

DOI:10.13232/j.cnki.jnju.2019.01.003

## 基于深度神经网络的网络安全实体识别方法

秦 娅<sup>1,2</sup>, 申国伟<sup>1,2\*</sup>, 赵文波<sup>1</sup>, 陈艳平<sup>1,2</sup>

(1. 贵州大学计算机科学与技术学院, 贵阳, 550025; 2. 贵州省公共大数据重点实验室, 贵阳, 550025)

**摘 要:** 基于安全知识图谱的网络安全威胁情报分析能够细粒度地分析多源威胁情报数据, 因此受到广泛关注. 传统的命名实体识别方法难以识别网络安全领域中新的或中英文混合的安全实体, 且提取的特征不充分, 因此难以准确地识别网络安全实体. 在深度神经网络模型的基础上, 提出一种结合特征模板的 CNN-BiLSTM-CRF 的网络安全实体识别方法, 利用人工特征模板提取局部上下文特征, 进一步利用神经网络模型自动提取字符特征和文本全局特征. 实验结果表明, 在大规模网络安全数据集上, 提出的网络安全实体识别方法, 相关评价指标优于其他算法,  $F$  值达到 86%.

**关键词:** 网络安全实体识别, 特征模板, CNN, BiLSTM, CRF

**中图分类号:** TP391

**文献标识码:** A

## Research on the method of network security entity recognition based on deep neural network

Qin Ya<sup>1,2</sup>, Shen Guowei<sup>1,2\*</sup>, Zhao Wenbo<sup>1</sup>, Chen Yanping<sup>1,2</sup>

(1. College of Computer Science and Technology, GuiZhou University, Guiyang, 550025, China;

2. Guizhou Provincial Key Laboratory of Public Big Data, Guiyang, 550025, China)

**Abstract:** With the continuous development of the Internet technology, network security threat intelligence analysis that base on security knowledge graph (SKG) can analyze multi-source threat intelligence data in a fine-grained manner, which has received extensive attention. Traditional named entity recognition (NER) methods are difficult to identify network security entity which mix Chinese and English in the field of network security, and can't fully extract some features, so it is difficult to accurately identify the network security entity. In this paper, we propose a novel CNN-BiLSTM-CRF security entity recognition method combining with feature template (FT-CNN-BiLSTM-CRF) on the basis of deep learning model. The feature template is used to extract local context features, and neural network model is used to automatically extract character features and text global features. Firstly, each character of the input sequence is converted into a corresponding character vector, and the convolutional neural network (CNN)

基金项目: 国家自然科学基金 (61802081), 贵州省自然科学基金 (20161052), 贵州省公共大数据重点实验室开放课题 (2017BDKFJJ024), 贵州大学博士基金 (201526)

收稿日期: 2018-09-01

\* 通讯联系人, E-mail: gwshen@gzu.edu.cn

extracts the character-level features. Secondly, the character-level features vectors are input into the BiLSTM (Bi-Long Short-Term Memory) together with the local context vectors extracted by the feature template. The global features of the security entity are automatically extracted by BiLSTM. Finally, the CRF (Conditional Random Fields) labels the network security entity to obtain the recognition result of the security entity. The experimental results show that our method reaches 86% F-scores on the large-scale network security dataset and outperforms other methods.

**Key words:** network security entity recognition, feature template, CNN, BiLSTM, CRF

随着信息技术的不断发展和网络环境的日趋复杂,网络空间安全已经成为世界各国共同关注的焦点.如何从碎片化、海量化的威胁情报数据中挖掘出关联关系、攻击模式等是情报分析研究的焦点.

网络安全知识图谱<sup>[1]</sup>可将海量的碎片化的多源异构安全数据进行细粒度的深度关联分析和挖掘.通过安全知识图谱分析,网络安全人员能更直观地洞悉网络安全威胁情报<sup>[2]</sup>和安全态势,发现复杂的网络攻击模式.网络安全知识图谱构建技术主要有安全实体识别、关系抽取和属性抽取等,其中安全实体识别技术是网络安全知识图谱构建的基础.

网络安全实体识别是命名实体识别领域<sup>[3]</sup>中一种特定领域的实体识别,主要任务是识别网络安全文本数据中的用户、恶意程序、黑客组织、漏洞等不同类型的安全实体,目的是对网络安全领域中的专业词汇进行确认和分类.

命名实体识别最初采用的是基于规则的识别方法,通过领域专家和语言学者手工制定的有效规则进行实体识别,制定的规则并不能适应于其他领域,领域移植性比较差.针对此类问题,研究者们提出利用人工特征的机器学习方法识别文本中命名实体,常用的机器学习方法主要有隐马尔科夫模型<sup>[4]</sup>、最大熵模型<sup>[5]</sup>和条件随机场<sup>[6]</sup>.但这些学习方法需要人工手动提取有效的语法特征,设定模型的特征模板进行识别,因此特征模板的选择直接影响命名实体识别的准确率.邱泉清等<sup>[7]</sup>提出使用 CRF (Conditional Random Field) 模型对微博数据进行命名实体识别,利用知识库和合适的特征

模板取得了良好的效果;Joshi *et al*<sup>[8]</sup>提出一种基于 SVM (Support Vector Machine) 算法的信息识别方法,从非结构化文本数据中识别网络安全相关术语和概念.

近年来,基于神经网络<sup>[9]</sup>的深度学习方法在通用领域的命名实体识别中达到了很好的效果.神经网络通过搭建多层网络结构实现数据的特征提取,目前比较常见的结构包括循环神经网络 (Recurrent Neural Network, RNN)、长短期记忆 (Long Short-Term Memory, LSTM) 神经网络和卷积神经网络 (Convolutional Neural Networks, CNN) 等. Collobert *et al*<sup>[10]</sup>首次使用 CNN 在通用命名实体识别领域上取得较好的结果,此后,基于神经网络提取特征的方法得到广泛应用. Hochreiter and Schmidhuber<sup>[11]</sup>提出一种利用门限机制对历史信息进行过滤的 LSTM 模型, Hammerton<sup>[12]</sup>将此模型运用到命名实体识别. Peng and Dredze<sup>[13]</sup>针对新浪微博文数据,利用 LSTM 提取特征,通过 CRF 进行分类,取得了较好的效果. Huang *et al*<sup>[14]</sup>, Dong *et al*<sup>[15]</sup> 和 Lample *et al*<sup>[16]</sup> 利用双向 LSTM (BiLSTM) 模型结合 CRF 模型进行命名实体识别,在中英文数据集上都取得了很好的效果. Chiu and Nichols<sup>[17]</sup> 针对 CoNLL 2003 数据集,提出一种新型的神经网络体系结构,使用 CNN 提取字符级特征,通过字符级特征与词向量相结合的方法来提高模型的性能,  $F$  值达到了 91.62%. Ma and Hovy<sup>[18]</sup> 提出一种基于 BiLSTM, CNN 和 CRF 的端到端的通用模型,在 CoNLL 2003 数据集上测试的  $F$  值达到了 91.21%.

虽然神经网络在通用命名实体识别领域具有较好的效果,但是网络安全领域的安全实体识别仍存在很多问题. 与传统的命名实体识别相比,网络安全实体识别主要存在以下难点:

(1)网络安全实体类型多样,数量众多,且变化频率非常高,不断地会有未登录词或短语作为新的安全实体出现,例如,新的恶意软件、漏洞以及补丁等. 因此基于分词的方法识别率较低.

(2)在不同的场景下,安全实体存在分类模糊的问题. 不同类型的安全实体之间界限不清晰,软件名也经常出现在组织名中. 例如,Oracle 既代表一个软件,又代表一个组织.

(3)网络文本数据中安全实体具有不同的结构,且中英文混合. 大量的软件和漏洞既有中文命名,又有英文命名,而且实体之间会出现大量嵌套、缩写使用不规范以及用词隐晦等问题,因此识别难度相对较大.

针对上述问题,本文在神经网络模型的基础上,提出一种结合特征模板和 CNN-BiLSTM-CRF 的网络安全实体识别方法. 该方法首先利用 CNN 对每一个单词进行字符特征提取;其次,制定少量特征放入特征模板,提取局部上下文特征;最后将字符向量特征和局部上下文特征进行组合,传入 BiLSTM-CRF 模型中进行训练,进而提高网络安全实体识别模型的准确率.

## 1 网络安全实体识别模型

针对网络安全文本数据,本文提出一种特征模板 (Feature Template, FT), CNN, BiLSTM 和 CRF 相结合的网络安全实体识别模型,如图 1 所示. 该模型首先根据中英文混合的网络安全语料数据,人工制定少量特征,形成特征模板,提取局部上下文特征;然后利用预先训练的字向量文件,将输入的语句转换为相应字向量序列;通过 CNN 对每一个单词进行卷积和池化,提取该单词的字符级特征;将字符特征向量和局部上下文特征相组合,传入到 BiLSTM 神经网络,利用 BiLSTM 神经网络来训练得到输入语句的信息特征;针对 BiLSTM

提取的语义特征,利用 CRF 来对每个字进行安全实体标注,标记出语句序列中的安全实体信息,得到最优的标记序列.

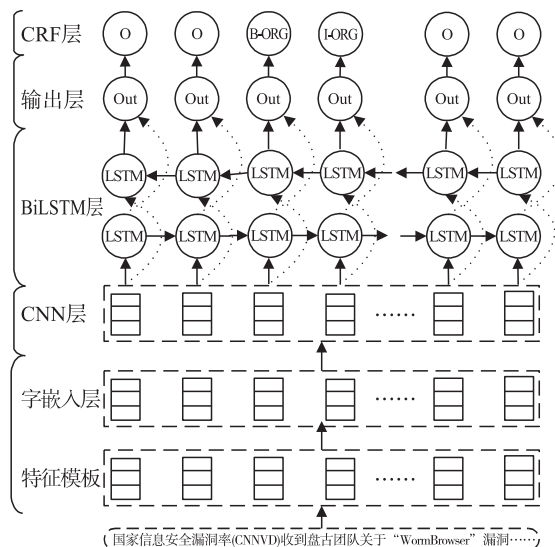


图 1 基于 FT-CNN-BiLSTM-CRF 的网络安全实体识别模型

Fig. 1 Network security entity recognition model based on FT-CNN-BiLSTM-CRF

**1.1 特征模板** 特征模板是根据所选用的特征设计出来的模板,是对数据的上下文的特定信息和信息特定位置的综合考虑. 本文基于 FT-CNN-BiLSTM-CRF 模型能够通过特征模板将网络安全文本中字的上下文信息加入到模型中. 上下文信息是指当前字以及前后的几个字的“观察窗口”,观察窗口越大,模板中包括的上下文信息就越多,但是观察窗口太大也会降低网络安全实体识别模型的效率,产生过拟合现象;观察窗口小自然能利用的信息也少,也会降低模型的识别效率. 所以合理选取模板的观察窗口大小是关键.

本文设计基于原子模板的特征模板,模板中制定的特征为原子特征,其模板如下:  $w[-2,0]$ ,  $w[-1,0]$ ,  $w[0,0]$ ,  $w[1,0]$ ,  $w[2,0]$ , 括号内的第一个数字表示它相对于当前字符的位置,第二个数字表示所选特征的列,包括词、词性和其他特征. 表 1 为原子特征示例,  $w$  表示当前的字,  $y$  表示字的标签.

表 1 原子特征

Table 1 Atomic feature

相对当前字符位置	$w$	$y$	当前 token
-2	找	O	
-1	到	O	
0	雅	B-ORG	✓
1	虎	I-ORG	
2	的	O	

本文根据上述特征模板中制定的特征,定义一组特征函数  $f_j(y_{i-1}, y_i, w, i)$  (其中  $y$  代表当前的标签,  $y_{i-1}$  表示下一个标签,  $i$  表示当前位置,  $w$  为当前字), 为每一个特征函数赋给一个权重  $\lambda_i$ . 如果一个特征函数被激活, 那么它的权重  $\lambda_i$  将被添加到一个累积值, 即该特征的得分值  $F_s \in \mathbf{R}^A$ ,  $F_s$  的分值越高, 说明对应的标签分数越高, 预测结果越准确.

**1.2 字向量** 基于神经网络模型的网络安全实体识别模型中的词或字向量通常具有较高的维度, 导致模型中具有大量的参数. 而在实际应用中, 有标注的监督语料通常不足, 难以学习到准确的参数. 本文受 Mikolov *et al.*<sup>[19-20]</sup> 的方法启发, 采用预训练字向量的方法. 首先用大规模的无标注网络安全语料训练得到字向量文件, 然后将其作为网络安全实体识别模型的字向量初始值, 比直接使用随机值作为标注模型初始值的性能要高.

word2vec 是 Google 在 2013 年提出的词向量表示学习方法, 将词语或字表示成一个固定长度的数值向量形式, 这个向量被认为具有一定的潜在语义信息. 近似词或者字之间具有一定的向量相似性. 词或字向量之间还可以进行加减操作, 获得词或字之间的语义联系. 本文将网络安全领域的原始语料通过 word2vec 的 CBOW 模型进行训练, 得到相应的字向量文件, 其中不同的字对应不同的数值向量, 同一个字对应唯一的数值向量.

本文利用 word2vec 训练字向量文件, 字向量文件中包括 16691 个汉字、英文字符和一些

特殊字符(包括数字, 标点符号等)以及它们的数值向量形式. 每个数值都有 100 维, 每一维表示一个特征, 维度大小对最终的网络安全实体识别有影响.

**1.3 CNN 算法** 本文根据 Chiu and Nichols<sup>[17]</sup> 和 Ma and Hovy<sup>[18]</sup> 的研究发现, CNN 能够有效地提取数据的字符级特征. CNN<sup>[21]</sup> 主要是处理英文, 英文单词是由更细粒度的字母组成, 这些字母潜藏着一些特征, 如前缀、后缀特征, 通过 CNN 的卷积操作能够提取这些特征. 因此, 本文针对网络安全领域中的英文安全实体, 例如 Microsoft, 提出利用 CNN 抽取安全实体的字符级特征, 通过字符级特征来提高模型的性能. 本文通过字嵌入的方式, 对于不同类型的字符设置了不同的字符向量, 以区分字符的大小写, 字符类型.

本文使用的 CNN 的结构如图 2 所示. 主要由字向量表、卷积层和池化层组成. 首先, 将输入序列的每个单词通过查询字向量表转化为对应的字符向量; 其次, 以最长的单词为基准, 通过在单词的左右两端填充占位符, 使得所有的字符向量矩阵大小一致; 然后, 将经过填充后的字符向量矩阵传入卷积层提取局部特征; 最后, 通过池化层, 对特征进行降维, 提取出字符级特征.

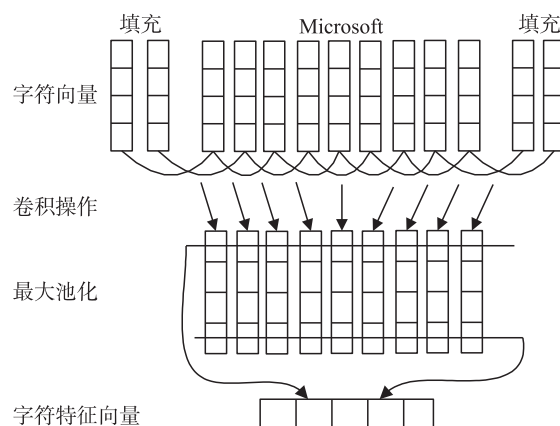


图 2 基于 CNN 的字符特征提取模型

Fig. 2 Character feature extraction model based on CNN

**1.4 BiLSTM 算法** LSTM<sup>[11]</sup> 是一种特定形式的循环神经网络 RNN,比一般的循环神经网络记忆能力强. LSTM 通过引入记忆单元和门限机制,在保留冗余上下文信息的同时实现了对长距离信息的有效利用,因此被广泛应用于网络安全实体识别. LSTM 记忆单元是通过一个细胞状态  $c_t$  来调解整个结构,其中细胞状态  $c_t$  的保存和更新由输入门、遗忘门和输出门决定. LSTM 记忆单元结构如图 3 所示.

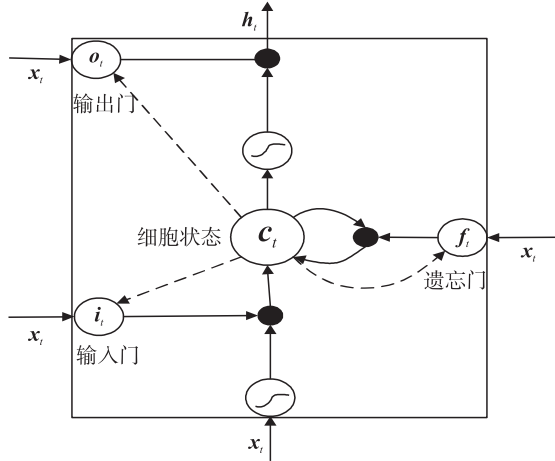


图 3 LSTM 记忆单元结构

Fig. 3 Structure of LSTM memory unit

LSTM 记忆单元具体计算方法如下:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (2)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (3)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (5)$$

其中  $i_t, f_t, o_t, c_t$  分别表示  $t$  时刻的输入门、遗忘门、输出门和细胞状态.  $x_t$  和  $h_t$  表示  $t$  时刻的输入向量和隐藏层向量.  $\sigma$  和  $\tanh$  表示两种不同的神经元激活函数.  $W$  表示连接两层的权重矩阵,如  $W_{xi}$  表示输入层到隐藏层的输入门的权重矩阵.  $b$  表示偏置向量,  $b_i$  表示隐藏层的输入门的偏置向量.

LSTM 只能访问过去的上下文信息,但是对于网络安全实体识别,未来的上下文信息也很重要,例如,识别“DDOS 攻击”这一网络相关

术语的时候,如果能像知道“O”这个字符之前的字符“D”,提前预测出“O”后面接下来将会出现的字符,就能够提高网络安全实体识别的性能.因此,为了能够有效利用过去和未来的上下文信息,本文采用双向 LSTM(BiLSTM)神经网络结构.

BiLSTM 通过 LSTM 记忆单元对输入的序列分别采用顺序(从第一个字开始,从左往右递归)和逆序(从最后一个字开始,从右向左递归)计算得到两层不同的隐藏层表示,然后通过向量拼接得到最终的隐藏层表示,其结构如图 4 所示.

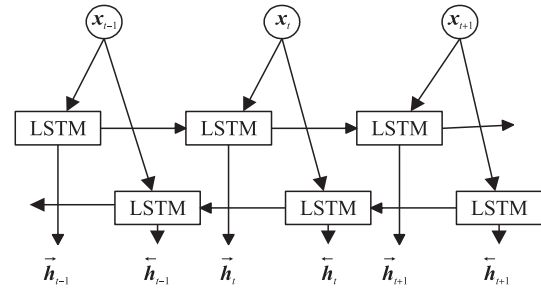


图 4 BiLSTM 模型结构

Fig. 4 Structure of BiLSTM model

从图 4 中可知, BiLSTM 接收通过 CNN 提取的字符特征向量序列,作为  $t$  时间的输入,再将正向 LSTM 输出的特征向量序列  $\vec{h} = (\vec{h}_1, \dots, \vec{h}_t)$  与反向 LSTM 的特征向量序列  $\vec{h} = (\vec{h}_1, \dots, \vec{h}_t)$  在  $t$  时刻进行拼接  $\vec{h}_t$ , 得到完整的特征向量序列:

$$\vec{h}_t = [\vec{h}_t; \vec{h}_t] \in \mathbf{R}^m \quad (6)$$

接着对隐藏层拼接后的特征向量  $\vec{h}_t$  通过  $\tanh$  激活函数做预处理,从而得到隐藏层的输出结果,如式(7)所示:

$$O_t = \tanh(W_h \vec{h}_t + b_o) \quad (7)$$

其中  $\vec{h}_t$  对应的权重为  $W_h \in \mathbf{R}^{|n| \times 2|n|}$ , 偏置向量为  $b_o \in \mathbf{R}^{|n|}$ ,  $O_t$  为隐藏层输出的结果,  $|n|$  表示隐藏层维数.

**1.5 CRF 算法** 本文将网络安全实体识别转换成序列标注问题. BiLSTM 在序列建模上很强大,能够获取长远的上下文信息,但是,网络安全实体识别任务的标签之间并不独立,而是



具有较强的依赖关系,特别是基于字的网络安全实体识别,如 B-ORG 标签后面是 I-ORG,但不可能是 I-PER. 因此,本文将在 BiLSTM 后接入 CRF,借鉴链式 CRF 的方法考虑标签之间的转移和计算整体上标签序列的概率,从而获得全局最优的标记序列.

本文模型输入序列为  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ , 其中  $\mathbf{x}_n$  为第  $n$  个字的输入向量,  $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n)$  为  $\mathbf{X}$  的标签序列. 序列  $\mathbf{X}$  中每个字所对应的标签都有相应的分数,分数越大,对应的标签越有可能;从标签序列  $\mathbf{y}$  的整体来看,前后标签有转移分数,分数越高,越有可能出现标签转移. 将这两个分数相加,分数最高的标签就是预测结果.

首先,求解在  $t$  时刻,每个字对应的所有标签分数,主要包括 BiLSTM 隐藏层计算的标签分数和特征模板计算的标签分数,计算公式如式(8)所示:

$$score = \mathbf{W}_c \mathbf{O}_t + \mathbf{b}_c + \mathbf{F}_s \quad (8)$$

其中  $\mathbf{W}_c \in \mathbf{R}^{n \times m}$  为权重矩阵,  $\mathbf{b}_c \in \mathbf{R}^m$ ,  $m$  表示标签个数.  $\mathbf{O}_t$  为 BiLSTM 的输出结果,如式(7)所示.  $\mathbf{F}_s \in \mathbf{R}^A$  是由特征模板得到的特征权重  $\lambda$  总和,  $score$  表示每个字对应的标签分数,分数越大越有可能.

其次,设置转移矩阵  $\mathbf{A}$ , 表示标签之间的转移分数,在  $m$  个标签的基础上增加了“开始”和“结束”两个标签. 一个标签序列的总分数是由每个标签的转移来决定的,如式(9)所示:

$$S(\mathbf{X}, \mathbf{y}) = \sum_{i=0}^l (\mathbf{A}_{\mathbf{y}_i, \mathbf{y}_{i+1}} + score(\mathbf{y}_{i+1})) \quad (9)$$

其中  $\mathbf{X}$  为输入序列,  $\mathbf{y}$  是预测标签序列,  $\mathbf{A}_{\mathbf{y}_i, \mathbf{y}_{i+1}}$  为由标签  $\mathbf{y}_i$  转移到  $\mathbf{y}_{i+1}$  的概率,即转移分数.  $score(\mathbf{y}_{i+1})$  表示当前标签  $\mathbf{y}_{i+1}$  的分数.  $\mathbf{y}_0$  和  $\mathbf{y}_l$  分别为“开始”和“结束”标签,它们对应的  $score$  分数为零.

然后,在输入序列  $\mathbf{X}$  的条件下,产生标签序列  $\mathbf{y}$  的概率,如式(10)所示:

$$p(\mathbf{y} | \mathbf{X}) = \frac{e^{S(\mathbf{X}, \mathbf{y})}}{\sum_{\mathbf{y}' \in Y_X} S(\mathbf{X}, \mathbf{y}')} \quad (10)$$

随后,本模型采用对数最大似然估计得到

损失函数使得正确的序列的概率最大,所以本模型采用对数最大似然估计得到损失函数,如式(11)所示:

$$\lg(p(\mathbf{y} | \mathbf{X})) = S(\mathbf{X}, \mathbf{y}) - \lg \left( \sum_{\mathbf{y}' \in Y_X} e^{S(\mathbf{X}, \mathbf{y}')} \right) = S(\mathbf{X}, \mathbf{y}) - \lg add S(\mathbf{X}, \mathbf{y}') \quad (11)$$

最后,利用随机梯度下降学习算法训练参数  $\theta$ , 求得参数  $\theta$  后,用 Viterbi 算法求得所有序列上打分最高的序列,作为最终的安全实体识别的标注结果,如式(12)所示:

$$\mathbf{y}^* = \operatorname{argmax}_{\mathbf{y}' \in Y_X} S(\mathbf{X}, \mathbf{y}') \quad (12)$$

## 2 模型训练

**2.1 训练过程** 本文在 TensorFlow 平台中实现基于 CNN-BiLSTM-CRF 的网络安全实体识别模型,选取的硬件平台为 Intel(R) Xeon(R) CPU 24 核,利用 FT-CNN-BiLSTM-CRF 训练算法构建网络安全实体识别模型. 表 2 为模型算法训练过程.

表 2 模型训练过程

Table 2 Model training process

FT-CNN-BiLSTM-CRF 模型训练过程

对于每个 epoch 循环:

对于每个 batch 循环:

- (1) 参数初始化;
- (2) CNN 模型前向传递,提取字符特征;
- (3) BiLSTM+CRF 模型前向传递,自动学习提取特征:
  - a. 向后状态的前向传递;
  - b. 向前状态的前向传递;
- (4) CRF 向前和向后传递,计算序列全局的似然概率;
- (5) BiLSTM+CRF 模型后向传递:
  - a. 向后状态的后向传递;
  - b. 向前状态的后向传递;
- (6) 更新参数;

结束 batch 循环;

结束 epoch 循环.

从表 2 中可以看出,对于每一个 epoch,模型在训练的过程中会将整个训练数据分成批次来处理,一个 epoch 处理一批,每一批训练数据的大小都是由参数 batch\_size 决定. 模型训练过程中,首先对模型需要的参数进行初始化;再通过

CNN 模型进行字符特征的提取;然后运行 BiLSTM 模型前向和后向传递,自动学习提取特征;接着通过运行 CRF 模型向前和向后传递来计算模型的输出状态;之后,可以将错误的状态从输出传回输入,其中包括 LSTM 的向前和向后的状态的反向传递;最后,重新更新模型参数.

**2.2 参数初始化** 基于 FT-CNN-BiLSTM-CRF 的网络安全实体识别模型具有大量参数,对于这些参数的选择是十分重要的,表 3 为模型设置的参数.

表 3 参数设置

Table 3 Parameter setting

参数	值
CNN window size	5
CNN filter size	30
BiLSTM hidden size	100
Learning rate	0.002
Batch size	100
Epoch	40
Fine tuning	True
Dropout rate	0.5
特征模板 window size	5

从表 3 可以看出,本文模型设置了九类参数. 在 CNN 层主要设置了窗口大小(window size)和过滤器的个数(filter size). 在 BiLSTM 层主要设置了隐藏层的神经单元数量(hidden size)为 100. 在特征模板 FT 中主要设置了基于上下文特征的窗口大小. 在字嵌入层对模型进行微调(Fine tuning),在模型梯度更新期间通过向后传播梯度来修改它们的参数. 本文在训练过程中,选取 Adam 优化器进行优化训练,初始学习率(Learning rate)设置为 0.002, batch 大小设置为 100,共训练 30 次. 为了减轻模型的过度拟合的问题,本文在 CNN 输入之前和 BiLSTM 的输入和输出部分都利用了 dropout 方法,dropout 值取 0.5.

### 3 实验及分析

**3.1 语料以及标注模式** 实验所用的数据主要来自于 Freebuf 网站和乌云漏洞数据库,主要包括技术分享、网络安全、漏洞信息等网络文

本数据,共计 15460 条. 选择其中 10000 条原始未标记数据用于训练字向量文件,5460 条数据作为网络安全实体识别标记语料.

本文主要识别的是网络安全文本数据中的人名(Person, PER)、地名(Location, LOC)、组织名(Orgnation, ORG)、软件名(Software, SW)、网络相关术语(Relevant Term, RT)以及漏洞编号(Vulnerability ID, VUL\_ID)等六类安全实体. 针对这六类安全实体,本文采用 BIO 的标注模式, B 代表实体开始, I 代表实体中间, O 表示不是实体. 本文采用半自动的标注的方式进行安全实体标注.

首先对网络安全文本数据进行分词,然后通过匹配人名库、地名库、漏洞编号库和网络相关术语库的方式,对分词后的数据进行标注,分别标为 B-PER, B-LOC, B-VUL\_ID, B-RT;接着通过代码对标注的数据进行处理,按照字的方式进行标注,即 B-PER 和 I-PER 代表人名的首字和非首字, B-LOC 和 I-LOC 代表地名的首字和非首字, B-VUL\_ID 和 I-VUL\_ID 代表漏洞编号的首字和非首字, B-RT 和 I-RT 代表相关术语的首字和非首字;最后,通过人工标注的方式,标注组织名和软件名, B-LOC 和 I-LOC 代表组织名的首字和非首字, B-SW 和 I-SW 代表软件名的首字和非首字.

将标注好的 70%网络安全语料数据作为训练数据, 10%作为验证数据, 20%作为测试数据,表 4 为网络安全语料数据统计信息.

表 4 语料统计信息

Table 4 Corpus statistics

数据集	训练数据	验证数据	测试数据
句子数	15090	1989	2697
标签总数	70324	9181	16669
人名	1139	230	249
地名	975	441	112
组织名	4059	374	1216
软件名	4236	666	1794
相关术语	59849	7428	13102
漏洞编号	66	42	196

**3.2 评价标准** 根据 BIO 的标注模式,将人名、地名、组织名、软件名、相关术语以及漏洞编号分为六个独立的问题,每一个问题分为一个三分类问题.以组织名为例,分为 B-ORG, I-ORG 和 O 等三类,将识别出的组织名的首字 B-ORG 和非首字 I-ORG 合并为一个实体,完整的识别一个安全实体为正确识别;而只识别出实体的首字 B-ORG、非首字 I-ORG 和 O 的不算正确识别.因此将采用多分类问题常用的评价指标体系<sup>[22]</sup>对本文研究的网络安全实体识别模型的识别效果进行评价.

本文主要采用的评价指标为精确率(Precision,  $P$ )、召回率(Recall,  $R$ )、 $F$  值、准确率(Accuracy,  $Acc$ )、宏平均(Macro-averaging)和微平均(Micro-averaging).具体的计算方式如下所示:

$$P = TP / (TP + FP) \quad (13)$$

$$R = TP / (TP + FN) \quad (14)$$

$$F = (2 \times P \times R) / (P + R) \quad (15)$$

$$Acc = (TP + TN) / (TP + TN + FP + FN) \quad (16)$$

宏平均(Macro-averaging),先计算每一个类统计指标值,然后再对所有类求算术平均值,计算公式如下:

$$Macro-P = \frac{1}{n} \sum_{i=1}^n P_i \quad (17)$$

$$Macro-R = \frac{1}{n} \sum_{i=1}^n R_i \quad (18)$$

$$Macro-F = \frac{1}{n} \sum_{i=1}^n F_i \quad (19)$$

其中  $n$  为实体类型的数量,本文共有六类实体,因此  $n$  为 6,  $P_i, R_i, F_i$  表示第  $i$  类实体的精确率、召回率和  $F$  值.

微平均(Micro-averaging),对测试数据中的每一个进行不分类统计,然后计算相应指标.计算公式如下:

$$Micro-P = \frac{\sum_{i=1}^n TP_i}{(\sum_{i=1}^n TP_i + \sum_{i=1}^n FP_i)} \quad (20)$$

$$Micro-R = \frac{\sum_{i=1}^n TP_i}{(\sum_{i=1}^n TP_i + \sum_{i=1}^n FN_i)} \quad (21)$$

$$Micro-F = \frac{(2 \times Micro-P \times Micro-R)}{(Micro-P + Micro-R)} \quad (22)$$

**3.3 模型对比分析** 为了验证本文提出的基于 FT-CNN-BiLSTM-CRF 的网络安全实体识别模型的识别效果,在不同的算法模型上进行对比实验.实验中的对比模型包括:CRF<sup>[6]</sup>, LSTM<sup>[11]</sup>, LSTM-CRF<sup>[13]</sup>, BiLSTM-CRF<sup>[16]</sup>, CNN-BiLSTM-CRF<sup>[18]</sup> 以及本文提出的模型 FT-CNN-BiLSTM-CRF.所有对比模型都在同一数据集上进行训练并测试,表 5 为测试集在不同模型上的实验对比结果(黑体字为最优),图 5 为六类安全实体在不同模型上的实验结果对比.

表 5 不同模型的实验结果对比

Table 5 Experimental results of the different models

模型名称	$Acc$	Macro-average			Micro-average		
		$P$	$R$	$F$	$P$	$R$	$F$
CRF	91.50	<b>75.34</b>	53.47	60.26	84.26	73.34	78.42
LSTM	92.36	67.38	52.21	56.60	83.75	80.62	82.16
LSTM-CRF	92.95	71.78	58.04	62.90	86.17	82.07	84.07
BiLSTM-CRF	92.83	69.98	<b>66.39</b>	67.81	84.70	<b>85.18</b>	84.94
CNN-BiLSTM-CRF	93.10	72.61	65.23	<b>68.11</b>	86.47	84.07	85.25
FT-CNN-BiLSTM-CRF	93.31	74.29	63.63	67.44	<b>88.45</b>	83.68	<b>86.00</b>



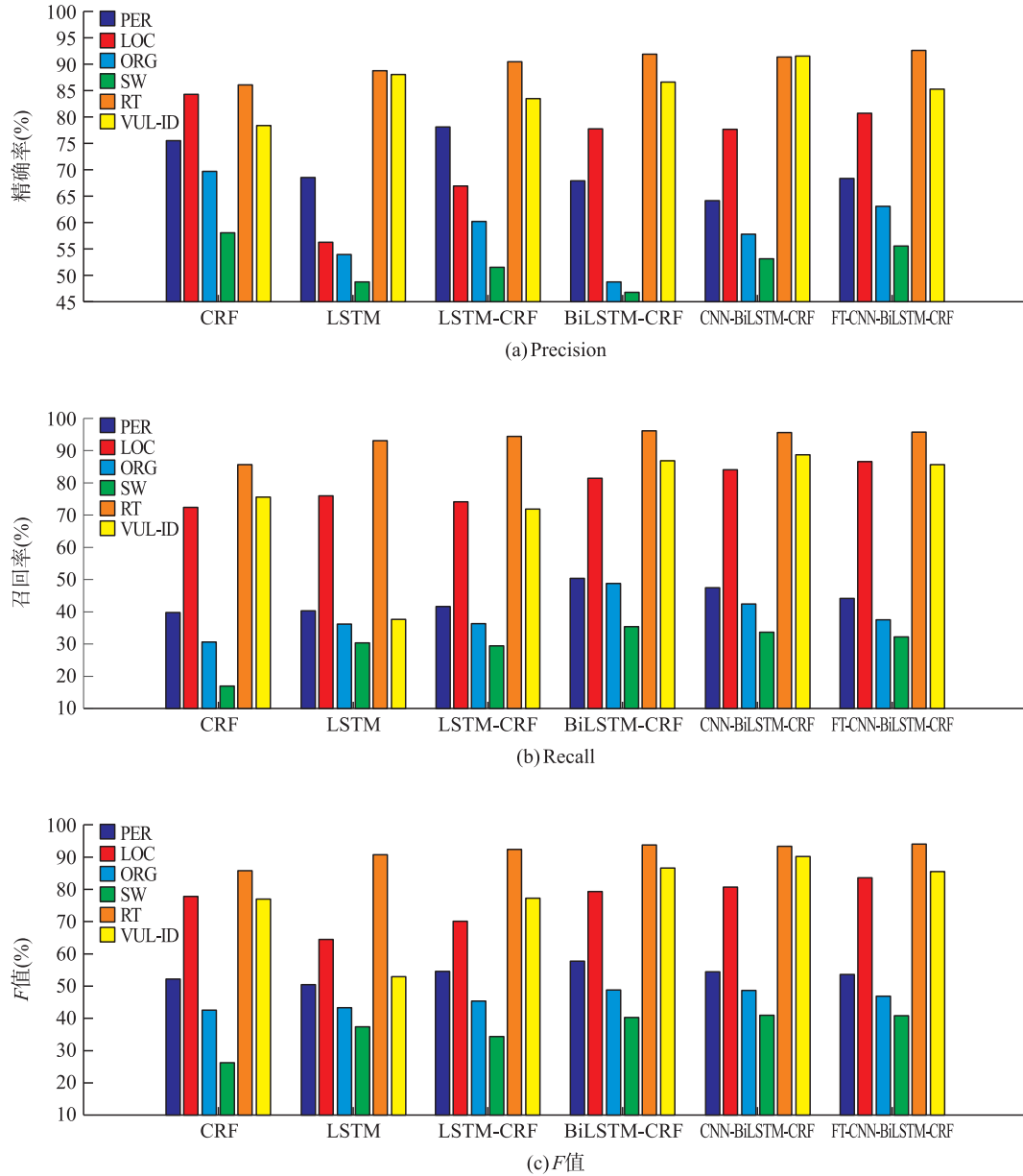


图 5 六类安全实体类型实验结果对比

Fig. 5 Experimental results of six types of security entities

从表 5 和图 5 中可以看出,通过对比 CRF 模型和 LSTM 模型,CRF 模型在 *Micro-P* 上要比模型 2 高出 0.5%,但是 *Micro-R* 低于 7.2%,其中 CRF 模型对软件名和组织名这两类安全实体的识别效果不强,召回率极低,是由于这两类安全实体大部分是由中英文组合而成,且大小写混合,因此 CRF 模型的对此类实

体的识别能力没有 LSTM 模型强. LSTM-CRF 结合了 LSTM 模型和 CRF 模型的优点,通过 LSTM 识别复杂的安全实体,同时 CRF 能够充分利用相邻标签的关系,在全局优化输出的标签序列,对长度较大以及带有修饰词汇的安全实体,识别性能较高,因此 *Micro-P* 和 *Micro-R* 都有所上升. BiLSTM-CRF 模型主要

是利用了双向 LSTM,提取长远上下文信息,对比 LSTM-CRF 模型,在 *Micro-R* 和 *Micro-F* 上高出 3.11% 和 0.82%。但是, *Micro-P* 有所下降,这是因为 BiLSTM-CRF 在软件名和组织名这两类实体的精确率较低。CNN-BiLSTM-CRF 模型相比 BiLSTM-CRF 模型的 *Micro-P* 上升了 1.77%, *Micro-F* 上升了 0.31%,表明了 CNN 抽取的字符特征的有效性。通过 CNN 模型抽取的字符级向量能够一定程度上表示形态特征,所以对于具有大小写混合,包含特殊字符,边界模糊特点的这类安全实体能够充分获取相关特征,从而提高实体识别的 *F* 值。FT-CNN-BiLSTM-CRF 模型加入了特征模板, *Micro-P* 和 *Micro-F* 相比其他五个模型达到最高,分别为 88.45% 和 86%,进一步验证了加入特征模板提取安全实体局部上下文特征能够提高网络安全实体的识别效果。

**3.4 参数调整分析** 本文提出的网络安全实体识别模型是基于神经网络模型,在对神经网络模型进行训练的过程中,模型训练的参数一般是随机初始化,但是训练效率和模型性能都不高,因此,大多数模型参数都需要调整,从而使模型的性能更高。在训练模型时,对参数的调整占了很大一部分工作时间,本文选取表 3 中的三个参数进行调整,分别是训练 epoch 数、是否设置 dropout 值以及是否对初始字向量进行 Fine-tuning。epoch 是一个完整的训练数据集通过神经网络模型训练一次并且返回一次的

过程,在模型中,epoch 一次是不够的,需要将训练数据在模型中传递很多次。训练数据经过多次 epoch 后,神经网络中的权重的更新次数也增加,曲线从欠拟合变得过拟合,因此选择适合 epoch 数是非常重要的。在训练数据较少的情况下,训练模型很容易引起过拟合,在测试集上的精确度会比较低,因此,本文设置了 dropout 参数,防止在训练过程中出现过拟合的现象。设置 dropout 之后,可以提高模型的泛化能力。

本文利用大规模的网络安全文本数据训练了字向量文件,作为模型的预训练模型,本文使用的训练数据集预训练模型的数据集都来自于网络安全文本数据。因此,本文使用模型微调(Fine-tuning),使用预训练模型中的权重来对模型进行训练,不必从头开始训练,预先训练的权重比随机初始化的权重要好,能够提高模型的计算效率和模型的精确率。

表 6 为本文对以上三类参数调整的实验结果对比。从表 6 可以看出,本文在训练模型中对 epoch, dropout 和 Fine-tuning 参数进行调整,其中 epoch 分别设置了 30, 40 和 50, dropout 设置 0 和 0.5, Fine-tuning 取值为 true 和 false。通过训练模型交叉验证可知,随着 epoch 数的增大,模型在 epoch 为 40, dropout 取 0.5, Fine-tuning 为 true 的时候,模型的对网络安全实体识别的性能最高,其中召回率和 *F* 值达到 83.68% 和 86%。

表 6 参数调整的实验结果对比

Table 6 Experimental results of parameter adjustment

Epoch	Dropout	Fine-tuning=False				Fine-tuning=True			
		Acc	Micro-average			Acc	Micro-average		
			<i>P</i>	<i>R</i>	<i>F</i>		<i>P</i>	<i>R</i>	<i>F</i>
30	0	92.69	86.81	79.50	82.99	92.89	88.21	81.20	84.56
	0.5	92.64	86.92	81.33	84.03	92.88	86.44	83.43	84.91
40	0	92.56	87.87	78.76	83.07	93.13	88.45	83.61	85.96
	0.5	92.20	85.90	81.90	83.86	93.31	88.45	83.68	86.00
50	0	92.59	88.09	78.78	83.17	93.14	88.99	81.12	84.87
	0.5	92.93	88.79	81.69	85.09	93.31	89.45	81.95	85.58



- recognition. *Proceedings of the IEEE*, 1989, 77(2):267–296.
- [5] Koeling R. Chunking with maximum entropy models // *Proceedings of the 2<sup>nd</sup> workshop on Learning Language in Logic and the 4<sup>th</sup> Conference on Computational Natural Language Learning*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2000:305–312.
- [6] Lafferty J D, McCallum A, Pereira F C N. Conditional random fields: probabilistic models for segmenting and labeling sequence data // *18<sup>th</sup> International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 2001:282–289.
- [7] 邱泉清, 苗夺谦, 张志飞. 中文微博命名实体识别. *计算机科学*, 2013, 40(6):196–198. (Qiu Q Q, Miao D Q, Zhang Z F. Named entity recognition on Chinese microblog. *Computer Science*, 2013, 40(6):196–198.)
- [8] Joshi A, Lal R, Finin T, *et al.* Extracting cybersecurity related linked data from text // *2013 IEEE Seventh International Conference on Semantic Computing*. Irvine, CA, USA: IEEE Computer Society, 2013:252–259.
- [9] Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning // *Proceedings of the 25<sup>th</sup> International Conference on Machine Learning*. Helsinki, Finland: ACM, 2008:160–167.
- [10] Collobert R, Weston J, Bottou L, *et al.* Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 2011, 12(1):2493–2537.
- [11] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8):1735–1780.
- [12] Hammerton J. Named entity recognition with long short-term memory // *Proceedings of the 7<sup>th</sup> Conference on Natural Language Learning at Hlt-Naacl*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003:172–175.
- [13] Peng N Y, Dredze M. Named entity recognition for Chinese social media with jointly trained embeddings // *Proceedings of 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: The Association for Computational Linguistics, 2015:548–554.
- [14] Huang Z H, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging. 2018, arXiv:1508.01991.
- [15] Dong C H, Zhang J J, Zong C Q, *et al.* Character-based LSTM-CRF with radical-level features for Chinese named entity recognition // *International Conference on Computer Processing of Oriental Languages*. Springer Berlin Heidelberg, 2016:239–250.
- [16] Lample G, Ballesteros M, Subramanian S, *et al.* Neural architectures for named entity recognition. 2016, arXiv:1603.01360.
- [17] Chiu J P C, Nichols E. Named entity recognition with bidirectional LSTM-CNNs. 2016, arXiv:1511.08308.
- [18] Ma X Z, Hovy E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. 2016, arXiv:1603.01354.
- [19] Mikolov T, Chen K, Corrado G, *et al.* Efficient estimation of word representations in vector space. 2013, arXiv:1301.3781.
- [20] Mikolov T, Sutskever I, Chen K, *et al.* Distributed representations of words and phrases and their compositionality // *Proceedings of the 26<sup>th</sup> International Conference on Neural Information Processing Systems*. Lake Tahoe, NV, USA: Curran Associates Inc, 2013, 26:3111–3119.
- [21] Lécun Y, Bottou L, Bengio Y, *et al.* Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998, 86(11):2278–2324.
- [22] Yang Y M. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1999, 1(1–2):69–90.

(责任编辑 杨可盛)