与字符向量的维度一致为64,卷积核的数量为200,卷积核滑动步长设置为1。对于某个卷积核 f 在第 i 个滑动窗口处获取的URL嵌入矩阵设为 $\boldsymbol{X}_i$:

$$\boldsymbol{X}_i=[\boldsymbol{x}_i,\boldsymbol{x}_{i+1},\cdots,\boldsymbol{x}_{i+h-1}]。\tag{3}$$

式中:$\boldsymbol{x}_i$ 为字符或敏感词的向量表示。

通过卷积操作产生的新特征设为 c_i^f:

$$c_i^f=\sigma(\boldsymbol{W}_f\cdot\boldsymbol{X}_i+b_f)。\tag{4}$$

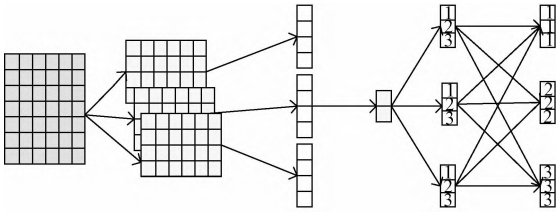

图1 卷积层网络结构

Figure 1 Convolution layer structure

式中:$\boldsymbol{W}_f$ 和 b_f 分别为权重矩阵和偏置项;$\sigma(\cdot)$ 为激活函数,使神经网络具有拟合非线性函数的能力。

卷积核遍历整个嵌入矩阵后产生一个特征图,记为 $\boldsymbol{c}^f$:

$$\boldsymbol{c}^f=[c_1^f,c_2^f,\cdots,c_{300-h+1}^f]。\tag{5}$$

将 $\boldsymbol{X}$ 经所有卷积核卷积池化后得到的新特征图堆叠得到一个序列矩阵,记为 $\boldsymbol{M}$:

$$\boldsymbol{M}=[\boldsymbol{m}_1,\boldsymbol{m}_2,\cdots,\boldsymbol{m}_s]。\tag{6}$$

式中:$s=\lceil(L-h+1)/pl\rceil$,pl 为池化窗口;$\boldsymbol{m}_i$ 为所有卷积核对URL词嵌入矩阵的同一区域经卷积、池化操作后的特征所组成的特征向量,$\boldsymbol{m}_i\in\mathbf{R}^{n\times1}$,$n$ 为卷积核个数。

2.3 BiLSTM层

双向长短记忆网络BiLSTM由2个方向相反的LSTM组成,二者网络结构相同,但权重参数不同。LSTM是RNN的一种变体,RNN由于梯度消失或梯度爆炸的原因只能获取短距离依赖信息,LSTM通过在网络节点上加上门结构以控制数据流动,避免梯度消失或梯度爆炸的问题。LSTM有3个门,自左向右分别为遗忘门、输入门、输出门,如图2所示。每个门都由一个激活函数 $\sigma(\cdot)$ 和一个点乘操作组成,其中 $\sigma(\cdot)$ 输出0~1的数值,描述了数据通过此门的比例程度,正向LSTM依时间顺序读入数据,以使信息沿时间起点正向传递,从而获取序列的前文信息,分为以下4个步骤。

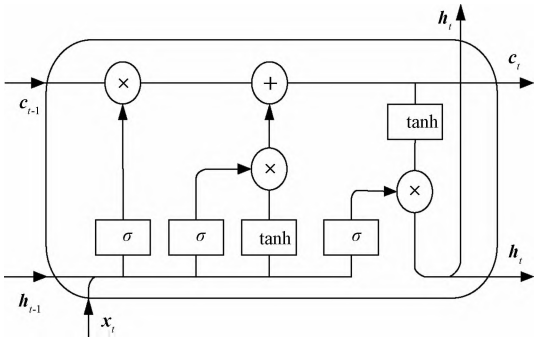

图2 LSTM网络结构

Figure 2 LSTM network structure

步骤 1　通过遗忘门从 ($t-1$) 时刻的细胞状态 $\boldsymbol{c}_{t-1}$ 中丢弃一定比例的信息。遗忘门 t 时刻的值为

$$f_t = \sigma(\boldsymbol{w}_f \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + b_f)。 \quad (7)$$

式中: $\boldsymbol{w}_f$ 为遗忘门权重矩阵; b_f 为遗忘门偏置; $\boldsymbol{h}_{t-1}$ 为($t-1$) 时刻的隐藏状态。

步骤 2　通过输入门控制当前时刻 t 的输入 $\boldsymbol{x}_t$, 确定细胞状态 $\boldsymbol{c}_t$ 中所保存的信息量。首先计算更新信息的比例 i_t, 然后通过激活函数 tanh 计算临时细胞状态 $\tilde{\boldsymbol{c}}_t$:

$$i_t = \sigma(\boldsymbol{w}_i \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + b_i); \quad (8)$$

$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{w}_c \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + b_c)。 \quad (9)$$

步骤 3　更新 ($t-1$) 时刻的细胞状态 $\boldsymbol{c}_{t-1}$, 计算 t 时刻的细胞状态。首先利用旧细胞状态与遗忘门输出点乘以丢弃旧细胞的部分信息,然后利用临时细胞状态与输入门输出点乘以得到需要加入细胞的新信息,最后利用二者的和得到新的细胞状态 $\boldsymbol{c}_t$:

$$\boldsymbol{c}_t = f_t \otimes \boldsymbol{c}_{t-1} + i_t \otimes \tilde{\boldsymbol{c}}_t。 \quad (10)$$

步骤 4　通过输出门的 σ 层计算输出比例 o_t, 然后将新的细胞状态输入 tanh 层进行处理,最后将二者进行点乘操作得到 t 时刻输出的值 $\boldsymbol{h}_t$:

$$o_t = \sigma(\boldsymbol{w}_o \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + b_o); \quad (11)$$

$$\boldsymbol{h}_t = o_t \otimes \tanh \boldsymbol{c}_t。 \quad (12)$$

细胞状态 $\boldsymbol{c}_t$ 水平方向自左向右移动,新的细胞状态是旧细胞状态的累加。这种细胞状态的累加方式会导致对神经网络进行训练时其导数也是一种累加形式而不是累乘,避免梯度消失或梯度爆炸的问题,能够对之前的信息进行长期记忆。逆向 LSTM 则沿时间终点逆向传递以获取序列的下文信息,信息传递过程与正向 LSTM 类似。

本文将卷积网络层的输出 $\boldsymbol{M}$ 看作时间轴上的序列信息作为 BiLSTM 的输入, $\boldsymbol{m}_i$ 与 BiLSTM 第 i 个时刻的输入对应。正向 LSTM 通过遗忘门、输入门、输出门来记忆 $i=s$ 时刻之前的信息,将此时刻的输出记为 $\boldsymbol{h}_{\mathrm{F}}$。反向 LSTM 通过遗忘门、输入门、输出门来记忆 $i=1$ 时刻之后的信息,将此时刻的输出记为 $\boldsymbol{h}_{\mathrm{R}}$。将 2 个不同方向的 LSTM 最后时刻的输出进行拼接,记为 $\boldsymbol{h} = \boldsymbol{h}_{\mathrm{F}} \oplus \boldsymbol{h}_{\mathrm{R}}$($\oplus$ 表示拼接运算符) ,以获取 URL 不同方向的长距离依赖特征。

2.4　全连接层

全连接层用于完成最终的分类功能,本文将其网络层数设置为 1,神经元个数设置为 2,通过 softmax 激活函数计算待测 URL 属于钓鱼或合法网页的概率:

$$p_i = \mathrm{e}^{z_i} \Big/ \sum_i^k \mathrm{e}^{z_i}。 \quad (13)$$

式中: $z_i = \boldsymbol{w}_i \boldsymbol{h} + b_i$, $\boldsymbol{w}_i$ 和 b_i 分别为权重和偏置参数; i 为 URL 类别索引(0 表示钓鱼 URL, 1 表示合法 URL) ; k 为 URL 类别总数,值为 2。

2.5　模型实现

首先基于敏感词分词方法对 URL 进行分词,并对分词后的数据进行整数编码,将其映射为 300×1 的一维矩阵; 通过词嵌入层转换为 300×64 的二维稠密矩阵; 通过一个卷积层进行卷积操作,并使用最大池化窗口获取更具有代表性的特征,实验中采用的卷积核个数为 200,池化窗口为 2,滑动步长为 1,将所有卷积核对词嵌入矩阵经卷积池化后形成的特征图按列堆叠形成 200×298 的矩阵,将其每行作为 BiLSTM 层对应时刻的输入; 利用 BiLSTM 的双向网络结构获取序列数据的上下文信息,充分学习特征之间的长距离依赖关系,实验中该网络的隐藏层神经元个数设置为 64,经过该网络后,特征矩阵被转化为一个 128 维的向量; 最后使用全连接层中的 softmax 函数将 BiLSTM 层输出的向量转换为 URL 属于合法或钓鱼的概率,根据交叉熵损失函数计算概率值和真实值之间的损失,通过反向传播算法更新网络模型参数。模型的整体结构如图 3 所示。

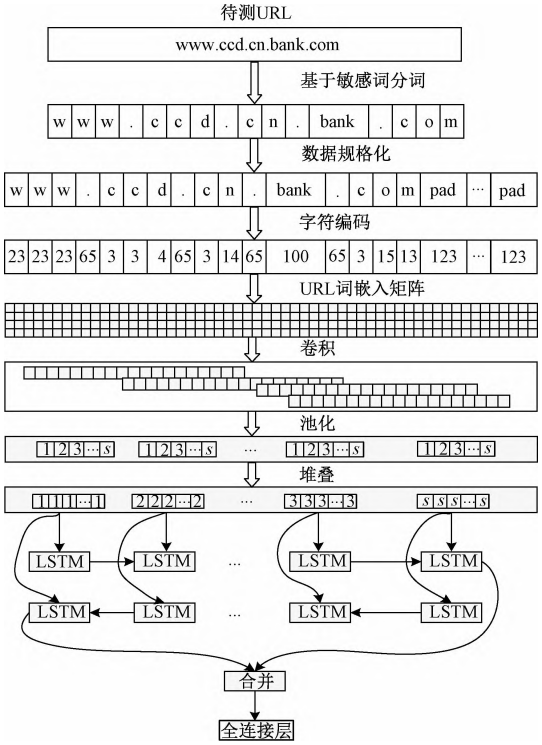

图 3　CNN-BiLSTM 网络结构

Figure 3　CNN-BiLSTM network structure

3 实验部分

3.1 实验数据

本文采用的数据集包括多个平台提供的开源样本,从 PhishTank 和 Malware Patrol 获取钓鱼 URL,从 Dmoz 和 Alexa 获取合法 URL,以此来丰富 URL 数据的来源。对数据去重后,数据集中共包含 206 200 条带标签的 URL 样本,其中钓鱼样本 105 100 条,合法样本 101 100 条,二者比例约为 1∶1。

3.2 评估标准

本文为了验证钓鱼网页检测方法的有效性,采用准确率 *Accuracy*、精确率 *Precision*、召回率 *Recall* 和 *F*1 值作为评价指标。*Precision* 表示被正确判断为钓鱼网页类别的网页占全部被判断为钓鱼网页类别的网页的比重,体现了检测方法对合法网页的区分能力,*Recall* 则体现了对钓鱼网页的识别能力,*F*1 值同时考虑到了精确率和准确率,是二者的加权平均,能综合评估检测模型的性能。计算式为

$$Accuracy = (TP + TN)/(TP + FP + TN + FN); \quad (14)$$

$$Precision = TP/(TP + FP); \quad (15)$$

$$Recall = TP/(TP + FN); \quad (16)$$

$$F1 = 2 \cdot Precision \cdot Recall/(Precision + Recall)。 \quad (17)$$

式中: *TP* 表示预测的钓鱼网页实际为钓鱼网页的数量; *FP* 表示预测的钓鱼网页实际为合法网页的数量; *TN* 表示预测的合法网页实际为合法网页的数量; *FN* 表示预测的合法网页实际为钓鱼网页的数量。

3.3 实验结果与分析

3.3.1 CNN-BiLSTM 在数据集上的准确率

本文对 URL 数据集采用十折交叉验证法,即将样本分为 10 组,其中 1 组包含 10 510 条钓鱼 URL 和 10 110 条合法 URL 作为测试集,另外 9 组包含 94 590 条钓鱼 URL 和 90 990 条合法 URL 作为训练集,该过程循环 10 次,保证每组样本数据都能作为测试集预测,将得到的 10 次测试结果取平均值评测模型的检测能力。图 4 是本文所提模型在十折交叉验证下,其准确率在训练集和测试集上的平均变化曲线。从图 4 中可以看出,训练过程中模型的参数收敛正常,当训练轮数为 30 时,模型的训练、测试准确率趋于稳定。

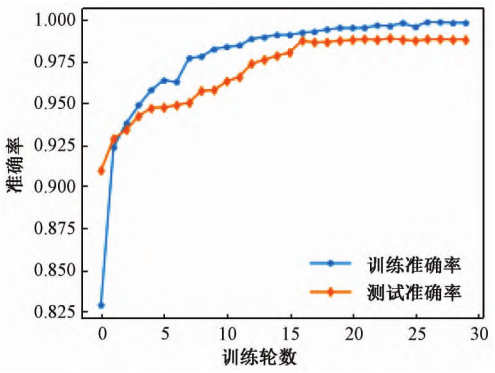

图 4 CNN-BiLSTM 在训练集和测试集上的准确率变化曲线

Figure 4 Accuracy curve of CNN-BiLSTM on training set and test set

3.3.2 不同模型在数据集上的检测效果

为了体现基于敏感词(sensitive word)分词方法的有效性,首先通过对 URL 数据采用 3 种不同的分词方法来训练 CNN 模型,分别为基于字符划分 URL 的字符级 CNN 模型 char-CNN、基于单词划分 URL 的词级 CNN 模型 word-CNN、基于敏感词划分 URL 的敏感词级 CNN 模型 sw-CNN,观察它们在测试集上的检测效果,如表 2 所示。与 char-CNN、word-CNN 相比,sw-CNN 在准确率、精确率、召回率和 *F*1 值这 4 个评估指标上均达到较好的检测效果,这表明本文所提出的基于敏感词分词的方法能够有效提升检测模型对钓鱼 URL 的检测能力。

表 2 所有模型在测试集上的最终检测结果

Table 2 Final test results of all models on test set

检测模型	准确率	精确率	召回率	*F*1 值
word-CNN	0.923 9	0.929 1	0.918 1	0.921 2
char-CNN	0.945 3	0.945 7	0.946 8	0.945 2
sw-CNN	**0.956 0**	**0.960 4**	**0.957 5**	**0.958 9**
sw-CNN-RNN	0.938 0	0.942 4	0.939 4	0.935 0
sw-CNN-BiLSTM	**0.988 4**	**0.997 1**	**0.980 4**	**0.988 6**

此外,为体现检测模型 CNN-BiLSTM 的优势,将其与深度学习模型 CNN、CNN-RNN 对比,通过对 URL 数据采用敏感词分词的方法来训练生成 2 个检测模型 sw-CNN-RNN、sw-CNN-BiLSTM,观察它们在测试集上的检测效果,如表 2 所示,同时对这些模型在训练集与测试集的准确率做了记录,如图 5、6 所示。结合表 2、图 5、图 6 可以看出,本文所涉及的 5 种检测模型在相同数据集上均获得了较高的检测准确率。其中,检测模型 char-CNN 在训练集及测试集上刚开始就达到了较高的准确率,但随着训练轮数的增加,准确率的提升程度不

大。word-CNN 在训练集与验证集上的准确率变化曲线与 char-CNN 类似,但准确率低于 char-CNN 模型,该结果可能源于以下 3 个方面: ①通过“.”“\”“?”等特殊字符对 URL 分词时忽略了特殊字符所具有的有效信息; ②为了避免内存受限,将数据集中仅出现一次的单词统一标记为<UNK>而忽略了这些单词的有效信息; ③无法获得新出现单词的有效信息。sw-CNN 由于能够获取到 URL 中敏感词的有效信息,其准确率高于 char-CNN。

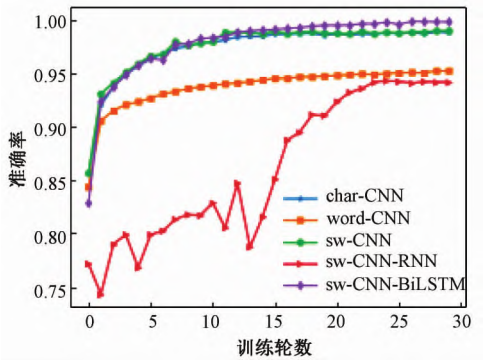

图 5　不同模型在训练集上的准确率

Figure 5　Accuracy of different models on training set

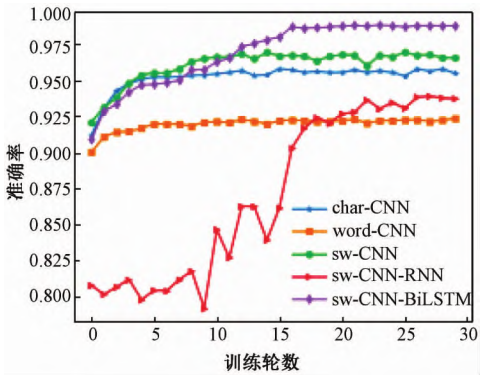

图 6　不同模型在测试集上的准确率

Figure 6　Accuracy of different models on test set

sw-CNN-RNN 虽然采用混合网络模型用于提取 URL 特征,但由于 RNN 无法获取到 URL 的长距离依赖特征,反而导致其检测准确率低于单模型结构的 char-CNN 和 sw-CNN。检测模型 sw-CNN-BiLSTM 相比以上模型能够获取到更为充分的 URL 特征,达到了最高的检测准确率、精确率、召回率和 $F1$ 值。

3.3.3　不同模型对不同长度的 URL 的检测效果

另外,在实验过程中发现,sw-CNN-BiLSTM 对 URL 短字符串也有较好的检测效果。为了研究其对短字符串的检测性能,在相同的实验环境下,将 URL 长度分别设置为 15、25、50、100、200、300、400,观察其检测效果,结果如图 7 所示。在 URL 长度降至 15 时,sw-CNN-BiLSTM 的检测准确率也能达到 87%,而 sw-CNN-RNN、sw-CNN 与 word-CNN 的准确率分别为 72%、78%、70%。实验结果表明,sw-CNN-BiLSTM 对 URL 短链接也有较好的检测效果。

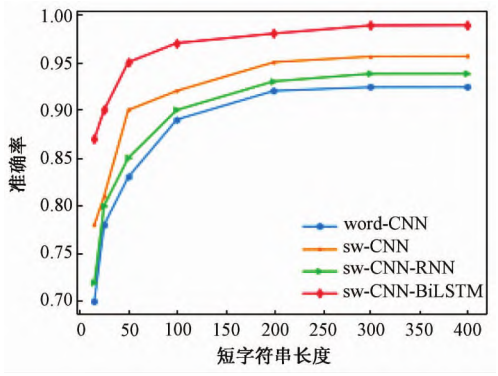

图 7　不同模型在测试集上对不同长度的 URL 的检测准确率

Figure 7　Detection accuracy of different models for URL of different length on test set

4　结论

(1) 提出了一种融合 CNN 与 BiLSTM 的检测模型,该模型能够兼顾 CNN 和 BiLSTM 的特点,充分提取 URL 数据的空间局部特征及长距离依赖特征。

(2) 提出了一种基于敏感词分词的方法,该方法能够获取新出现单词的嵌入向量,也能获取 URL 中敏感词、特殊字符的有效信息,提升了 URL 数据信息的利用程度。

(3) 在数据集上的实验结果表明,本文所提出的基于 CNN-BiLSTM 的钓鱼 URL 检测方法可以有效提升对钓鱼网页检测的能力。

参考文献:

[1] 中国互联网络信息中心.第 45 次中国互联网络发展状况统计报告[R/OL].(2017-02-17)[2020-03-25].http://www.cnnic.cn/gywm/xwzx/rdxw/20172017_7057/202004/t20200427_70973.htm.

[2] 中国反钓鱼网站联盟.2020 年 8 月钓鱼网站处理简报[EB/OL].(2020-03-20)[2020-10-08].http://www.apac.cn/gzdt/202003/P020200320392664104846.pdf.

[3] CANALI D,COVA M,VIGNA G,et al.Prophiler: a fast filter for the large-scale detection of malicious web pages [C]//Proceedings of the 20th International Conference on World Wide Web-WWW'11.New York: ACM,2011: 197-206.

[4] THOMAS K, GRIER C, MA J, et al. Design and evaluation of a real-time URL spam filtering service [C]// 2011 IEEE Symposium on Security and Privacy. Piscataway: IEEE, 2011: 447-462.

[5] SHENG S, WARDMAN B, WARNER G, et al. An empirical analysis of phishing blacklists [EB/OL]. (2009-01-01) [2020-04-08]. http://citeseerx.ist.psu.edu/viewdoc/summary? doi=10.1.1.165.520.

[6] ALEROUD A, ZHOU L N. Phishing environments, techniques, and countermeasures: a survey [J]. Computers & security, 2017, 68: 160-196.

[7] LIU G, QIU B T, WENYIN L. Automatic detection of phishing target from phishing webpage [C]//20th International Conference on Pattern Recognition. Piscataway: IEEE, 2010: 4153-4156.

[8] MA J, SAUL L K, SAVAGE S, et al. Beyond blacklists: learning to detect malicious web sites from suspicious URLs [C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD´09. New York: ACM, 2009: 681-688.

[9] 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.

[10] KIM Y. Convolutional neural networks for sentence classification [C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha: Association for Computational Linguistics, 2014: 1746-1751.

[11] ZHANG M, XU B Y, BAI S, et al. A deep learning method to detect web attacks using a specially designed CNN [C]//Neural Information Processing. Berlin: Springer, 2017: 828-836.

[12] CUI J P, LIU M, HU J W. Malicious web request detection technology based on CNN [J]. Computer science, 2020, 47(2): 281-286.

[13] YU B, PAN J, HU J M, et al. Character level based detection of DGA domain names [C]//2018 International Joint Conference on Neural Networks (IJCNN). Piscataway: IEEE, 2018: 1-8.

Research on Phishing URL Detection Technology Based on CNN-BiLSTM

BU Youjun[1], ZHANG Qiao[1,2], CHEN Bo[1], ZHANG Surong[1], WANG Fangyu[2]

(1.PLA Strategic Support Force Information Engineering University, Zhengzhou 450001, China; 2.Zhongyuan Network Security Research Institute, Zhengzhou University, Zhengzhou 450001, China)

Abstract: In order to solve the increasingly serious problem of phishing, a phishing URL detection method based on convolution neural network (CNN) and bi-directional long short termmemory (BiLSTM) was proposed. This method first classified the URL based on the sensitive word segmentation method; classified the URL according to the special characters and sensitive words; and classified the non-sensitive words in the character level, so as to obtain the effective information of the special characters and sensitive words, and improve the use of URL data information. Then the segmented URL was input into CNN and BiLSTM, to obtain the spatial local features of the URL through CNN, to obtain the bidirectional long-distance dependent features of the URL through BiLSTM, and to detect phishing webpages based on the automatically extracted features. Compared with traditional machine learning and blacklist detection methods. Experimental results showed that the phishing URL detection method based on CNN and BiLSTM could achieve better detection results, the accuracy rate was 98.84%, the precision rate was 99.71%, the recall rate was 98.04%, and the $F1$ value was 98.86%. This method did not require manual feature extraction and could identify newly emerging phishing webpages.

Keywords: phishing URL; URL segmentation; CNN; BiLSTM