

基于 CNN-BiLSTM 模型的日志异常检测方法

孙 嘉^{1,2}, 张建辉², 卜佑军², 陈 博², 胡 楠^{1,2}, 王方玉^{1,2}

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

摘 要: 目前日志异常检测领域存在数据量大、故障和攻击威胁隐蔽性高、传统方法特征工程复杂等困难, 研究卷积神经网络(CNN)、循环神经网络等迅速发展的深度学习技术, 能够为解决这些问题提供新的思路。提出结合 CNN 和双向长短期记忆循环神经网络(Bi-LSTM)优势的 CNN-BiLSTM 深度学习模型, 在考虑日志键显著时间序列特征基础上, 兼顾日志参数的空间位置特征, 通过拼接映射方法进行最大程度避免特征淹没的融合处理。在此基础上, 分析模型复杂度, 同时在 Hadoop 日志 HDFS 数据集上进行实验, 对比支持向量机(SVM)、CNN 和 Bi-LSTM 验证 CNN-BiLSTM 模型的分类效果。分析和实验结果表明, CNN-BiLSTM 达到平均 91% 的日志异常检测准确度, 并在 WC98_day 网络日志数据集上达到 94% 检测准确度, 验证了模型良好的泛化能力, 与 SVM CNN 和 Bi-LSTM 相比具有更优的检测性能。此外, 通过消融实验表明, 词嵌入和全连接层结构对于提升模型准确率具有重要作用。

关键词: 日志异常检测; 深度学习; 特征融合; 泛化能力; 消融实验

开放科学(资源服务)标志码(OSID):



中文引用格式: 孙嘉, 张建辉, 卜佑军, 等. 基于 CNN-BiLSTM 模型的日志异常检测方法[J]. 计算机工程, 2022, 48(7): 151-158, 167.

英文引用格式: SUN J, ZHANG J H, BU Y J, et al. Log anomaly detection method based on CNN-BiLSTM model[J]. Computer Engineering, 2022, 48(7): 151-158, 167.

Log Anomaly Detection Method Based on CNN-BiLSTM Model

SUN Jia^{1,2}, ZHANG Jianhui², BU Youjun², CHEN Bo², HU Nan^{1,2}, WANG Fangyu^{1,2}

(1. Zhong Yuan Network Security Research Institute, Zhengzhou University, Zhengzhou 450001, China;

2. PLA Strategic Support Force Information Engineering University, Zhengzhou 450001, China)

[Abstract] At present, the field of log anomaly detection has difficulties such as large data volume, high concealment of faults and attack threats, and complex feature engineering of traditional methods. The rapid research and development of deep learning provides new ideas for solving these problems. Here we propose to combine Convolutional Neural Network (CNN) and Bi-LSTM. The superior CNN-BiLSTM deep learning model not only considers the significant time series characteristics of the log key, but also takes into account the spatial location characteristics of the log parameters, and uses the splicing mapping method to perform feature fusion processing to avoid mutual inundation to the greatest extent, which is feasible in analyzing model complexity. After the performance, based on the Hadoop log HDFS data set, comparing CNN and Bi-LSTM to verify the superior CNN-BiLSTM massification effect of the CNN-BiLSTM model, reaching about 91% log anomaly detection accuracy, and reaching 94% detection accuracy on the WC98_day Web log data set. Verify the good generalization ability of the CNN-BiLSTM model, and finally analyze the importance of word embedding and fully connected layer structure in the CNN-BiLSTM model through ablation experiments.

[Key words] log anomaly detection; deep learning; feature fusion; generalization ability; ablation experiment

DOI: 10.19678/j.issn.1000-3428.0061750

0 概述

随着共享、开放的互联网飞速发展, 网络攻击方式也呈现出自动化、多样化的发展趋势, 网络安全面

临着前所未有的挑战。网络安全威胁主要包括系统内部漏洞威胁、误操作威胁和外部攻击威胁。目前多数网络系统都会输出记录系统运行状态和执行操作的日志文件, 日志文件可以在入侵检测、故障处

基金项目: 国家自然科学基金(62176264); 郑州市协同创新重大专项(20XTZX-X010)。

作者简介: 孙 嘉(1995—), 男, 硕士研究生, 主研方向为网络安全、机器学习; 张建辉、卜佑军, 副研究员; 陈 博, 博士; 胡 楠、王方玉, 硕士研究生。

收稿日期: 2021-05-25 修回日期: 2021-08-25 E-mail: sun1851265208@163.com

理、事件关联、事故处理、事后追究等诸多方面提供帮助。对日志进行分析应用于在线监视和威胁检测,是计算机安全领域中的研究热点之一^[1]。

传统的日志分析方法多是开发人员根据专业领域知识,使用手动检查、编写规则、应用统计学分析或聚类等方法,人工进行特征识别和建立规则,但是随着网络入侵攻击由独立、简单、直接、易暴露逐渐演变成有组织、有目标、持续时间长的APT等攻击,以及逐渐规模化发展、分布式部署、高并行和冗余运行的系统应用发展^[2],海量的日志数据和高隐蔽性的攻击手段导致人工选取特征、制定规则困难和检测方法适用性低,而深度学习可为解决这些问题提供新的思路。

在大数据时代背景下,深度学习技术蓬勃发展,如聚焦学习样本空间特征的卷积神经网络(Convolutional Neural Network, CNN)^[3]、挖掘发现时间序列特征的循环神经网络(Recurrent Neural Network, RNN)^[4]。深度学习模型在参数合适的情况下不需要人工提取特征,模型本身就能完成特征提取与检测工作,在保证准确率的同时大幅减少工作量。为保证深度学习检测模型的准确率,模型结构应与数据结构相适应,并且需要足够的数据进行训练,而海量的日志数据正适用于训练深度学习模

型。此外,还应设置合适的模型参数,目前存在网格搜索、随机搜索、贝叶斯优化等多种调参算法^[5],可以协助定义模型参数,进一步降低工作量。

本文针对海量日志数据的特点,结合CNN和Bi-LSTM提取时空序列特征的优势,构建适用于日志异常检测的CNN-BiLSTM深度学习模型。通过解析日志键和日志参数,根据各自特点分别提取空间和时间序列特征,提高日志异常检测准确率。对比传统日志异常检测方法和单核结构的深度学习模型,在HDFS和WC_98day数据集上进行实验,验证CNN-BiLSTM模型的普适性。同时,通过消融实验从检测效果和模型收敛速度两个方面出发,分别测评词嵌入结构和全连接层结构对于CNN-BiLSTM模型的重要意义。

1 相关工作

1.1 传统日志异常检测方法

传统日志分析方法对比如表1所示,一般由开发人员根据相关领域知识或手动检查,或编写规则,或应用统计学分析、聚类等方法分析日志数据,但是传统方法不仅存在高度依赖特征工程、普适性差等缺点,而且对处理海量数据、检测复杂网络攻击缺少高效的解决方案。

表1 传统日志异常检测方法分析

Table 1 Analysis of traditional log anomaly detection methods

方法	过程	优点	缺点
基于人工规则处理	专家手工制定	精准查找符合规则的攻击异常	受限于专家知识,时常更新,查找耗费时间
基于关联规则挖掘	寻找高频项目组,产生关联规则(Apriori, FP-Growth)	挖掘深层次规则	依赖于离散化效果,支持度和置信度合理设置,查找耗费时间
基于统计分析	3 σ 准则, Grubbs 检验, 时间序列建模、混合建模	适合低维数据,鲁棒性高	严重依赖假设
基于聚类分析	距离,密度,互联性	多个分类簇,无需假设分布,估测度量值	不适合高维数据,需要设置合适的距离函数,易受异常值干扰

1.2 深度学习方法

目前,深度学习在日志异常检测领域的应用主要有以下3个方面:

1)RNN模型的应用研究。在诸多模型中,RNN以其时间序列强大的学习能力而受到关注:MENG等通过加入同义词和反义词训练DLCE词向量,词向量引入语义信息用以LSTM顺序检测,并通过词频统计得到例如打开和关闭此类的量化关系用于定量检测^[6];YUAN等利用LSTM神经网络提出一种无监督的在线日志异常检测框架,并设计一种能够动态调节历史信息输入长度的动态阈值算法。该算法可以根据最近的检测事件决定输入长度,在Los Alamos National 实验室网络安全日志数据集上的实验结果表明,其达到F1值约0.95的优越效果^[7];DU等基于双向长短时记忆循环神经网络(Bi-directional Long Short-Term Memory network, Bi-LSTM)构建根据任务分类的工作流模型,对日志异常实施在线检

测达到92%的正确率^[8]。

2)CNN模型的应用研究。CNN以其在图像视觉领域的卓越效果引起诸多研究者的注意:HASHEMI等将解析器、向量化器和分类器集成为一个深度网络学习模型,并采用字符级CNN处理日志事件,在综合单项目、多项目的数据集上实验评估该模型的鲁棒性,并通过提前异常检测测试评估模型预测异常的能力,达到F1值约为0.99的优越性能^[9];梅御东等使用CNN-Text基于日志数据检测软件异常,在不同数据集上达到90%左右的准确率^[10]。

3)基于注意力机制的模型。注意力机制不仅能并行计算,提高处理效率,而且可以解决日志键长距离上的梯度消失问题^[11];HUANG等基于注意力机制分别设计了日志序列编码器和参数值编码器,用以捕获日志中蕴含的语义信息,并通过不稳定的日志数据集实验评估该方法的性能和鲁棒性^[12];GUO等构建基于多头注意力的序列模型,将日志流作为

模板事件序列进行处理,通过下一个事件的预测任务训练模型进行日志异常检测,在 HDFS 数据集上不同注意力头数的实验 F1 值均稳定在 0.97 左右^[13]; NEDELKOSKI 等构建基于自注意力的编码器模型,并使用系统切换时间等辅助信息增强训练数据,通过密集的日志数据训练使模型能够更有效地区分正常和异常日志之间的差异,同时,在 BGL 等公开数据集上和 PCA 方法进行对比实验,准确率达到约 90%,F1 值约为 0.67^[14]。

上述研究面对海量日志数据分析深度学习普遍优于手工特征提取的传统方法,但是依然存在不足,例如:将日志整体作为分析对象,忽视日志键和日志参数特征不同;单核模型偏重于处理时间序列或空间位置特征;模型普适性差等。因此,深度学习在日志异常检测方面仍有较大的研究与提升空间。

本文分析日志数据的特征和传统日志异常检测方法的局限性,结合 CNN 和 Bi-LSTM 模型的优势,构建 CNN-BiLSTM 模型应用于日志异常检测。

2 理论基础

2.1 词向量表示

深度学习模型的输入只能是数值化的张量, Word2Vec 是目前自然语言处理领域常用的词向量编码方式,包含 CBOW 和 Skip-Gram 两种训练方式,其中 CBOW 适合小规模语料,而 Skip-Gram 在大规模语料上表现更佳。根据数据量大这一特点,日志编码选择 Skip-Gram 方式,其结构中的词嵌入层由包含线性变换的隐含层组成^[15],通过根据当前词预测上下文可能出现的词,最大化日志训练语料相关词出现的概率,以学习语料之间的相关关系。

令 W 是对应的词向量矩阵, C 为单个数据长度,则 W 矩阵中的每一行为单个切分词组的 N_c 维词向量 M_i :

$$M_i = W^T X_i \quad (1)$$

其中: X_i 为切词后的单个词组 one-hot 编码,第 i 位为 1,其余位皆为 0。

2.2 卷积神经网络

文本卷积神经网络 (Text-CNN) 提取文本主要特征以进行分类,不同于图像的卷积过程,为能多角度提取特征和维护文本完整特征表达,Text-CNN 使用多种规格卷积窗口进行一维滑动,且卷积窗口宽度与词向量长度相等。

如图 1 所示,Text-CNN 是只构建输入层、卷积层、池化层、全连接层四层的卷积模型,其中 $M_i \in W_c$ 是数据中第 i 个切分词组对应的 N_c 维词向量,卷积运算类似滤波器 $C \in W_{h-c}$,该过程使用 RELU 激活函数,将其应用于 h 个词组的词向量矩阵提取特征,如从 $M_{i:i+h-1}$ 的窗口内提取特征 Y_i :

$$Y_i = f(C \cdot M_{i:i+h-1} + b) \quad (2)$$

其中: b 为偏置项; f 为非线性函数。为防止训练过程

中过度依赖局部特征,同时避免过拟合和增强模型泛化能力,在反向传播过程中使用 Dropout 函数,使神经元激活值按照概率 P 随机丢失,如式 (3) 所示:

$$y_c = w \left(r \circ \bar{Y}_i \right) + b \quad (3)$$

其中: \circ 是逐元素乘法运算符; r 是 Bernoulli 函数按照概率 P 随机生成的 0 或 1。

通过多种规格的卷积窗口提取数据不同角度的空间特征,并经过池化层选取价值最高的特征向量。

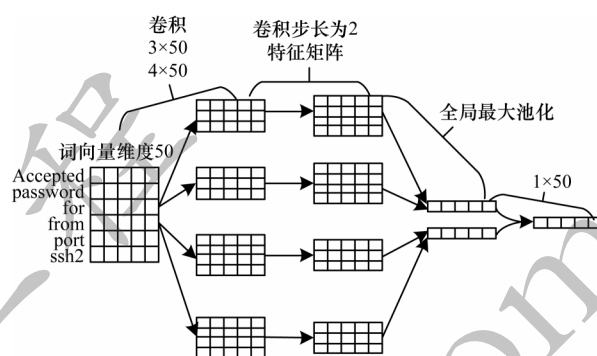


图1 Text-CNN 特征提取过程

Fig.1 Text-CNN feature extraction process

2.3 双向长短时记忆循环神经网络

双向长短时记忆循环神经网络 (Bi-LSTM) 是在长短时记忆神经网络的基础上增加双向输入进行改进^[16]。Bi-LSTM 不仅设置输入门、遗忘门、输出门以解决循环神经网络长期依赖缺失的问题,而且利用双向输入同时捕获序列正反方向特征信息,从更多角度学习序列特征信息。

如图 2 所示,Bi-LSTM 借鉴双向 RNN 输入方式,将 RNN 中的循环单元替换为带有门控单元的 LSTM 循环单元,等同于在序列两端各构建单向 LSTM,且都连接于同一层,这个结构提供输出层输入序列中完整的上下文信息,从正反两个方向学习序列特征。

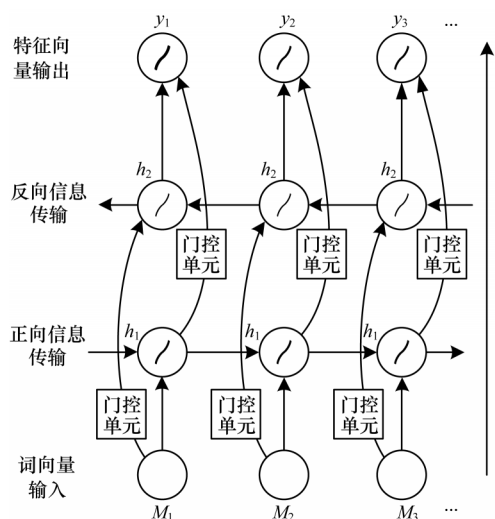


图2 Bi-LSTM 特征提取过程

Fig.2 Bi-LSTM feature extraction process

神经网络正向更新为:

$$\vec{h}_t = H\left(W_{x\vec{h}_t}x_t + W_{\vec{h}\vec{h}}\vec{h}_{t-1} + b_{\vec{h}}\right) \quad (4)$$

神经网络反向更新为:

$$\overleftarrow{h}_t = H\left(W_{x\overleftarrow{h}_t}x_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}\right) \quad (5)$$

因此,正反双向循环神经网络层结合输出为:

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_y \quad (6)$$

其中: t 是时间序列; \vec{h} 对应下标时间的隐层向量; x 对应下标时间的输入; y 对应下标时间的输出; W 表示对应下标输入和隐层、隐层和隐层、隐层和输出之间的权重矩阵; b 为对应下标隐层或输出层偏置向量; H 为隐层 sigmoid 激活函数。

3 CNN-BiLSTM 模型

如图3所示,基于深度学习进行日志异常检测实际是序列预测任务^[17],本节根据日志结构特点,结合 CNN 和 Bi-LSTM 模型的优势构建 CNN-BiLSTM 模型应用于日志异常检测。

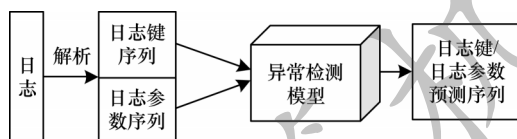


图3 日志异常检测流程

Fig.3 Procedure of log anomaly detection

3.1 模型结构

CNN 和 Bi-LSTM 存在两种融合结构:一种是并行处理结构,两种模型提取特征向量后,拼接融合输入分类器;另一种是先 CNN 后 Bi-LSTM 或先 Bi-LSTM 后 CNN 的串行处理结构,特征叠加到同一向量进行表达。选择融合方式需要深入分析日志特征。

日志格式各不相同,但都是由源代码输出语句生成^[18],如源代码中的日志打印语句为 `printf("Accepted password for %s from %s port %d ssh2 \n", user, host, port)`,意图打印使用安全外壳协议 ssh2 从访问主机名、IP 地址、端口接收到密码,就会生成如 Feb 28 04:48:54 combo sshd (pam_unix)[6741]: Accepted password for root from 112.64.243.186 port 2371 ssh2 的日志记录,其中源代码生成的固定内容称为日志键或日志常量,随系统状态变化而生成的变量称为日志参数,同一源代码生成的日志记录为相同类型。日志的基本特征可总结如下:

1) 日志是具有一定格式的文本数据,日志内容可以分为常量的日志键以及变量的日志参数。

2) 日志键决定日志生成顺序,且由于部分任务时间跨度长、并发或并行执行的原因,其在较长的时间维度上与前后日志条目存在较强的相关性。

3) 同一日志条目内日志参数之间存在关联意义,不同日志条目之间的日志参数相关性较弱。

由上分析可知,日志键和日志参数特征存在较

大的差异,需要分别处理。一方面,在并联结构中,并行处理不仅有利于提高处理效率,而且卷积和循环神经网络互不影响,能够使特征向量更纯粹地表达日志键的长距离序列特征或日志参数的短距离特征信息;另一方面,在串联结构中,日志键特征向量经过 CNN 卷积结构后,只能保留卷积窗口长度内的时间序列特征,日志参数特征向量经过 Bi-LSTM 后也会受到冗余的长距离依赖关系的影响,不利于短距离上的空间特征表达^[19-20],而且较深的网络结构也不利于日志浅层特征的表达。综上,选择并行的模型结构更有利于日志键和日志参数的特征表达。

CNN-BiLSTM 模型结构如图4所示,其主要由词嵌入层、卷积-循环层、特征融合层、分类层四部分组成。模型主要参数如表2所示。

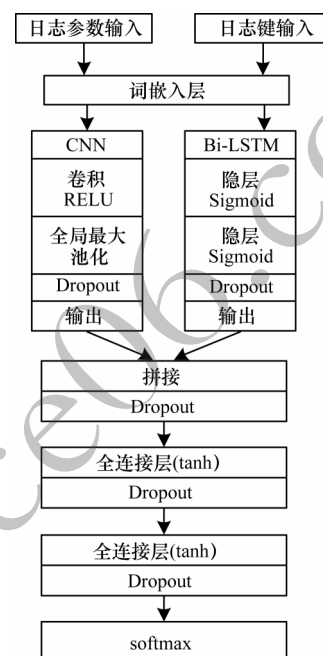


图4 CNN-BiLSTM 模型结构

Fig.4 CNN-BiLSTM model structure

表2 CNN-BiLSTM 参数设置

Table 2 CNN-BiLSTM parameters setting

参数项	参数设置
词嵌入维度	50
卷积核数	100
卷积窗口大小	3/50, 4/50
卷积步长	2
卷积激活函数	RELU
池化类型	全局最大池化
循环层大小	50
循环层激活函数	Sigmoid
全连接层隐层大小	1×100, 1×50
全连接层激活函数	tanh
优化器	Adam(1e-5)
Dropout	0.5/5

词嵌入层将解析后的日志键和日志参数通过 Word2Vec 映射为向量矩阵, 即将切分后的日志键和日志参数中的词组映射为具有固定长度 L 的一列向量, 则单个日志条目映射为 $N \times L$ 大小向量矩阵。在 CNN-BiLSTM 模型实验中, 词嵌入层输出维度为 50, 即一条日志键或日志参数经词嵌入层映射为 10×50 的向量矩阵作为卷积-循环层输入。

卷积-循环层使用卷积神经网络提取日志参数的短距离序列特征, 同时使用循环神经网络提取日志键的时间序列特征。CNN-BiLSTM 模型卷积神经网络使用宽为 3, 4 的卷积窗口各 50 个, 且卷积步长为 2。为得到区分度明显的特征, 使用全局最大池化, 单个日志条目经过最大池化层后得到长度为 50 的输出向量。双向循环神经网络隐层神经元各为 25, 即得到蕴含时间特征信息长度为 50 的特征向量。

特征融合层主要是拼接卷积-循环层得到的两种特征向量, 并且为增强特征融合度和模型拟合能力, 连接两层使用 tanh 激活函数的全连接神经网络。根据卷积-循环层的输出, CNN-BiLSTM 模型特征向量融合后会得到 1×100 的融合特征向量。

分类层使用 softmax 函数, 给予预测序列概率分布判断日志异常与否。在测试阶段, 当实际日志序列与分布概率最高的预测日志序列不同, 即判断为异常。

模型使用 Adam 优化器, 学习率为 $1e-5$ 。为进行正则化和防止过拟合, 全局使用值为 0.5 的 5 层 Dropout。

3.2 复杂度分析

由于卷积神经网络主要操作时间为各个卷积层上的卷积窗口滑动时间, 层内相乘, 层间累加, 因此

卷积神经网络时间复杂度定义为 $O\left(\sum_{l=1}^D \left(\frac{m}{k}\right)^s \cdot C_{l-1} \cdot C_l\right)$,

其中: D 是卷积层数; l 是第 l 个卷积层; m 是输入向量维度; k 是卷积步长; s 是卷积核滑动维度; C 是通道数即卷积核个数。CNN-BiLSTM 模型中使用两种规格的卷积核各 50 个, 且窗口只进行一维滑动, 因此,

时间复杂度为 $O\left(\frac{m_1 + m_2}{k}\right)$ 。

循环神经网络主要操作时间是输入向量在各个隐藏单元之间的映射运算时间, 因为每个单元都与其他单元之间存在映射关系, 所以循环神经网络时间复杂度定义为 $O(nd^2)$, 其中: n 为序列操作次数; d 为隐藏单元数。CNN-BiLSTM 模型使用两层全连接层实现模型特征融合, 由于全连接层实现线性映射运算, 因此全连接层时间复杂度为 $O(f_1 \cdot f_2 + f_2 \cdot f_3 + f_3 \cdot 1)$, 其中: f_1 是输入向量长度; f_2, f_3 是全连接层神经元数。对于问题规模 f , 各层神经元个数为常量, 则时间复杂度为 $O(f)$ 。

根据以上分析可知, CNN-BiLSTM 模型中卷积和循环为并行结构, 全连接层映射为串行结构, 因此, CNN-BiLSTM 模型整体时间复杂度为

$$O\left(\max\left[\frac{m_1 + m_2}{k}, nd^2\right] + f\right)。$$

相对于 CNN 和 LSTM 模型, CNN-BiLSTM 模型在兼顾两种模型优势的同时, 其时间复杂度并未有较高提升, 而是于合理时间范畴内趋于卷积和循环结构中较高的时间复杂度。关于 CNN-BiLSTM 模型的空间复杂度, 考虑到目前系统较高的硬件性能与计算能力, 本文不作分析。

4 实验验证与分析

本节对 CNN-BiLSTM 深度学习模型应用于日志异常检测的性能进行验证。实验以主流用于日志异常检测的 CNN 和 Bi-LSTM 深度学习模型为基准模型, 并使用 HDFS 和 WC_day13 两个数据集作为实验数据集, 以验证 CNN-BiLSTM 模型的准确率和普适性。实验所涉及深度学习模型均基于编程框架 keras 实现。

4.1 数据集分析与处理

实验使用从 Amazon EC2 平台收集的公开 Hadoop 日志的 HDFS 数据集, 以及包含 1998 年世界杯赛官网 92 天访问信息的 WC98_day 公开日志数据集。

HDFS 日志数据集通过 200 多个 Amazon EC2 节点上运行的基于 Hadoop 的 map-reduce 作业生成, 共包含 11 175 629 条日志信息, 信息包括时间(年月日、时分秒)、源 IP、数据大小等字段, 根据 Block 操作码可以划分为 575 062 组操作序列, 其中 2.9% 被 Hadoop 领域相关专家标记为异常, 类型包括写入异常等事件, 常用于在线主成分分析研究使用^[21], 后被研究者应用于日志异常识别的深度学习模型训练。

在 HDFS 数据预处理过程中, 首先删除重复、空白数据, 原 11 175 629 条日志信息余 11 173 720 条日志信息, 之后基于标点符号、空格信息, 利用正则表达式解析日志键和日志参数以及 Block 标签信息, 并删除不存在区分度的 INFO、dfs、数字等字段, 然后根据 Block 标签信息对应日志键和日志参数。

WC98_day 数据集用以验证 CNN-BiLSTM 模型的普适性, 数据收集 1998 年 4 月 26 日—1998 年 7 月 26 日的 92 天世界杯期间赛事官网的 1 352 804 107 次请求信息, 数据记录包括时间、用户临时 ID、登录协议、登录状态码、登录详情等信息, 其中约 10% 为异常日志。

4.2 评价方法与指标

为避免出现测试数据中预测样本数据全为正常以致高准确率的现象, 实验选用准确率 A_{Accuracy} 、查准率 $P_{\text{Precision}}$ 、召回率 R_{Recall} 和 F1 值 4 个指标^[22], 计算公式如下:

$$A_{\text{Accuracy}} = \frac{T}{N_{\text{ALL}}} \quad (7)$$

$$P_{\text{Precision}} = \frac{T_p}{T_p + F_p} \quad (8)$$

$$R_{\text{Recall}} = \frac{T_p}{T_p + F_N} \tag{9}$$

$$F_1 = \frac{2 \times R_{\text{Recall}} \times P_{\text{Precision}}}{R_{\text{Recall}} + P_{\text{Precision}}} \tag{10}$$

其中： N_{ALL} 是总样本数； T 是预测正确的样本数； T_p 是正常且预测为正常的样本数； F_p 是正常预测为异常的样本数； F_N 是异常预测为正常的样本数。

准确率可以直观体现模型的准确性。查准率和召回率可以反映模型是否处于过拟合状态：查准率较低说明模型偏向于输出异常标签，召回率较低说明模型偏向于输出正常标签；F1值则综合反映查准率和召回率两个指标，其值越高说明模型拟合效果越好。

4.3 实验分析

本节进行3组对比实验，验证CNN-BiLSTM深度学习模型检测异常日志能力：基于HDFS数据集对比SVM、CNN-BiLSTM、CNN和Bi-LSTM模型检测效果的实验；基于WC98_day数据集的检验CNN-BiLSTM、CNN和Bi-LSTM模型普适性的对比实验；基于HDFS数据集的CNN-BiLSTM模型消融实验。

实验1 基于HDFS数据集对比SVM、CNN-BiLSTM、CNN和Bi-LSTM模型检测效果。通过检测HDSF日志数据集中的异常日志，并对比常用于海量日志异常检测的两种深度学习模型CNN和Bi-LSTM，衡量CNN-BiLSTM深度学习模型的检测能力。HDFS数据集中60%用于训练数据，其他40%作为测试数据。考虑到数据不平衡问题，实验过程中调试class_weight设置为正常：异常=0.930 9：0.069 1时，模型训练效果较好。

为更客观地衡量CNN-BiLSTM模型检测能力，如表3所示，CNN和Bi-LSTM各模型超参数分别与CNN-BiLSTM模型卷积和循环结构超参数相同，且输入是Word2Vec编码后的相同词向量，模型后同样连接两层全连接神经网络。

表3 对比模型参数
Table 3 Comparison model parameters

对比模型	参数项	参数值
CNN	输入维度	50
	卷积核个数	50+50
	卷积核大小	3/50, 4/50
	卷积步长	2
	池化类型	最大值
	池化窗口大小	1×50
	激活函数	RELU
	全连接层隐层大小	1×100, 1×50
	全连接层激活函数	tanh
	输入维度	50
Bi-LSTM	循环层大小	25+25
	激活函数	Sigmoid
	全连接层隐层大小	100/1, 50/1
	全连接层激活函数	tanh

训练使用K折交叉验证方法，K值为5，最终得到各模型平均准确率、查准率、召回率、F₁值指标如表4所示。

表4 5折交叉实验结果对比
Table 4 Comparison of 5-fold crossover experiment results

模型	K	准确率	查准率	召回率	F ₁ 值
CNN	5	0.85	0.88	0.94	0.90
Bi-LSTM	5	0.89	0.88	0.98	0.92
CNN-BiLSTM	5	0.92	0.92	0.98	0.94
SVM	5	0.83	0.00	0.00	0.00

将4种模型用于测试数据进行对比分析，结果如图5所示。可以看出，在超参数相同的情况下，CNN-BiLSTM深度学习模型较单个CNN、传统SVM方法准确率约提高0.14，其查准率和F1值也处于领先水平，这说明CNN-BiLSTM深度学习模型兼顾CNN和Bi-LSTM模型优势，对比单核模型，性能提升显著。但其召回率低于最高水平Bi-LSTM模型约0.11，低于模型本身查准率约0.15，这是由于数据集中正常数据远多于异常数据造成的影响，由此可见，训练过程中CNN-BiLSTM相比于对比模型更容易受到数据集不平衡的影响，更倾向于将异常样本预测为正常样本。

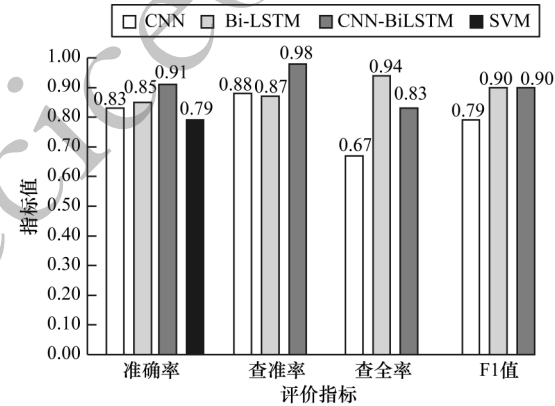


图5 HDFS日志异常检测结果
Fig.5 HDFS log anomaly detection result

实验2 基于WC98_day数据集检验CNN-BiLSTM、CNN和Bi-LSTM模型的普适性，以及使用欠采样方法解决数据不平衡问题的对比实验。CNN-BiLSTM、CNN、Bi-LSTM三种深度学习模型在保证公共超参数相同的情况下，在WC98_day日志数据集上进行异常检测实验。数据集中欠采样正常和异常数据各100 000条，其中60%作为训练数据，40%作为测试数据，class_weight参数调整为正常：异常=0.5：0.5，得到各模型准确率、查准率、召回率、F1值指标如图6所示。

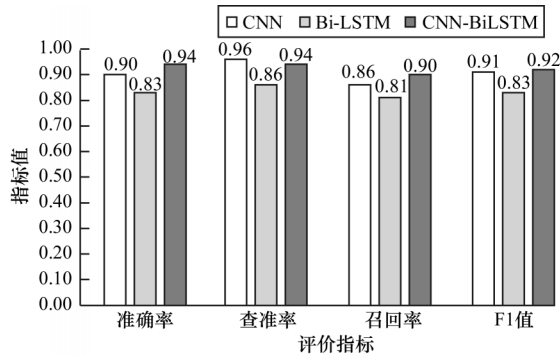


图 6 WC98_day 日志异常检测结果

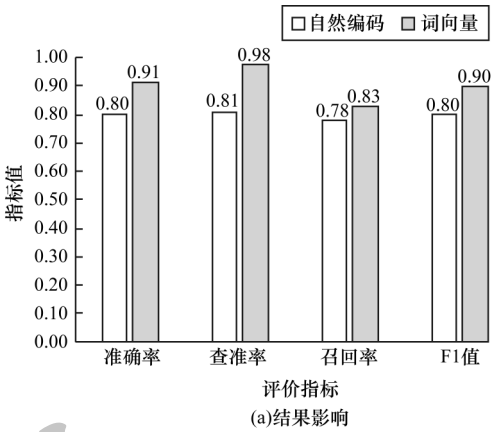
Fig.6 WC98_day log abnormal detection result

由实验结果分析可知,从 HDFS 数据集至 WC98_day 数据集的模型迁移实验中,CNN-BiLSTM 模型依然保持较高的性能优势,其准确率、召回率、F1 值三个指标分别高于次优模型 CNN 为约 0.04、0.05、0.01,同时模型自身召回率低于查准率 0.04,但远低于实验 1 中 15% 的差值,这说明过采样、欠采样等这类调整数据数量但不丰富数据特征的方法,能够在一定程度上解决数据集不平衡问题。

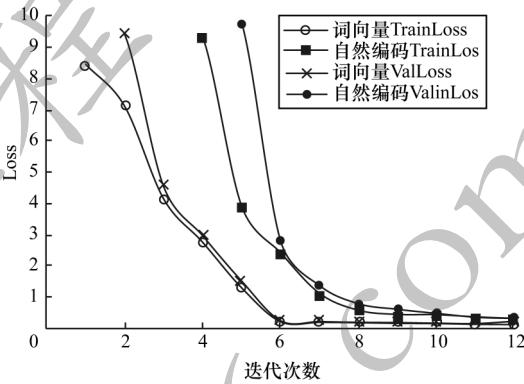
此外,对比 HDFS 和 WC_98day 两个数据集上 3 个模型的实验结果,结果表明,CNN 和 CNN-BiLSTM 模型在 WC98_day 数据集的上准确率高于 HDFS 数据集,WC98_day 相比于 HDFS 数据集包含更多的文本文字特征,而 HDFS 数据集日志参数包含更多的数值特征,由此可见,CNN 和 CNN-BiLSTM 模型更擅长提取文本特征,HDFS 数据集上的检测效果受到丢弃的数值特征影响而降低。

实验 3 基于 HDFS 数据集的 CNN-BiLSTM 模型消融实验。在不影响 CNN-BiLSTM 模型提取日志空间和时间序列特征能力的情况下,通过消融 CNN-BiLSTM 深度学习模型部分结构,研究其对于模型的价值:

1) 消融 CNN-BiLSTM 模型词嵌入层,将 HDFS 数据集中的训练数据自然编码后作为 CNN-BiLSTM 模型的输入,实验结果如图 7 所示。可以看出,词向量 CNN-BiLSTM 模型准确率、查准率、召回率、F1 值指标分别高于自然编码 CNN-BiLSTM 模型 13%、8%、6%、13%,由此可见词嵌入层对模型性能具有积极影响。根据训练 Loss (TrainLoss) 和验证 Loss (ValLoss) 变化趋势,可以判断词向量 CNN-BiLSTM 模型在第 6 次迭代就已经达到收敛平衡,而自然编码 CNN-BiLSTM 模型在第 8 次达到收敛平衡,证明经过日志语料训练 Word2Vec 产生的词向量相比于自然编码更能凸显日志语料特征,其在放大特征的同时亦保持良好语料间依赖关系,比较适合 CNN-BiLSTM 模型的输入。



(a)结果影响



(b)收敛影响

图 7 消融实验结果 (词嵌入层)

Fig.7 Ablation experiment result (word embedding layer)

2) 消融 CNN-BiLSTM 模型全连接层,分别设计两层全连接层、无全连接层、四层全连接层 3 种 CNN-BiLSTM 模型,根据 Loss 判断其对模型收敛速度的影响,实验结果如图 8 所示。可以看出,2 层全连接网络 CNN-BiLSTM 模型在第 6 次迭代达到全局收敛,而无全连接层的 CNN-BiLSTM 模型在第 11 次迭代达到收敛,4 层全连接层在第 7 次迭代会产生过拟合问题,且含有 2 层全连接层网络的 CNN-BiLSTM 深度学习模型准确率为 0.91,高于无全连接网络模型 0.83 的准确率。实验结果表明,合适的全连接网络层数能够提高模型的拟合能力与特征表达能力,且在一定程度上促进日志键时序特征与日志参数空间特征产生融合关系,而添加过深的全连接层网络,容易在训练过程中产生过拟合问题。

以上 3 组实验结果表明,对比同等深度学习模型,CNN-BiLSTM 深度学习模型在针对日志异常检测任务中,不仅能够提取日志时间和空间序列特征,达到领先的检测水准,而且通过迁移模型至另一数据集的实验证明,其在普适性方面优于单核模型,并在保证一定检测能力的情况下,基本不需要特征工程工作,降低工作量的优势明显。另一方面,根据 CNN-BiLSTM 模型两种数据集的不同检测效果对比,亦说明 CNN-BiLSTM 模型比较适用于文本特征的学习与检测。

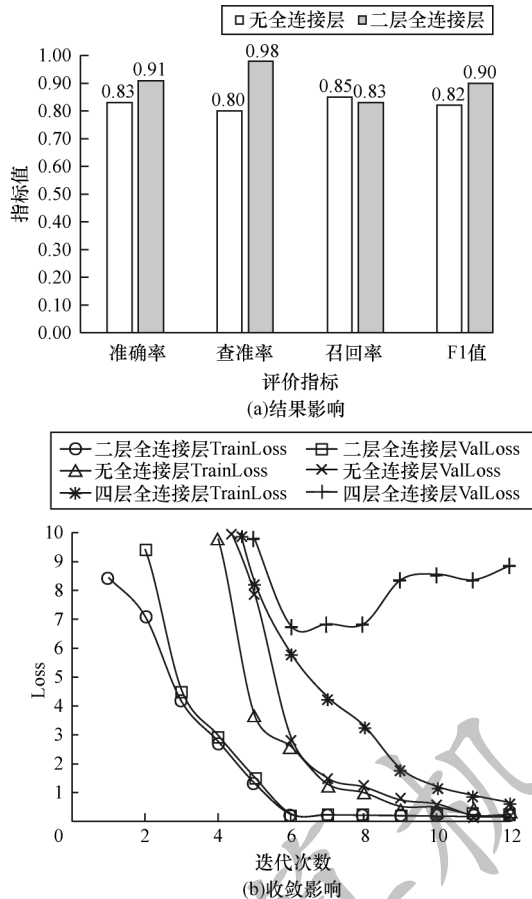


图8 消融实验结果(全连接层)

Fig.8 Ablation experiment result (fully connected layer)

5 结束语

本文结合CNN和Bi-LSTM构建双核CNN-BiLSTM模型用于完成日志异常检测任务,解决海量日志异常检测模型普适性差、准确率低等问题。实验结果表明,CNN-BiLSTM模型相比于CNN和Bi-LSTM模型,准确率、查准率、召回率和F1值指标均有不同程度的提升,能够在提高准确率的同时降低误报率。后续将针对数据集不平衡使模型预测产生偏向的问题,从训练数据角度进一步提高CNN-BiLSTM模型性能,同时结合具体工作场景,在CNN-BiLSTM模型的基础上研究多核模型在多源日志融合检测方面的适用性,提高模型辅助解决系统异常问题的能力。

参考文献

[1] AGOSTI M, CRIVELLARI F, DI NUNZIO G M. Web log analysis; a review of a decade of studies about information acquisition, inspection and interpretation of user interaction [J]. Data Mining and Knowledge Discovery, 2012, 24(3): 663-696.

[2] HE S L, ZHU J M, HE P J, et al. Experience report: system log analysis for anomaly detection[C]//Proceedings of the 27th International Symposium on Software Reliability Engineering. Washington D. C., USA: IEEE Press, 2016:

207-218.

[3] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[4] MIKOLOV T, KARAFIA T M, BURGET L, et al. Recurrent neural network based language model [C]//Proceedings of 2010 Conference of the International Speech Communication Association. Makuhari, Japan: [s. n.], 2010: 1045-1048.

[5] GOLOVIN D, SOLNIK B, MOITRA S, et al. Google Vizier: a service for black-box optimization[C]//Proceedings of the 23rd ACM SIGKDD International Conference. New York, USA: ACM Press, 2017: 1-10.

[6] MENG W B, LIU Y, ZHU Y C, et al. LogAnomaly: unsupervised detection of sequential and quantitative anomalies in unstructured logs[EB/OL]. [2021-04-10]. https://blog.csdn.net/qq_37660745/article/details/108471442.

[7] YUAN Y L, SRIKANT ADHATARA S, LIN M K, et al. ADA: adaptive deep log anomaly detector[C]//Proceedings of IEEE Conference on Computer Communications. Washington D. C., USA: IEEE Press, 2020: 2449-2458.

[8] DU M, LI F F, ZHENG G N, et al. DeepLog: anomaly detection and diagnosis from system logs through deep learning [C]//Proceedings of 2017 ACM SIGSAC Conference on Computer and Communications Security. New York, USA: ACM Press, 2017: 1-10.

[9] HASHEMI S, MÄNTYLÄ M. OneLog: towards end-to-end training in software log anomaly detection[EB/OL]. [2021-04-10]. <https://arxiv.org/abs/2104.07324>.

[10] 梅御东, 陈旭, 孙毓忠, 等. 一种基于日志信息和CNN-text的软件系统异常检测方法[J]. 计算机学报, 2020, 43(2): 366-380.

MEI Y D, CHEN X, SUN Y Z, et al. A method for software system anomaly detection based on log information and CNN-text[J]. Chinese Journal of Computers, 2020, 43(2): 366-380. (in Chinese)

[11] DUAN S F, ZHAO H. Attention is all You need for Chinese word segmentation[C]//Proceedings of 2020 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, USA: Association for Computational Linguistics, 2020: 1-10.

[12] HUANG S H, LIU Y, FUNG C, et al. HitAnomaly: hierarchical transformers for anomaly detection in system log [J]. IEEE Transactions on Network and Service Management, 2020, 17(4): 2064-2076.

[13] GUO Y C, WEN Y J, JIANG C W, et al. Detecting log anomalies with multi-head attention (LAMA)[EB/OL]. [2021-04-10]. <https://arxiv.org/abs/2101.02392>.

[14] NEDELKOSKI S, BOGATINOVSKI J, ACKER A, et al. Self-attentive classification-based anomaly detection in unstructured logs [C]//Proceedings of 2020 IEEE International Conference on Data Mining. Washington D. C., USA: IEEE Press, 2020: 1196-1201.

[15] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality [C]//Proceedings of the 26th Advances in Neural Information Processing Systems. New York, USA: Curran Associates, 2013: 3111-3119.

(下转第167页)

(上接第158页)

- [16] ZHENG J. A novel computer-aided multi-label emotion recognition of text method based on word embedding and BiLSTM[C]//Proceedings of International Informatization and Engineering Associations. [S. l.]: Computer Science and Electronic Technology International Society, 2019: 10.
- [17] 王勇,李战怀,张阳. 基于序列关联规则挖掘的Web日志预测精度研究[J]. 计算机工程, 2006, 32(12): 39-41.
WANG Y, LI Z H, ZHANG Y. Mining sequential association rule for improving Web document prediction [J]. Computer Engineering, 2006, 32(12): 39-41. (in Chinese)
- [18] 任肖肖. 基于多源报警日志的网络安全威胁态势感知关键技术研究[D]. 郑州:解放军信息工程大学, 2014.
REN X X. Research on crucial technologies of network security threat situation awareness based on multi-source alerts [D]. Zhengzhou: PLA Information Engineering University, 2014. (in Chinese)
- [19] YANG J, YANG J Y, ZHANG D, et al. Feature fusion: parallel strategy vs. serial strategy[J]. Pattern Recognition, 2003, 36(6): 1369-1381.
- [20] SUN Q S, ZHONG J, HENG P A, et al. A novel feature fusion method based on partial least squares regression[C]//Proceedings of ICAPR'05. Berlin, Germany: Springer, 2005: 268-277.
- [21] WEI X, LING H, ARMANDO F, et al. Detecting large-scale system problems by mining console logs[C]//Proceedings of 2009 ACM Symposium on Operating Systems Principles. New York, USA: ACM Press, 2009: 117-132.
- [22] SHI C Y, XU C J, YANG X J. Study of TFIDF algorithm [J]. Journal of Computer Applications, 2009, 29(s1): 167-180.

编辑 金胡考