网络出版时间：2019-07-09 09:24:02
网络出版地址：http://kns.cnki.net/kcms/detail/11.1826.TP.20190708.1745.006.html

一种基于日志信息和 CNN-text 的软件系统异常检测方法

梅御东[1,2)] 陈旭[1,2)] 孙毓忠[1)] 牛逸翔[1)] 肖立[1)] 王海荣[3)] 冯百明[4)]

[1)](中国科学院计算技术研究所,计算机体系结构国家重点实验室,北京 100080)

[2)](中国科学院大学, 北京 101400)

[3)](北方民族大学，计算机科学与工程学院，银川 750021)

[4)](西北师范大学，计算机科学与工程学院，兰州 730070)

摘 要 当前，数据挖掘作为一种高时效性、高真实性的分析方法，正在社会中扮演着越发重要的角色，其在大型数据中快速挖掘模式，发现规律的能力正逐步取代人工的作用。而在当前各个计算机领域大行其道的大型分布式系统（如 Hadoop、Spark 等）的日志中，每天都产生着数以百万计的系统日志，这些日志的数据量之庞杂、关系之混乱，已大大影响了程序员对系统的人工监控效率，同时也提高了新程序员的培养成本。为解决以上问题，数据挖掘及系统分析两个领域相结合是一种必然的趋势，也因此，机器学习模型也越来越多地被业界提及用于做系统日志分析。然而大多情况下系统日志中，报告系统运行状态为“严重”的日志占少数，而这些少数信息才是程序员最需要关注的，然而大多用于系统日志分析的机器学习模型都假设训练集的数据是均衡数据，因此这些模型在做系统日志预警时容易过度偏向大样本数据，以至于效果不够理想。本文将从深度学习角度出发，探究深度学习中的 CNN-text（CT）在系统日志分析方面的应用能力，通过将 CT 与主流的系统日志分析机器学习模型 SVM、决策树对比，探究 CT 相对于这些算法的优越性；将 CT 与 CNN-RNN-text（CRT）进行对比，分析 CT 对特征的处理方式，证实 CT 在深度学习模型中处理系统日志类文本的优越性；最后将所有模型应用至两套不同的日志类文本数据中进行对比，证明 CT 的普适性。在 CT 同日志分析的主流机器学习模型对比的实验中，CT 相较于最优模型的结果召回率提升了近 15%；在 CT 同 CRT 模型对比的实验中，CT 相较于更为先进的 CRT 模型准确率高出约 20%，召回率高出约 80%、查准率高出约 60%；在 CT 的普适性实验中，将各类模型融入到本文的实验数据集 logstash 和公开数据集 WC85_1 中，在准确率同其他表现较优的模型同为 100%的情况下，CT 的召回率高出其余召回率最高的模型（DT-Bi）近 14%。从中可看出，相较于主流系统日志分析机器学习模型，如支持向量机、决策树、朴素贝叶斯等，CNN-text 的局部特征提取能力及非线性拟合能力都有更为优异的表现；同时相较于同为深度学习 CNN 簇的 CNN-RNN-text 由于将大量权重投入到系统日志的序列特征中的特点，CNN-text 则报以较少的关注，反而在序列不规则的系统日志中展现出比 CNN-RNN-text 更优秀的表现。最终证明了 CNN-text 是本文所提到的方法中最适合进行软件系统异常检测的方法。

关键词 系统日志分析；系统异常预警；不均衡数据；机器学习；深度学习；CNN-text

中图法分类号 TP181

A Method for Software System Anomaly Detection Based on Log Information and CNN-text

MEI Yu-Dong[1,2)] CHEN Xu[1,2)] SUN Yu-Zhong[1)] NIU Yi-Xiang[1)] XIAO Li[1)] WANG Hai-Rong[3)] FENG Bai-Ming[4)]

[1)](State Key Laboratory of Computer Architecture,Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)

[2)](University of Chinese Academy of Sciences, Beijing 101400)

本课题得到面向云计算的网络化操作系统(No. 2016YFB1000505)、国家自然科学基金委员会(NSFC)-广东省人民政府联合基金超级计算科学应用研究专项计划(第二期) (U1611261)、宁夏自治区重点研发计划（引才专项）（2018BEB04002）资助. 梅御东，男，1996 年生，博士研究生，主要研究兴趣为机器学习和数据挖掘. E-mail：201436009@uibe.edu.cn. 陈旭，男，1993 年生，硕士研究生，主要研究兴趣为机器学习和数据挖掘. E-mail：chenxuict@163.com. 孙毓忠，男，1968 年生，博士，研究员，计算机学会(CCF)会员(06433D)，主要研究领域为大数据智能（机器学习）分析与计算. E-mail: yuzhongsun@ict.ac.cn. 牛逸翔，男，1995 年生，硕士研究生，主要研究兴趣为大数据和云计算，E-mail：786465269@qq.com. 肖立（**通信作者**），男，1987 年生，博士，主要研究兴趣为人工智能、计算医学. E-mail: xiaoli@ict.ac.cn. 王海荣，女，1977，博士，主要研究兴趣为大数据知识工程，E-mail:bmdwhr@163.com. 冯百明，男，1966 年生，博士，主要研究兴趣为云计算与智能计算.Email:solluno@nwnu.edu.cn

[3)](North Minzu University, School of Computer Science and Engineering, Yinchuan ,750021)

[4)] (Northwest Normal University, School of Computer Science and Engineering, Lanzhou 730070)

Abstract At present, data mining, as a method of analysis with high timeliness and high fidelity, is playing an increasingly important role in society. Its quick pattern-discovering ability in large-scale data, and the ability to quickly discover laws is gradually replacing the role of manpower. In the current large-scale distributed systems (such as Hadoop, Spark, etc.), there are tens of thousands of system logs every day. The amount of data in these logs and the chaos of the relationship have greatly affected the programmers. Manual monitoring of the system's efficiency also increases the cost of training for new programmers. To solve the above problems, the combination of data mining and system analysis is an inevitable trend. Therefore, the machine learning model is also increasingly mentioned by the industry for system log analysis. But in most cases, the system logs will report really few "serious" logs of the system, which are the programmers most concerned about. However, since most machine learning models used for system log analysis are assumed to train on balanced data, these models are prone to overfitting when they do syslog warnings, so that the results are not ideal enough. This paper will explore the application capabilities of CNN-text (CT) in system log analysis from the perspective of deep learning. By comparing CT with the mainstream system log analysis machine learning model Support Vector Machine and Decision Tree, we will explore the superiority of CT, comparing with these algorithms; we will compare CT with CRT, analyzes the treatment of CT features, and verifies the superiority of CT in processing deep-learning models to process syslog class texts; finally applies all models to two different log class texts. Contract the data to prove the universality of CT. In the experiments comparing CT with the mainstream machine learning model of log analysis, the recall rate of CT compared with the optimal model has increased by nearly 15%. In experiments comparing CT with CRT model, CT is more advanced than The accuracy of CRT model is about 20% higher, the recall rate is about 80% higher, and the precision rate is about 60% higher; in the universal experiment of CT, various types of models are integrated into the experimental dataset of this paper, logstash and public. In the data set WC85_1, the recall rate of CT was higher than that of the model with the highest remaining recall (DT-Bi) by nearly 14% when the accuracy rate was 100% with other models with better performance. According to the results above, the ability to abstract feature sets from plots of texts and the ability of non-linear regression are better than mainstream system log analysis machine learning model. Meanwhile, comparing to CNN-RNN-text, which is also a sort of CNN model and pays too much attention to the sequential features of the texts of systematical logs, CNN-text concentrates less on that. This difference, however, makes CNN-text maintains much better performance than that of CNN-RNN-text. Finally, it is argued that CNN-text is the best method among the methods mentioned in this paper.

Key words System Log Analysis; System Abnormal Warning; Unbalanced Data; Machine Learning; Deep Learning; CNN-text;

1 引言

当前，数据挖掘作为一种高时效性、高真实性的分析方法，正在社会中扮演着越发重要的角色。而基于文本挖掘的预警系统（EWS）也在很多领域发挥着重要作用，例如市场风险预警分析[1]、水利工程移民预警[2]、健康状况预警[3]，甚至板块漂移带来的地震预警[4]。另一方面，信息系统的日志数据异常预警的研究目前还处于起步阶段，绝大多数的系统运营过程中的预警都还在凭借经验性的知识支持运维人员对系统进行的维护[5]，而借助统计学[6]或一般主流机器学习[7]的方法进行判别的相关研究尚处于起步阶段，主要工作集中于针对系统的关键词挖掘、借助业务经验以及统计学的方法进行判别与预测，大多情况下还是只考虑关键词汇相关的规则对日志所属类别的影响，而对于词汇间的关系及权重、日志信息语法等考虑较少。[8]

随着大型分布式系统（例如：Apache Hadoop[1]、Spark[2]等）对软件系统市场的占领以及大型超级计算机的兴起，每天这些计算机会产生着数以万计的

[1] http://hadoop.apache.org/
[2] http://spark.apache.org/

日志条目，这些日志信息条目通过对系统运行状况的监测，为大量的互联网业务提供服务。一旦系统产生了类似资源占满、数据冲突等问题未能及时解决，将很容易导致应用层的软件崩溃，造成巨大的损失[9]。因此，针对系统日志进行分析并进行异常判别的时效性和高效性，在互联网行业中扮演着越来越重要的角色。

异常判别，主要指针对正运行系统的运行状况进行监控，采取特定词汇检索或凭借知识经验积累等一系列方式对杂乱、繁多、关联复杂的系统运行状态进行分析，并对系统是否运行正常进行判别的业务，这对软件系统的运维人员来说是一项重要的工作。[8-11]而基于系统日志的系统异常判别（Log-based Anomaly Detection, LAD）是指随着操作系统的运行，系统及时反馈自身运行的状况及过程信息，并将这些信息可以作为系统异常判定的主要数据依据。[6,12] LAD 的相关方法现已逐步受到学界和工业界的重视。随着更多大规模分布式系统的引入，运维人员及程序员已很难再通过以往的肉眼监测或者简单的关键词检索就能够马上发现系统中所出现的问题。而通过肉眼监测或者简单的关键词检索发现问题又需要消耗大量的学习成本，因此如何根据系统生成的日志让系统对自身进行监控在当下行业发展中是非常必要且重要的，而机器学习就是解决这类问题的重要方法之一。[9]机器学习通过一定的规则将程序员的业务抽象为数学模型，通过适当的特征工程以及参数调整，经过一定数据量的训练后，可在远超人类效率情况下得到接近人类识别的效果。[13]

但是根据大量的机器学习训练经验，类似 SVM、决策树等传统机器学习在训练时需要将大量时间用在特征工程上，这需要占用大部分的实验及测试时间，同时在抽取特征过程中，也需要相关人员的行业知识和经验的积累，这样抽取到的特征才能更容易反映出分类的逻辑，也有助于机器学习更准确地对日志信息进行判别。[14]为减少特征调整带来的人力成本、资金成本的损失，近年关于由传统机器学习演化出来的深度学习方面的研究也进展的如火如荼。[15]一方面基于系统运行产生的数据量通常是足够深度学习模型进行学习的，另一方面，深度学习将主要目光集中在调参上能从实质上降低所需投入的人力，提高工作效率。[16,17]

近年来随着大数据技术的发展、机器学习相关行业的兴起，深度学习的模型获得了很大的进步，如可聚焦样本局部特征的卷积神经网络(Convolutional Neural Network, CNN)[17]、可发现时间序列特征的循环神经网络(Recurrent Neural Network, RNN)[18]、改善了 RNN 在长序列预测中发生遗忘问题的长短记忆网络(Long Short-Term Memory, LSTM)[19]以及将这些深度学习算法结合起来使用的其他算法，如同时发掘文本局部特征及序列特征的卷积循环神经网络(C-LSTM)[20]、同时考虑正反序序列特征的双向循环神经网络(Bi-LSTM)[21]。

这些深度学习模型已被广泛应用于很多领域，如文本挖掘领域，通过 CNN 改进的 Tb-CNN 模型可用于情感分析,并借助多层 CNN 的特性挖掘语法特征及逻辑结构特征，再基于这些特征针对语料所反应的信息进行多分类学习及预测[22];金融领域，通过 EPCNNs 将深度神经网络（MLP）和遗传算法结合，进行股价的预测[23]；物流领域，将 CNN 用于物联网 APP 中，通过 CS-CNN 对 CNN 进行优化，对新上线的物联网 APP 相关数据这类不足量样本数据进行学习及分类以提高准确率[24]。然而目前深度学习在大规模分布式系统日志的异常判别中的应用还处于起步阶段。

而本文将从深度学习角度出发，探究深度学习中的 CT 在系统日志分析方面的应用能力，通过将 CT 与主流的系统日志分析模型 SVM、决策树对比，探究 CT 相对这些算法的优越性；之后将 CT 与 CRT 进行对比，分析 CT 对特征的处理方式；最后将所有模型应用至两套不同的日志类文本数据中进行对比，证明 CT 的普适性。综上，本文在系统异常检测的相关领域作出以下三点创新点：

创新点一：首次对**百万量级**的系统日志数据应用多种方法进行异常检测并进行性能比较；

创新点二：首次将 **CNN 网络**用于**系统日志分析异常检测**领域；

创新点三：对 CNN 神经网络卷积核形状进行半定长调整，同时调整滑动窗口的滑动方向，使之适用于**应用到系统日志文本分析**中，进行系统异常的判别。[25]

2 相关工作

随着信息技术的不断发展、大型分布式系统（Large scale distributed operating system, LSDOS）的广泛引入，基于机器学习的系统日志分析方法逐渐受到学界和业界的关注和重视，相关模型及算法的进步也促进着整个运营人员的业务能力不断提高。

2.1 关于日志分析方法的研究

Haoyu Chen 等学者证实了基于原始日志分析对系统进行维护效果是优于其他方法的，[9]传统通过日志对操作系统运行情况严重程度的判别，主要将目光集中于独立系统（standalone systems），独立系统的分析数据量较小、系统复杂度较低，因而分析方法主要集中于基于知识经验的分析以及关键词的匹配。[25]而随着系统规模不断扩大，将这些技巧结合起来形成新的方法的研究也层出不穷。Shun-Fa Yang 等学者将 hadoop 的 mapreduce 方法同传统甄别方法结合，提高了系统错误判别的效率[26]，但是并未将注意力集中到判别效果中去；Jorge Herrerias 等学者通过提取系统反馈的输入数据种类、签名、数据存在检验、时间戳、ip 地址以及登录文件等数据，借助一系列规则的设定以定位当前登入系统的数据是否安全[27]，但这些规则主要还是基于数据数值及部分正则表达式完成，即需要相关人员的扎实行业知识和深厚经验的积累；Senthil Mani 等人在前人的基础上测试了多种有监督和无监督模型在识别编译器 debug 模块信息的效果，发现一般情况下有监督模型效果会好很多，但是若给无监督模型加上去噪机制，无监督模型运行效果有可能超越有监督模型，但是无论何种情况，召回率、查准率和准确率都较低（40%-60%），数据量太小或所使用的模型不合适都有可能造成这样的问题[28]；Sarah Rastkar 等学者对 Eclipse, Gnome, Mozilla 和 KDE 等编译器的 debug 模块进行人工标注，并进行分析，基于统计学理论对 bug report 进行分类[29]，初步使用了具有一定的智能方法（EC,EMC,BRC），但是工作量巨大，并且没有验证方法的可扩展性。

2.2 可用于日志分析的机器学习方法研究

当今随着大规模分布式系统的引入，每天由这些系统产生的日志数据量数以万计，并且这些日志信息在经过日志整合软件（如：logstash 等）处理后不仅包含了底层操作系统，还包含大量数据层、应用层软件的日志数据[30]，因而即使是简单的判断系统运行是否正常的业务，对于过于混乱、复杂、广泛且弱联系的数据来说传统的基于经验及关键词检索的方法来说已不在适用。因此，通过机器学习模型来抓取数据的重要特征，并对日志信息反应的信息所属类型进行预测，可在一定程度上解决这一问题。

而随着机器学习相关研究的发展，各类模型的应用也逐渐在系统异常检测这一领域扩展开来。Wei Xu 等学者首次将决策树引入异常检测的过程中，使得最终预测及检测结果根据有可解释性[31]；Fei Wu 等学者通过在异常事件的概率图模型上进行蒙特卡洛马尔科夫链运算获取最小能量概率图模型，以定位各个事件间的关系[32]；Min Du 等学者基于 LSTM 设计了 DeepLog 模型，该模型主要用于日志事件发生序列的预测，日志事件区分的方式是通过提取常量字符串对日志类型进行分类并进行预测[31,33]；Fan Jing Meng 等学者基于 DBSCAN 的方法对大型 PaaS 云的部件状态进行降维，并基于设置时长门槛的方法对系统异常进行判别，该方法能够以较快的速度以较高的准确度判断系统的问题部件，但是准确度主要依赖于时间门槛的设置，即依然非常依赖从业人员的相关知识及经验[30]； Shilin He 等学者引入了多种有监督和无监督的模型进行对比实验，通过实验证明了 SVM、决策树在分析过程中的结果较优[25]；Ariel Rabkin 等学者通过图谱的方式对系统日志关系进行识别，并展示出可视化界面，给予程序员以清晰地分析思路[34]，但是以上这些研究方法或在特征工程这个步骤都需要付出大量的人力工作，或并未对日志的特征进行深入的分析，使得最终得到的模型在预测过程中非常依赖于系统历史的日志，因而很难在同类型新系统上进行推广。并且往往准确率提高的过程复杂而缓慢。

针对以上问题，学界采取了很多方法，其中就有深度学习这一类的方法，通过调整深度学习框架及相关参数以达到几乎不对特征进行人工提取就获得比较好结果的效果。

深度学习从感知器在 1957 年被美国计算机科学家罗森布拉特[35]提出后，经过了几十年的艰难发展，于近十年左右在各个领域获得了大量的突破。近些年，卷积神经网络逐渐被计算语言学学者所利用。起初，RNN 以其特有的序列特征而在文本分类中起到了重要作用。Garen Arevian 等学者通过多层 RNN 神经网络对文本进行分类，这样的方法能有效地减少噪声对模型表现的干扰。[35]Cheng J 等学者将基于 RNN 改进的 GRNN 同传统的 bag-of-words 方法对比，证实了将文本序列顺序纳入特征范围内的 GRNN 对微博的情感预测效果比传统的基于 bag-of-words 的方法正确率平均提高了 4%左右[36]。随着 CNN 在图像模式识别领域近几年的快速发展及技术突破，CNN 也逐步用到文本模式识别领域中。当前已经有学者提出借助 CNN 神经网络的卷积层提取文本中的不同位置的不同特征，调整相应词向

量所赋予的权重，以进行分类器训练，最终获取相应的 CNN 模型[36,37]，但是这段时间提出的 CNN 模型起到的主要是一种探索功能，即搭建较为简单的模型，独立地探索文本各方面特征（句长特征、序列特征等）在模型中起到的作用。Wenpeng Yin 等学者通过对比证明了在短句(平均句长小于 20 词）文本分类中，CNN 比 RNN 拥有更优良的表现；[38]SHIN J 等学者将 CNN 引用至语境挖掘中，将序列不同时间段内 CNN 抓取到的特征化为语境，所得到的预测结果正确率比一般的 CNN 模型和传统机器模型高 1%左右[39]；Siwei Lai 等学者搭建了 R-CNN 神经网络模型，借助 RNN 的结果受序列先后性影响的特性将词向量嵌入到语境向量内，再借助 CNN 神经网络提取语境向量的特征以进行文本分类，这种分类方式将语境考虑到实际场景中来，因而实际上更适合较长的自然语言文本[40]；HE S 等学者设计了基于 CNN-text(CT)的文本分类神经网络，而 CT 对重要特征及其临近词汇的聚焦很适合于系统日志这类由变量字符串及常量字符串组成的文本[41]。综合以上各个学者的研究成果及观点，再观察本文所处理的语料特点，考虑到本文语料涉及到的系统日志文档大部分是长度为中型或短型的文档，本文的主要模型将主要使用 CNN 簇的深度学习模型。

本文引入了深度学习领域的 CT 模型[41]对系统日志数据进行分析，并对系统运行状况的严重程度进行判别。在和当前用于日志分析的主流的用于系统日志分析的机器学习模型（SVM, DT）[42]进行对比测试中，该模型对日志判别的准确率、召回率和查准率分别提高至 90%以上，相对主流日志分析的机器学习平均提高了近 12%，可见用于系统日志分析的 CT 模型表现优于大部分用于日志分析的主流机器学习模型；在和近期受到较大关注的 CRT 对比中，性能上大大超越了考虑了序列问题的 CRT，准确率、召回率和查准率平均分别高出近 22%，可见在面对系统日志这类文本时，重点考虑重要特征及邻近词汇的 CT 更有优势；将每个模型引入 WC85_1 数据集中进行测试，同样 CT 表现要优于其他各个模型，在大部分模型过拟合的情况下，CT 保持了较为稳定的准确率召回率和查准率，召回率和查准率较表现最好的模型（DT-Bi）平均高出 12.5%。

3 用于系统日志分析的机器学习模型

本文将针对以上相关工作对现有系统日志信息进行分析，通过引入对应的机器学习模型对日志的 message 和日志信息所反映的系统运行严重程度进行学习，以达到对系统所信息反映的系统运行严重程度进行及时监控的目的。

为证实前文提到的创新点，同时研究与其他主流机器模型、深度学习模型相比，CT 模型在分析百万量级系统日志异常检测过程中所起到的作用，本文将两类系统日志数据嵌入多种主流机器学习模型以及文本挖掘模型（SVM、决策树、CRT、CT）中进行对比实验，根据不同的数据结果对应的 CT 模型的特性进行解释以证明 CT 在类似系统日志的文本分析上各方面的优势，相关的实验流程如图 1 所示。

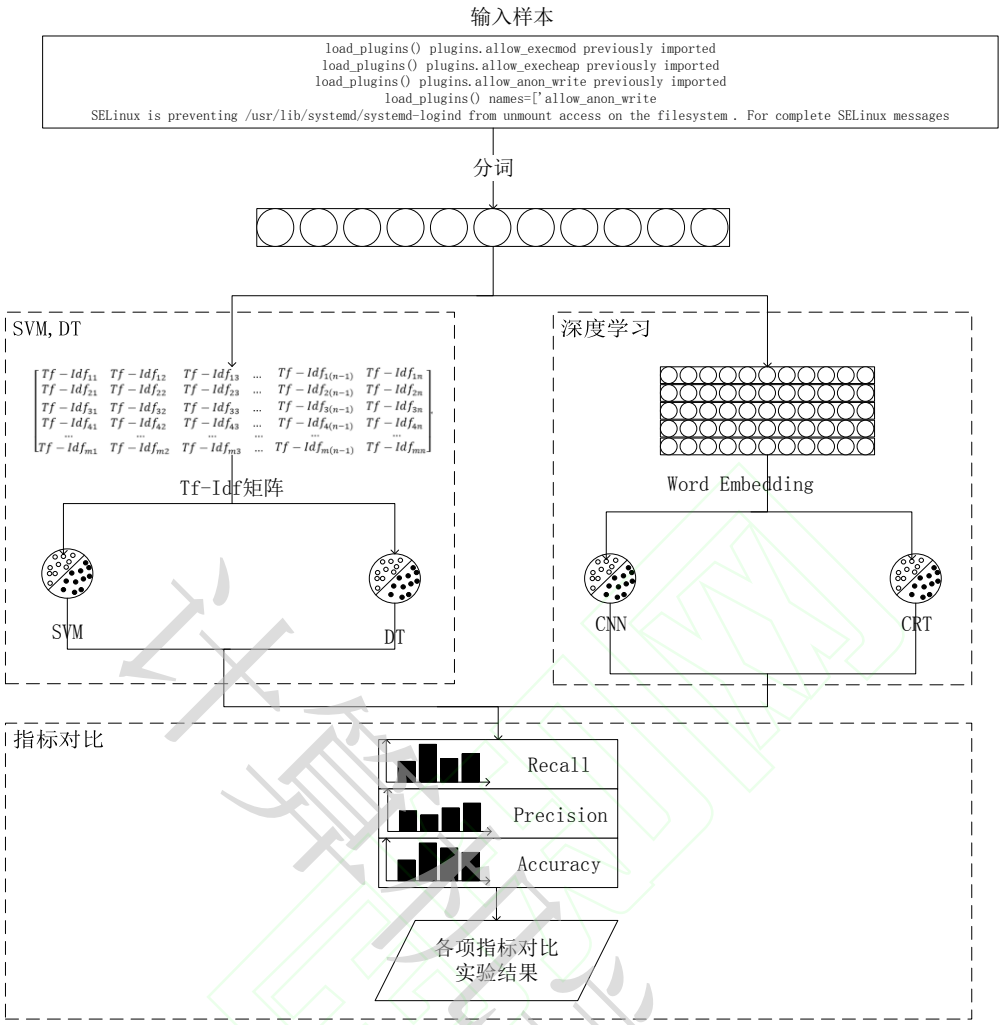

图1 实验整体流程图

3.1 深度学习

传统主流的用于系统日志分析的机器学习模型所需要的大规模稀疏矩阵（Tf-Idf 矩阵等）大小与传统机器学习的样本量和词汇量成正相关，即：

$$Size_{matrix\ 1} \propto m * n \ (1)$$

式中，m 代表样本量，n 代表词汇量，也就是说，词汇每增加一个，矩阵曾佳 m 条数据，样本每增加一条，矩阵曾佳 n 条数据，而一般文本的词汇量都是上百万级别的，样本量根据情况也非常之大，这样的空间开销不仅不值，同时稀疏矩阵对模型的训练也会产生不好的影响。而相对于传统机器学习所需要的大规模稀疏矩阵，深度学习相对消耗的空间要小很多,若词汇使用的是 rnn-word-embedding 方法进行嵌入，则输入模型的张量大小为：

$$Size_{\text{matrix }2} \propto m * l * d \ (2)$$

其中 m 为样本量、l 为最长句子长度、d 为词嵌入维度，也就是说模型的训练与词汇量再无关系，而由于以上公式中的 l 和 d 为超参数，因而式 3.13 还可化简为

$$Size_{matrix\ 2} \propto m \ (3)$$

即，输入模型的数据大小仅与样本量大小呈正相关。因而深度学习的空间消耗增长较少，同时为了解决前一小节所述的传统机器学习矩阵运算量过大的问题。[43]

3.2 基于 CNN-RNN-text 进行分类

CNN-RNN-text（CRT）是将 CNN 与 RNN 结合起来对文本进行分析的一种新的深度学习模型，该模型通过 CNN 抽取文本的重要特征并将这些特征存储于池化层中，再通过 RNN 对池化层进行学习及预测以得到最终结果，通过 CNN 将长文本转化为短文本，提高了文本的重要特征影响力同时考虑了文本序列的影响。同时 CNN 的卷积和池化作用也剪短了序列长度，有效帮助 RNN 避免了过长序列导致的遗忘效应（梯度消失）。

CRT 的基本原理图如图 2 所示：

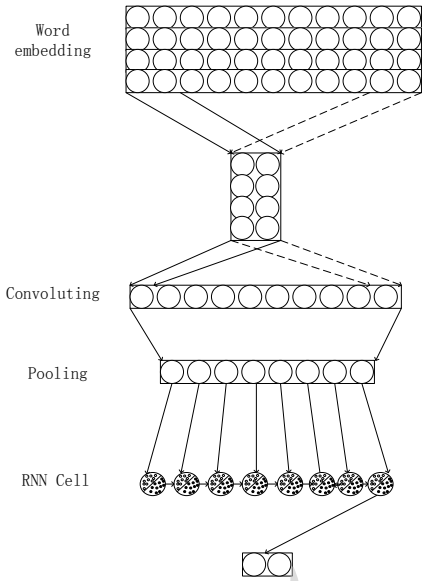

图2 CRT 基本原理图

但是在针对系统日志文件处理中，由于日志信息最开始创造出来时是由人为输入了一定的语意构造的，主要由常量型字符串和变量型字符串组成，如图 3 所示：

图3 logstash 提取的系统日志文件（节选）

图中被矩形框住的是常量字符串（人为输入的文本，同一类异常不随异常具体内容变化而变化），未被框住的是变量字符串（人为输入的变量，异常会给变量赋予不同的值，因而会随异常具体内容变化而变化），而识别问题错误与否主要受常量字符串影响，次要受变量字符串影响，因此，语序对分类效果不会造成太大影响，若过于关注语序特征甚至有可能导致分类结果受到影响。

3.2.1 基于 CNN-text 进行分类

CNN-text（CT）的主要特点是将 CNN 应用到文本分类中，将分类器目光集中于文本的主要特征，以进行分类，同图像卷积神经网络结构不同在于，CT 对文本的处理方式，即，在本文所涉及的数据中，CT 处理方式是将 embedding 后的特征矩阵看做一张等大的图像，如词嵌入维度为 100 层，句子最大长度为 1003 个单位，那么 CT 就将该文本视为 1003*100 的图像，相应的卷积核就为 n*100 的张量。

CT 的基本原理图如图 4 所示：

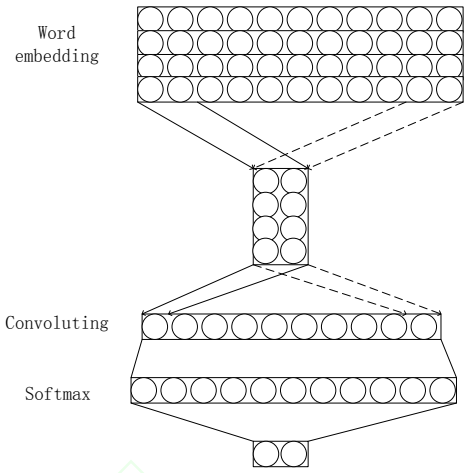

图4 CT 基本原理图

仅把卷积核的宽设为常数的这种方式被称为**半定长调整**。由于词嵌入时整个词向量被视为一个整体，即矩阵的每一列表示一个特征，在经过卷积核的一维滑动后特征可以被明显地提取出来，而图片则需要通过二维的滑动才能确定好特征矩阵，因此这种半定长的调整方式使得卷积核的滑动方向适应于文本的特征提取，不会像传统 CNN 一般扰乱语言特征发现过程，以减少 CNN 对特征的发现难度。[25]同时由于 CT 是一种单次遍历的过程，因此相对于常用 CNN 模型的时间复杂度，CT 的时间复杂度降低为：

$$O\left(\frac{m}{n}\right) \quad (4)$$

其中 m 为常规 CNN 模型的卷积核遍历步数，n 为卷积核跳跃次数。

这种神经网络将关注点集中于文本的重要特征以及临近词间的关系，几乎不关注序列特征的关系，而系统日志文件中的语句所反映的问题主要由变量字符串决定，因此，CT 理应适合应对这类更为注重词汇特征的文本数据。

3.2.2 CNN-text 在对系统日志文本类型进行判断时与 CRT 过程上的对比

根据以上对 CNN-text 和 CRT 的描述可知，CRT 在对文本进行预测时对文本序列顺序的关注度远高于 CNN-text 的关注度。

在训练过程中，处理每一条文本时，CRT 会存在如下过程：

过程 1 CRT 处理每一条文本时的过程

第一步：滑动卷积核提取重要特征集合 F1，并通过线性函数 Con1(F1)将卷积核映射到特征向量 Fv1 上；

第二步：对 Fv1 进行最大池化获取池化向量 Fpv1；

第三步：依次将 Fpv1 的元素 *FpvItem1* 带入 RNN1 中，并根据结果优劣，更新 RNN1 神经元的参数：

FOR *FpvItem1* IN *Fpv1*:

RNN1=RNN1(FpvItem1)

而相应的，CNN-text 的处理方式如下过程所示：

过程 2 CNN 处理每一条文本时的过程

第一步：滑动卷积核提取重要特征集合 F2，并通过线性函数 Con2(F2)将卷积核映射到特征向量 Fv2 上；

第二步：对 Fv2 进行最大池化获取池化向量 Fpv2；

由以上流程对比可知，CRT 和 CNN-Text 都会使用 CNN 的卷积核遍历文本向量，并且通过特征向量及池化向量来提取相应的权重。二者差别在于是否将池化向量带入 RNN 中。

根据卷积神经网络原理，池化层以前各个向量一直是以一种线性关系进行变换，甚至于说池化层也同样与重要特征成线性关系（尤其采取最大池化方法的池化层）。因此可知

$$Fpv \propto f_{max} \quad (5)$$

其中，f_{max} 表示权重最大的特征，因此 CNN 将大量的注意力集中于对重要特征进行处理的工作上。

而在将池化特征放入 RNN 后，由于 RNN 每一次转移后的输出结果都受上一次转移结果的影响，因而在 RNN 内进行了最后一次转移后输出的结果与f_{max} 的关系就变为

$$\text{result}_{RNN} = \sigma_{RNNSingle}\left(a * \text{result}_{T-1} + b * input_T\right) \quad (6)$$

其中result_{RNN}表示 RNN 最终的输出结果，T 表示在 RNN 内传递的次数，result_{T-1}表示倒数第二次 RNN 输出的结果，input 表示最后一次输入的特征，a, b 分别为参数$\sigma_{RNNSingle}$ 表示 RNN 的单次处理过程。将result_{T-1}无限展开可得到

$$\text{result}_{RNN} = \sigma_{RNN}\left(\sigma_{RNNSingle}\left(\sigma_{RNNSingle}\left(\ldots\left(f_{max}\right)\ldots\right)\right)\right) \quad (7)$$

可见，在输出最终结果前f_{max} 被带入了多个非线性函数中，这个过程中尽管语言先后顺序得到了关注而重要特征却被忽略了。以报错日志文本“exiting.,daemon,dhclient[24820]”为例，该文本仅出现了三个单词，其意义表示 dhclient 指令被重复输入的错误，但是从自然语言理解角度却很难看出这其中意义，因而语序在这其中发挥的作用不会很大，同时根据统计，样本的语句长度最长为 94 个单词，且句子长度分布如图 5 所示：

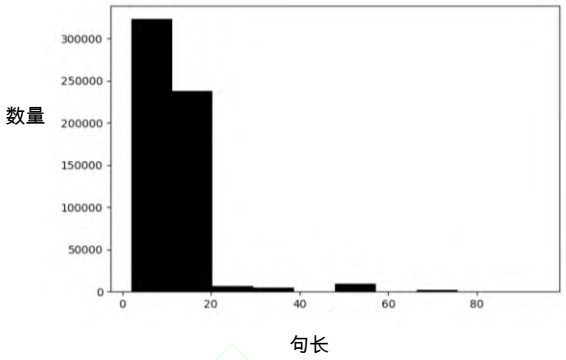

图5 句子长度分布图

大部分句子长度都在 20 个单词以内，其中以 5 个单词见多，同时长句内容往往以罗列相关部件及 API 为主，因而很难训练出合格的 RNN 簇模型，而再结合前节所提到的，日志文本大多由常量字符和变量字符组成，因而各个单词之间独立性较强且各个文本句式较为单一，强行在其中寻找先后次序反而容易扰乱重要特征所占有的权重，在不均衡数据下，大部分句式都是非问题文本的句式，易导致模型的欠拟合[44]，最终使结果不如意。

相反，CT 仅关注文本中的个别重要特征及重要词汇，减少对特征间关系的关注，一切分类结果都与特征选取呈较大关联。即 CT 仅关注特征的有或无，在不考虑序列关系时候，决定各个类别的特征向量形成的子矩阵之间存在接近正交的关系，对存在正交关系的特征进行训练，模型的效果受样本量影响非常小，因而 CT 模型可以一定程度上解决样本分布不均的问题。因此 CT 在处理这类文本的时候能够有效避免数据不均衡带来的影响，而抓住日志内容的本质，从而拥有更好的训练及预测效果。

3.3 SVM 和决策树

本文所做实验中用作对比的传统机器学习模型包括 SVM 和决策树两种。根据文献[6,7,26]进行的对比实验结果，SVM 和决策树算是系统日志分析的主流模型，在不同的系统日志分析的细分场合发挥的作用效果不同。[33]

3.3.1 数据输入模型

格式一：Tf-Idf 模型矩阵

是一种基于概率和信息熵原理进行文本特征筛选的方法，在大量的文档分类过程中都起到了较为明显的提升作用。[34,40,42]其核心公式如下：

$$Tf - Idf = \frac{n_{w_i s_j}}{\sum_{w_k \in s_j} n_{w_k s_j}} \ln\left(\frac{\sum_{s_r \in c} n_{s_r}}{\sum n_{s_m w_i}}\right) \quad (8)$$

式中，$n_{w_i s_j}$表示文档s_j中单词w_j的个数，w_k表

示文档s_j中所包含的单词，$\sum_{w_k \in s_j} n_{w_k s_j}$表示文档$s_j$中的单词总数，因此，$\frac{n_{w_i s_j}}{\sum_{w_k \in s_j} n_{w_k s_j}}$表示的是检索词频率（term frequency, Tf）。$n_{s_m w_i}$表示的是包含单词w_i的文档在语料 c 中的数目，$\sum_{s_r \in c} n_{s_r}$表示语料 c 中文档的总数，则$ln(\frac{\sum_{s_r \in c} n_{s_r}}{\sum n_{s_m w_i}})$表示对数逆文档频率（inverse document frequency, Idf）。将检索词频率和对数逆文档频率相乘即可得到。[45]

根据该公式我们可以看出，针对每个文本，某一单词出现的频率越高，而这个单词在其他文本中出现的频率越低则这个单词的 Tf-Idf 值越高，即这个单词对这个文本来说越重要。因此面对类似 SVM、决策树等传统机器学习模型，可通过将文本转化为 Tf-Idf 模型矩阵，将矩阵输入到这些模型中进行学习及预测。[46]

格式二：Bi 模型矩阵

Bi 矩阵是一种 0-1 矩阵，矩阵中每一行表示一个文档，每一列表示每个词汇，每个单元的数值表示这个单元所对应的词汇是否在这个单元所对应的文档中。因此矩阵中每个单元的数值如下：

$$BiItem_{ij} = \begin{cases} 1 \; if \; w_j \in s_i \\ 0 \; if \; w_j \notin s_i \end{cases} (9)$$

其中，$BiItem_{ij}$表示 Bi 模型矩阵的第 i 行第 j 列元素，w_j表示第 j 个词汇。

显然，当文档中每个单词存在与否的区分度很大时，Bi 矩阵的效果会优于 $Tf\text{-}Idf$ 矩阵。

格式三：词嵌入模型张量

词嵌入（word embedding）是应用于文本挖掘的深度学习常用的输入格式，词嵌入得到的词向量形式多为[词维度,次数]。[39]

通过将 one-hot embedding 的词向量同随机生成的矩阵相乘，获得的内积就是经过 word embedding 后的词嵌入向量。

3.3.2 基于 SVM 模型进行分类

支持向量机（Support Vector Machine, SVM）是一种有监督的分类器，通过建立超平面对数据进行分类。借助超平面分类的过程可以看做一个优化问题，即使两个类别中距离最近的两个点距离超平面距离最远，而借助这个超平面，由超平面分割出来的两个类就得以分辨出来了。

以文本二分类为例，设$x1$和$x2$为样本特征，γ表示每个分类的边界超平面 a、b 间距，直线 c 为 svm 的分类超平面，a、b 为两个类别之间的边界超平面，设超平面 a、b、c 公式分别为：

$$a: \omega^T x_i + \varepsilon = +1 \; (10)$$

$$b: \omega^T x_i + \varepsilon = -1 \; (11)$$

$$c: \quad \omega^T x_i + \varepsilon = 0 \; (12)$$

其中，ω^T为系数项、ε为超平面偏置。由图可知，SVM 的优化问题实质上为：

$$\begin{cases} \max(\gamma) \\ s.t. \, y_i(\omega_i x_i + \varepsilon) \geq +1 \; i\epsilon 1,2,\dots,n \end{cases} (13)$$

而求取 SVM 模型的核心就是求取分类的超平面，这个超平面满足到两个类的距离最大且处于两个类最相近的样本之间。

综上所述，x_i的数量决定了规划矩阵的大小，同时x_i的值本身也会对SVM超平面的系数和偏置向量造成影响，也对 SVM 是一种对特征非常敏感的模型，因此特征工程对 SVM 来说非常重要。

3.3.3 基于决策树进行分类

决策树（Decision Tree, DT）的原理是基于数据的属性进行多层分支，形成树状数据结构，不同的属性独占一层，根据不同属性在树上的组合结果进行分类，决策树选用的属性排序标准有较多种，如信息增益、gini 系数等，一般是按照这些标准从大到小进行排列，即，以信息增益为例，属性的信息增益越大，则该属性在分类过程中越重要。

信息增益的计算方式如下：

$$E_y = -\sum_{i=1}^{l} P_{y_i} ln(P_{y_i}) \; (14)$$

$$E_{event_j} = -\sum_{i=1}^{n} P_{event_{j y_i}} \; ln\left(P_{event_{j y_i}}\right) (15)$$

$$IG = E_y - \sum_{j}^{m} P_{event_j} E_{event_j} \; (16)$$

上例中，E_y表示样本整体熵，y表示样本所属的类别的整体集合，l表示样本所属类别的总量，y_i表示第i类，P_{y_i}表示第i类出现的概率，E_{event_j}表示属性$event_j$的个体熵，实验中每个$event_j$代表每个词汇，$event_{j_y_i}$表示属性$event_j$中的第i类，$P_{event_{j_y_i}}$表示在属性$event_j$中个体所属类别为$event_{j_y_i}$的概率，IG 表示信息增益，信息增益是整体熵与个体熵的加权均值的差，即个体熵越大，混乱度越高，信息增益越小，越不容易被排到前面进行对比，反之越容易被排到前面进行对比。

综上所述，由于词汇数据量过大导致生成的 Tf-Idf 矩阵及 *Bi* 矩阵是一种大规模稀疏矩阵，导致大部分的$P_{event_{j_{y_i}}}$差别不大，以至于非关键特征覆盖关键特征，导致特征之间重要程度难以区分，决策树过拟合的现象发生。

4 实验结果及分析[3]

本文主要展示 CT 模型与其他模型得到的实验结果。我们首先介绍了从实验室 COS 采集的基于 logstash 整理出来的与底层、数据层和应用层相关的数据集和相关的评价指标和评价方法;然后基于这些指标和方法设计了对比实验探究与各个模型相比，CT 模型在学习和预测过程中所表现的特点和优越性;最后，我们选择了 WC85_1 的数据使用相同的实验过程进行对比实验获取相应指标，以确定基于 CT 的系统日志分析方法的普适性。

4. 1 数据集及描述性分析

本文的原始数据集取自 logstash 提取的来自实验室的 COS 底层、数据层和应用层的相关日志信息数据，从中截取了一段时间内的 600000 条数据，这些数据共涉及 11 台主机，涵盖 authpriv、cron、daemon、kern、local、mail、syslog、user 等层次的运行日志。以“反映系统运行状态严重”的日志信息标注为正样本，以“反映系统运行状态正常”的日志信息标注为负样本，则正负样本分别为 10693 条、589307 条，是一个典型的不均衡样本分类问题。样本内容覆盖实验室 COS 的各个层面，可基本反应整个系统运行过程中的大部分信息。除去这些日志的无关变量，如：主机名、事件哈希码等，留下日志文本信息 message，去除 message 的重复项后依然剩余 600000 条，对 message 中的字母进行小化处理，之后进行分词，得到词汇量 27444 个。另一方面，将这些 message 做序列化处理，获得句子最大长度为 1253，将小于这个长度的句子用 0 补齐。

本文选择的证实本文方法普适性的数据为 WC85_1，这组数据是 1985 年 1 月世界杯官网受世界网民访问日志的公开数据集，该数据集记录了用户登录网站的用户临时 ID、时间、登录方式、登录界面、登录协议、登录状态代码以及登录状态详情代码。提取重要信息（登录方式、登录界面、登录协议及登录状态代码），删除次要信息（用户临时 ID、时间、登录状态详情代码）并删除空缺数据和重复数据后，该数据留下共 1905106 条有效数据从中同样随机取样 600000 条进行对比实验，后续的预处理方式与针对 logstash 的处理方式类似，WC85_1 的数据大致情况如图 6 所示，可见该数据同样具有常量字符串和变量字符串两个部分。

```
"""GET /images/icon_quotes.gif HTTP/1.0""",0
"""GET /images/hm_btm_arw02.gif HTTP/1.0""",0
"""GET / HTTP/1.0""",0
```

图6 WC85_1 数据大致情况（节选）

4. 2 评价指标和评价方法

本文使用的对比模型包括传统机器学习模型 SVM 和决策树，深度学习模型 CRT 以及 CT，使用的数据来自从云操作系统获取的系统日志信息。在对比的评价指标选择上，由于大量的不均衡样本训练研究发现，由于大部分模型在设计出来时都假设训练数据是均衡的，在不均衡数据集中，单纯的准确率无法正确地反应模型的训练学习能力[47]。例如，在正负样本比为 1:9 的数据中，若模型将所有的样本全部预测为负，那么模型的准确率依然可以保证在 90%，但往往我们更加关心的是那 10%的小样本数据。

考虑到以上问题，本文选用的评价指标包括准确率（Accuracy）、召回率（Recall）、查准率（Precision）以及 *F1* 值，这四个指标的计算公式如下：

$$Accuracy = \frac{T}{total} \quad (17)$$

$$Recall = \frac{TP}{TP + FN} \quad (18)$$

$$Precision = \frac{\mathrm{TP}}{TP + FP} \quad (19)$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision} \quad (20)$$

其中，T 为预测正确的样本数，*total* 为总的样本数，TP 为预测正确的正样本数，FN 为预测错误的负样本数，FP 为预测错误的正样本数。

根据准确率，我们可以直观了解到模型训练的准确性，而召回率和查全率则让我们能够查看模型是否处于过拟合状态，当 *Recall* 较低则反映了有较大量的正样本被模型预测为负样本，当 *Precision* 较低则反映了模型将较大量的负样本预测为正样本。而 $F1$ 值可综合反映二者的状态，$F1$ 值越接近 1，模型拟合效果越好，越接近 0，模型拟合效果越差。

[3] 相关实验代码已上传至:
https://github.com/Timaos123/LogstashAI

4.3 实验过程及结果分析

我们进行了三组实验来评估 CT 的各方面效果，以对 CT 相对主流机器学习模型、CRT 深度学习模型在系统日志异常判别方面的表现以及 CT 模型在这一方面使用的普适性进行检测。为减少数据偏移导致评估结果的差异，本文设定每个模型的训练集（training dataset）和开发集（validation dataset）比率分别为总样本的 60%和 40%。

4.3.1 CT 同主流机器学习模型

首先在做传统机器学习与深度学习的对比上，我们采取了 *Tf-Idf* 模型矩阵和 *Bi* 模型矩阵两种将文本转化为矩阵的方法，作为我们的对比方法。使用的数据是从实验室获取的云操作系统的日志数据。经过转化后的矩阵大小为 360000*27444。我们将 *Tf-Idf* 矩阵和数量矩阵输入到 SVM、决策树的模型中，将经过 RNN-embedding 后的词向量张量输入 CT 中进行训练和预测，CT 的相关参数设置如表 1 所示：

表1 CT 参数设置

词嵌入维度	100
卷积核数	1
卷积核大小	100*3
池化层类别	最大池化
池化窗口大小	100*4

如上表所示，我们选择从简单架构像复杂架构进行扩展，根据文本挖掘一般经验我们将词嵌入维度设定为 100 维；根据一般的词汇三元组的组成形式，将卷积核的宽设定为 3 个字符单位；池化方式选择最大池化以提高池化区域的重要特征对结果影响的权重；根据一般经验，池化窗口宽度选择为 4。另外各个层次张量的参数初始化方式均为随机初始化。

本文训练这些模型时采取 K 折交叉验证的方式进行训练，K 值取 3，最终得到的模型的训练集平均准确率、召回率、查准率如下表（表 2）所示。

表2 传统机器学习模型与 CT 训练时 K 折交叉验证对比

模型	K	准确率	召回率	查准率	F1
CNN	3	0.99	0.98	0.98	0.98
DT	3	0.99	0.89	0.74	0.81
SVM	3	0.98	0	0	\

将得到的模型带入开发集进行预测及评估，所得结果如表 3 所示。

表3 传统机器学习模型与 CT 对比实验结果数据

模型	输入	准确率	召回率	查准率	F1
CT	RNN-Embedding	0.92	**0.92**	**0.92**	**0.92**
SVM	Tf-idf 矩阵	0.98	0	0	\
	Bi 矩阵	0.97	0	0	\
决策树	Tf-idf 矩阵	**0.99**	0.73	1	0.84
	Bi 矩阵	**0.99**	0.69	1	0.82

根据以上表格数据可看出，SVM 的表现非常不理想，即，准确率接近 100%而召回率和查准率都为 0，恰好印证了本文在之前提到的，SVM 对特征过于敏感导致最终结果容易被大样本影响而出现过拟合的现象。

决策树的结果相对于 SVM 会好一些，无论准确率、召回率还是查准率都有大幅的提升，说明决策树在处理操作系统日志这类数据的过程中有着明显的优势，而这一结论与文献[26]的结论是一致的。但是，决策树在处理这类数据过程中所采用的，将特征完全分割开并依方法，导致特征排序变化敏感，也在一定程度上影响了召回率。

再看 CT，在针对这些数据的预测过程中表现优于其他两个模型，大部分指标在几个机器学习模型中表现数值最好。准确率同其他模型相近，召回率和查准率却高于其他模型的准确率最高的结果，即，未经过特征工程 CT 就得到了优于 SVM 和决策树的结果。也就印证了：相对于 SVM 和决策树，CT 能够减少更多的人为的特征工程及规则修订的工作，并取得较好的学习效果

4.3.2 CT 与 CRT 对比

另一方面在深度学习中，我们尝试了将 CRT 和 CT 进行实验对比，两个深度学习大部分参数相同，但是 CRT 比 CT 多了一层 RNN 模型。

经过对多次调参测试的结果进行分析，drop 率为 0.5，RNN 选用 LSTM 的 CRT 模型在训练集开发集数据量比为 6:4 时训练结果较好（如图 7 所示），即在训练集测试集样本量比率为 6:4 的情况下，模型准确率没有特别低，并且召回率和准确率都处于较高水平的情况，由于 CRT 的性能提升不明显，因而基本可判断其大致指标范围，同时考虑到训练 CRT 的时间成本较高因而选择调整其训练集合测试集比率而非直接带入交叉检验进行训练。

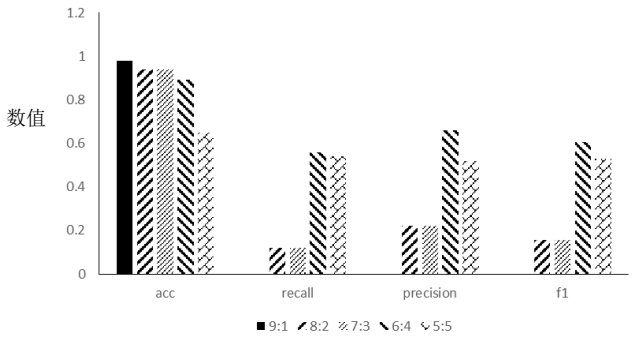

图7 不同训练集、测试集比例下的 CRT 训练结果数据

因此我们保持其他参数不变情况下，在这一过程中同样采用训练集：开发集为 6:4 的方式进行训练及评估，对比了 CNN 与不同 RNN 参数下的 CRT 的预测结果。所得实验结果数据如表 4 所示：

表4 CRT 与 CT 对比试验结果数据

模型	输入		准确率	召回率	查准率	F1
CT	RNN-Embedding		**0.92**	**0.92**	**0.92**	**0.92**
CRT	drop: 0.2	RNN: RNN	0.78	0.11	1	0.20
	drop: 0.2	RNN: LSTM	0.65	0.02	0.37	0.04
	drop: 0.5	RNN: RNN	0.82	0.51	0.37	0.43
	drop: 0.5	RNN: LSTM	**0.89**	**0.56**	**0.66**	**0.61**

可见，无论 CRT 以何种参数进行实验，所得到的各项指标都差于 CT，取 CRT 最好的一次实验（drop:0.5; RNN:RNN）结果为例，准确率 0.89，同 CT 结果相差不大，而召回率和查准率都在 0.6 左右，也就是说，该模型判断出的平均 10 个异常中有 4 个是错误的，而这段时间内发生的 10 个异常中平均只能找到其中的 6 个，而与之相对应的 CT 则保持着较好的结果（Precision:0.92, Recall:0.92）。究其原因，我们以 CRT 模型中运行效果最好的一次实验的训练过程的准确率变化为例，所得结果如图 8 所示。

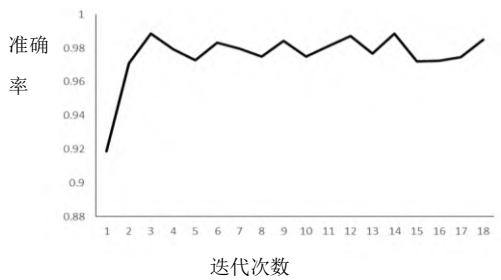

图8 训练集准确率变化图（部分）

可见，CRT 在训练的过程中迭代到第三次就开始发生过拟合现象，大量相似的语序特征出现覆盖了文本的重要特征及重要词汇相关联的关系特征，再由于大量的负样本的出现，正样本几乎无法识别，导致最终训练结果过拟合。

综上，本轮实验对比证明了：相对于把较多精力放在处理序列关系的 RNN 族的深度学习模型（如 CRT 等），将更多精力放在处理重要特征及相关临近词汇的 CT 在系统日志的文本上的处理能力展现出更多的优势。

4.3.3 证实 CT 的普适性

将 WC85_1 的数据导入 CT、SVM、决策树及 CRT 中，训练集与开发集的数量比均为 6:4，保证 CT 同 CRT 的公用参数相同，所得结果如表 5 所示:

表5 各个模型在 logstash 及 WC85_1 数据上对比试验结果数据

数据集	模型	矩阵	准确率	召回率	查准率
Logstash	CT	RNN-Embedding	0.92	**0.92**	**0.92**
	SVM	Tf-Idf	0.98	0	0
		Bi	0.97	0	0
	DT	Tf-Idf	**0.99**	0.73	1
		Bi	**0.99**	0.69	1
	CRT	RNN-Embedding	0.99	0	0
WC85_1	CT	RNN-Embedding	**0.99**	**0.67**	**1**
	SVM	Tf-Idf	**0.99**	0	0
		Bi	**0.99**	0	0
	DT	Tf-Idf	**0.99**	0.42	**1**
		Bi	**0.99**	0.59	**1**
	CRT	RNN-Embedding	**0.99**	0	0

可见，在两个数据集中，CT 都保持着较为优秀的表现（准确率、召回率和查准率都名列前茅）。因而从实验角度，可以说明：CT 相对于多个其他模型在百万量级的系统日志分析异常判别方面具有普适性的优势。

综合以上三个实验，我们可以看到 CT 在处理操作系统日志信息的时候，仅考虑日志信息中重要的特征词汇及这些特征词汇之间的关系，弥补了 SVM 和决策树仅考虑单个词汇对文本所属类别所做的缺陷，同时无需特征工程的特点也更加体现了

CT 的优势；由于系统日志信息大多是程序员预先定义信息格式，即程序员主要关注于日志信息的常量字符串部分该如何措辞，而变量特征部分则由系统自行识别生成，因而日志的语法及语序和自然语言相比可能会发生变化，不能用常规的语序去理解，而 CT 正好忽略了词汇之间的词序特征，这一特点让 CT 的表现超越了较为先进的 CRT。

5 结论与工作展望

在 CT 同日志分析的主流机器学习模型对比的实验中，CT 相较于最优模型的结果召回率提升了近 15%；在 CT 同 CRT 模型对比的实验中，CT 相较于更为先进的 CRT 模型准确率高出约 20%，召回率高出约 80%、查准率高出约 60%；在 CT 的普适性实验中各项指标 CT 同样名列前茅，在公开数据集中，在准确率同其他表现较优的模型同为 100%的情况下，CT 的召回率高出其余召回率最高的模型（DT-Bi）近 14%。

本文率先将 CT 引入百万量级系统日志分析领域，通过半定长的卷积核对文本特征进行抓取，突出重点词汇及其临近词间关系的特征，同时减少了运算的时间复杂度。

通过实验，本文从纵向对比了基于主流用于系统日志分析的机器学习方法和基于 CT 的系统日志分析方法，证明了 CT 相对于主流的做系统日志分析的机器学习算法在这一领域的优势；从横向对比了 CT 与最新发展出来的 CRT 在系统日志分析方面的性能，证明了 CT 相对于其他深度学习模型在这一领域的优势；最后本文通过引入公开数据集 WC85_1 的数据，证实了 CT 在系统日志这类由常量字符串和变量字符串组成的文本处理上优越性能的普适性。

可见，CT 模型在进行系统日志分析的过程中，具有少调参、高精度、高适应性的特点。能够在少量的人工工作参与下训练出的性能不亚于甚至于超过其他算法的模型。CT 将主要计算精力集中于文本中的重要词汇及其临近词汇间关系的特征，而大大忽略了这些词汇的词序特征，因此既不像 SVM 或决策树一般仅考虑单个词汇的特征，又不像 CRT 一样过度重视词序特征。

本文重点在于证实 CT 在系统日志分析中异常判别业务上的突出能力，在未来工作中，将从以下角度进行进一步的探索及挖掘。1）由于系统日志异常监控是一种典型的不平衡数据学习的业务，未来也会将精力投入到防止在不均衡数据中模型过拟合的现象；2）借助系统日志的时序特征同 CT 结合形成类 RCT 的深度学习模型，做到系统异常预警机制，以帮助程序员提前预知系统存在风险；3）对系统日志进行更加深入分析，通过最大熵或条件随机场等模型对 CT 发现的异常日志信息进行抽取和归档，以帮助程序员确定一个异常的异常信息簇，便于程序员更方便地找到系统的具体问题所在；4）尝试通过 CSDN 或 Stack Overflow 等获取系统日志异常的解决方法，并通过机器学习的方式为程序员生成可参考的解决方案，协助系统异常的修复工作。

致 谢 感谢面向云计算的网络化操作系统(No. 2016YFB1000505)、国家自然科学基金委员会(NSFC)-广东省人民政府联合基金超级计算科学应用研究专项计划(第二期)（U1611261)、宁夏自治区重点研发计划（引才专项）（2018BEB04002）资助，感谢《计算机学报》编辑部老师们的辛勤工作。

参 考 文 献

[1] *ACAR M, KARAHOCA A. Designing an early warning system for stock market crashes based on adaptive neuro fuzzy inference system forecasting//Proceedings of the 2nd international conference on risk analysis and crisis response, Beijing, China, 2009: 79-85.*

[2] *LOU Y, SHEN J, YUAN S. The development and application of hydraulic engineering migration risk early warning system based on data mining// Proceedings of the IEEE International Conference on Computer Communication and the Internet, Wuhan, China, 2016: 346-349.*

[3] *MAO Y, CHEN Y, HACKMANN G, et al. Medical data mining for early deterioration warning in general hospital wards// Proceedings of the IEEE International Conference on Data Mining Workshops, Vancouver, BC, Canada, 2012: 1042-1049.*

[4] *VENKATESAN M, THANGAVELU A, PRABHAVATHY P. An improved bayesian classification data mining method for early warning landslide susceptibility model using GIS. Advances in Intelligent Systems & Computing, 2013, 202: 277-288.*

[5] *DONG N D, IUHASZ G. Tuning logstash garbage collection for high throughput in a monitoring platform//Proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, 2017: 359-365.*

[6] *FU Q, LOU J G, WANG Y, et al. Execution anomaly detection in distributed systems through unstructured log analysis// International conference on Data Mining, Miami, USA, 2009: 149-158.*

[7] *NI Z, LI Q, GUO Y. Ensemble forecasting algorithm for anomaly*

detection onelectric-power big data log analysis platform. Journal of Nanjing University of Science and Technology 2017(5):634-645.

[8] *RAZAVI A, KONTOGIANNIS K. Pattern and policy driven log analysis for software monitoring// Proceedings of the IEEE International Computer Software and Applications Conference, Turku, Finland, 2008: 108-111.*

[9] *CHEN H, TU S, ZHAO C, et al. Provenance cloud security auditing system based on log analysis// Proceedings of the Online Analysis and Computing Science, Chongqing, China, 2016: 155-159.*

[10] *KRUEGEL C, VIGNA G. Anomaly detection of web-based attacks// Proceedings of the ACM Conference on Computer and Communications Security, Washington, USA, 2003: 251-261.*

[11] *MASHIMA D, AHAMAD M. Using identity credential usage logs to detect anomalous service accesses// Proceedings of the ACM Workshop on Digital Identity Management, Chicago, USA,2009: 73-80.*

[12] *YANG E, GU B, YOON T. Intensified analysis and comparison of 5 flacicirus with the use of decision tree and support vector machine (SVM)// Proceedings of the International Conference on Advanced Communication Technology, Bongpyeong, South Korea, 2017: 526-529.*

[13] *LEWIS D D. Feature selection and feature extraction for text categorization// Proceedings of the Workshop on Speech & Natural Language, New York, USA,2013: 212-217.*

[14] *MA Shi-long, WU Nirijiji-ge, LI Xiao-ping. Overview of Data and Deep Learning. CAAL Transactions on Intelligent Systems, 2016, 11(6): 728-742.*
(马世龙，乌尼日其其格，李小平. 大数据与深度学习综述[J]. 智能系统学报，2016，11(6)：728-742.)

[15] *ERHAN D, BENGIO Y, COURVILLE A, et al. Why DOES unsupervised Pre-training help deep learning?. Journal of Machine Learning Research, 2010, 11(3): 625-660.*

[16] *SUTSKEVER I, MARTENS J, DAHL G, et al. On the importance of initialization and momentum in deep learning// Proceedings of the International Conference on International Conference on Machine Learning, Atlanta, USA, 2013: III-1139--III-1147.*

[17] *KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks// Proceedings of the International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, 2012: 1097-1105.*

[18] *MIKOLOV T, KARAFIáT M, BURGET L, et al. Recurrent neural network based language model// Proceedings of the Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 2010: 1045-1048.*

[19] *GRAVES A. Long Short-Term memory. Berlin,Germany: Springer, 2012.*

[20] *ZHOU C, SUN C, LIU Z, et al. A C-LSTM neural network for text classification. Computer Science, 2015, 1(4): 39-44.*

[21] *Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014..*

[22] *YANG T, LI Y, PAN Q, et al. Tb-CNN: joint tree-bank information for sentiment analysis using CNN// Proceedings of the Control Conference, Chengdu, China, 2016: 7042-7044.*

[23] *CHANG P C, WANG D D, ZHOU C L. A novel model by evolving partially connected neural network for stock price trend forecasting. Expert Systems With Applications, 2012, 39(1): 611-620.*

[24] *SHEN Y, HAN T, YANG Q, et al. CS-CNN: enabling robust and efficient convolutional neural networks inference for Internet-of-Things applications. IEEE Access, 2018, PP(99): 1.*

[25] *HE S, ZHU J, HE P, et al. Experience report: system log analysis for anomaly detection// Proceedings of the IEEE International Symposium on Software Reliability Engineering, Ottawa, Canada, 2016: 207-218.*

[26] *SF Yang, WY Chen, YT Wang. Icas: an inter-VM IDS log cloud analysis system// Proceedings of the EEE International Conference on Cloud Computing and Intelligence Systems, Beijing, China, 2011: 285-289.*

[27] *HERRER AS J, GóMEZ R. Log analysis towards an automated forensic diagnosis system// Proceedings of the Ares '10 International Conference on Availability, Reliability, and Security, Krakow, Poland, 2010: 659-664.*

[28] *MANI S, CATHERINE R, SINHA V S, et al. AUSUM:approach for unsupervised bug report summarization// Proceedings of the ACM Sigsoft International Symposium on the Foundations of Software Engineering, Cary, North Carolina, 2012: 1-11.*

[29] *DU M, LI F, ZHENG G, et al. Deeplog: anomaly detection and diagnosis from system LOGS through deep learning// Proceedings of the ACM Sigsac Conference on Computer and Communications Security, Dallas, USA, 2017: 1285-1298.*

[30] *MENG F J, ZHANG X, CHEN P, et al. Driftinsight: detecting anomalous behaviors in Large-Scale cloud platform// Proceedings of the IEEE International Conference on Cloud Computing, Honolulu, USA, 2017: 230-237.*

[31] *XU W, HUANG L, FOX A, et al. Detecting large-scale system problems by mining console logs// Proceedings of the ACM Sigops Symposium on Operating Systems Principles, Big Sky, USA,2009: 117-132.*

[32] *WU F, ANCHURI P, LI Z. Structural event detection from log messages// Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 2017-08-13, 2017: 1175-1184.*

[33] *Chandola V , Banerjee A , Kumar V . Anomaly Detection: A Survey. ACM Computing Surveys, 2009, 41(3):15.*

[34] *RABKIN A, XU W, WILDANI A, et al. A graphical representation for identifier structure in logs// Proceedings of the Workshop on Managing Systems Via Log Analysis & Machine Learning, Vancouver, Canada, 2010: 3-12*

[35] *AREVIAN G. Recurrent neural networks for robust Real-World text classification// Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence,Fremont,USA, 2007-11-02, 2007: 326-329.*

[36] *Rie Johnson, Tong Zhang. Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. . Eprint Arxiv, 2014.12(1): 1058-1067.*

[37] *JOHNSON R, ZHANG T. Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding. ADV Neural INF Process SYST, 2015(28): 919-927.*

[38] *YIN W, KANN K, YU M, et al. Comparative study of CNN and RNN for natural language processing. LMU Munich & IBM Research, 2017,2(7): 923-2001.*

[39] *SHIN J, KIM Y, YOON S, et al. Contextual-CNN: a novel architecture capturing unified meaning for sentence classification// Proceedings of the IEEE International Conference on Big Data and Smart Computing, Shanghai, China, 2018: 491-494.*

[40] *LAI S, XU L, LIU K, et al. Recurrent convolutional neural networks for text classification. AAAI, 2015(1): 2267-2273.*

[41] *Kim Y. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.*

[42] *OLINER A, GANAPATHI A, XU W. Advances and challenges in log analysis. Communications of the ACM, 2012, 55(2): 55-61.*

[43] *GAL Y, GHAHRAMANI Z. A theoretically grounded application of dropout in recurrent neural networks. Statistics, 2015(4): 285-290.*

[44] *car M, Karahoca A. Designing an Early Warning System for Stock Market Crashes Based On Adaptive Neuro Fuzzy Inference System Forecasting// Proceedings of the 2nd International Conference on Risk Analysis and Crisis Response, Beijing, China, 2009:79-85.*

[45] *JING L P, HUANG H K, SHI H B. Improved feature selection approach TFIDF in text mining// Proceedings of the International Conference on Machine Learning and Cybernetics, Beijing, China, 2003,2: 944-946*

[46] *JOACHIMS T. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization// Proceedings of the International Conference on Machine Learning, San Francisco, USA, 1997: 143-151.*

[47] *SHI C Y, CHAO-JUN X U, YANG X J. Study of TFIDF algorithm. Journal of Computer Applications, 2009(29): 167-180.*

MEI Yudong, born in 1996, Ph.D candidate. His current research interests include machine learning and data mining.

CHEN Xu, born in 1993, M.S. candidate. His current research interests include medical diagnosis and data mining.

SUN Yu-Zhong, born in 1968, Ph.D.. professor. His main research interests include big data intelligence analysis (machine learning) and calculation.

NIU Yixiang, born in 1995, M.S. candidate. His current research interests include big data and cloud computing.

XIAO Li, born in 1987, Ph.D.. His current research interests include artificial intelligence and computational medicine.

WANG Hairong, born in 1977, Ph.D.. Her main research interests include knowledge engineering of big data.

FENG Baiming, born in 1966, Ph.D.. His main research interests include cloud computing and artificial computing.

Background

At present, data mining, as a method of analysis with high timeliness and high fidelity, is playing an increasingly important role in society. Its quick pattern-discovering ability in large-scale data, and the ability to quickly discover laws is gradually replacing the role of manpower. In the current large-scale distributed systems (such as Hadoop, Spark, etc.), there are tens of thousands of system logs every day. The amount of data in these logs and the chaos of the relationship have greatly affected the programmers. Manual monitoring of the system's efficiency also increases the cost of training for new programmers.

Therefore, the machine learning model is also increasingly mentioned by the industry for system log analysis.But in most cases, the system logs will report really few “serious” logs of the system, which are the programmers most concerned about. However, since most machine learning models used for system log analysis are assumed to train on balanced data, these models are prone to overfitting when they do syslog warnings, so that the results are not ideal enough.

This paper will explore the application capabilities of CNN-text (CT) in system log analysis from the perspective of deep learning. By comparing CT with the mainstream system log analysis machine learning model SVM and decision tree, we will explore the superiority of CT, comparing with these algorithms; we will compare CT with CRT, analyzes the treatment of CT features, and verifies the superiority of CT in processing deep-learning models to process syslog class texts; finally applies all models to two different log class texts. Contract the data to prove the universality of CT.

We evaluate the proposed method with some other method on a set of data of logs collected by logstash in our laboratory, to prove that CT performs better than many other models; and then we evaluate them on a set of public data, WC85_1, to prove the Universality of the CT.

The proposed work is under the support of the Networked Operating System for Cloud Computing (Grant No. 2016YFB1000505) and the Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (U1611261). The project is to investigate advanced statistical models for the intelligent log records analysis for predicting log case labels in a large multi-label, heterogeneous and imbalanced data set. The authors have proposed a deep learning model for the multi-label prediction in system log analysis. This work is a significant extension and improvement over system log analysis.