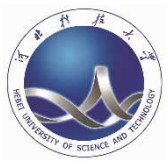
中图分类号：TP393.0　　密级：公开
UDC：621.39　　学校代码：10082



河北科技大学
HEBEI UNIVERSITY OF SCIENCE AND TECHNOLOGY

硕士学位论文

基于用户和实体行为分析的威胁检测技术研究

论 文 作 者：闫风如
指 导 教 师：张冬雯　教授
副 指 导 教 师：张光华　教授
申请学位类别：工学硕士
学 科 、 领 域：计算机科学与技术
所 在 单 位：信息科学与工程学院
答 辩 日 期：2023 年 5 月

Classified Index： TP393.0 **Secrecy Rate：** Publicized

UDC： 621.39 **University Code：** 10082

Hebei University of Science and Technology

Dissertation for the Master Degree

Research on Threat Detection Technology Based on User and Entity Behavior Analysis

Candidate： Yan Fengru

Supervisor: Prof. Zhang Dongwen

Associate Supervisor： Prof. Zhang Guanghua

Academic Degree Applied for: Master of Engineering

Speciality： Computer Science and Technology

Employer： School of Information Science and Engineering

Date of Oral Examination: May, 2023

摘 要

随着各种新型网络技术的兴起和网络应用的发展，网络威胁变得复杂多样，威胁检测技术随着攻击行为的不断出现而提高和完善，但是其一直受新技术的挑战。因此提出能够有效识别未知攻击的威胁检测技术是网络安全的重要任务之一。现有的内部威胁检测技术主要采用的是机器学习方法，需要复杂的特征工程，且多数忽略了用户行为的时序性和多维性。入侵检测是一种行之有效的外部威胁检测技术，而现有的入侵检测算法需要大量标记样本来进行模型训练，且当新的攻击类型出现时需要重新训练模型。针对以上问题，本文将用户和实体行为分析技术应用到威胁检测中，对内部威胁和外部威胁的检测算法进行了深入的研究，具体的研究内容如下。

（1）提出基于 LSTM-Attention 用户行为分析的内部威胁检测算法。该方法综合考虑用户行为、角色行为和用户心理数据，通过多维度的用户日常活动来刻画用户的行为，并考虑到了用户行为存在时序性的特点。首先，采用 LSTM-Attention 算法对用户及角色行为进行建模。其次采用 MLP 算法对用户行为、角色行为以及用户心理评分进行综合决策，从而实现内部威胁检测。在 CERT 数据集上进行评估，该算法取得了 96.4%的 AUC 分数，优于现有的内部威胁检测算法。其解决了当出现新用户时无法进行内部威胁检测的弊端，同时还提高了算法的 AUC 分数。

（2）提出基于 IG-FCBF-TL 流量行为分析的网络入侵检测算法。该算法首先采用数据预处理和 IG-FCBF 特征工程对样本进行处理。然后选择 VGG16、Inception 和 Xception 三种 CNN 模型作为基础学习模型，并采用超参数优化方法 Tree-Structured Parzen Estimator(TPE)算法在数据集上寻求最优参数。最后采用置信度平均的集成方法对优化后的三个 CNN 模型进行集成，从而实现更高准确率的网络入侵检测。其在 CICIDS2017 和 NSL-KDD 数据集上均取得超 99%的准确率，也证明该模型具有较好的泛化能力。其解决了入侵检测领域网络流量数据量不足、收集样本代价大的问题。

（3）提出基于 TfidfVectorizer 和孪生 LSTM 网络实体行为分析的主机入侵检测算法。首先，该算法采用 TfidfVectorizer 对系统调用序列进行向量化处理，然后采用基于相似度的孪生 LSTM 网络的小样本学习算法对主机行为进行分析，从而达到主机入侵检测的目的。通过在 ADFA-LD 数据集上进行评估，其采用少量的训练样本对各个攻击类型的检测准确率均超过 96%。其不仅能够识别未知攻击，也解决了现有主机入侵检测算法依赖大量高质量有标注的样本进行模型训练的问题。

关键词 网络安全；用户和实体行为分析；内部威胁检测；主机入侵检测；网络入侵检测；深度学习；小样本学习

Abstract

With the emergence of various network technologies and the development of network applications, network threats have become more complex and diverse. While threat detection technology has continuously improved with the emergence of new attacks, it has been challenged by new technologies. As a result, proposing an effective threat detection technology to identify unknown attacks has become a critical task for network security. Currently, most existing insider threat detection technologies rely on machine learning methods. However, these methods require complex feature engineering, and often disregard the timing sequence and multidimensionality of user behavior. Intrusion detection is a proven technique for detecting external threats, but existing algorithms require a significant number of labeled samples for model training, and often need to retrain the model when new types of attacks emerge. To address these issues, this thesis applies user and entity behavior analysis techniques to detect threat, and conducts an in-depth study of the detection algorithms for internal and external threats. The specific research contents are as follows.

(1) An insider threat detection model based on LSTM-Attention user behavior analysis is proposed. This method comprehensively takes into account user behavior, role behavior, and psychological data. It describes user behavior through multi-dimensional daily activities and also considers the time sequence of user behavior. Firstly, the LSTM-Attention algorithm is employed to model both user and role behaviors. Secondly, the MLP algorithm is utilized to make a comprehensive decision on user behavior, role behavior, and user psychological score, enabling the detection of insider threats. Evaluated on the CERT dataset, the algorithm achieves a 96.4% AUC score, which is superior to existing Insider threat detection algorithms. This not only addresses the drawback of being unable to perform insider threat detection when a new user appears but also improves the AUC score of the algorithm.

(2) A network intrusion detection algorithm based on IG-FCBF-TL traffic behavior analysis is proposed. The proposed algorithm initially preprocesses the data and performs IG-FCBF feature engineering. Subsequently, three CNN models, namely VGG16, Inception, and Xception, are selected as basic learning models. The Tree-Structured Parzen Estimator (TPE) algorithm is then employed to optimize the hyperparameters on the datasets. Finally, the ensemble method of confidence averaging is utilized to integrate the three optimized CNN models, achieving higher accuracy in network intrusion detection. It

achieved over 99% accuracy on both CICIDS2017 and NSL-KDD datasets, which also proves that the model has good generalization ability. This method addresses the issues of insufficient network traffic data and the high cost associated with collecting samples in the field of intrusion detection.

(3) The host intrusion detection algorithm based on TfidfVectorizer and Siamese LSTM network entity behavior analysis is proposed. Firstly, the algorithm utilizes TfidfVectorizer to vectorize the system call sequence. It then applies the few-shot learning algorithm of the Siamese LSTM network based on similarity to analyze the behavior of the host and achieve the purpose of host intrusion detection. Through evaluation of the ADFA-LD dataset, its detection accuracy for various attack types using a small number of training samples exceeded 96%. This approach can not only identify unknown attacks but also resolves the issue of existing host intrusion detection algorithms relying on a large number of high-quality labeled samples for model training.

Key words Network security; User and entity behavior analysis; Insider threat detection; Host intrusion detection; Network intrusion detection; Deep learning; Few-shot learning

目 录

第 1 章　绪　论

1.1　研究背景与意义

随着各种新型网络技术的高速发展，人们的日常生活和工作变得更加信息化与便捷。在享受互联网对生活带来便利的同时，也面临着更加复杂的安全威胁和攻击，系统入侵、信息泄露和病毒植入无时无刻影响着正常的网络服务。攻击者通过网络窃取、传播和破坏未经授权的用户信息，给企业造成巨大的财产损失和名誉损失。因此，企业通过购买形式各样的安全产品和服务，如防火墙[1]、入侵检测系统[2]或防杀毒产品来维护公司的网络安全。与此同时，由于各大项目下属各类承包商，服务和合作趋向多元化，公司员工结构不稳定，导致企业遭受严重的内部威胁。据安全机构 RBS 发布的数据泄露报告显示，2021 年共发生 4 145 起公开披露的数据泄露事件，造成 227.7 亿条数据泄露，庞大的数字揭示了网络安全问题的严峻性。据 Verizon 发布的《2022 年数据泄露调查报告》可知，2022 年的数据泄露事件有 82%的涉及人为因素，其中勒索软件增加了 13%，超前五年的总和。与 2021 年相比，全球网络安全攻击增加了 1.885%[3]。

由此可见，无论是外部攻击还是内部威胁都使得企业或个人遭受巨额的财产损失，此外还会对民族国家的网络安全造成威胁。因此，如何采取安全有效的威胁检测技术来应对日益严峻的网络安全问题是当前安全领域的研究重点。内部威胁指来自组织内部人员的恶意威胁，通过故意欺诈等手段换取商业信息或破坏企业的计算机系统，其与外部威胁不同，需要识别被授权用户的异常行为，具有复杂性和隐蔽性。针对外部威胁，构建入侵检测系统是最有效的方法之一，其能够帮助识别系统异常行为和来自网络的恶意攻击。

传统的威胁检测技术主要是基于规则或基于签名的，采用的大多是机器学习模型。随着攻击技术的不断发展，其不能有效的检测网络中的威胁，而且机器学习模型主要面向的是安全事件本身，当出现新的安全事件形式，则需要重新训练模型，同时在此过程中也易受到新的网络攻击。在 2016 年，Gartner 安全和风险管理峰会提出的用户和实体行为分析（User and Entity Behavior Analytics，UEBA）技术，其关注的是用户和实体的行为而不是安全事件本身，能够有效进行威胁检测。本文研究基于用户和实体行为分析的威胁检测技术，采用深度学习和小样本学习算法来减少威胁检测过程中的漏报和误报。

1.2　国内外研究现状

近年来，随着计算机网络的高速发展以及新型互联网技术的出现，使得网络威

胁变得越来越复杂，网络攻击给个人或企业带来巨大的损失。目前威胁检测的相关研究主要分为内部威胁检测和外部威胁检测，采用的主要技术为用户和实体行为分析技术，而外部威胁检测最有效的是入侵检测技术，根据数据来源可以将其分为网络入侵检测和主机入侵检测。

1.2.1 用户和实体行为分析的研究现状

用户和实体行为分析是一种检测用户行为和组织内部攻击的新方法，它使用先进的数据分析方法来检测用户和实体行为中的异常。UEBA 是在用户行为分析的基础上形成的，首先被应用于电商领域，其通过数据分析建立用户行为基线，为客户提供更有针对性的服务，从而促进商品销量。在 2014 年，Gartner 将其应用到网络安全领域，于 2015 年，正式命名为用户和实体行为分析。由于威胁究其原因是由于人有意或无意的行为，但其所有的行为在应用程序、数据资产和主机等实体上都有体现，因此持续监控用户和实体的行为并构建行为基线，就能识别其异常行为。聚焦用户和实体的行为，可以提高准确率并降低误报率，UEBA 的实施是一个迭代优化、持续改进的过程，通过探索不同的特征工程、异常检测算法来改进威胁检测的性能。

文献[4]提出了 Niara 安全分析平台的用户和实体行为分析的解决方案，其跟踪和监控企业中用户、IP 地址和设备的行为，使用奇异值分解算法从 Niara 内部网络收集的流量数据中检测异常行为。文献[5]采用了用户和实体行为分析中不同的方法，包括基于用户和角色的检测、用户和实体活动映射、用户分析技术和个人风险评分计算，从而实现威胁检测。文献[6]提出了一种基于模糊粒子群聚类的多宿主异常行为检测模型，利用 BF-IEF 技术，优化用户和实体行为相似度的度量方法，其异常检测能力显著提高，提高了信息系统在实际应用中抵御未知威胁的能力。文献[7]从应用窗口的使用角度出发，对用户行为进行研究，采集与分析用户在应用窗口上的行为数据，从中提取面向异常用户检测与用户变化行为识别的行为特征。

综上所述，用户和实体行为分析技术已在网络安全领域被广泛使用，并在医疗行业、金融行业、能源行业和政企行业得到验证，其有比传统威胁检测技术更好的效果。故本文也将 UEBA 技术应用到内外部威胁检测中，从而提高威胁检测的性能。

1.2.2 内部威胁检测的研究现状

内部人员位于组织内部，距离私有数据或私有服务器更近，且具有一定的合法操作权限。内部威胁的高危性、隐蔽性和复杂性等特点，使得内部威胁检测难度加大。传统的内部威胁检测技术主要有基于用户命令检测[8]、基于生物特征认证[9]、基于大数据和机器学习[10]等，主要检测对象是安全事件本身，如木马和病毒。但是随着攻击方式越来越复杂，传统的威胁检测技术已不能有效检测网络中的威胁。相关领域的专家学者针对内部威胁提出了不同解决方案，其中基于规则的方法是使用较

广泛的方法之一。其一般过程是：首先通过挖掘行为的关联规则建立正常的行为画像；然后通过分析传入的实例和现有的规则执行异常检测。文献[11]提出一种改进物理安全和内部威胁检测的本体框架和方法，其利用基于规则的异常检测促进数据取证分析，并主动减少内部威胁。基于规则的方法具有过程简单和响应迅速的优点，其缺点是需要大量专家知识建立规则数据库，效果取决于行为库的更新，无法识别未知模式的威胁。

随着神经网络的发展，基于深度学习的用户和实体行为分析技术在内部威胁检测中得到广泛应用，内部威胁检测的关键是建立用户正常行为模型，从而通过行为偏差识别异常行为。文献[12]采用基于 Transformer 的双向编码器表示(BERT)来检测 APT 攻击序列，充分考虑了长攻击序列和长期连续 APT 攻击的特点，提高 APT 攻击序列检测的准确率。用户行为可以看作长期的时间序列数据，而 LSTM 网络有较强的学习长期序列模式的能力，可以发现内部用户行为中的隐含行为特征，大大提高检测率。文献[13]在流数据上使用各种 LSTM 网络进行评分，并综合评分实现异常检测。文献[14]使用 LSTM-CNN 框架发现用户的异常行为，首先使用 LSTM 网络学习用户行为，提取抽象的时间特征；然后使用卷积神经网络检测内部威胁。上述方案存在一些不足之处，这些用户行为建模方法大多忽视了内部威胁的特殊性，没有充分利用心理数据和同一角色行为的相似性等因素。

综上所述，机器学习算法在内部威胁检测领域得到广泛应用，其取得较好的成果。但因为内部威胁的特殊性，现在算法的误报率较高，AUC 分数较低。如何进行数据处理发现数据间的内在关联并选择适合用户行为模式的算法进行内部威胁检测是当前安全领域的一项重要任务。

1.2.3　网络入侵检测的研究现状

网络入侵检测系统（Network Intrusion Detection System, NIDS）对危害系统安全的行为进行检测，如漏洞信息收集、拒绝访问和非法获取系统权限等，其配置简单、适用于大部分的系统。随着人工智能的发展，大量学者将机器学习和深度学习两大人工智能方法用于网络入侵检测领域，传统的机器学习方法是浅层学习，适用于样本数量和特征维度较小的情况。文献[15]提出一种基于随机森林和 SVM 的机器学习方法来预测网络入侵，通过随机森林计算变量重要性得分来进行特征选择从而发现潜在特征。文献[16]采用 KNN 分类器的特征选择和模型选择，通过集成各种加速和改进分类器的技术进行网络入侵检测，与传统 KNN 比较，其运行时间和错误率均减少了。文献[17]采用决策树（J48）算法对网络入侵检测系统中的网络流量包进行分类，并采用 Kyoto 2006+数据集进行评估，其较 KDD99 更能代表当今网络攻击现状。

随着技术的不断发展，深度学习算法已被广泛用于开发 IDS 来检测企业网络遭受的各种攻击，如欺骗攻击、拒绝服务和模糊攻击等。该方法依赖大量有标注的网

络流量数据。文献[18]提出了一种将流计算和深度学习相结合的网络攻击检测方法，其采用支持向量机和深度信念网络两种分类算法，在 CICIDS2017 数据集上进行了一系列对比实验，表明实时检测效率高于传统机器学习算法。文献[19]基于 NSL-KDD 数据集，提出了一种结合了双向长短期记忆和注意力机制的恶意流量检测模型 BAT，用于解决入侵检测中存在的特征工程和准确率低的问题。文献[20]提出了一种 SDAE-ELM 的集成深度入侵检测模型，用于实现对入侵行为的及时响应，并通过小批量梯度下降法进行网络训练和优化，在 NSL-KDD、UNSW-NB15 和 CIDDS-001 等多个数据集上进行了实验。但是基于深度学习的入侵检测模型需要大量的数据支撑且往往伴随着较高的计算复杂度。

由于恶意流量样本较少且收集大量训练样本代价较大，安全研究员开始将研究转向迁移学习。在目标域的数据量较小的情况下，可以通过数据量充足或相似任务中进行迁移学习，结合微调可以获得较好的性能，此外还能减少模型训练时间。文献[21]提出了基于深度迁移学习的车内网络入侵检测的 P-LeNet 方法。P-LeNet 模型在汽车黑客数据集上获得了高达 97.83%的 f1-分数。文献[22]采用非监督学习的深度自编码器进行迁移学习，其框架由嵌入层和标签层来实现编码和解码，源域与目标域共享权重，用于知识的迁移，从而实现网络的入侵检测。文献[23]提出一种互信息加权的集成迁移学习方法用于网络入侵检测。该方法利用迁移策略对多个特征组进行建模，在迁移模型中运用互信息度量评估特征集在不同领域内的数据分布，最后依据度量值，集成加权迁移模型，得到集成迁移模型。因此，在网络入侵检测领域数据量不足的情况下，使用迁移学习进行模型训练并采用有效的集成策略对各个模型进行集成，是解决新环境下入侵检测问题的突破点。

1.2.4 主机入侵检测的研究现状

主机入侵检测系统（Host Intrusion Detection System, HIDS）与网络入侵检测系统有显著的不同，因为 NIDS 检测组织网络中的攻击，而 HIDS 检测在 NIDS 失败或被绕过后对组织主机系统的攻击，其具有检测内部攻击和高级持续威胁的能力。HIDS 是一种安装在主机上的 IDS，通过分析程序行为、日志、系统调用序列等系统信息来检测主机中的威胁，某些情况下特通过网络流量分析检测主机中的威胁[24]。现有的主机入侵检测系统主要分为两大类，即基于误用的 HIDS 和基于异常的 HIDS。

基于误用的 HIDS[25~27]通过规则数据库中预定义的签名模板来识别异常的系统进程，其规则库的建立离不开专家的经验知识。文献[28]使用开源软件 Snort 实现基于签名的入侵检测，其被广泛用于入侵检测和防御领域，Snort 的组件如图 1-1 所示，所有组件一起工作，从而检测各种攻击并产生输出，其具有嗅探、过滤的能力。文献[29]提出基于主机 Linux 操作日志误用检测系统，采用主成分分析特征提取技术和基于系统调用序列的 KNN 分类方法。虽然这种方法很有效，假阳性率也较低，但是

它们只能识别已知的攻击行为，而且准确率高度依赖专家建立规则库的质量。

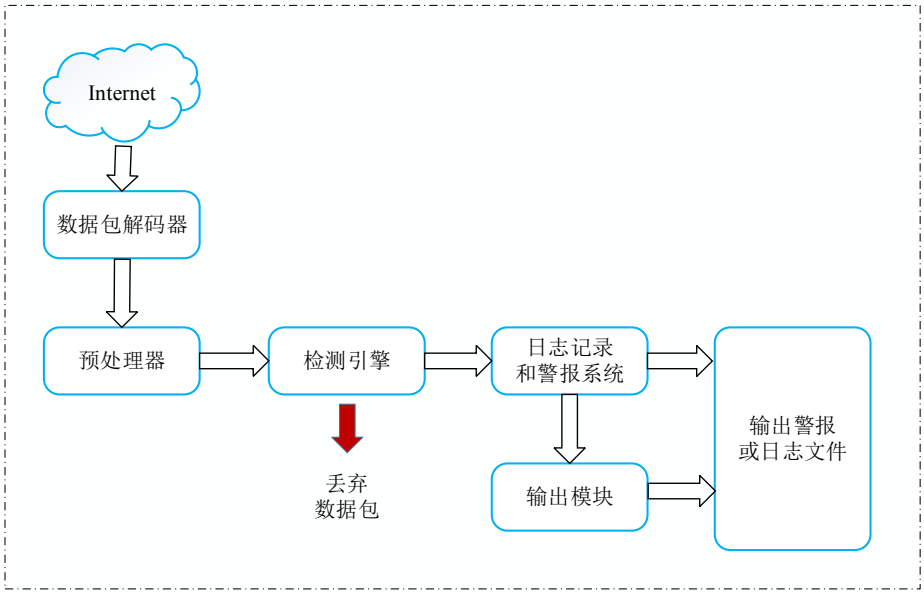


图 1-1　Snort 的组件

另一种主机入侵检测方法是基于异常的 HIDS[30~32]。它是基于主机的行为变化，而不是特定的攻击类型，通过建立正常行为基线，将偏离基线的行为视为异常。它为主机构造一个正常的配置文件，当出现的事件行为与正常的配置文件不同时，就会触发异常警报。文献[33]提出了一种基于长短期记忆网络的主机入侵检测模型，建立系统调用序列的正常模型然后采用多个阈值的分类进行入侵检测。文献[34]提出了基于循环神经网络的异常入侵检测系统，其使用的是堆叠的 CNN 与 GRU，其合并导致了异常 IDS 的改进，将发生概率低的序列被归类为异常。基于异常的 HIDS 可以检测到零日攻击，但其具有更高的假阳性率，而且需要大量的样本进行模型训练，此外还需要定期进行重训练，以将系统活动模式的变化归入正常的基线模型。因此，迫切需要研究一种高效、可移植性强且不依赖于大量数据样本的主机入侵检测算法。

为了能够全面检测主机上的入侵行为，弥补传统入侵检测的不足，研究人员提出基于误用和基于异常混合的主机入侵检测方法，其结合了基于签名 IDS 的检测速度快的优势和基于异常 IDS 能够检测未知攻击的优势。文献[35]提出一种 Holt-winter 异常算法和基于签名算法混合实现的 IDS，并对其进行案例研究，实验结果显示其能够减少误报与漏报，明显提高入侵检测系统的性能。文献[36]提出一种基于混合的 IDS 来检测已知和未知的攻击，并采用多区安全方法来保护云安全，第一区使用基于签名的 IDS 来检测已知的攻击，第二区使用基于异常的 IDS 检测未知攻击。然而从现有的规模和框架来看，基于混合的 IDS 仍需进一步开发，以在行业中实现。

1.3　主要研究内容

威胁检测是预防和抵御网络攻击的重要手段，为了给企业和个人提供更安全的

网络环境，国内外学者围绕威胁检测中的内部威胁检测、网络入侵检测和主机入侵检测进行了广泛深入的研究，从而减少网络威胁对企业及个人造成的危害。在实际应用中，目前的威胁检测技术还存在以下问题：（1）目前的内部威胁检测算法仅使用用户行为建模，没有综合考虑多个维度的行为，如用户对应的角色行为和心理数据等；其次，目前的内部威胁检测算法的准确率和 AUC 分数不高。（2）目前的网络入侵检测算法多数是基于深度学习模型的，其需要大量有代表性的训练样本，收集样本代价大，其次存在类不平衡问题，恶意流量的数量远远小于正常流量。（3）目前的主机入侵检测算法的出发点是安全事件本身，而不是主机行为（系统调用序列），当出现新的安全事件时，无法准确的检出；其次，标注大量训练样本耗费资源巨大，而在样本较少的情况下，原有主机入侵检测算法准确率低、误报率高。针对以上问题，本文的具体研究内容如下。

（1）研究并实现基于 LSTM-Attention 的用户行为分析算法用于内部威胁检测。首先，提取用户的行为序列、用户行为特征、角色行为特征和心理数据来描述用户的日常活动，通过多维度的用户日常活动来描述用户的行为；其次，使用长短期记忆网络和注意力机制学习用户的行为模式，并计算真实行为与预测行为之间的偏差；最后，使用多层感知机（Multilayer Perceptron，MLP）根据这些偏差进行综合决策来识别异常行为。

（2）研究并实现基于 IG-FCBF-TL 的流量行为分析算法用于网络入侵检测。首先，采用信息增益和快速相关性滤波算法组合的特征工程方法进行预处理，并将其转换为适合 CNN 模型输入的图像形式；其次，选择 VGG16、Inception 和 Xception 三种 CNN 模型作为基础学习模型，并采用 Tree-Structured Parzen Estimator(TPE)算法的超参数优化方法在目标数据集上寻求最佳模型；最后采用置信度平均的集成方法对优化后的三个 CNN 模型进行集成。在入侵检测领域网络数据量不足的情况下，采用迁移学习进行模型训练，提高训练效率，并实现网络流量的正确分类。

（3）研究并实现基于 TfidfVectorizer 和孪生 LSTM 网络的主机行为分析算法用于主机入侵检测。通过分析系统调用序列，而避免安全事件本身，进行主机入侵检测。首先，使用 TfidfVectorizer 对样本特征进行词向量化处理，然后将二维数据转化图像数据，最后将其输入孪生 LSTM 网络进行模型训练，并采用对比损失函数作为优化函数。通过小样本学习算法来分析主机行为，解决主机入侵检测领域中现有深度学习模型的小样本问题和可移植性较差的问题。

1.4 论文组织结构

如图 1-2 所示，首先，在本文的绪论部分阐述研究背景及意义、国内外研究现状以及主要研究内容。其次，对本文相关基础理论进行介绍。然后针对威胁检测中的内部威胁和外部威胁检测进行深入研究并提出了相应的解决方案，其中外部威胁检

测采用的是最广泛的入侵检测方法，其包括网络入侵检测和主机入侵检测。最后，对全文进行总结，并指出下一步的研究工作方向，各章节的主要内容如下。

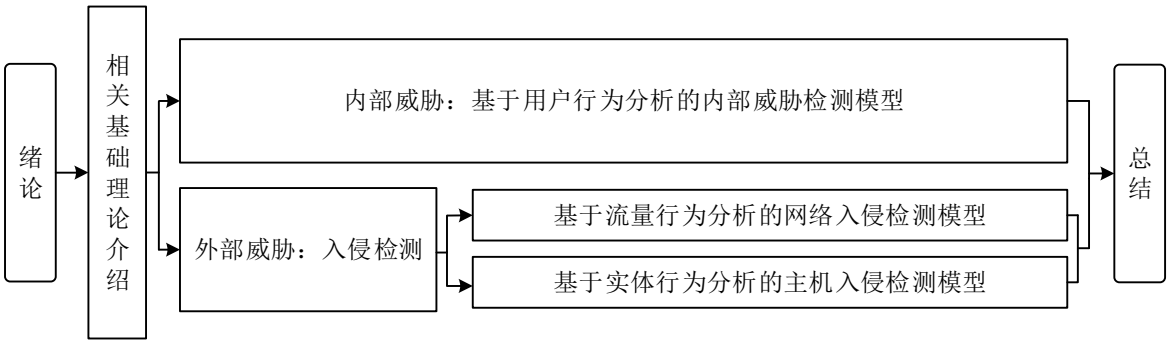


图 1-2　论文的组织结构

第一章，绪论。首先，介绍了本文的研究背景及意义，然后阐述了用户和实体行为分析、内部威胁检测、网络入侵检测和主机入侵检测的国内外研究现状，分析了传统机器学习和深度学习在威胁检测方面的适用性，最后对本文的主要研究内容和章节安排进行阐述。

第二章，相关基础理论介绍。首先，介绍用户和实体行为分析的定义和应用。其次，介绍了网络威胁中的内部威胁和外部威胁以及其检测模型。然后，介绍威胁检测相关算法包括深度学习、迁移学习和小样本学习。

第三章，基于用户行为的内部威胁检测模型。首先，介绍了内部威胁检测数据集 CERT 数据集，其次，构建基于 LSTM-Attention 的内部威胁检测算法模型，最后对实验结果进行分析。

第四章，基于流量行为分析的网络入侵检测模型。首先，对网络入侵检测数据集 CICIDS2017 进行描述，其次，对基于 IG-FCBF-TL 的网络入侵检测框架进行描述，包括数据预处理、IG-FCBF 特征工程、迁移学习集成学习等过程，最后是实验结果的分析。

第五章，基于实体行为分析的主机入侵检测模型。首先，对主机入侵实验数据集 ADFA-LD 进行介绍。其次，构建基于 TfidfVectorizer 和孪生 LSTM 网络的主机入侵检测算法，最后是实验结果对比分析，包括 LSTM 层数损失函数等。

最后对本文的研究内容进行总结与分析，并对未来工作进行展望。

第 2 章　相关基础理论介绍

研究威胁检测需要深入了解相关基础理论。本章主要包括四个部分。第一部分介绍了内部威胁的定义、分类和检测模型。第二部分对入侵检测的相关概念进行描述。第三部分对用户和实体行为分析技术进行阐述。第四部分对威胁检测相关算法进行论述。

2.1　内部威胁概述

2.1.1　内部威胁的定义

至今为止内部威胁还没有一个统一的定义，其是相对于外部威胁而言的，一般指来自组织内部人员的恶意威胁，通常包括故意欺诈，盗窃具有商业价值的信息或破坏计算机系统。随着对内部威胁研究的不断深入，不同的专家给出不同的理解与定义，但是主要是从内部人的角度上出发的。早在 1997 年，Tugular 等人[37]就将能够使用特定计算机系统且被授予权限级别并违反组织安全策略的人定义为内部攻击者。Schultz 等人[38]从是否有计算机、网络的授权和是否对其故意滥用造成损失来定义了内部威胁。Ning 等人[39]依据计算机网络节点来定义内部人员，将拥有其完全控制权的人视为内部人员，其采取的恶意行为视为内部威胁。综上所述，内部人员是对组织有一定了解和权限并能对其造成有意或无意的威胁。CERT/CC 的最新技术报告[40]将内部威胁定义为“一个员工、承包商或者商业合作伙伴被授权或曾被授权访问组织的系统、网络以及数据，并且故意使用以对组织信息的机密性、完整性和可用性产生负面影响的方式访问信息系统”。这一概念被广大学者认同并沿用，从定义中可以发现内部威胁具有隐蔽性、高危性和多样性的特点。

隐蔽性：内部攻击行为不经常发生，属于小概率事件，其被大量正常的行为数据所覆盖，一般情况下难以发现；其次攻击人员熟悉内部组织结构，在一定程度上可以采取措施主动规避安全检测。

高危性：由于内部攻击者距离私有数据或私有服务器更近，且具有一定的合法权限，从而更容易对组织进行有针对性的破坏，内部威胁往往会造成比外部威胁更加严重的后果。

多样性：首先，内部威胁攻击主体呈现多样化的特点。攻击主体包括组织内的员工、服务提供商和商业合作伙伴等；其次，内部威胁的攻击手段也是多样化的。攻击者通过自身权限窃取敏感信息，或者通过职务之便进行内部欺诈等。

2.1.2　内部威胁的分类

内部人员分为滥用其特权进行恶意活动的叛徒，代表合法雇员进行非法行为的

伪装者以及无意中犯错的犯罪者三种类型。根据内部人员进行的恶意活动，可以将内部威胁也分为以下三种类型：信息系统破坏、信息窃取和内部欺诈，其主要通过恶意攻击和内部人员的无意误操作产生的。

（1）信息系统破坏是指具有技术才能的内部人员直接采用信息技术对组织进行破坏。攻击者具有组织系统的专业知识，如程序员、数据库管理员、系统管理员，通常以不满报复为行为动机，通过密码破解、远程入侵、木马植入等技术手段在系统或服务器上埋置逻辑炸弹或删除重要数据。信息系统破坏的影响具有毁灭性，其可用性一旦遭到破坏，业务运行、信息化办公随之受到影响，组织核心数据也可能遭到损毁，给社会稳定造成恶劣影响。若在政府、军工以及基础设施等部门发生系统破坏，带来的影响更加严重，其不仅是经济损失，还可能对整个国家安全造成威胁。

（2）信息窃取是指从企业或组织中窃取关键信息，如源代码、知识产权或客户信息，组织中的技术人员或非技术人员均能实施窃取。其依据攻击人数的不同分为个体与群体信息窃取，后者无法独立窃取，需他人的配合。信息窃取这种内部威胁主要来源于具有核心数据访问权的内部人员，如开发人员、科研人员以及高层管理者，其通过窃取高密度具有商业价值的信息谋取更好的发展机会。一般采用的攻击方式利用内部人员的合法权限，通过拷贝或发送到网盘、邮箱等方式将信息带离组织。

（3）内部欺诈通常指出于经济利益内部人员未经授权修改、删除、添加组织相关数据或通过窃取身份信息进行电子欺诈。内部欺诈主要分为资产侵占欺诈、数据欺诈和身份欺诈三类。其中资产侵占欺诈形式包括贪污、盗窃、偷窃等，数据欺诈指员工篡改、窃取或销毁组织的数据，以获得不当利益；身份欺诈为员工冒充他人身份，或使用他人身份证件等，以获得未经授权的访问权限。其与信息系统破坏、信息窃取不同的是其攻击人员不在局限于技术人员，而是普通员工（如行政秘书、客户经理、人事专员等）。内部欺诈不仅会给个人安全带来威胁，还会给企业造成严重的经济损失甚至对国家安全造成威胁。

2.1.3 内部威胁检测模型

产生内部威胁的原因分为主体原因和客体原因两种，根据其可将内部威胁检测模型分为基于主体和基于客体的检测模型。基于主体的内部威胁检测模型从心理学和社会学的角度出发对心理状态、社会关系及动机等主观因素进行分析并通过计算与行为基线的偏差度来预测是否发生了内部威胁，其代表模型为 CMO 模型和 SKRAM 模型。而基于客体的检测模型是基于客观事实进行分析建模，如电脑的日志记录、日常打卡、网页浏览记录等真实存在的记录，其代表模型有 CRBM 模型。随着人工智能的发展，专家学者开始采用各种机器学习算法来建立内部威胁检测模型。

(1)　基于主体的检测模型　Wood B[41]提出的 CMO 模型是内部威胁最早的通用模型。其将内部威胁定义为由内部威胁用户、内部威胁工具和环境三个因素构成的主观要素。同时，内部攻击者需要满足能力、动机和机会三个主观要素才能实施内部攻击。这个模型可以用来帮助组织评估和管理内部威胁，从而提高组织的安全性。PARKER[42]提出的 SKRAM 模型将其扩展为技能、知识、资源、授权、动机五个元素。基于主体的内部威胁检测模型，贴合实际，充分考虑了主观因素对用户行为的影响，但是存在主观因素难以量化的困难。

(2)　基于客体的检测模型　与基于主体的检测模型相对应，基于客体的检测模型从客观因素出发，对用户行为的真实数据进行分析建模。Park 等人[43]提出的 CRBM 模型是基于角色访问控制的威胁检测模型，根据其所属的角色来监视内部人员在操作系统、组织中的行为。当内部人员行为不符合其所属角色的行为，则视为该内部人员造成了内部威胁。

(3)　基于人工智能的检测模型　该模型利用机器学习、深度学习等人工智能技术，从大量的数据中提取特征并进行分类，以识别内部威胁行为。具体来说，基于人工智能的检测模型通常分为基于规则、基于统计、基于机器学习[44]和基于深度学习[45]的检测模型，在内部威胁检测领域具有很大潜力。其通用模型如图 2-1 所示。

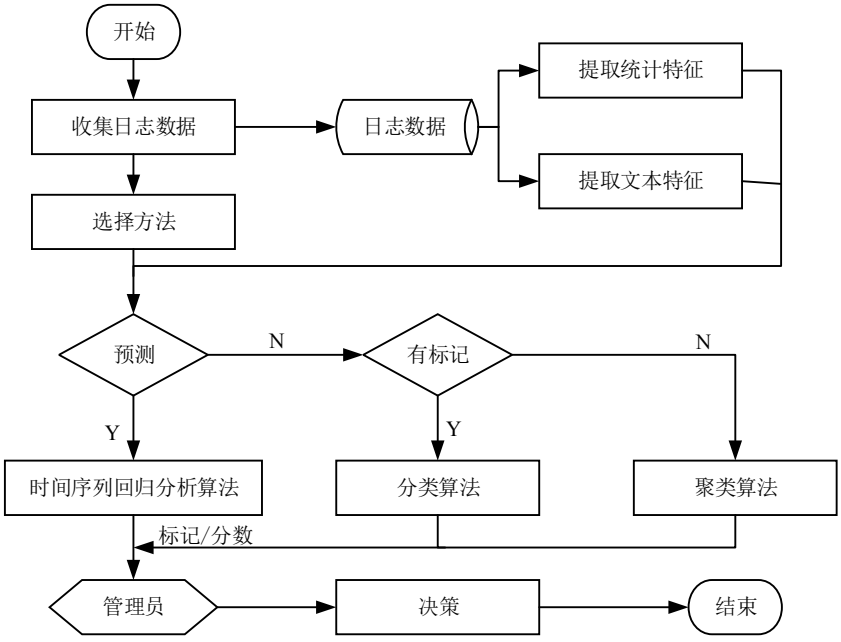


图 2-1　基于人工智能的通用检测模型

2.2　入侵检测概述

2.2.1　入侵检测系统

入侵通常是指未经授权地访问或操作计算机系统、网络或应用程序等，以执行恶意活动或窃取敏感信息的行为。入侵检测是指通过系统日志、网络流量等信息进

行实时监控和分析，从而发现对系统的入侵或尝试入侵的企图。入侵检测系统是一种安全防护设施，用于监测计算机网络或系统中的异常行为或攻击行为，并及时警告或阻止这些行为。Denning[46]针对入侵首次提出一个由主体、客体、审计记录、活动简档、异常记录和活动规则六个主要部分构成的通用入侵检测模型，如图 2-2 所示。其对后续的研究工作具有指导意义，提供了一个通用的参考框架。

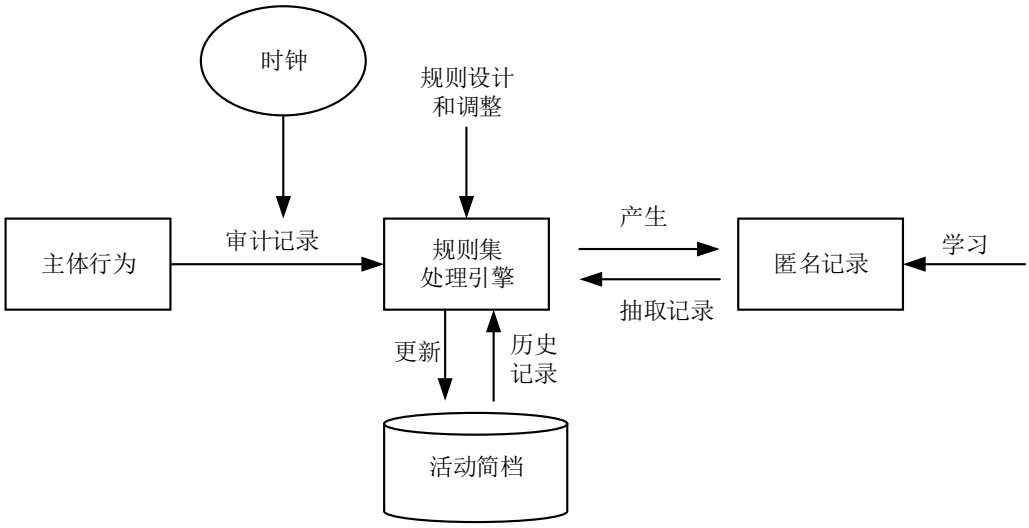


图 2-2　Denning 通用入侵检测模型

2.2.2　入侵检测系统分类

根据用于检测异常活动的输入数据源进行分类，入侵检测系统被分成基于网络的 IDS（NIDS）和基于主机的 IDS（HIDS）两类。根据识别入侵的方法，IDS 系统被分成基于签名的入侵检测系统（SIDS）和基于异常的入侵检测系统（AIDS）两类。IDS 分类框架如图 2-3 所示。

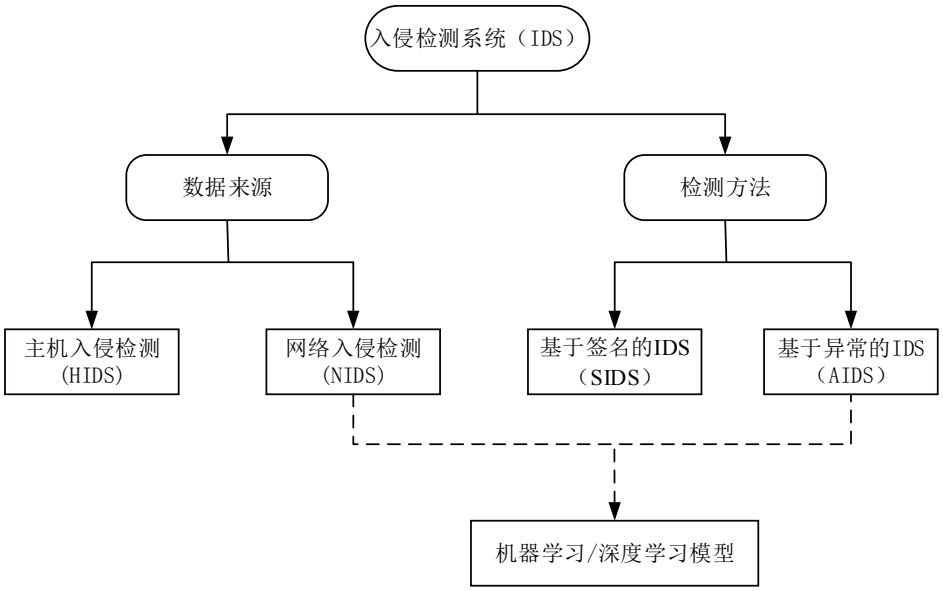


图 2-3　IDS 分类框架图

基于网络的入侵检测系统可以监视连接到一个网络中的所有计算机以及流量。

该系统能够实时监控外部的恶意行为，并在早期阶段检测和阻止外部威胁进一步传播到其他计算机系统。一般采用模式匹配、统计分析、机器学习等方法检测异常行为，其检测速度快、部署简单，同时存在难以定位入侵者和无法直接检测主机入侵的弊端。基于主机的入侵检测系统检查来自主机系统和审计源的数据，如操作系统、防火墙日志、数据库日志等。HIDS 可以检测到不涉及网络流量的内部攻击，检测粒度细，主要缺点是占用主机和服务器，增加额外开销，可移植性差。

基于签名的入侵检测系统，也被称为基于知识的检测或误用检测[47]。其通过模式匹配识别攻击行为，当入侵签名与签名数据库中存在的入侵签名相匹配时，则认为有攻击行为。SIDS 对已知类型的入侵表现出极好的检测精度，但是无法检测零日攻击，因为数据库中没有可匹配的签名。随着零日攻击率的增加，基于异常的入侵检测系统逐步发展壮大。基于异常的入侵检测系统（AIDS）是基于明确定义的正常行为来进行检测，任何偏离正常行为基线的情况都被视为异常[48]。AIDS 的主要优点是它能够检测到未知的和新的攻击，缺点是难以明确正常和异常特征之间的界限。

2.3　用户和实体行为分析技术

2.3.1　UEBA 架构与技术

UEBA 系统是一个包含数据中心层、算法分析层和场景应用层的完整系统，它支持传统的引擎的同时也支持基线分析以及群组分析、集成学习、强化学习、异常检测、知识图谱等人工智能引擎。除此之外，其还会使用如特征工程、身份识别等技术。UEBA 系统架构通过采集各种数据如安全类数据、资产结构类数据、流量行为数据等，分析用户行为的同时也能分析应用实体的行为，采用机器学习和异常检测的技术来检测异常行为。UEBA 的典型系统架构如图 2-4 所示。

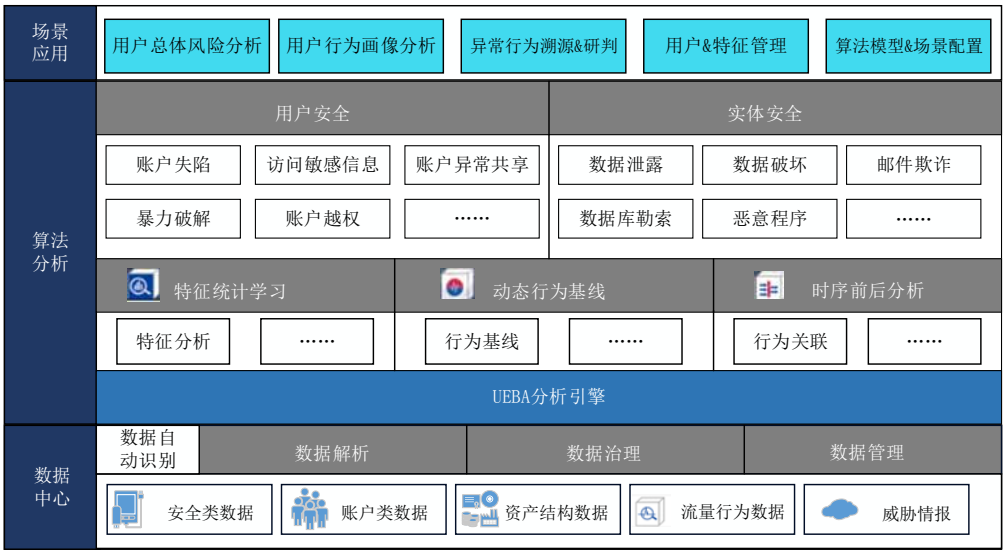


图 2-4　UEBA 系统架构图

UEBA 聚焦用户和实体，以机器学习驱动行为分析，在全时空的上下文中进行异常检测，从而发现恶意行为和恶意用户。其具有发现未知、提高风险可视性、降低成本和提升能效的优点。

2.3.2 UEBA 现状与应用

UEBA 技术日渐成熟，并在各中大型企业中得到应用，国内外多个安全公司自主研发了基于 UEBA 的安全产品，GURUCUL[49]致力于检测签名、规则之外的威胁，计算风险评分并终止威胁行为的发生，Exabeam[50]采用各种日志记录和专家规则来预测事件发生的顺序 ；国内观安、启明星辰等也有类似产品用来检测数据泄露等员工异常行为。作为安全信息与事件管理（SIEM）的有效补充，UEBA 能够识别基于日志方案无法识别的异常，随着技术方案的演进，UEBA、SIEM 和安全编排自动化响应（SOAR）逐步融合，SOAR 侧重于识别后对威胁的处理，UEBA 与 SIME 区别如表 2-1 所示。

表 2-1　UEBA 和 SIME 的区别

名称	分析视角	方法
SIEM	流量和请求	专家经验、已知规则、人为设定的阈值
UEBA	用户行为	关联分析、行为建模、异常检测

UEBA 技术有着广泛的应用前景，在医疗行业、金融行业、能源行业以及政务行业都有应用，为其解决数字化推进过程中带来的网络安全问题如数据泄露、金融欺诈。UEBA 技术可以解决但不限于的网络安全问题有：（1）内部威胁检测。选取相关特征，构建用户正常行为基线，通过基线判断内部人员是否进行恶意操作。（2）入侵检测。通过对主机、应用程序、网络流量等实体进行行为分析和判断，从而发现异常入侵行为。（3）风险定级排序。使用基线模型、威胁模型以及报警信息，对用户和实体的行为进行时间排序，进行风险聚合。（4）远程办公安全。UEBA 收集远程办公使用的 VPN 日志，从而构建包括登录时间、时长、地点、网络行为等特征的行为基线，对比同角色人员发现可疑账号。

2.4 威胁检测算法

威胁检测算法是指用于识别和捕获计算机系统中的恶意行为和攻击的算法。本章对研究中涉及到的威胁检测算法包括深度学习、迁移学习和小样本学习的理论知识进行阐述，为后续的威胁检测模型的构建建立基础。

2.4.1 深度学习

深度学习是一种机器学习算法，通过多层次的神经网络来学习和识别数据的特征，以实现分类、回归、聚类等任务。随着数据量得不断增加，传统机器学习性能

的提升受到局限，深度学习模型因而得到发展，用于解决复杂的应用问题，其在威胁检测领域也有广泛应用。深度学习算法中常用的模型包括卷积神经网络和循环神经网络等。

卷积神经网络（Convolutional Neural Network, CNN）是至少在一层网络层中使用卷积运算来替代矩阵乘法运算的多层网络结构，其主要包含输入层卷积层、池化层和输出层，CNN 的完整结构如图 2-5 所示。针对卷积操作其核心为步长，通过变换步长，能够得到不同类型的卷积操作；针对池化，其核心为滤波器的尺寸，其操作与卷积类似，在保持原有特征的基础上缩小数组的维度，即降维。CNN 可以扩展到高维度的图像，甚至是大小可变的图像，其经典模型有 Alexnet、VGG、GoogleNet 等。

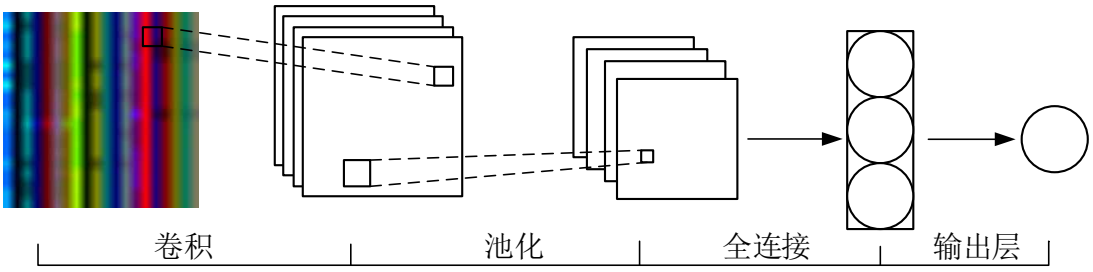


图 2-5　CNN 的完整网络结构

循环神经网络（Recurrent Neural Network, RNN）是一类能够对序列数据进行处理的神经网络，高效学习序列中的非线性特征，能够对序列中之前的信息进行建模，因此很适合处理威胁检测中基于时间序列的异常检测问题。RNN 将上一个网络的输出保存在记忆单元中，之后将记忆单元中的信息与下一次的输入一起输入的网络中，即网络的输出取决于当前的输入和记忆。其计算方法如公式（2-1）和公式（2-2）所示。

$$O_t = g\left(V \cdot S_t\right) \tag{2-1}$$

$$S_t = f\left(U \cdot X_t + W \cdot S_{t-1}\right) \tag{2-2}$$

其中公式（2-1）是输出层的计算公式，其是一个全连接层，V 是输出层的权重矩阵，g 是激活函数，公式（2-2）是隐藏层的计算公式，即循环层。U 是输入 X 的权重矩阵，W 是上一次的值作为这一次的输入的权重矩阵，f 是激活函数。

由于激活函数的局限性，使得 RNN 无法学习更长序列的数据，故提出 RNN 的变体来学习更长的序列特征，如长短时记忆网络（Long Short Term Memory，LSTM）、GRU 等。LSTM 通过遗忘门、输入门、输出门的特殊设计来灵活选择状态的更新和是否全部参与输入。GRU 也是一种带有门控制机制的 RNN，只有两个门控单元，相较于 LSTM 的三个门控单元，参数较少，计算速度较快。

2.4.2 迁移学习

迁移学习属于机器学习的研究领域，通过知识迁移，解决目标领域的新问题并取得较好的学习效果。一般来说，源域的数据较为充足，能够很好地训练模型，而目标领域数据量较小。其与传统机器学习的区别是它利用之前任务中学出到的数据特征或模型参数来辅助新应用中的训练过程，从而得到新的训练模型。迁移学习的优势是可以加快训练速度，减少所需得训练数据量，并提高模型得准确性。该技术在自然语言处理、文本识别、计算机视觉等领域都有应用，其不限于特定领域，本文将其应用在网络入侵检测领域。可以从迁移定义、迁移方法或领域数据的维度对迁移学习进行分类，其分类如图 2-6 所示。

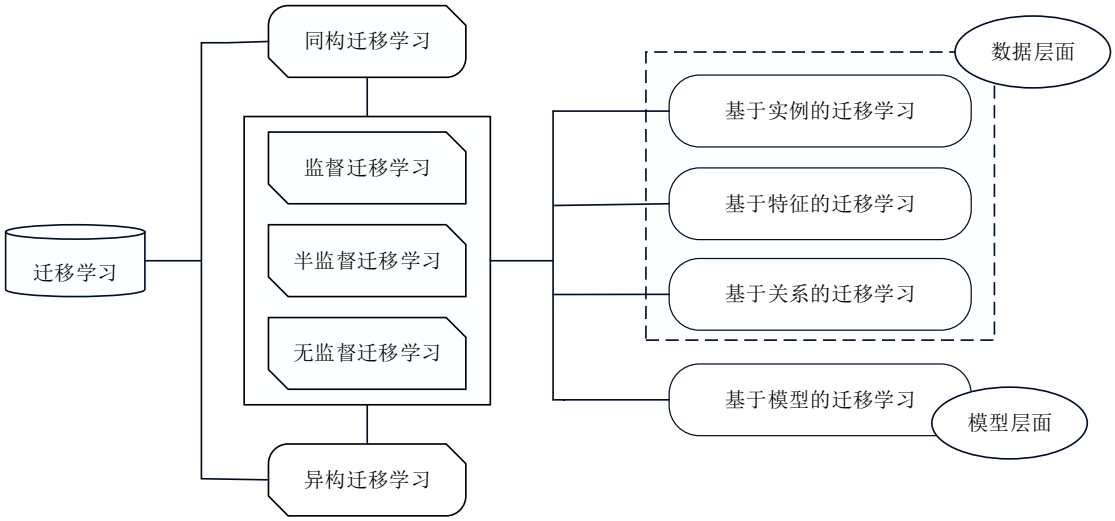


图 2-6　迁移学习分类

根据迁移方法将迁移学习分为基于实例[51]、基于特征[52]、基于关系[53]和基于模型[54]四大类，而前三类是基于数据层面的，最后一个基于模型层面的，本研究采用是基于模型的迁移学习，通过复用模型中的参数，结合优化的方法来解决现有的问题。

2.4.3 小样本学习

小样本学习是机器学习的一个分支，用于研究和解决标注样本数量少和模型可移植性差的问题。当出现新的类别时，不需要改变训练好的原有模型，只通过极少的数据即可以学习新类别的特征[55]。小样本学习是首先用较大数量的样本训练一个能区别“异同”的模型，到了测试阶段，支持集中含有训练数据集中不包含的数据类型，让模型判断当前的数据样本是支持集中的哪一类。根据实现方式的不同，将小样本学习分为基于模型、基于度量方法和基于优化三种类型。

基于模型的小样本学习算法通过建立的元模型获取经验知识，然后再通过经验知识去评估分类任务，减少对数据的依赖性。其中一种常见的基于模型的小样本学

习算法是贝叶斯方法，通过先验概率和后验概率的推导来进行分类。在小样本学习中，由于数据量较小，先验概率的影响会更加显著。因此，利用领域知识或先验信息可以更准确地推导出先验概率，从而提高分类的准确率。

基于度量的小样本学习算法通过计算不同样本之间的距离或相似度，将相似的样本归为一类。其可以采用固定距离度量如欧氏距离[56]，也可以使用非固定距离度量[57]如使用 Sigmoid 函数计算距离。其具有较高的准确性和泛化能力，并且可以在数据集较小的情况下进行有效的学习。

基于优化的小样本学习通过系统的学习初始化，使得训练以好的初始化开始，其在图像处理领域的应用仍较少。常见方法为原型网络，它学习一个函数，将不同类别的例子映射到一个共同的表示空间，在这个空间里，类的原型被计算为每个类的例子的平均值，而新的例子则根据它们与这些原型的接近程度进行分类。

2.5　本章小结

本章针对安全威胁与相关技术展开介绍，首先阐述了内部威胁的定义、分类和检测模型，对内部威胁这一研究内容进行深入了解，其次是对外部威胁检测相关技术之一入侵检测进行描述，然后对用户和实体行为分析技术进行阐述。最后，对威胁检测相关算法进行论述，其是威胁检测技术研究的核心部分。

第 3 章　基于用户行为分析的内部威胁检测模型

信息被内部人员非法泄露、复制、篡改，给企业、政府带来巨额的经济损失，为了防止信息被内部人员非法窃取，本章提出了基于 LSTM-Attention 用户行为分析算法的内部威胁检测模型—ITDBLA。首先，提取用户的行为序列、用户行为特征、角色行为特征和心理数据描述用户的日常活动；其次，使用长短期记忆网络和注意力机制学习用户的行为模式，并计算真实行为与预测行为之间的偏差；最后，使用多层感知机根据这些偏差进行综合决策来识别异常行为。在 CERT 内部威胁数据集上进行实验，实验结果表明，ITDBLA 模型具有较强的学习用户活动模式和检测异常行为的能力。

3.1　总体框架

ITDBLA 模型由数据预处理模块、基于 LSTM-Attention 的用户行为分析模块两个模块构成。其中，后者包括 4 个模型：行为特征模型、角色特征模型、行为序列模型和综合决策模型，而角色特征是同一角色下行为特征的均值，因此采用相同的模型对角色特征和行为特征进行建模。ITDBLA 模型的工作流程如图 3-1 所示。

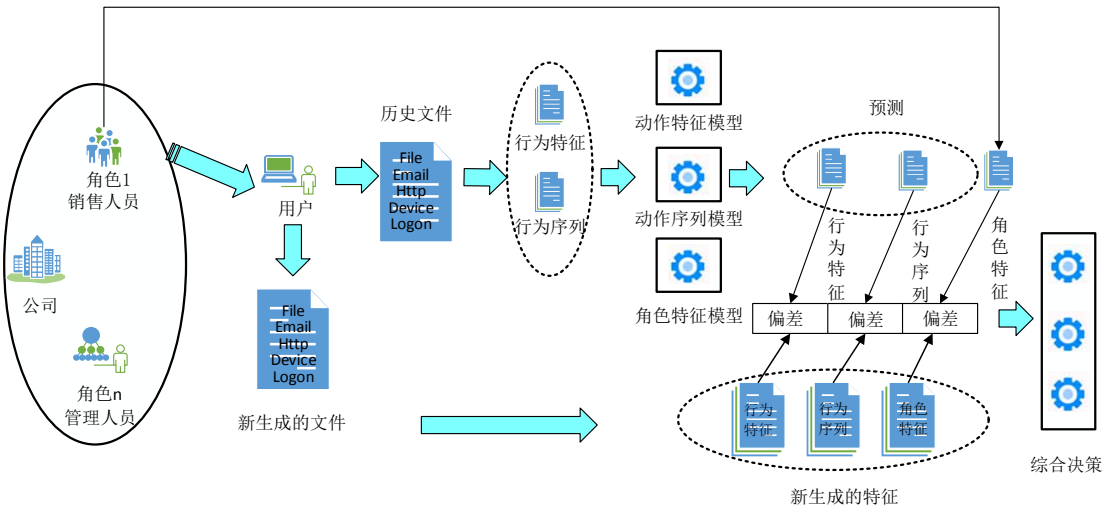


图 3-1　TDBLA 模型的工作流程

在数据预处理模块中，首先对数据进行清理、集成、编码；然后，进行特征提取，该模型的设计考虑同一角色下用户的工作相似，可以通过其日常工作等行为数据从工作角色中提取角色特征。因此，将组织中的员工根据角色进行分组，如技术人员、人力资源、生产线工人等。最后，分别生成模型所需的行为序列、行为特征、角色特征和心理数据。

在基于 LSTM-Attention 的用户行为分析模块中，该模型考虑 3 种类型特征，分别为行为特征、行为序列和角色特征。根据这 3 种类型的特征设计 3 个模型，并通

过历史数据学习用户的正常行为模式，对下一个状态进行预测。当内部人员发生异常行为时，其真实行为会偏离预测的行为模式，模型计算行为之间的偏差。用户行为分析模块的具体工作流程为：首先，使用 LSTM 对用户行为和角色行为进行分析，分别建立用户行为和角色行为模型；其次，使用注意力机制对用户行为序列进行分析，并建立用户行为序列模型；最后，将 3 种类型的偏差和心理数据输入训练的综合决策模型 MLP 中，实现用户行为的异常检测，当模型表现出较高的均方误差时，说明该用户在此时间段发生异常行为。

3.2 基于 LSTM-Attention 的内部威胁检测算法

3.2.1 数据预处理

数据预处理的目的是从多源异构的日志文件中提取相关信息，并将其转换为规范化表示，当异常行为发生时，深度学习算法可以从中检测偏差。预处理阶段的主要流程为：首先，对每个日志文件进行数据清理，包括错误信息的修改、多余字段的删除和缺失值的填充。其次，将多个日志文件进行数据集成，数据集中的每个日志文件都是所有用户的操作，需要按照用户名进行信息提取，然后将提取到的信息进行集成并按照时间先后顺序排序，形成用户行为序列特征。由于用户一天的行为相对于小时这一时间粒度能更好地体现用户的行为习惯，因此，本实验的用户行为以天为单位进行描述。最后，对提取的数据进行编码，数值型变量可以直接作为深度学习的输入，类别型数据不能直接作为深度学习算法的输入，需要对类别型数据进行编码如{‘logon’:1, ‘Connect’:2, ‘Disconnect’:3, ‘http’:4, ‘email’:5, ‘logoff’:6}。

合适的特征对于捕获真实行为和模型预测的用户行为之间的偏差有重要作用，这些偏差可以表示异常行为，并且可以描述威胁用户的可疑程度。识别异常用户需要对用户的各种行为进行综合分析，本实验从不同来源的日志文件（如 logon.csv、email.csv 和 http.csv 等文件）中提取行为特征、行为序列、心理数据和角色特征，特征描述如表 3-1 所示。

表 3-1　实验特征描述

特征	数据	描述
行为特征	logon.csv	登录总次数、登录总时长 第一次/最后一次登录时间
	email.csv	工作/非工作时间发送的邮件数量 接收内部、外部电子邮件数量 邮件大小、附件数量
	http.csv	浏览网站总时间 浏览不同类别网站的时间

(续表)

特征	数据	描述
行为特征	device.csv	USB 连接总数 工作/非工作时间连接 USB 数量
	file.csv	访问的文件数量 访问的不同格式的文件数量
序列特征	一段时间内的行为序列的序列数据	logon:1　Connect:2 Disconnect:3　http:4 email:5　logoff:6
角色特征	角色下所有用户的共同行为特征	同一角色员工的行为特征的平均值
心理数据	大五人格特质	O 开放性、C 责任性、 E 外倾性、A 宜人性、N 情绪性

3.2.2　行为和角色特征模型

ITDBLA 模型的行为特征模型与角色特征模型是基于 LSTM 网络构建的。LSTM 由文献[58]提出，是循环神经网络的一个变体，与单一的 tanh 循环结构不同，LSTM 是一种拥有 3 个门结构的特殊网络结构。LSTM 可以有效解决简单循环神经网络的梯度爆炸或消失问题，已广泛应用于自然语言处理和语音识别。工作日行为和非工作日行为的正常情况有本质区别，由于实验条件限制，本实验只保留了工作日行为，没有做非工作日行为的模型。具体来说，LSTM 的任务是预测序列中的下一个向量，使用前 4 天的行为特征预测第 5 天的特征。LSTM 网络的循环单元结构如图 3-2 所示，其计算过程为：首先，利用上一时刻的外部状态 h_t 和当前时刻的输入 x_t，计算出 3 个门以及候选状态；然后，结合遗忘门 f_t 和输入门 i_t 更新记忆单元 c_t；最后，结合输出门 o_t 将内部状态的信息传递给外部状态 h_t。

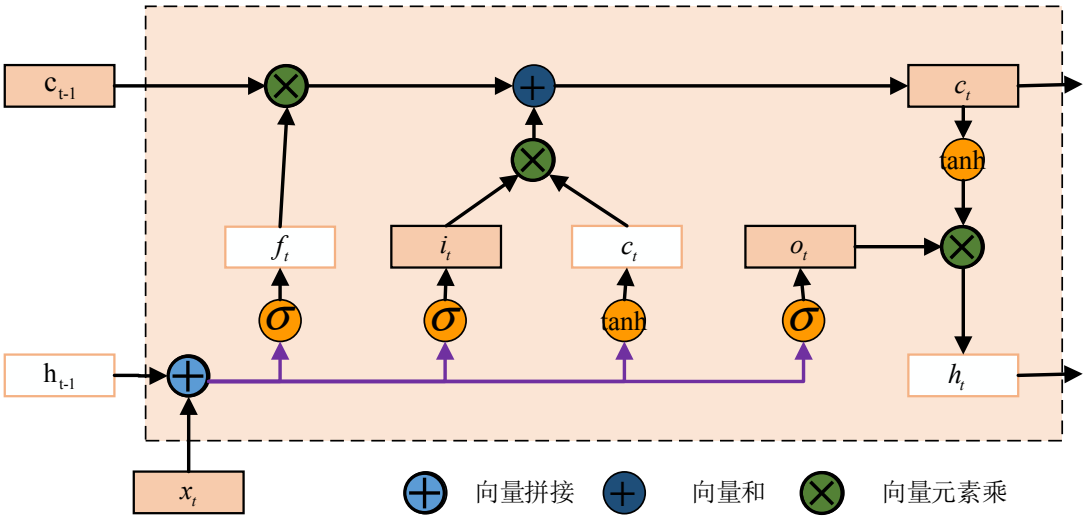


图 3-2　LSTM 网络的循环单元结构

3 个门的计算过程如公式（3-1）、公式（3-2）、公式（3-3）所示。

$$i_t = \sigma\left(W_i x_i + U_i h_{t-1} + b_i\right) \tag{3-1}$$

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{3-2}$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{3-3}$$

其中，$\sigma(\cdot)$为 Logistic 函数，其输出区间为（0,1），x_t为当前时刻的输入，h_{t-1}为上一时刻的外部状态。

通过 LSTM 循环单元，整个网络可以建立较长距离的时序依赖关系，如公式（3-4）、公式（3-5）、公式（3-6）所示。

$$\begin{bmatrix} \text{illegible} \\ o_t \\ i_t \\ f_t \end{bmatrix} = \begin{bmatrix} tanh \\ \sigma \\ \sigma \\ \sigma \end{bmatrix} \left(W \begin{bmatrix} x_t \\ h_{t\text{-}1} \end{bmatrix} + b \right) \tag{3-4}$$

$$c_t = f_t \text{ e } c_{t-1} + i_t \text{ e illegible} \tag{3-5}$$

$$h_t = o_t \text{ e } tanh\left(c_t\right) \tag{3-6}$$

其中，$x_t \in R^l$为当前时刻的输入，$W \in R^{4d(d+e)}$和$b \in R^{4d}$为网络参数。

3.2.3 行为序列模型

ITDBLA 模型的行为序列模块是搭建在加入注意力机制的全连接网络上。注意力机制最早应用于图像领域，2014 年 Google Mind 团队提出了在 RNN 模型上使用注意力机制进行图像分类，取得了很好的分类效果。注意力机制的核心是分配权重，其是一种思想，本身不依赖于任何框架，可以嵌入到神经网络中，用来自动学习和计算输入数据对输出数据的贡献大小。训练加入注意力机制的全连接网络学习用户正常的行为序列，并根据历史记录预测下一个状态的行为序列，含注意力机制的全连接网络结构如图 3-3 所示。

注意力层由一个 Dense 层、Multiply 操作和一个 Softmax 激活层组成。其具体的计算过程分为三个阶段。第一个阶段，通过采用不同的计算机制计算查询向量与键向量的相似性或相关性，本实验采用的计算机制为向量点积，通过计算得到每个 Key 对应 Value 的权重系数，计算公式如（3-7）所示。

$$Similarity\left(Q,K\right) = Q \cdot K \tag{3-7}$$

其中，Q为注意力机制中的 Query，即查询向量，K为注意力机制中的 Source 中的 Key，$Similarity\left(Q,K\right)$值越大，表明这两个向量相关性越大，从而该 Key 值对应的 Value 的权重越大。

第二阶段引入 Softmax 函数对相似性得分进行归一化处理，将数值转化到[0,1]范围内，通过权重系数找出对 Query 向量重要的 Key 值。采用的计算公式如（3-8）

所示。

$$a_i = Softmax(Sim_i) = \frac{e^{Sim_i}}{\sum_{j=1}^{L_x} e^{Sim_j}} \tag{3-8}$$

第三阶段，根据权重系数对V执行加权求和操作得到*Attention*值，具体操作为权重系数乘以对应的V后相加，计算公式如（3-9）所示。

$$Attention(Q,S) = \sum_{i=1}^{L_x} Similarity(Q,K_i) * V_i \tag{3-9}$$

其中，S为注意力机制中的 Source，V_i为K_i对应的 Value 值。

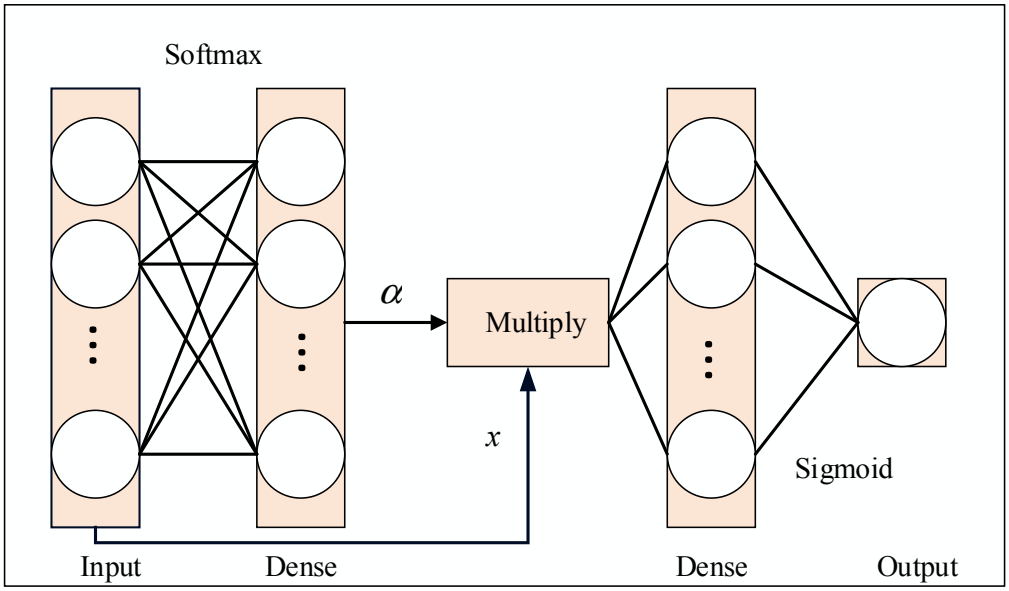


图 3-3　含注意力机制的全连接神经网络结构

在本实验中，使用 N 天的行为序列预测下一个状态的行为序列，每个用户的行为序列长度不同。用户的操作为{登录，网页，网页，网页，驱动器连接，驱动器断开，……，电子邮件，注销}，将其进行编码并输入模型中。

3.2.4　综合决策模型

ITDBLA 模型的最后环节是综合决策，本实验采用多层感知机将上述深度学习模型的结果和用户心理数据进行综合决策，从而实现内部威胁检测。MLP 是由第一层输入层、中间隐藏层和最后输出层组成的神经网络，输入层用向量 X 表示，隐藏层的输出则为 $f(W_1X+b_1)$，其中权重用W_1表示，偏置用b_1表示，本实验函数为*sigmoid*函数，输出层的输出就是 $softmax(W_2X_1+b_2)$，X_1表示隐藏层的输出 $f(W_1X+b_1)$。MLP 整个模型为 $f(x)=G(b_2+W_2(s(b_1+W_1x)))$，其中$G$为*softmax*函数。MLP 主要优势在于其快速解决复杂问题的能力强，常用于解决非线性问题。在 ITDBLA 模型中，需要使用历史数据训练 MLP 来学习这些特征之间的关系。MLP 根据行为序列、行为特征和角色特征的偏差判断是否存在异常行为，类似于执行分类任务。

3.3 实验过程及结果分析

3.3.1 实验数据集

本章在包含用户属性元数据等分类信息的 CERT 内部威胁数据集上进行实验，其由卡内基梅隆大学提出，被广泛用于内部威胁检测方法的研究、开发与测试[59]。CERT 内部威胁数据集的数据收集于一个真实企业，使用各种模型包括主题模型、行为模型和心理测量模型生成，模拟恶意内部人员实施的系统破坏、信息窃取和内部欺诈 3 类攻击行为，包括 1 000 名用户，其中有 70 名内部攻击人员。CERT 内部威胁数据集中包括登录数据、设备数据、HTTP 数据、电子邮件数据和文件数据 5 个不同类别的 csv 文件，CERT 数据集中文件描述如表 3-2 所示。此外，该数据集还提供用户的职位信息、心理数据以及员工月度考勤情况，使得本文可以根据员工个性和工作角色对用户进行分析。

表 3-2　CERT 数据集的文件描述

文件名	描述
LDAP	描述每个模拟用户的本体（角色、电子邮件部门、主管等）LDAP 文件
device.csv	可移动设备(如 USB 硬盘)的连接和断开
email.csv	用户邮件的日志，包括邮件的接收、发送和大小等
file.csv	文件访问活动，包括文件打开、复制、写入、删除
http.csv	用户网页浏览、上传、下载
logon.csv	基于登录和注销计算设备的用户活动
psychometric.csv	为 1 000 名模拟用户提供工作满意度变量

3.3.2 评价标准

由于 CERT 内部威胁检测数据集中存在正负类别严重不平衡的现象，且测试数据中的正负样本的分布也可能随着时间的变化而改变，受试者工作特征曲线（Receiver Operating Characteristic Curve, ROC）以及 ROC 曲线下与坐标轴围成的面积（Area Under Curve, AUC）可以很好地消除样本类别不平衡对指标结果产生的影响。ROC 曲线纵坐标是真正例率（True Positive Rate，TPR），横坐标是假正例率（False Positive Rate，FPR）。综合考虑精度和召回率，故本文使用 AUC 作为评价指标，AUC 是位于 ROC 曲线下方的面积，ROC 曲线下方面积越大，即 AUC 越大，表明模型效果越好。

3.3.3　实验环境与参数设置

本实验采用 Keras 建立 LSTM 和注意力机制的网络模型，Keras 是由 Python 编写的高级神经网络 API，能够在 Tensorflow、CNTK 或 Theano 上运行，可以进行深度学习模型的设计、调试、评估、应用和可视化。模型训练和测试使用的计算机配置处理器为 i7-5500U，CPU 2.40GHz，RAM 12GB。在本实验中，采用加权偏差度（the weighted deviation degree，WDD）测量预测值与真实值之间的偏差，使用 Production Line Worker 角色下的用户文件进行测试。本章对训练集的顺序进行随机打乱，以此提高模型的鲁棒性。本实验中的 LSTM 模型由两个 LSTM 层组成，各层的单元数量分别为 100 个和 120 个，LSTM 层后有一个 tanh 激活层、一个具有 37 个单元的 Dense 层和一个 relu 激活层。 Attention 模型是在全连接网络上搭建的注意力机制，注意力层由一个 Dense 层、Multiply 操作和一个 Softmax 激活层组成，模型的超参数设置如表 3-3 所示。

表 3-3　模型的超参数设置

模型	网络层	超参数设置
行为特征模型-LSTM	Input	*Dim* = 148
	Reshape	*Dim* = (4,37)
	LSTM	Units = 100
	Activation	*Function* = *tanh*
	LSTM	*Units* = 120
	Activation	*Function* = *tanh*
	Dense	*Dim* = 37
	Activation	*Function* = *relu*
行为序列模型-Attention	Input	*Dim* = 132
	Attention_vec: Dense	*Dim* = 132
	Activation	*Function* = *Softmax*
	Multiply	*Dim* =[(None,132), (None,132)]
	Dense	*Dim* = 33
	Activation	*Function* = *Softmax*

3.3.4　结果分析

本实验在训练模型的过程中使用 4 天的特征来预测第五天的特征，并采用加权偏差度 WDD 计算真实特征与预测之间的偏差，根据一个加权对平方误差进行线性加

权。WDD 的计算过程如公式（3-10）所示。为了使模型可以学习用户正常的行为模式，训练 LSTM 和注意力机制的数据都是良性的样本。

$$WDD = \frac{1}{|V|}\sum_{\forall y \in V} w\left(y - \hat{y}\right)^2 \tag{3-10}$$

其中，V 是真实特征中所有特征的集合，y 是属于 V 的单个特征，$\hat{y}$ 是与 y 相同的特征，但属于预测的特征，w 是根据特征 y 专门设计的值。

分析 LSTM、注意力机制和 MLP 在 CERT r4.2 上的结果，具体如下。

(1) 用户的行为特征的结果分析 用户动作特征是多样的，并且这些不同的特征之间有潜在联系。本文利用以 5 天为一个时间单位的正常用户特征训练 LSTM，其中前 4 天的特征用于预测第五天的数据，真实的第五天数据与来自 LSTM 的预测数据之间的误差在训练阶段进行计算和优化。需要注意的是，同一角色下所有用户定义的特征是相同的，LSTM 模型也是相同的，所有用户共享同一个 LSTM 模型，用户各自保存自己的参数。行为特征的 WDD 如图 3-4 所示，前 200 天的偏差基本为 0~2，200 天之后的偏差大幅增加，与之前的偏差显著不同。由图 3-4 可以看到，200 天后异常偏差很多，与实际情况相对应，说明 LSTM 网络具有较强的学习用户行为模式的能力。

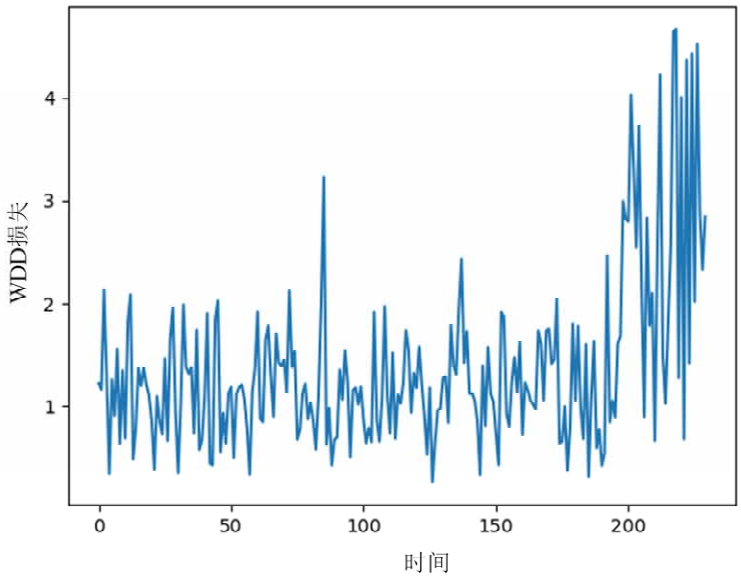


图 3-4 行为特征的 WDD

(2) 用户行为序列的结果分析 如 3.2.3 所述，加入 Attention 的全连接网络被用来训练学习用户行为序列的正常模式，由于时间和数据的限制，使用以 5 天为一个时间单位的行为序列训练 Attention 网络模型，前 4 天作为已知数据预测第 5 天的数据，然后与第 5 天的实际数据进行比较，并对 Attention 网络模型进行误差计算和优化。行为序列的 WDD 如图 3-5 所示，图 3-5 中前 40 天的测试数据与前 160 天的训练数据具有相似的分布，损失范围为 0~4，说明 Attention 网络模型已经很好地学习到了用户的行为序列。从图 3-5 中发现，用户在 200

天附近有一些异常行为，导致预测与实际序列之间的异常偏差变大。根据上述获得的结果和分析表明，注意力机制可以用于学习行为序列的正常模式，并且在实际应用中具有重要意义，但是要注意过拟合的情况。

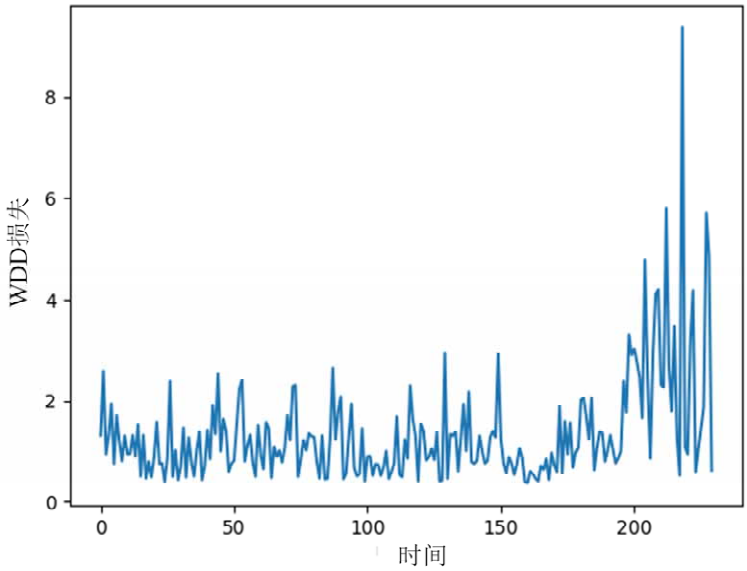


图 3-5　行为序列的 WDD

(3) MLP 的综合决策的结果分析　ITDBLA 模型从行为序列、行为特征和角色特征 3 个角度获得真实与预测之间的偏差，正常和异常数据的偏差分布如图 3-6 所示，可以发现正常点和异常点是可分离的，虽然存在一定的假阳性和假阴性，但这 3 个特征在很大程度上可以反映用户的异常行为。为了在异常行为发生时更准确地发出警报，本实验使用 MLP 学习 3 个偏差之间的关系，从而确定用户在某天是否有异常行为发生。

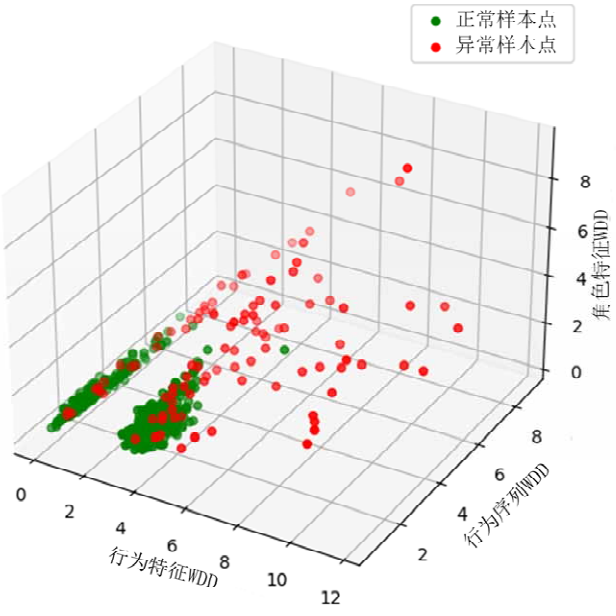


图 3-6　正常和异常数据的偏差分布

ITDBLA 和其基线模型的实验结果对比如表 3-4 所示，ITDBLA 和其他模型的 AUC 分数、ITDBLA 和 SUS 模型的 ROC 曲线分别如图 3-7 和图 3-8 所示。表 3-4 和图 3-8 显示了 ITDBLA 和文献[60]中 SUS 的实验结果对比，表明 ITDBLA 准确率和 AUC 分数上取得了很好的结果，不足之处是误报率有所提升，未来工作的重点就是降低误报率，增加可解释性。在 ROC 曲线上，SUS 和 ITDBLA 的轨迹相似，但是 ITDBLA 的 ROC 曲线下方的面积达 0.964，明显优于 SUS。本文还与其他多个近年来工作在 CERT r4.2 数据集上的内部威胁检测模型进行了 AUC 分数比较，ITDBLA 模型的 AUC 分数明显均优于其他模型，这也是本文提出 ITDBLA 模型有效性的证明，证明其具有很强的异常检测能力。

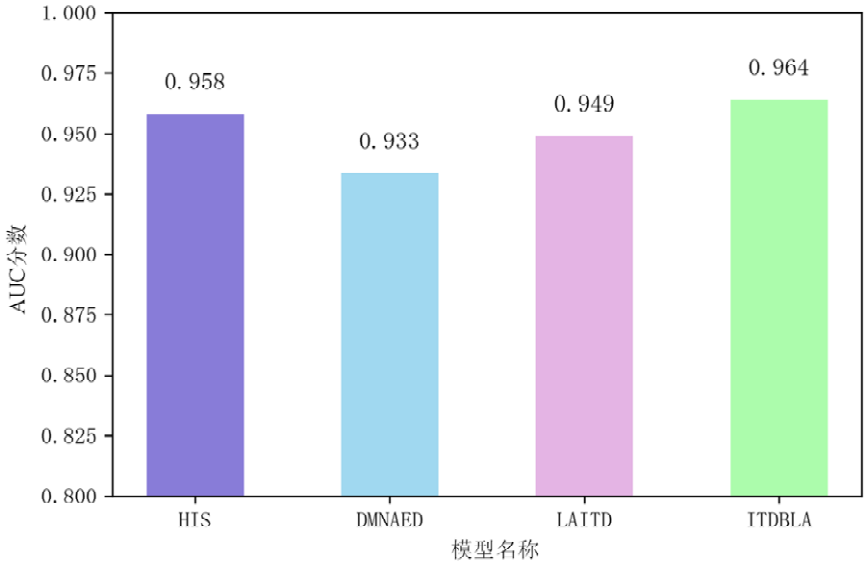


图 3-7　ITDBLA 和其他模型的 AUC 分数

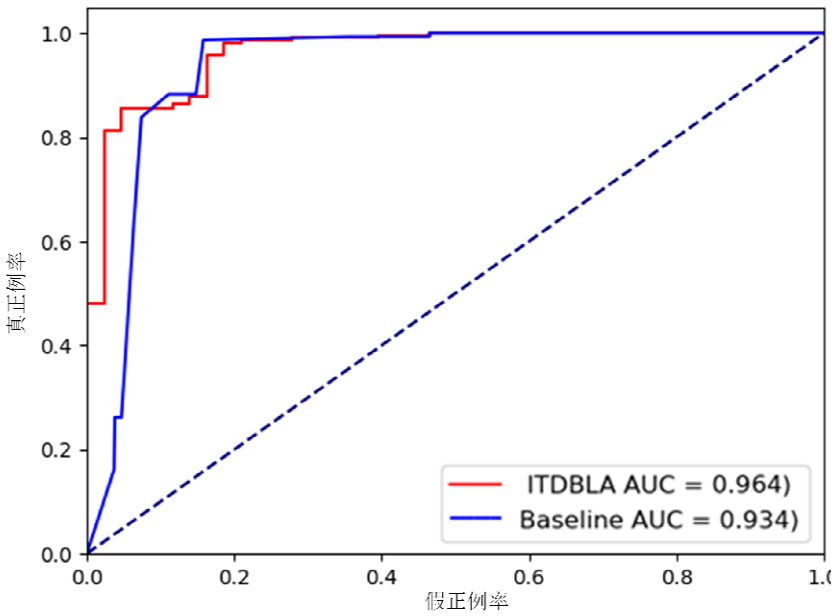


图 3-8　ITDBLA 和 SUS 模型的 ROC 曲线

表 3-4　ITDBLA 和其基线模型的实验结果对比

模型	准确率	AUC 分数	TPR	FPR
ITDBLA	0.980	0.964	0.987	0.281
SUS	0.940	0.934	0.965	0.222

3.4　本章小结

本章提出了一种基于 LSTM-Attention 用户行为分析的内部威胁检测模型 ITDBLA，充分利用 LSTM 和注意力机制能够很好地处理长序列的优势，从行为特征、行为序列、角色特征和心理数据等多个角度对内部人员的日常行为进行全方位建模，学习用户的行为模式，MLP 利用多种特征的偏差执行综合决策，从而实现异常检测。将本章提出的 ITDBLA 模型与其他模型进行对比分析表明，ITDBLA 模型的异常检测能力优于单一机器学习模型和绝大多数的融合模型。

第 4 章　基于流量行为分析的网络入侵检测模型

针对真实网络环境下入侵检测训练样本不足和样本类别不平衡的问题，本章提出了一种基于迁移学习和集成学习的入侵检测模型 TL-CNN-IDS。首先，采用 IG-FCBF 特征工程进行预处理并将获得的数据集转换为适合 CNN 模型输入的图像形式；其次，选择 VGG16、Inception 和 Xception 三种 CNN 模型作为迁移学习的基础学习模型，并采用 Tree-Structured Parzen Estimator(TPE)算法在目标数据集进行参数优化；最后采用置信度平均的集成学习方法对优化后的三个 CNN 模型进行集成，进一步提高模型性能，并利用 CICIDS2017 和 NSL-KDD 数据集对模型性能进行评估。

4.1　总体框架

本章设计了基于 IG-FCBF 特征工程与 CNN 迁移学习的模型用于检测企业网络中的各种攻击类型。图 4-1 展示了本章所提出的网络入侵检测模型 TL-CNN-IDS 的工作流程。首先，对数据集进行数据预处理并使用分位数变化方法将表格数据转换为彩色图像，详细的过程在数据预处理部分进行描述；然后，使用 VGG16、Inception 和 Xception 三种 CNN 模型作为基础模型训练由上一步生成的图像数据集，并使用贝叶斯超参数优化方法 Tree Parzen Estimator（BO-TPE）算法进行参数寻优；最后，采用置信平均集成策略构建集成模型进行最终的入侵检测。

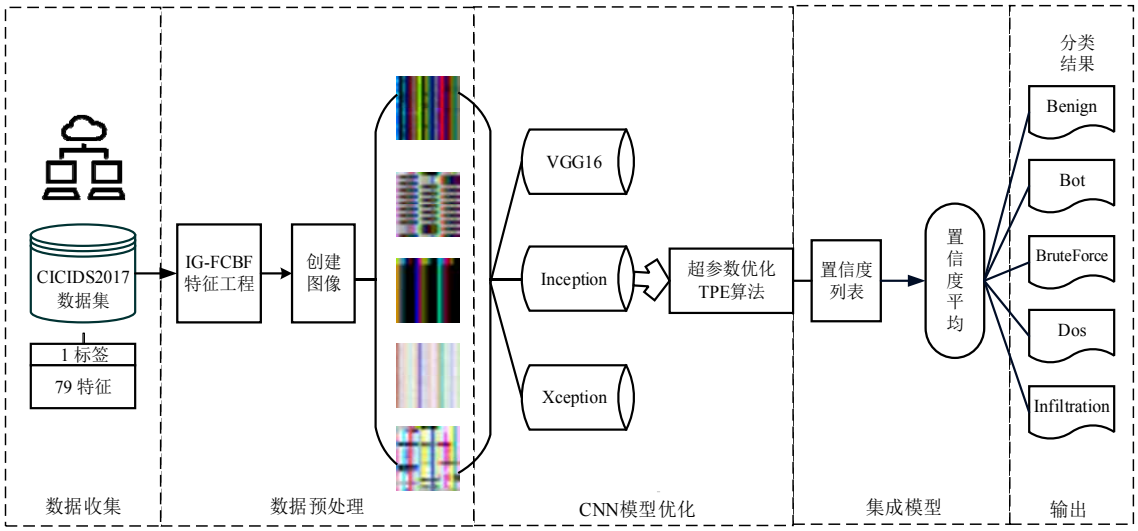


图 4-1　TL-CNN-IDS 网络入侵检测模型工作流程

4.2　基于 IG-FCBF-TL 的网络入侵检测算法

4.2.1　数据预处理

在获取数据后，首先应该对其进行预处理，使其适合提出的 IDS 的输入。由于 CNN 模型在图像集上表现得更好，而网络流量数据集通常是表格数据，因此应将原

始网络数据转换为图像形式。与 CICIDS2017 数据集不同，NSL-KDD 数据集中存在非数值型特征，需首先通过编码的方式将其转换为数值型特征，后续处理则相同。网络流量数据集的数据是庞大和冗余的，受资源和实验设备计算能力的限制，且考虑到现实生活中不会花费大量的时间，采用数量级这么大的网络流量来训练网络模型，因此需要对原数据集进行一些预处理。下面将详细描述数据预处理的完整过程，如图 4-2 所示，以 CICIDS2017 数据集为例。

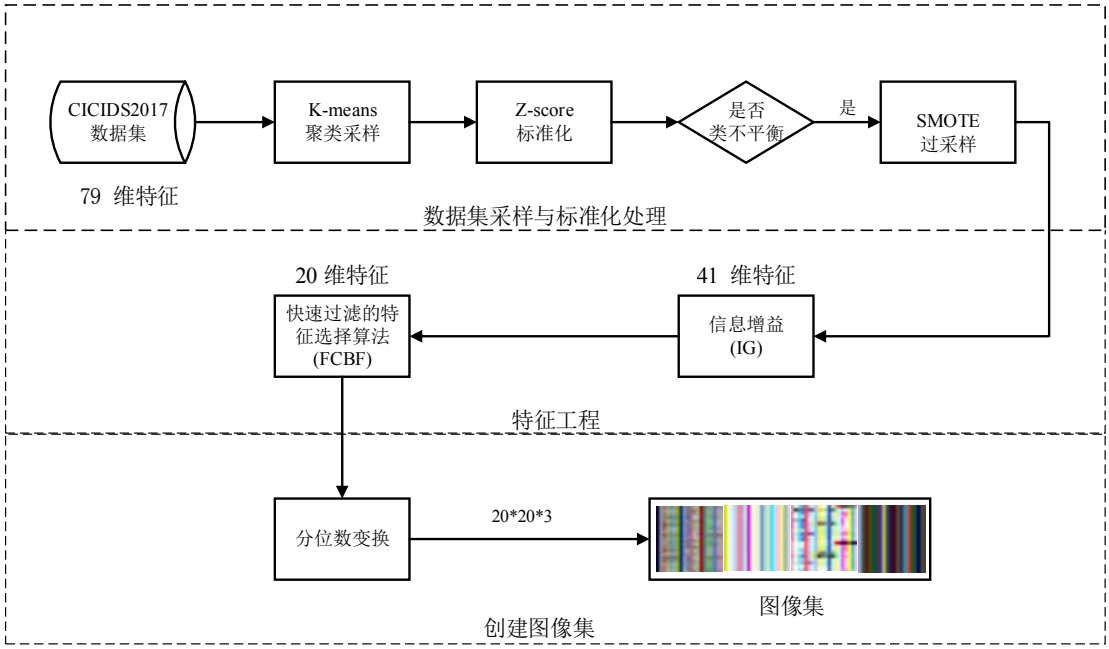


图 4-2　数据预处理过程

(1)　采样与标准化处理　数据预处理的第一阶段是数据采样与标准化处理。首先，对数据集进行 K-means 聚类采样，删除冗余数据，生成一个具有高度代表性的训练子集，从而提高模型训练的效率。K-means 是一种无监督的数据分析技术，其目的是对数据相似度最大的数据进行分组，并减小聚类之间的相似度。在集群中使用距离函数作为相似度的度量函数，从而最大化数据到聚类中心的最短距离来获得数据的相似性，采用的距离函数如公式（4-1）所示。K-means 聚类算法的步骤：首先随机选择 K 个数据作为初始的聚类中心；然后对于每一个数据点计算它与 K 个聚类中心之间的距离（可以使用欧氏距离、曼哈顿距离等），并将该数据点分配给距离最近的聚类中心所属的簇中，将所有数据分为 K 组；最后从 K 组里随机抽取数据样本，由计算资源和数据规模来决定抽取的比例。本实验保留少数类，对多数类进行 K-means 聚类采样，其中 CICIDS2017 数据集对 Benign、Dos 和 Sniffing 进行聚类采样，NSL-KDD 数据集对 Normal、Dos、Probe 进行采样。

$$d_{ik}=\sqrt{\sum_{j=1}^{m}(x_{ij}-c_{kj})^2} \tag{4-1}$$

其中，x_{ij} 是需要聚类的数据，c_{kj} 为聚类的中心，m 是变量的值，d_{ik} 为数据和每

个聚类中心之间的距离。

其次，由于网络流量数据集是一个极度不平衡的数据集，其恶意流量的数量远远小于正常流量，为了提高模型的检测精度 ，该实验采用了 SMOTE 采样的方法对少数类进行过采样从而避免类不平衡问题对模型效果带来的不良效果。最后，对数据进行 Z-score 标准化操作，因为数据集中不同的特征处于不同的数值范围，这使得模型训练有偏差，而 Z-score 标准化可以将各个特征归一化到相同的范围并处理异常值。

(2)　IG-FCBF 特征工程　经过数据采样处理后，生成了具有高度代表性的训练子集 Tr，然后通过适当的特征工程来获得最优的特征，从而实现更准确和更高效的模型学习。在模型训练之前，本实验采用了由信息增益(Information Gain, IG)和快速相关性滤波算法 (Fast Correlation Based Filter, FCBF)组合的特征工程方法，在保留重要特征的同时，去除无关的、冗余的和有噪声的特征。

第一步采用信息增益的方法选择重要的特征。IG 即获得的信息量或熵的变化，可以用来衡量一个特征可以带来多少信息给目标变量。首先计算每个特征的重要性得分，然后从上到下选择重要特征，直到累计重要性达到 90%。经过此过程 CICIDS2017 数据集剩余 49 个特征，NSL-KDD 数据集剩余 29 个特征。假设 T 为目标变量，对于用随机变量 X 表示的每个特征，使用特征 X 的 IG 值表示如公式（4-2）所示。

$$IG(T \mid X) = H(T) - H(T \mid X) \tag{4-2}$$

其中，$H(T)$ 为目标变量 T 的熵，$H(T \mid X)$ 为 T 超过 X 的条件熵。$IG(T \mid X)$ 可以表示特征 X 的重要性，即 X 与目标 T 之间的相关性，值越大则说明特征 X 对目标 T 越重要。

第二步采用 FCBF 特征工程方法进一步消除冗余的特征。基于 IG 的特征工程方法消除了不重要的特征，但是仍然存在许多冗余的特征。快速相关性滤波算法在保留信息特征的同时有效去除冗余特征，可以进一步提高模型的性能和效率。在 FCBF 中通过归一化 IG 值来测量特征之间的相关性，计算方法如公式（4-3）所示。

$$SU(X,Y) = 2\left[\frac{IG(X \mid Y)}{H(X) + H(Y)}\right] \tag{4-3}$$

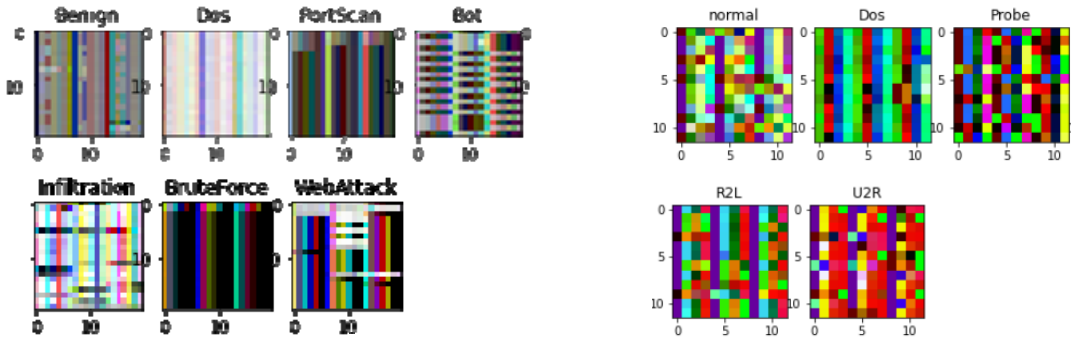
其中，X 和 Y 表示不同的特征，$SU(X,Y)$ 在[0,1]范围内，值 1 表示两个特征 X 和 Y 之间的完美相关性，值 0 表示这两个特征完全独立。重复计算相关性和删除特征的过程，直到特征列表中的每一对特征都没有高度的相关性。最终，CICIDS2017 数据集剩余 20 个特征如表 4-1 所示，NSL-KDD 数据集剩余 13 个特征。

表 4-1　IG-FCBF 特征工程提取 CICIIDS2017 数据集的特征及描述

特征	描述
Flow Duration	流持续时间
Total Fwd Packets	上行包的总数量
Total Backward Packets	下行包的总数量
Flow Bytes/s	流字节率，即每秒传输的数据包字节数
Flow LAT Min	两个流到达的最小时间间隔
FIN Flag Count	带有 FIN 包的数量
SYN Flag Count	带有 SYN 包的数量
ACK Flag Count	带有 ACK 包的数量
URG Flag Count	带有 URG 包的数量
ECE Flag Count	带有 ECE 包的数量
Min_seg_size_forword	上行包的最小分割大小
Fwd Avg Packets/Bulk	上行数据包的平均数量
Fwd Avg Bulk Rate	上行平均块速率
Bwd Avg Packets/Bulk	下行数据包的平均数量
Bwd Avg Bulk Rate	下行平均块速率
Init Win bytes forward	上行初始窗口中发送的字节数
Init Win bytes backward	下行初始窗口中发送的字节数
Destination port	目的端口号
Bwd Packet Length Std	下行数据包的标准大小
Subflow Fwd Bytes	转发子流中的字节数

(3)　创建图像集　由于卷积神经网络模型在图像集上表现得更好，而网络流量数据集是表格数据，因此应将其转换为图像形式。由于图像像素的值范围为 0 到 255，所以该实验采用了分位数归一化[61]将网络流量数据归一化到[0,255]范围内。分位数归一化方法是将多种分布映射到同一种分布，其将特征分布转换为正态分布。因此，大多数变量值都接近于中值，这对处理异常值是有效的。其具体过程如下：首先，对于每个特征，将样本按照从小到大的顺序进行排序。然后，计算每个样本的累积分布函数，即该样本在整个数据集中的累积比例。最后，将每个样本的原始值替换为对应分位数的值。根据特征取值与像素点的对应，CICIDS2017 数据集生成的 20 个重要特征被转换为 20×20×3 的彩色图像，NSL-KDD 转化为 13×13×3 的图像数据集。经过上述图像转换的过程，将最终生成的图像集作为 CNN 模型的输入。图 4-3 展示了图像转换后生成的 CICIDS2017 数据集和 NSL-KDD 数据集中各种攻击类型的

样本。



a) CICIDS2017 数据集　　b) NSL-KDD 数据集

图 4-3　网络流量各种攻击类型样本的可视化

4.2.2　基于 CNN 模型的迁移学习

卷积神经网络是一种被广泛用于图像分类和图像识别的深度学习模型，因为图像可以在不需要额外的特征提取和数据重建的过程直接输入到 CNN 模型中。在提出的 TL-CNN-IDS 中，采用了 VGG16、Inception 和 Xception 深度卷积神经网络作为基础学习模型，它们在大多数图像分类问题中都取得了成功。这些 CNN 模型在 ImageNet 数据集上进行了预训练，并在一般的图像分类上表现出了良好的性能。

VGG16 网络[62]是 2014 年 VGG 团队基于 ImageNet 挑战提出的。其网络结构可以分为 5 个卷积块（用于特征提取）和 3 个全连接层（用于完成分类任务），如图 4-4 所示。每个卷积块包含两到三个卷积层，后面跟着一个最大池化层。每个卷积层之后都跟着一个 Relu 激活函数。在最后一个卷积块之后，VGG16 有两个完全连接的层，每个层都有 4 096 个神经元。最后，网络的输出是一个 1 000 维的向量，对应于 ImageNet 数据集中的 1 000 个类别。VGG16 的主要特点是其深度和卷积核大小的统一性，这种结构的好处是，通过多次重复使用小的卷积核和池化层，可以学习到更多的特征，同时减少了参数数量，从而减少了过拟合的风险。

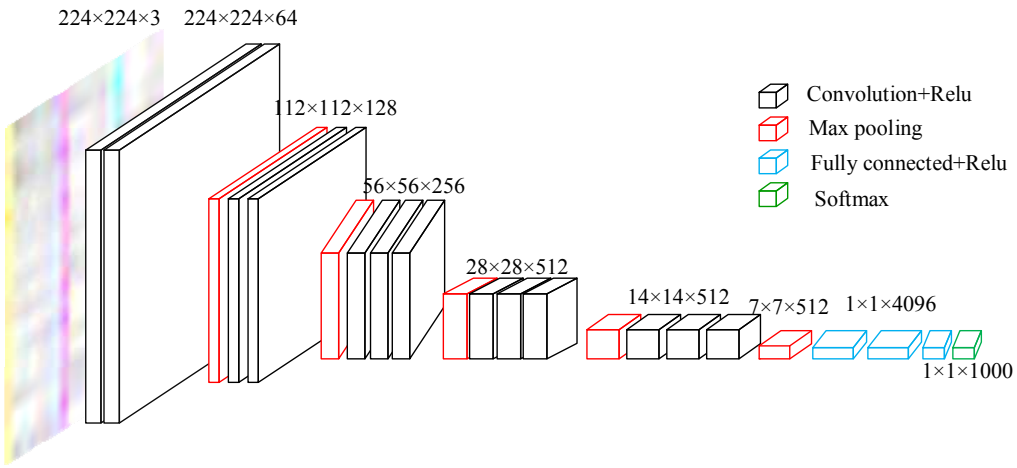


图 4-4　VGG16 网络结构图

Inception 网络[63]是 Google 在 2014 年提出的一种卷积神经网络，旨在解决卷积神经网络中深度加深导致的计算量大和过拟合问题。Inception 网络的基本结构包括三种类型的模块，分别是1×1 卷积模块、3×3 卷积模块和 5×5 卷积模块。其中1×1 卷积模块主要用于降维，即减少特征图的深度；3×3 和 5×5 卷积模块则用于提取特征。此外，Inception 网络还使用了池化层和全连接层。其次 Inception 网络的一个重要改进是引入了“批标准化”技术，即在每个批次的训练数据中对每个特征进行归一化，以加速训练过程并提高模型的精度。

Xception 网络[64]是 Inception 的扩展，基于深度可分离卷积的思想，是一种轻量级的卷积神经网络架构，于 2016 年被提出。Xception 网络的基本思想是将标准的卷积操作分解为两个步骤：深度卷积和逐点卷积。深度卷积通过将每个卷积核分别应用于每个输入通道来计算输出通道中的特征图，然后逐点卷积是将1×1 卷积核应用于深度卷积的输出通道，以产生最终的输出特征图。由于深度可分离卷积可以减少计算量和模型大小，并且能够更好地捕捉特征之间的相关性，使得 Xception 网络相对于传统的卷积神经网络架构在计算性能和准确性上都有所提高，此外，由于使用了更少的参数，Xception 网络还能够在相对较小的数据集上进行有效训练。

对于深度学习模型，迁移学习是将在一个数据集上训练的深度神经网络模型的权值迁移到另一个数据集的过程。迁移学习技术已成功地应用于许多图像处理的任务中，这是由于 CNN 模型底层学习到的特征模式通常适用于许多不同任务的一般模式，而且只有顶层学习到的特征是特定数据集的特定特征，故 CNN 模型的底层可以直接迁移到不同的任务中。为了提高迁移学习的有效性，可以在深度学习模型的迁移学习过程中进行微调。在微调中，预先训练过的模型的大多数层被冻结，也就是说这些层的权重被保留，而一些顶层被解除冻结，在一个新的数据集上重新训练模型。微调使学习模型能够更新预训练模型中的高阶特征，从而更好地适应目标任务或目标数据集。本研究将训练好的 VGG16、Inception 和 Xception 模型迁移到网络入侵检测领域，然后对采用 BO-TPE 算法对模型进行优化，从而使得其在 CICIDS2017 和 NSL-KDD 数据集上表现出较好的性能，实现了训练速度和模型性能的平衡。

4.2.3 基于 BO-TPE 的超参数优化算法

为了更好地将基础模型与本实验的数据集匹配，并进一步提高模型的性能，需要对 CNN 模型的超参数进行调整和优化。超参数优化是一个使用优化技术来调整机器学习或深度学习模型的超参数的自动化过程。CNN 模型有大量需要调优的超参数，这些超参数可以分为模型设计超参数和模型训练超参数。模型设计超参数是在模型设计过程中应设置的超参数。在所提出的 TL 框架中，模型设计的超参数包括冻结层的数量 、学习率和 dropout 率。另一方面，模型训练超参数用于平衡训练速度和模型性能，包括批处理大小、期数和早期停止耐心。基于贝叶斯的超参数优化方法较随

机搜索的优化方法需要更少的验证次数，即意味着更少的时间投入，故本研究采用了改进的贝叶斯超参数优化 TPE 算法。其优化过程主要包括以下步骤，首先定义超参数搜索空间，确定需要优化的超参数及其可能的取值范围，以 CICIDS2017 数据集为例。BO-TPE 算法在基础 CNN 模型需要寻优的参数范围及取值如表 4-2 所示。其次，选择损失函数，本实验选择的损失函数是分类交叉熵。再次，运行 TPE 算法，通过 TPE 算法在搜索空间内不断采样，更新候选超参数，直到达到预设条件。最后确定模型的最优参数。

表 4-2　CNN 模型的超参数设置

CNN 模型	超参数	寻优范围	最优值
VGG16	Num of epochs	[1,20]	10
	Learning rate	[0.001,0.1]	0.03
	Dropout rate	[0.2,0.8]	0.5
	Early stop patience	[2,5]	3
	Frozen layers	[15,18]	17
Inception	Num of epochs	[1,20]	20
	Learning rate	[0.001,0.1]	0.002
	Dropout rate	[0.2,0.8]	0.6
	Early stop patience	[2,5]	3
	Frozen layers	[35,150]	141
Xception	Num of epochs	[1,20]	15
	Learning rate	[0.001,0.1]	0.003
	Dropout rate	[0.2,0.8]	0.3
	Early stop patience	[2,5]	3
	Frozen layers	[50,131]	60

基于贝叶斯的超参数优化即建立目标函数的概率模型，并用它来选择最好的超参数在真实的目标函数中进行评估。BO-TPE（Bayesian Optimization-Tree Parzen Estimator）是一种基于模型的序贯优化方法，是传统贝叶斯优化的一种变体。它将构型空间转换为非参数密度分布。配置空间可以用均匀分布、离散均匀分布和对数均匀分布来表示。因此，TPE 比传统的贝叶斯优化更灵活。引入贝叶斯，根据已有

数据选取一个 loss 的阈值 y*，对于大于阈值或小于阈值的数据 BO-TPE 进行学习并创建$l(x)$和$g(x)$两个概率密度函数，以作为变量的生成模型。TPE 的目标函数由 Parzen 窗口来模拟，BO-TPE 通过最大化比率$l(x)/g(x)$来检测最优超参数值。计算公式如（4-4）所示。

$$p(x|y)=\begin{cases} l(x) & y<y^* \\ g(x) & y\geq y^* \end{cases} \tag{4-4}$$

其中，$p(x|y)$是模型 loss 为y时，超参数为x的条件概率。$l(x)$是通过使用观测值$\{x^{(i)}\}$形成的密度，使相应的损失小于y^*，$g(x)$是使用其余观测值形成的密度。

4.2.4 基于置信平均度的集成模型

集成学习是一种集成多个基础学习模型来构建具有改进性能的集成模型的技术。集成学习广泛应用于数据分析问题，因为多个学习者的集成通常比单个学习者表现更好。在使用迁移学习和微调在网络安全数据集上训练三个 CNN 模型后，将其作为基础模型；然后来构建引入的集成模型。本研究采用的集成策略是置信度平均，置信度平均使集成模型能够检测到不确定的分类结果，并通过使用分类信任来纠正错误分类的样本。每个 CNN 模型的 Softmax 层可以输出一个包含每个类的分类置信度的后验概率表。置信度平均方法结合基础学习模型的分类概率值，找到置信度最高的类，置信度平均方法计算每个类的基础学习者的平均分类概率，然后返回平均置信度最高的类标签，作为最终的分类结果。每个类的置信度值使用的*Softmax*函数如公式（4-5）所示。

$$Softmax(v)_i=\frac{e^{vi}}{\sum_{j=1}^{N}e^{vj}} \tag{4-5}$$

其中v为输入向量，N为数据集中的类数，CICIDS2017 数据集中N为 7，NSL-KDD 数据集中N为 5，e^{v_i}和e^{v_j}分别为输入向量和输出向量的标准指数函数。置信平均法得到的预测类标签如公式（4-6）所示。

$$\hat{y}=\underset{i\in\{1,\cdots,N\}}{argmax}\frac{\sum_{j=1}^{K}p_j\left(y=i|L_j,x\right)}{K} \tag{4-6}$$

其中L_j是第j个基学习者，j的取值为 1、2、3。K是选择的基 CNN 学习者的数量，本实验的 IDS 中的$k=3$，选取了 VGG16, Inception 和 Xception 三个 CNN 模型作为基础模型；$p_j\left(y=i|L_j,x\right)$表示使用$L_j$的数据样本$x$中类值$i$的预测置信度。

4.3 实验过程及结果分析

本实验采用准确率、精度、召回率和 F1-score 四种不同的评价指标对 TL-CNN-IDS 模型的性能进行评估。其中，准确率是指分类器正确分类的样本数占总样本数的比例，计算方法如公式（4-7）所示；精度是指分类器正确预测为正样本的

样本数占被预测为正样本的样本总数的比例，计算方法如公式（4-8）所示；召回率是指分类器正确预测为正样本的样本数占实际为正样本的样本总数的比例，计算方法如公式（4-9）所示。由于网络流量通常是高度不平衡的数据，只有少部分的攻击样本。同时，由于入侵检测需要较高精准度的同时也需要较高的召回率，F1-score 是精确度和召回率的调和平均数，计算方法如公式（4-10）所示，其兼顾了入侵检测模型精确度和召回率的要求。

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4-7}$$

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP} \tag{4-8}$$

$$\mathrm{Re}call = \frac{TP}{TP + FN} \tag{4-9}$$

$$F1 - score = \frac{2 * \mathrm{Re}call * \mathrm{Pr}ecision}{\mathrm{Re}call + \mathrm{Pr}ecision} \tag{4-10}$$

4.3.1　实验数据集

在网络入侵检测领域，被广泛应用的数据集有 KDD99、NSL-KDD、UNSW-NB15 和 CICIDS2017。本实验采用 CICIDS2017 数据集作为模型评估的基准数据集，并在 NSL-KDD 数据集上进行验证。

CICIDS2017 数据集[65]包含良性样本和最新的常见攻击样本，和真实世界网络流量数据类似。它的每一条记录包含 79 个特征和一个标签，与其他安全数据集相比，它具有更多的特征、实例和网络攻击类型。它的数据采集于 2017 年 7 月 7 日星期五下午 5 时截至，共计 5 天。根据文献[66, 67]对 CICIDS2017 数据集的分析，其攻击模式可以归纳为六种主要的攻击类型：Botnets、Dos、Sniffing、Brute-Force、Infiltration 和 Web Attack，其攻击类型及数量如表 4-3 所示。

表 4-3　CICIDS2017 数据集攻击类型和数量

原始标签	攻击类型	样本数量	训练集（平衡处理后）
Benign	Benign	2 273 097	1 363 893
Bot	Botnets	1 966	8 000
Dos Golden Eye, Dos Hulk, Dos Slow-httptest, Dos Slowloris, Heartbleed	Dos	380 699	228 367
Port-Scan	Sniffing	158 930	95 355
SSH-Patator, FTP-Patator	Brute-Force	13 835	8 307
Infiltration	Infiltration	36	8 000
Web Attack-Brute Force, Web Attack-Sql Injection, Web Attack-XSS	Web Attack	2 180	8 000

NSL-KDD 数据集是 KDD99 的修正版本，经过预处理删除了一些冗余数据和一些已知问题，对入侵的检测更加准确，对各种入侵检测技术的评估更有效。数据集中每条记录包含 43 个特征，其中 41 个特征是流量本身，可以分为内在、内容、基于主机和基于时间四类特征，最后两个是标签（正常或攻击）和分数（流量输入本身的严重性）。该数据集中有 33 种攻击类型，被分为 DoS、Probe、U2R 和 R2L 攻击四种类型，每种攻击类型的数据分布如表 4-4 所示。

表 4-4　NSL-KDD 数据集攻击类型数据分布

数据集	数量					
	总计	Normal	DoS	Probe	U2R	R2L
KDD Train+	125 973	67 343	45 927	11 656	52	995
KDD Test+	22 544	9 711	7 458	2 421	200	2 654

4.3.2　CICIDS2017 数据集性能分析

为了评估所提出的 TL-CNN-IDS 入侵检测系统的性能，首先在 CICIDS2017 数据集上进行了训练和测试。通过实现提出的 IG-FCFB 特征工程方法，从 80 个原始特征中选择了 20 个优质特征，然后将其转换为适合 CNN 模型输入的图像形式。使用迁移学习将 VGG16、Inception 和 Xception 作为基础模型对各种攻击图像进行分类，为了构建最优模型，使用 BO-TPE 算法对所采用的 CNN 模型的主要参数进行优化，其不仅影响 CNN 模型的结构和有效性，也提高了模型的训练速度和性能。如表 4-5 所示，TL-CNN-IDS 入侵检测系统通过 IG-FCBF 方法大大缩短了模型执行的时间，执行时间从 5 785.9s 到 1 875.6s，时间缩减了 67.5%，并对模型准确率有一定的提升。

表 4-5　加入 IG-FCBF 方法前后 IDS 性能对比

方法	准确率(%)	TIME（s）
TL-CNN-IDS（without IG-FCBF）	99.446	5 785.9
TL-CNN-IDS	99.856	1 875.6

经过 BO-TPE 算法对参数进行寻优后，以优化后的 CNN 模型作为基础训练模型，模型的精度得到进一步提升，最后采用置信平均的集成策略来构建所提出的集成模型。在 CICIDS2017 数据集上 TL-CNN-IDS 的分类情况在混淆矩阵中进行显示，如图 4-5 所示。

图 4-5　TL-CNN-IDS 系统的混淆矩阵

经过 BO-TPE 算法优化的 CNN 模型和提出的最终的集成模型 TL-CNN-IDS 的对比结果如图 4-6 所示。置信平均使集成模型能够检测到不确定的分类结果，并通过使用分类信任来纠正错误分类的样本。优化后的 CNN 模型在实现数据转换和超参数优化后，F1 分数达到了 99.059%-99.724%，最终置信平均集成模型 TL-CNN-IDS 的 F1 分数达到 99.854%。准确率、精确度和召回率三个评估指标，置信平均的集成模型 TL-CNN-IDS 较优化后的 CNN 均有不同程度的提高，体现了置信平均这一集成策略的有效性。

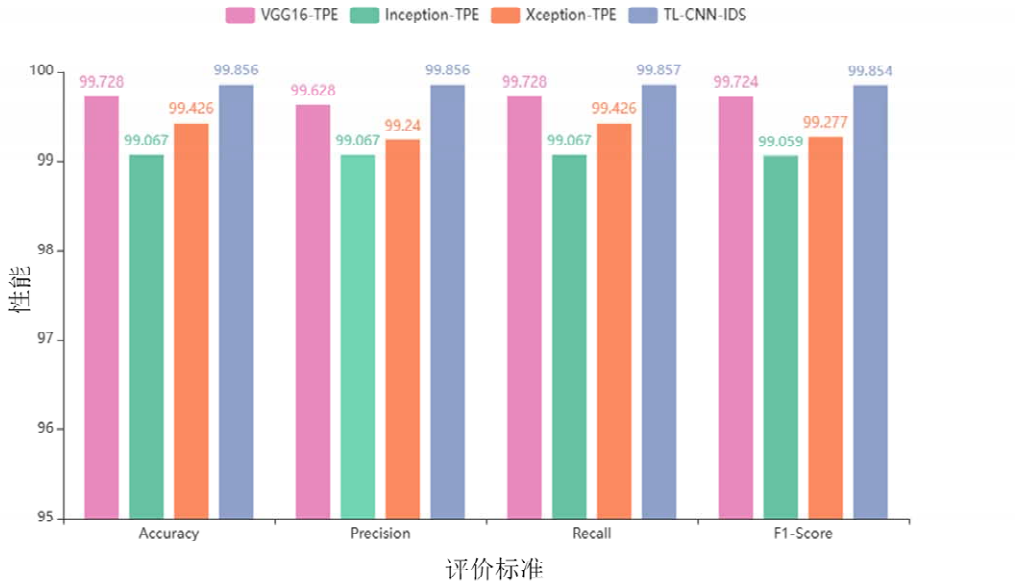


图 4-6　CNN 模型和 TL-CNN-IDS 性能对比

为了进一步证明所提出的 TL-CNN-IDS 模型的有效性，本文与其他最新在 CICIDS2017 数据集上提出 IDS 的文献[68,69]进行了对比分析，文献[68]提出了一种改进的前馈神经网络 MLP 模型的入侵检测方法，实验表明其优于一般深度模型，文献[69]评估深度信任网络结合超参数优化方法和 Double 超参数优化方法在 CICIIDS2017 数据集上的性能。对每个模型的准确率、精确度、召回率和 F1-score 四个共同指标进行了对比分析如表 4-6 所示。结果表明，所提出的模型 TL-CNN-IDS 具有更高的性能，其充分发挥了 CNN 模型在图像分类的优势，也证明了本文提出的 IG-FCBF 特征工程和基于 CNN 模型的迁移学习和集成学习方法在网络入侵检测领域有效的。

表 4-6　在 CICIDS2017 数据集上的模型的性能评估

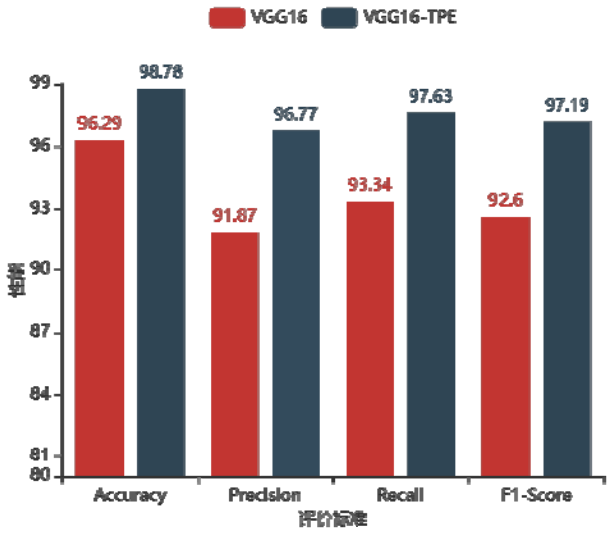
模型	准确率 (%)	召回率 (%)	精度 (%)	F1-分数(%)
改进的 MLP[68]	99.46	99.40	99.52	99.46
DBN-PSO [69]	95.81	95.81	95.82	95.81
DNN [69]	88.04	88.04	88.08	88.06
LSTM- RNN [69]	92.41	92.41	92.44	92.43
TL-CNN-IDS	99.86	99.86	99.86	99.85

4.3.3　NSL-KDD 数据集性能分析

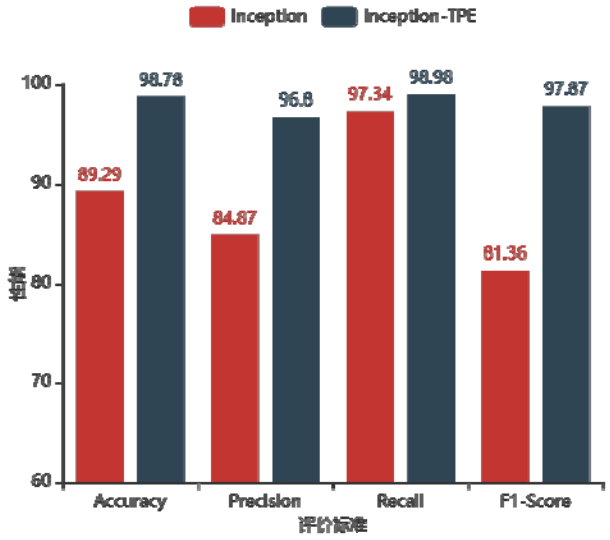
为了评估模型的有效性和普适性，在 NSL-KDD 数据集上进行补充实验。数据集 NSL-KDD 经过预处理后，为构建迁移后的最优 CNN 模型，采用 BO-TPE 算法对 CNN 基础模型的主要超参数进行优化，优化后 CNN 模型的超参数取值如表 4-7 所示。对比迁移学习 CNN 基础模型优化前后在 NSL-KDD 数据集上的性能如图 4-7 所示，从图中可以得知，BO-TPE 优化算法能显著提高 CNN 模型的恶意流量分类的准确率等系列评价指标，VGG16 在默认超参数下就表现出较好的性能，通过优化算法后，准确率提升了 2%左右，而 Xception 模型经过优化后准确率提升了 11%，从而验证了 BO-TPE 方法的有效性。

表 4-7　CNN 模型在 NSL-KDD 数据集上的最优参数

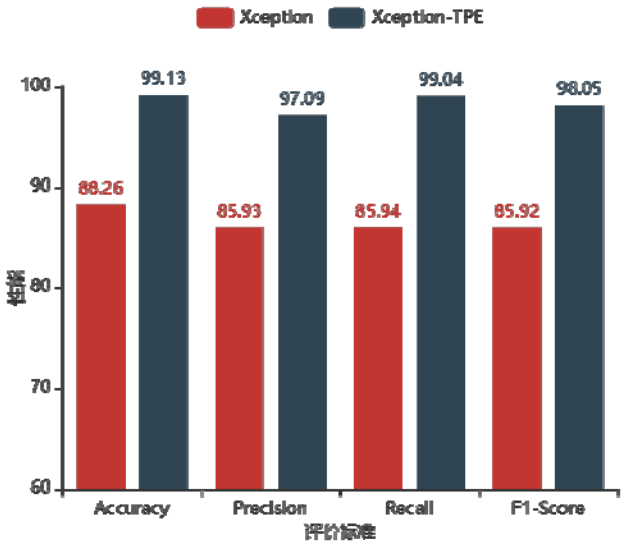
Model	Num of epochs	Learning rate	Dropout rate	Early stop patience	Frozen layers
VGG16	15	0.03	0.5	3	15
Inception	15	0.02	0.3	3	148
Xception	20	0.03	0.6	2	86



a) VGG16



b) Inception



c) Xception

图 4-7　迁移学习 CNN 基础模型优化前后在 NSL-KDD 数据集上的性能

其次，目前有很多工作在 NSL-KDD 数据集上评估他们的模型，本实验与文献[69,70]中提出的方法进行了对比，对比结果如表 4-8 所示。文献[69]采用的深度信任网络结合超参数优化方法和 Double 超参数优化的方法，文献[70]采用的是由深度自动编码器（AE）与长短期记忆（LSTM）和双向长短期记忆（Bi-LSTM）结合的方法来将网络流量分为正常样本和异常样本。实验结果表明，所提出的 IDS 能够有效区分正常流量和各类恶意流量，有效检测系统入侵中的各种网络攻击，且具有普适性，能在不同的网络流量数据集上具有较好的性能。

表 4-8　在 NSL-KDD 数据集上的模型的性能评估

模型	准确率 (%)	召回率 (%)	精度 (%)	F1-分数 (%)
DNN-PSO [65]	90.63	80.61	93.86	86.73
DBN-PSO [65]	96.91	92.29	98.10	95.11
AE-LSTM [66]	89.00	94.00	88.00	91.00
AE-BiLSTM [66]	87.00	88.00	89.00	89.00
TL-CNN-IDS	99.53	97.63	96.77	97.13

4.4　本章小结

受迁移学习和集成学习应用于图像分类领域的启发，本章提出了一种基于迁移学习和集成学习的 IDS 模型，用于实现入侵检测。该模型使用 VGG16、Inception 和 Xception 三个 CNN 模型作为基础模型，然后对其进行超参数优化，最后采用置信平均的集成策略对优化的 CNN 模型进行集成，从而实现企业网络各种类型的攻击的检测。此外，本章还提出了 IG-FCBF 的特征工程用于入侵检测数据的特征提取，并将网络流量表格数据可视化为图像后作为 CNN 模型的输入。所提出的 IDS 模型在 CICIDS2017 和 NSL-KDD 数据集上进行评估，实验结果表明，该 IDS 模型能够有效地识别各种类型的网络攻击，且具有普适性，能在不同的数据集上都表现出较好的性能。

第 5 章　基于实体行为分析的主机入侵检测模型

虽然针对主机入侵检测投了大量研究人员和工业者的努力，但越来越复杂的攻击载体导致主机入侵检测系统遭受了很高的误报率，HIDS 的高误报率容易导致安全团队预警疲劳[71]。目前，主机入侵检测算法主要分为两类：基于异常和基于误用的，基于异常的主机入侵算法无法对具体的攻击类型进行分类，而基于误用的关注的是安全事件本身，当出现新的安全事件时算法无法工作。同时，由于收集训练数据和标注样本需要付出高昂的代价，如何采用较少的训练样本得到较高准确率和较低的误报率。因此，本章提出了基于小样本学习的主机行为分析算法来进行主机入侵检测，通过系统调用序列反映主机行为，即小样本下的基于系统调用序列的主机入侵检测。该算法解决了小样本问题并提高了模型的鲁棒性和可移植性，当新的攻击行为出现也不需要重新训练模型。

5.1　总体框架

基于实体行为分析的主机入侵检测算法模型由两个核心模块组成，即数据预处理模块和基于孪生 LSTM 网络的主机入侵检测模块，其整体框架如图 5-1 所示。数据预处理模块进一步分为词向量化和样本图像化处理两个子模块，其主要步骤为：首先，使用 N-gram 对数据集进行词向量化处理，生成系统调用序列的 N-gram 特征表示模型；其次，计算每个 N-gram 项的 Tfidf 值；最后，通过数据填充和标准化后将其转换为二维图像数据。基于孪生 LSTM 网络的主机入侵检测模块的主要步骤为：首先，在训练阶段根据训练集生成等数量的相似对和不同对，将其输入孪生 LSTM 网络中进行模型训练；然后，在测试阶段对处理好的主机行为数据进行相似度计算，并进行结果分析；最后，通过分析结果的独热编码值判断是否发生了主机入侵。

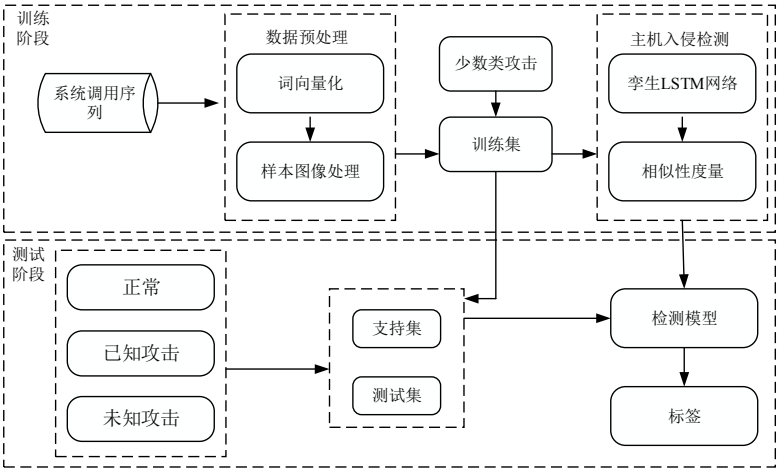


图 5-1　基于实体行为分析的主机入侵检测框架图

5.2 基于 TfidfVectorizer 和孪生 LSTM 网络的主机入侵检测算法

5.2.1 数据预处理

基于实体行为分析的主机入侵检测模型的数据预处理模块主要包括词向量化和样本图像处理。词向量化包括 N-gram 特征向量表示和 TfidfVectorizer 向量化两个步骤，样本图像处理包括数据标准化和数据填充和生成图像样本三个部分，下边将详细阐述 ADFA-LD 数据集的预处理过程。

(1) 词向量化 系统调用跟踪文件是主机进程行为的记录，能够反映其是正常还是异常。本章提出的主机入侵检测框架将系统调用跟踪文件的语料库看作文档，将单个系统调用看作一个词，故本算法先将其转化为 N-gram 特征向量表示模型，并计算其频率。利用表 5-1 中的示例来阐明从系统调用跟踪文件中生成 N-gram 语法项，假设系统调用跟踪文件 S 中包含五个唯一的系统调用（0、1、2、3 和 4），其二元、三元项和其频率计数如下，在真正的 N-gram 模型中用概率来计算。获得整个系统调用跟踪文件的N元语法特征向量表示模型后使用TfidfVectorizer的向量化技术来计算转换后的特征向量的 N-gram 项的 Tfidf 值，其是一种数值模型，其用 TF-IDF 表示。TF-IDF 即 TF*IDF，其中 TF 为词频，IDF 为逆文档频率。其中，逆文档频率能衡量给定的 N-gram 术语在系统调用跟踪文件语料库中出现的稀有程度。本实验用 TF-IDF 来评估一个 N-gram 术语对于一个系统调用跟踪文件的重要程度。

表 5-1 系统调用的 n-gram 特征向量表示

系统调用系列 S:0,0,1,1,1,1,2,3,4,0,0
2-gram：{0,0:2}, {0,1:1}, {1,1:3}, {1,2:1}, {2,3:1}, {3,4:1}, {4,1:1}
3-gram：{0,0,1:1}, {0,1,1:1}, {1,1,1:2}, {1,1,2:1}, {1,2,3:1}, {2,3,4:1}, {3,4,0:1}, {4,0,0:1}

转化后的N-gram特征向量的每一个元素都代表了相应的跟踪文件中存在的唯一的 N-gram 术语，下一步计算 Tfidf 值。TfidfVectorizer 方法的具体步骤如下：

Step1 构建语料库

根据系统调用序列特征化后的样本，构建语料库，形式如公式 5-1 所示。

$$Text = \begin{pmatrix} S_{00} & \cdots & S_{0j} \\ \vdots & \ddots & \vdots \\ S_{i0} & \cdots & S_{ij} \end{pmatrix}, 0 \le i < n, 0 < j < L \tag{5-1}$$

其中，$S_{00}, S_{i0}, \cdots, S_{ij}$ 为系统调用文档，n 为样本数，L 为序列长度的最大值。

Step2 生成词典

统计语料库中所有文档中的系统调用序列以及其出现的次数，生成的词典如公式 5-2 所示。

$$dict=\begin{pmatrix} d_{00} & \cdots & d_{0j} \\ \vdots & \ddots & \vdots \\ d_{i0} & \cdots & d_{ij} \end{pmatrix}, 0\le i<n, 0\le j<m \tag{5-2}$$

其中，d_{ij} 为第 i 个文档中第 j 个系统调用序列的次数，n 为样本数，m 为词典中系统调用序列的总个数。

Step3 TF 矩阵计算

计算每个文档中的系统调用序列出现的频率，即词频，计算方式见公式（5-3）。

$$tf_{ij}=d_{ij}\Big/\sum_{k=0}^{n=1}d_{kj}, 0\le i<n, 0\le j<m \tag{5-3}$$

然后生成样本词频矩阵如公式（5-4）所示。

$$TF=\begin{bmatrix} tf_{00} & \cdots & tf_{0j} \\ \vdots & \ddots & \vdots \\ tf_{i0} & \cdots & tf_{ij} \end{bmatrix}, 0\le i<n, 0\le j<m \tag{5-4}$$

其中，tf_{ij} 为第 i 文档中第 j 系统调用序列的词频。

Step4 计算 IDF

一个系统调用序列的重要性随着其出现频率的增加而提高，也会随着在语料库中出现的频率（即 IDF）而反向下降，其计算方法如公式（5-5）所示。

$$IDF=log_{10}\left(\frac{N}{df}\right), df\ne 0 \tag{5-5}$$

其中，N 是语料库中文档的总数，df 是给定 N-gram 系统调用序列的文档频率。

Step5 计算 TF-IDF

TF-IDF 综合考虑了 TF 和 IDF 对词语重要性的影响，其是自然语言处理中常见的衡量指标，其计算方法如公式（5-6）所示。

$$TF-IDF=TF*IDF \tag{5-6}$$

(2)　样本图像化处理　词向量化后，采用 Z-score 方法对数据样本进行标准化，其标准化过程如公式（5-7）、公式（5-8）和公式（5-9）所示。

$$\overline{x}=\frac{1}{n}\sum_{i=1}^{n}x_i \tag{5-7}$$

$$v=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{x})^2} \tag{5-8}$$

$$x_i^{'}=(x_i-\overline{x})/v \tag{5-9}$$

其中，x_i 表示原始数据样本的第 i 维特征，$x_i^{'}$ 表示标准化后样本的第 i 维特征。

本算法采用的是二维数据，且孪生网络在处理图像问题具有成熟的研究，故将一维向量转化为二维图像数据，具体方法与第四章中生成图像样本的过程相同。

5.2.2 生成训练样本对

孪生神经网络的输入是样本对，通过对样本对的学习来训练模型，因此本文生成 ADFA-LD 数据集的训练样本对。给定一个 N 类数据集，首先，选择一个攻击类别 e 作为未知攻击，这个类别将不再用于模型训练；其次，剩下的 N-1 个类，每个类的实例被分成两部分，一部分用于生成训练集中的正负样本对，另一部分用于模型评估。训练样本对生成过程如图 5-2 所示。该方法相当于扩大了数据集的规模，因此较小数据量的数据集也能通过深度学习来进行学习训练而得到较好的结果。攻击类别 e 用于模拟真实情况，在这种情况下，只有少数标记的样本可用检测未知攻击。该模型依赖于随机对的生成，且要保证配对的唯一性，没有重复项。

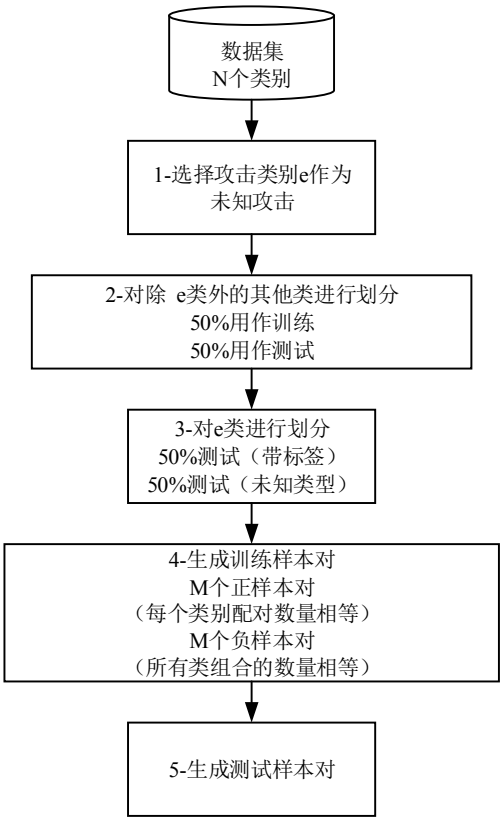


图 5-2　训练样本对生成过程

生成训练样本对的数量计算方法如下。假设 $C\text{lass}=\{C_1,C_2,...,C_n\}$ 表示训练集中的 n 类，本文进行主机入侵检测采用的数据集中包含 7 种样本数据，其中包括 6 种攻击类型和 1 种正常系统调用样本，即 n 为 7。通过从 C_i 中任选两个样本组成正类样本，假设 C_i 类中有 K 个样本，则 C_i 类产生的正类样本对的数量为 C_K^2，其他的类计算方式相同，正类样本总数为各类产生的正类样本对累加。负类样本对则是选择 C_i 类和另一个非 C_i 类的样本组合而成，其数量为 $C_i \times \overline{C}_i$，其中×为笛卡尔积。

5.2.3　构建孪生 LSTM 网络

孪生神经网络由两个结构相同的网络，共享权值而构成，其基础网络结构如图 5-3 所示。孪生网络对数据样本进行检测的本质上是学习样本之间的相似性的过程，其在变化检测[72]、目标跟踪[73]等领域有广泛应用，其淡化了标签，能够利用有限数量的标记样本表示的新攻击类别来评估网络在不需要重新训练的情况下对新的攻击类型进行分类的性能。孪生网络是一种思想，基本的网络结构可以是 CNN 也可以是 RNN，本节选择 LSTM 作为孪生网络的基础模型，通过最小化损失函数使样本之间的距离变大。首先将训练样本对输入该神经网络，然后将输入映射到新的空间并形成新的表示，通过计算损失来评估两个输入的相似度。

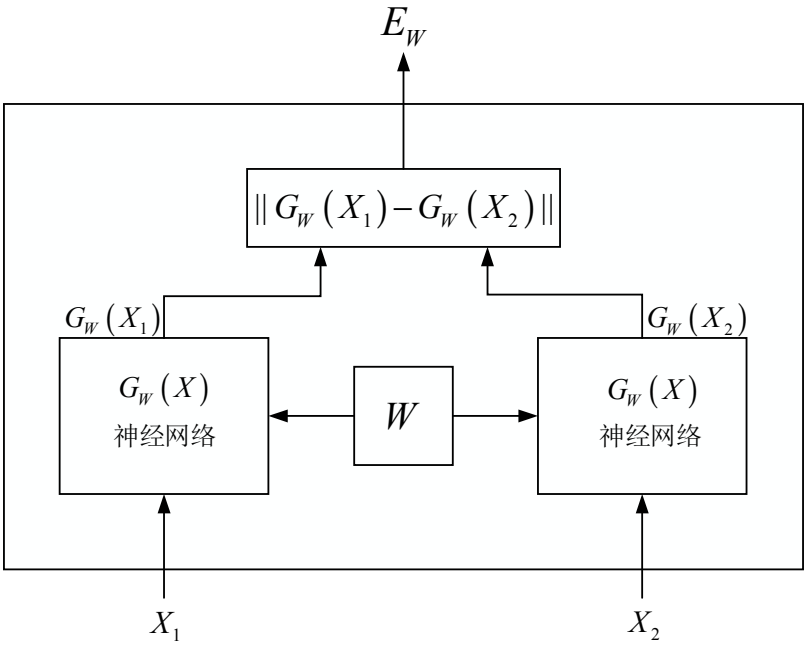


图 5-3　孪生神经网络结构图

如图所示，X_1、X_2 为输入，通过神经网络得到 $G_W(X_1)$、$G_W(X_2)$ 两个特征向量，计算这两个向量的距离 E_W，即两个向量的相似度，计算方法如公式 5-10 所示。

$$S(X_1, X_2) = \| G_W(X_1) - G_W(X_2) \| \tag{5-10}$$

本章在孪生神经网络的基础上采用了基于孪生 LSTM 的网络模型进行主机入侵检测，孪生 LSTM 网络的结构如图 5-4 所示。将图像视为序列信息，采用 LSTM 可以保障数据维度之间的相关性，对比不同层数的 LSTM 网络模型效果，发现 4 层网络具有更好的性能，故本文的 LSTM 是一个四层网络结构。

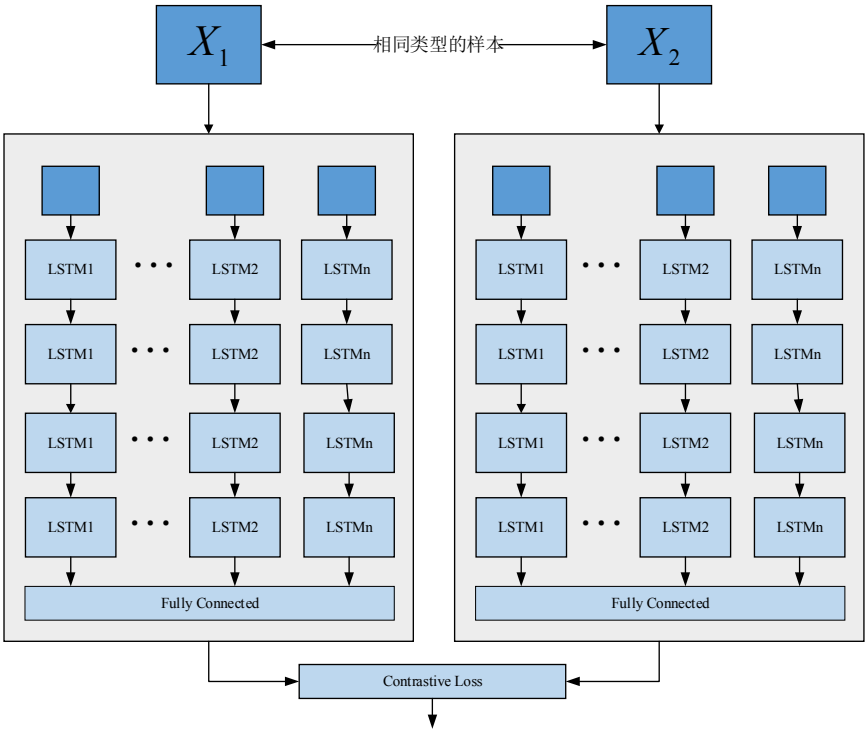


图 5-4　孪生 LSTM 网络结构图

5.2.4　对比损失函数

损失函数用于评价预测值和实际值之间的差异程度。损失函数越小，模型性能越好，一般来说不同的模型的最佳损失函数也不同。该实验的孪生 LSTM 网络采用欧氏距离来计算数据样本之间的相似度，采用对比损失函数来衡量，同一个类型的样本的相似度大于不同类型样本的相似度，两个样本的相似度量越大，其属于同一类型的概率越大。对比损失函数[74]的计算公式如（5-11）所示，欧氏距离的计算方式如（5-12）所示。

$$L\left(W,\left(Y,X_1,X_2\right)\right)=\frac{1}{2N}\sum_{n=1}^{N}YD_W^2+\left(1-Y\right)max\left(m-D_W,0\right)^2 \tag{5-11}$$

$$D_W\left(X_1,X_2\right)=\|X_1-X_2\|_2=\left(\sum_{i=1}^{P}\left(X_1^i-X_2^i\right)^2\right)^{\frac{1}{2}} \tag{5-12}$$

其中，$D_W\left(X_1,X_2\right)$代表样本X_1与样本X_2的欧氏距离，P为样本的维数，N为样本数，m表示设定阈值。Y表示样本标签是否匹配，若两个数据样本相似，则$Y=1$，此时，对比损失函数$L\left(W,\left(Y,X_1,X_2\right)\right)$为$1/2N\sum_{n=1}^{N}YD_M^2$，两个数据样本差别较大时$Y=0$，损失函数为$1/2N\sum_{n=1}^{N}YD^2+max\left(m-D_M,0\right)^2$。表明当两个数据样本不相似，欧氏距离变小，而对比损失值会变大，这样更有利于检测。

为了验证对比损失函数的有效性，本章还采用其他的损失函数进行了实验，包括平方误差损失函数和余弦损失函数。平方误差损失是预测值与真实值之差的平方，也称为 L2 Loss，其存在梯度消失的问题，计算方式如（5-13）所示，其中δ函数见

公式（5-14）所示。

$$E_{sq}=\frac{1}{mb}\sum_{i=1}^{mb}\left[\left(1-f^{(i)}\right)*\left(\frac{1}{2}-\delta\left(d^{(i)}\right)^{2}\right)^{2}+f^{(i)}*\left(1-\delta\left(d^{(i)}\right)^{2}\right)^{2}\right] \tag{5-13}$$

$$\delta(x)=\frac{1}{1+\ell^{-x}} \tag{5-14}$$

其中，m 为边界值，δ 为逻辑函数，$d^{(i)}$ 为第 i 个样本之间的相似性度量。

余弦损失函数，也被称为余弦相似度损失函数，其通过计算两个向量夹角的余弦值来判断输入是否相似，其计算方式如公式（5-15）所示。

$$E_{cos}=\frac{1}{mb}\sum_{i=1}^{mb}[\left(1-f^{(i)}\right)*cos\left(X_1^{(i)},X_2^{(i)}\right)+f^{(i)}\left(m+cos\left(X_1^{(i)},X_2^{(i)}\right)\right)] \tag{5-15}$$

5.3　实验过程及结果分析

5.3.1　实验数据集

关于主机行为的数据集有很多，但是 DARPA、KDD 和 UNM 这些数据集已经过时，无法有效地捕获现代主机的进程和攻击，不能很好地代表现代的互联网环境。本研究采用的是 ADFA-LD 数据集[30]，主要包括在 Linux 服务器（Ubuntu11.04）上运行的各种活动服务的系统调用跟踪文件，其具有最新攻击向量属性，被广泛用于最近的主机入侵检测研究。ADFA-LD 数据集通过系统调用序列来反映主机的行为，其包括 325 个系统调用并完成特征化，每个跟踪只记录相应系统调用表中出现的系统调用标识号，而不是其名称，如表 5-2 所示。

表 5-2　系统调用特征化

系统调用名称	标识号
sys_restart_syscall	0
sys_exit	1
…	…
sys_poll	168
…	…
sys_clock_gettime	265
…	…
sys_eventfd	323
sys_fallocate	324

通过分析系统调用的进程来识别攻击，例如第 168 和 265 号常以‘168 265’调用序列出现在 Adduser 攻击中，表明创建了套接字，程序 poll 和 select 等待到达的数据或准备发送的数据。同时，数据集对样本中正常和攻击行为序列进行了标注，为后续的模型训练提供支持，ADFA-LD 训练集中有 833 个正常的系统调用序列，验证

集中包含 746 个攻击和 4 372 个正常系统调用序列，其攻击行为类型和数量如表 5-3 所示。

表 5-3　ADFA-LD 攻击行为类型和数量

攻击类型	描述	样本数
Hydra_FTP	Hydra 暴力破解 FTP	162
Hydra_SSH	Hydra 暴力破解 SSH	148
Java_Meterpreter	TikiWiki 漏洞利用	125
Meterpreter	客户端投毒	75
C100 WebShell	C100 Webshell PHP 远程文件包含漏洞	118
Adduser	添加 root 权限用户、客户端投毒	91

ADFA-LD 数据集中包含的六种攻击类型代表了中级黑客使用的常见攻击方法，从低级的密码猜测如暴力破解到基于 Web 的攻击和社会工程，再到远程攻击。这些攻击包括在 SSH 和 FTP 的开放端口上分别使用两种蛮力猜测密码的方法，通过将恶意有效载荷编码到正常的可执行文件中，未经授权试图使用中毒的可执行文件创建一个新的超级用户，上传 Java 和 Linux 可执行 Meterpreter 有效载荷以远程控制目标主机，以及使用 C100 Webshell 进行控制和权限升级。ADFA-LD 数据集中，有 10 个文件夹对应于每种攻击类型。每个攻击类型的文件夹包含 5 到 20 个不同的异常系统调用跟踪。因此可以采用 ADFA-LD 数据集对现代化的 HIDS 进行评估

5.3.2　实验过程

基于 TfidfVectorizer 和孪生 LSTM 网络的主机入侵检测算法的实验过程被分为训练过程和测试过程。训练过程主要分为三个步骤，首先使用数据预处理方法对原始日志文件进行处理，其次，构建适合孪生 LSTM 网络输入的训练样本对，最后进行模型训练和微调并将训练好的模型进行保存。测试过程首先将处理好的数据分为测试集和支持集，然后加载训练好的模型进行测试，如图 5-5 所示。

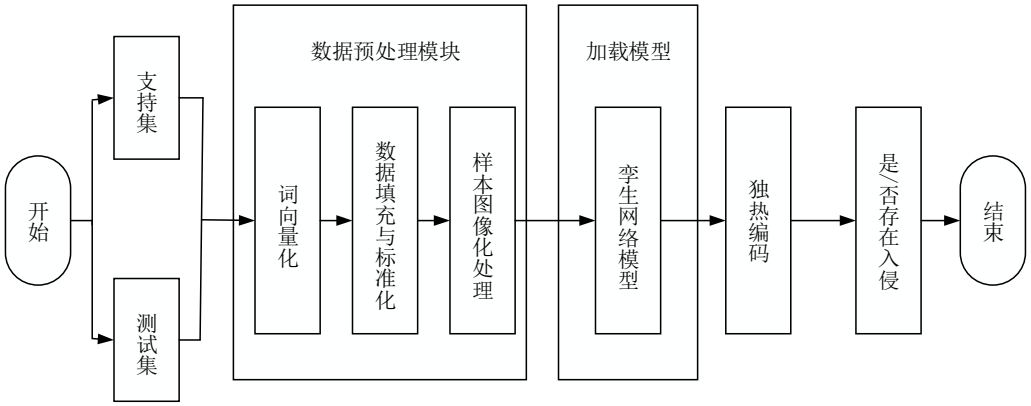


图 5-5　主机入侵检测模型测试流程图

通过十折交叉验证方法调节训练模型参数使其达到最优，其中学习率为 0.001，并将保留训练后的权重对孪生 LSTM 网络进行微调。训练过程迭代次数为 100 000，测试过程中迭代次数为 10*800。由于主要为了突出该算法可以检测未知的恶意主机行为，本章将 ADFA-LD 数据集中的 Meterpreter 类型设为未知攻击类型，不会在训练集中出现，仅用于测试，其余的五个攻击类型和正常类型每种随机抽取 50 个生成训练样本对，其他的用作测试。

5.3.3　实验结果分析

(1)　数据预处理和特征表示分析　为了分析本实验采用的数据预处理方法对所提出的 HIDS 模型性能的影响，对不同的 N-gram 特征向量表示模型进行对比分析。在本研究中考虑了 1-gram、2-gram 和 3gram 的系统调用跟踪文件。在 ADFA-LD 数据集的训练数据集中，共识别出 167 个唯一的 1-gram 术语，3 396 个唯一的 2-gram 术语和 18 316 个唯一的 3-gram 术语。然后将训练数据集中的所有系统调用序列转换为 N-gram 的特征向量，并使用 TfidfVectorizer 的向量化技术来计算转换后的特征向量的 N-gram 项 Tfidf 值，其中 N 的取值为 1、2、3。以 3-gram 为例，ADFA-LD 训练数据集的 3-gram 特征向量表示模型如表 5-4 所示。

表 5-4　ADFA-LD 数据集的 3-gram 特征向量表示模型

特征向量	F_1(‘63,42, 120’)	F_2(‘42,120,195’)	…	F_{18316}(‘174,175,120’)	攻击类型
1	0.004 3	0.078 4	…	6.760 9	Normal
2	2.013 2	1.105 3	…	1.369	Webshell
…	…	…	…	…	…
5 951	0.843 6	0.518 2	…	7.942 9	Adduser

表中的每一行表示一个转换后的 N-gram 特征向量，其对应于 ADFA-LD 训练数据集中的一个系统调用跟踪文件。如表所示，共有 5 951 个转换后的 3-gram 特征向量，每个向量包含 18 316 个特征。特征向量的每一列（F_k）表示在相应的系统调用跟踪文件中出现的 3-gram 项的 Tfidf 值。然后经过标准化处理和数据填充，将其转化为适合的图像形式。保证采用相同的孪生 LSTM 网络和损失函数，对不同的词向量化操作进行模型训练和测试，特征向量表示模型的对比结果如图 5-6 所示。

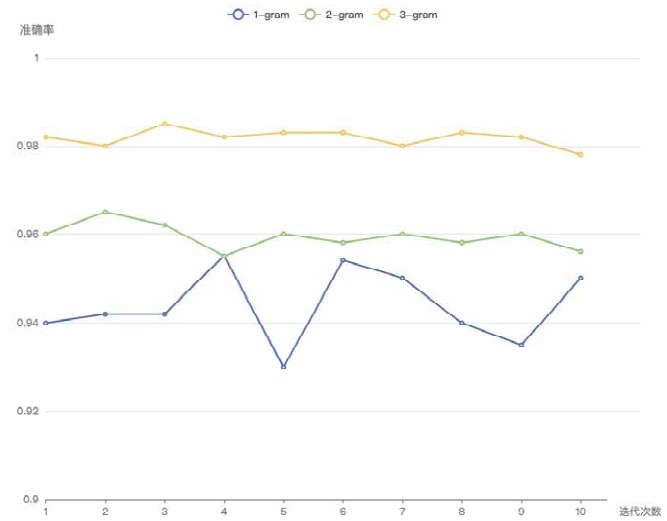


图 5-6　不同特征向量表示模型对比结果

其中，纵坐标的准确率为小样本学习多次测试准确率的平均值，迭代次数为10*800。根据图 5-6 结果显示，采用 3-gram 特征向量表示模型能够使模型取得更好的效果，故本研究采用 3-gram 的特征向量表示模型。

(2)　LSTM 层数对检测结果的影响　由于本算法搭建的孪生网络是基于 LSTM 的，因此需测试 LSTM 层数对检测结果的影响。在实验过程中，影响实验结果的其他因素保持不变，LSTM 层的数量为唯一变量，增加孪生网络中 LSTM 的层数，观察模型检测的准确率。实验结果如图 5-7 所示。从图中得知，一开始随着 LSTM 层数的递增模型的准确率在提高，在层数为 4 时模型准确率达到最高 98.55%，该 LSTM 网络模型达到最优。当 LSTM 层数继续递增时，模型出现过拟合现象，导致模型的检测准确率逐步降低。因此本算法中的孪生 LSTM 网络使用 4 层网络结构，用于提升主机入侵检测系统的准确率。

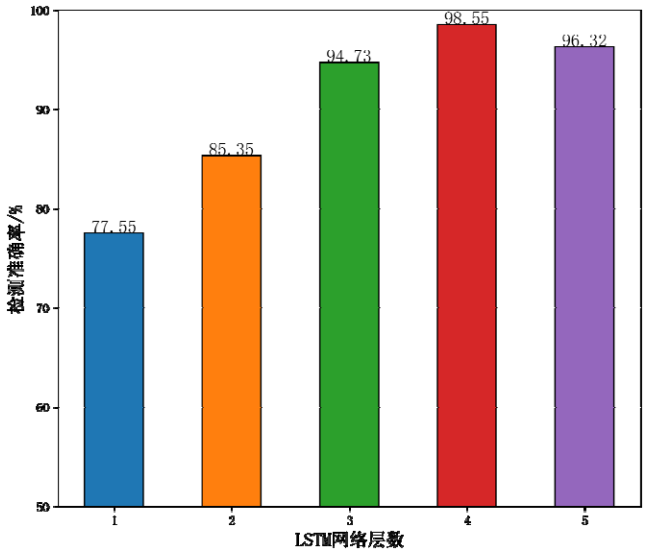


图 5-7　不同 LSTM 层构成孪生网络的检测结果

(3) 损失函数对比分析　本实验为了选择最佳损失函数，对 5.2.4 节中提到的平方误差损失函数、余弦损失函数和对比损失函数分别进行测试，比较不同损失函数给主机入侵检测结果带来的影响。本实验采用准确率作为评价标准，不同损失函数的检测结果如表 5-5 示。据检测结果显示，在孪生 LSTM 网络上使用不同的损失函数，表现出不同的检测准确率。经比较可以发现，使用对比损失函数的模型效果明显优于平方误差损失函数和余弦损失函数，其准确率高达 98.5%，故本算法模型采用对比损失函数来进行模型参数的调优过程。

表 5-5　三种损失函数检测结果

损失函数	准确率
平方误差损失函数	94.1%
余弦损失函数	93.7%
对比损失函数	98.5%

(4) 本算法与其他文献算法对比实验　本算法采用了对比损失函数和四层 LSTM 搭建的孪生网络，为证明本文提出方法的有效性，与最新研究文献[75]中采用的 MLP 和 dCNN 算法进行对比，其首先用词袋模型对 ADFA-LD 数据集进行预处理，然后再使用 MLP 和 dCNN 算法进行分类。每个算法在不同类别的检测准确率的结果见表 5-6 示。

表 5-6　本算法与文献[71]模型检测准确率

算法	Normal	Hydra_FTP	Hydra_SSH	Java_Meterpreter	Meterpreter	WebShell	Adduser
本文	97.45%	98.65%	98.62%	98.50%	96.43%	97.85%	98.75%
MLP	94.57%	98.45%	98.15%	98.26%	87.85%	97.82%	98.71%
dCNN	100%	100%	99.94%	99.83%	99.83%	99.94%	100%

从表 5-6 以得知，该算法对各个类别的检测准确率均优于文献[75]中的 MLP 算法，特别是在 Meterpreter，MLP 算法仅取得了 87.85%的准确率，而该算法在训练过程并未使用 Meterpreter 的情况下，模型检测准确率仍明显高于 MLP 算法，这表明本文提出的孪生 LSTM 网络具有较强检测未知样本的能力。尽管该算法检测的准确率比 dCNN 算法低，但该算法仅抽取了每类样本中的 50 个样本用来生成训练集，比 dCNN 需要的训练样本少得多。故该算法仍表现出重要的价值，一方面该算法具有检测未知攻击的能力，能够很好得应对新出现的各种主机行为，一定程度上解决了现有算法可移植性差的问题，另一方面，该算法在训练样本很少的情况下，仍有较好的效果，解决了主机入侵检测中的小样本问题。

5.4 本章小结

本章主要构建了基于实体行为分析的主机入侵检测算法。首先对本章所采用的主机入侵检测数据集 ADFA-LD 进行展示。然后阐述了基于孪生 LSTM 网络模型的主机入侵检测算法，算法主要流程为数据预处理、生成训练样本对，然后构建孪生 LSTM 网络用于主机行为分析。最后对基于 TfidfVectorizer 和孪生 LSTM 网络的主机入侵检测算法进行实验，选择最优的 LSTM 的层数和损失函数，然后对实验结果进行分析。实验结果表明，本章算法具有识别未知攻击的能力，而且解决了传统的深度学习模型因训练样本较少而出现过拟合现象，导致模型准确率降低的问题。

结 论

随着科学技术的发展，网络规模日益庞大，信息容量迅速膨胀，网络威胁层出不穷，网络安全显得愈发重要。以系统破坏、信息窃取以及电子欺诈为首的内部威胁给国家、企业以及个人造成巨大损失。同时，还存在网络病毒、黑客入侵等外部攻击的威胁。但是传统的威胁检测技术不能应对快速更新的网络环境，解决现有的安全问题，而用户和实体行为技术能够从多角度、多维度来对各种网络威胁进行建模，故利用用户和实体行为分析技术来解决威胁检测问题能降低误报率，提高检测准确率。本文从以下三个方面对网络威胁的检测方法展开深入研究。

（1）针对现有的内部威胁检测方法没有从多个维度来考虑内部攻击行为和算法AUC 分数较低的问题，本文研究并实现了基于 LSTM-Attention 的用户行为分析算法用于内部威胁检测。首先，提取用户的行为序列、用户行为特征、角色行为特征和心理数据来描述用户的日常活动，通过多维度的用户日常活动来描述用户的行为，发现行为之间的逻辑关系；其次，使用长短期记忆网络和注意力机制学习用户的行为模式，并计算真实行为与预测行为之间的偏差；最后，使用多层感知机根据这些偏差进行综合决策来识别异常行为。通过 CERT 内部威胁检测数据集进行评估，并取得了 96.4%的 AUC 分数。

（2）针对真实的网络环境下防御者难以获得攻击流量，导致样本类别不平衡，且训练样本不足的问题，本文研究并实现了基于 IG-FCBF-TL 的流量行为分析算法用于网络入侵检测。首先，对所获得的数据集进行预处理（包括 IG-FCBF 特征工程和 SMOTE 过采样）并将其转换为适合 CNN 模型输入的图像形式；其次，选择 VGG16、Inception 和 Xception 三种 CNN 模型作为基础学习模型，并采用 Tree-Structured Parzen Estimator 算法的超参数优化方法在目标数据集上寻求最佳模型；最后采用置信度平均的集成方法对优化后的三个 CNN 模型进行集成。在入侵检测领域网络数据量不足的情况下，采用迁移学习进行模型训练，提高训练效率，并实现网络流量的正确分类。通过 CICIDS2017 和 NSL-KDD 网络入侵检测数据集对模型进行评估，其均取得超 99%的准确率，验证了方案的可行性与有效性。

（3）针对样本数量较少的情况下深度学习模型训练过程中容易出现过拟合导致算法准确率低和因安全事件形式改变原有算法无法工作的问题，研究并实现了基于孪生 LSTM 网络的主机入侵检测算法。为避免安全事件本身，本文通过分析系统调用序列即主机行为，进行主机入侵检测。首先，使用 TfidfVectorizer 对样本特征进行词向量化处理，然后将二维数据转化图像数据，最后将其输入孪生 LSTM 网络进行模型训练，并采用对比损失函数作为优化函数。通过改进小样本学习算法来分析主

机行为，解决了深度学习算法的小样本问题。通过 ADFA-LD 主机入侵检测数据集对算法进行评估，通过较少的样本达到了 98.5%的准确率，验证了该算法在小样本条件下的有效性。

针对威胁检测领域目前存在的问题，基于用户和实体行为分析技术，本文给出了相应的解决方案，这三个模型在公共数据集上都取得了很好的效果，然而真实的网络环境瞬息万变，基于此情况计划下一步工作如下。

（1）考虑到数据样本分布不平衡，恶意样本极少的情况，下一步工作主要考虑两个方向来处理类不平衡问题，一是数据层面，通过改进的数据增强的方法来减弱少数类样本对评估结果的影响；二是模型层面，通过改进小样本学习算法实现对未知威胁的检测。

（2）针对检测方法实时性的问题，下一步工作是改进实验方案，从而实现网络威胁的实时检测，因为现有的工作是针对已有的公共数据集进行的威胁检测，其是静态的数据文件，但真实的网络环境瞬息万变，每刻都有新的情况产生，而且真实环境下的数据量要比实验数据大的多。

参考文献

[1] X. YUE, W. CHEN, Y. WANG. The Research of Firewall Technology in Computer Network Security. 2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications (PACIIA). Wuhan, CN: IEEE, 2009(2): 421-424

[2] E. KABIR, J. K. HU, H. WANG, et al. A Novel Statistical Technique for Intrusion Detection Systems. Future Generation Computer Systems, 2018, 79(1): 303-318

[3] SONICWALL, 2022 Sonicwall Cyber Threat Report. https://tinyurl.com/2f9m5vks,2022-10-10

[4] M. SHASHANKA, M. Y. SHEN, J. WANG. User and Entity Behavior Analytics for Enterprise Security. 2016 IEEE International Conference on Big Data (Big Data). Washington DC, USA: IEEE, 2016: 1867-1874

[5] S. KHALIQ, Z. U. A. TARIQ, A. MASOOD. Role of User and Entity Behavior Analytics in Detecting Insider Attacks. 2020 International Conference on Cyber Warfare and Security (ICCWS). IEEE, 2020: 1-6

[6] J. CUI, G. ZHANG, Z. CHEN, et al. Multi-homed Abnormal Behavior Detection Algorithm Based on Fuzzy Particle Swarm Cluster in User and Entity Behavior Analytics. Scientific Reports, 2022, 12(1): 22349

[7] 李志, 宋礼鹏. 基于用户窗口行为的内部威胁检测研究. 计算机工程, 2020, 46(4): 135-142, 150

[8] 吴驰, 帅俊岚, 龙涛, 等. 基于 Linux Shell 命令的用户异常操作检测方法研究. 信息网络安全, 2021, 21(5): 31-38

[9] 姚海龙, 王彩芬, 许钦百, 等. 一种基于同态加密的分布式生物特征认证协议. 计算机研究与发展, 2019, 56(11): 2375-2383

[10] X. Y. YE, S. S. HONG, M. M. HAN. Feature Engineering Method Using Double-layer Hidden Markov Model for Insider Threat Detection. International Journal of Fuzzy Logic and Intelligent Systems, 2020, 20(1): 17-25

[11] V. MAVROEIDIS, K. VISHI, A. JOSANG. A Framework for Data-driven Physical Security and Insider Threat Detection. 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). Barcelona, Spain. Piscataway: IEEE, 2018: 1108-1115

[12] K. YU, L. TAN, S. MUMTAZ, et al. Securing Critical Infrastructures: Deep-Learning-Based Threat Detection in IIoT. IEEE Communications Magazine, 2021, 59(10): 76-82

[13] S. B. TEODORA, B. CAGLAYAN, A. HAYTHAM. DeepAD: A Generic Framework Based on Deep Learning for Time Series Anomaly Detection. Berlin: Springer, 2018: 577-588

[14] F. F. YUAN, Y. N. CAO, Y. M. SHANG, et al. Insider Threat Detection with Deep Neural Network.

International Conference on Computational Science (ICCS), Wuxi, CN. Berlin: Springer, 2018: 43-54

[15] Y. CHANG, W. LI, Z. YANG. Network Intrusion Detection based on Random Forest and Support Vector Machine. 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). Guangzhou, CN: IEEE, 2017, 1: 635-638

[16] M. GOVINDARAJAN, R. M. CHANDRASEKARAN. Intrusion Detection Using K-Nearest Neighbor. 2009 First International Conference on Advanced Computing. Chennai, IN: IEEE, 2009: 13-20

[17] S. SAHU, B. M. MEHTRE. Network Intrusion Detection System Using J48 Decision Tree. 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). Kochi, IN: IEEE, 2015: 2023-2026

[18] H. ZHANG, Y. LI, Z. LV, et al. A Real-Time and Ubiquitous Network Attack Detection Based on Deep Belief Network and Support Vector Machine. IEEE/CAA Journal of Automatica Sinica, 2020, 7(3): 790-799

[19] T. SU, H. SUN, J. ZHU, et al. BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset. IEEE Access, 2020, 8: 29575-29585

[20] Z. WANG, Y. LIU, H. E. DAOJING, et al. Intrusion Detection Methods Based on Integrated Deep Learning Model. Computers & Security, 2021,103:102177

[21] S. T. MEHEDI, A. ANWAR, Z. RAHMAN, et al. Deep Transfer Learning Based Intrusion Detection System for Electric Vehicular Networks. Sensors, 2021, 21(14): 4736

[22] M. X. LU, G. Z. DU, Z. X. JI. Network Intrusion Detection Based on Deep Transfer Learning. Application Research of Computers, 2020,37(9):4

[23] J. HU, Y. D. SU, W. Z. HUANG, et al. Intrusion Detection Method Based on Ensemble Transfer Learning via Weighted Mutual Information. Journal of Computer Applications, 2019, 39(11):3310-3315

[24] P. DESHPANDE, S. C. SHARMA, S. K. PEDDOJU, et al. HIDS: A Host Based Intrusion Detection System for Cloud Computing Environment. International Journal of System Assurance Engineering and Management, 2018, 9: 567-576

[25] Z. S. MALEK, B. TRIVEDI, A. SHAH. User Behavior Pattern-Signature Based Intrusion Detection. 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4). IEEE, 2020: 549-552

[26] V. KUMAR, D. SINHA, A. K. DAS, et al. An Integrated Rule Based Intrusion Detection System: Analysis on UNSW-NB15 Data Set and the Real Time Online Dataset. Cluster Computing, 2020,

23: 1397-1418

[27] A. SHAIKH, P. GUPTA. Advanced Signature-Based Intrusion Detection System. Intelligent Communication Technologies and Virtual Mobile Networks: Proceedings of ICICV 2022. Singapore: Springer Nature Singapore, 2022: 305-321

[28] V. KUMAR, O. P. SANGWAN. Signature Based Intrusion Detection System Using SNORT. International Journal of Computer Applications & Information Technology, 2012, 1(3): 35-41

[29] G. SERPEN, E. AGHAEI. Host-based Misuse Intrusion Detection Using PCA Feature Extraction and KNN Classification Algorithms. Intelligent Data Analysis, 2018, 22(5): 1101-1114

[30] G. CREECH, J. HU. A Semantic Approach to Host-Based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns. IEEE Transactions on Computers, 2013, 63(4): 807-819

[31] P. K. KESERWANI, M. C. GOVIL, E. S. PILLI, et al. A Smart Anomaly-Based Intrusion Detection System for the Internet of Things (IoT) Network Using GWO–PSO–RF Model. Journal of Reliable Intelligent Environments, 2021, 7: 3-21

[32] T. SABA, A. REHMAN, T. SADAD, et al. Anomaly-Based Intrusion Detection System for IoT Networks through Deep Learning Model. Computers and Electrical Engineering, 2022, 99: 107810

[33] G. KIM, H. YI, J. LEE, et al. LSTM-based System-Call Language Modeling and Robust Ensemble Method for Designing Host-Based Intrusion Detection Systems. 2016, arXiv preprint arXiv:1611.01726

[34] A. CHAWLA, B. LEE, S. FALLON, et al. Host Based Intrusion Detection System with Combined CNN/RNN Model. Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Bilbao, ES: Springer, Cham, 2019: 149-158

[35] M. ELMUBARAK, A. KARRAR, N. HASSAN. Implementation Hybrid (NIDS) System Using Anomaly Holt-winter Algorithm and Signature Based Scheme. International Journal of Advances in Scientific Research and Engineering (IJASRE), 2019, 5(6): 141-148

[36] M. JELIDI, A. GHOURABI, K. GASMI. A Hybrid Intrusion Detection System for Cloud Computing Environments. 2019 International Conference on Computer and Information Sciences (ICCIS). Aljouf, Kingdom of Saudi Arabia. IEEE, 2019: 1-6

[37] T. TUGLULAR, E. H. SPAFFORD. A Framework for Characterization of Insider Computer Misuse. Unpublished paper, Purdue University, 1997

[38] E. E. SCHULTZ. A Framework for Understanding and Predicting Insider Attacks. Computers & security, 2002, 21(6): 526-531

[39] P. NING, K. SUN. How to Misuse AODV: A Case Study of Insider Attacks Against Mobile Ad-Hoc Routing Protocols. Ad Hoc Networks, 2005, 3(6): 795-819

[40] C. DANIEL, A. MICHAEL, C. MATTHEW, et al. An Insider Threat Indicator Ontology. Technical Report CMU/SEI-2016-TR-007. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA

[41] B. WOOD. An Insider Threat Model for Adversary Simulation. SRI International, Research on Mitigating the Insider Threat to Information Systems.2000,2:1-3

[42] D. B. PARKER. Fighting Computer Crime: A New Framework for Protecting Information. New York, NY, USA: John Wiley & Sons, Inc.1998.

[43] J. S. PARK, S. M. HO. Composite Role-Based Monitoring (CRBM) for Countering Insider Threats. H. CHEN, D. D. ZENG, R. MOORE, et al. Intelligence and Security Informatics, Proceedings. 2004: 201-213

[44] D. C. LE, N. ZINCIR-HEYWOOD, M. I. HEYWOOD. Analyzing Data Granularity Levels for Insider Threat Detection Using Machine Learning. IEEE Transactions on Network and Service Management, 2020, 17(1): 30-44

[45] S. Yuan, X. Wu. Deep Learning for Insider Threat Detection: Review, Challenges and Opportunities. Computers & Security, 2021, 104: 102221

[46] D. E. DENNING. An Intrusion-Detection Model. IEEE Transactions on Software Engineering, 1987, 13(2): 222-232

[47] A. KHRAISAT, I. GONDAL, P. VAMPLEW. An Anomaly Intrusion Detection System Using C5 Decision Tree Classifier. Trends and Applications in Knowledge Discovery and Data Mining. Springer, Cham, 2018: 149–155

[48] A. KHRAISAT, I. GONDAL, P. VAMPLEW, et al. Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges. Cybersecurity, 2019, 2(1): 1-22

[49] GURUCUL. UEBA. https://gurucul.com/products/user-entity-behavior-analytics- ueba, 2020-3-30

[50] EXABEAM. User and Entity Behavior Analytics. https://www.exabeam.com/siem- guide/ueba, 2020-3-30

[51] Q. CHEN, B. XUE, M. ZHANG. Genetic Programming for Instance Transfer Learning in Symbolic Regression. IEEE Transactions on Cybernetics, 2020, 52(1): 25-38

[52] W. MAO, J. CHEN, Y. CHEN, et al. Construction of Health Indicators for Rotating Machinery Using Deep Transfer Learning with Multiscale Feature Representation. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1-13

[53] M. TENG, W. PEDRYCZ. Knowledge Transfer in Project-Based Organisations: A Dynamic Granular Cognitive Maps Approach. Knowledge Management Research & Practice, 2022, 20(2): 233-250

[54] K. K. BALI, Y. S. ONG, A. GUPTA, et al. Multifactorial Evolutionary Algorithm with Online

Transfer Parameter Estimation: MFEA-II. IEEE Transactions on Evolutionary Computation, 2019, 24(1): 69-83

[55] F. SUNG, Y. YANG, L. ZHANG, et al. Learning to Compare: Relation Network for Few-Shot Learning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE,2018: 1199-1208

[56] S. BARTUNOV, D. VETROV. Few-shot Generative Modelling with Generative Matching Networks. International Conference on Artificial Intelligence and Statistics. Playa Blanca, Lanzarote, Canary Islands: PMLR: 2018: 670-678

[57] J. CHOI, J. KRISHNAMURTHY, A. KEMBHAVI, et al. Structured Set Matching Networks for One-Shot Part Labeling. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018: 3627-3636

[58] S. HOCHREITER, J. SCHMIDHUBER. Long Short-term Memory. Neural Computation, 1997, 9(8): 1735-1780

[59] J. GLASSER, B LINDAUER. Bridging the Gap: A Pragmatic Approach to Generating Insider Threat Data. IEEE. 2013 IEEE Security and Privacy Workshops. San Francisco, CA, USA. Piscataway: IEEE, 2013: 98-104

[60] D. C. LE, A. N. ZINCIR-HEYWOOD. Evaluating Insider Threat Detection Workflow Using Supervised and Unsupervised Learning. 2018 IEEE Security and Privacy Workshops. San Francisco, CA, USA. Piscataway: IEEE, 2018: 270-275

[61] S. F. LOKMAN, A. T. OTHMAN, M. H. A. BAKAR, et al. The Impact of Different Feature Scaling Methods on Intrusion Detection for In-Vehicle Controller Area Network (CAN). Advances in Cyber Security: First International Conference, ACeS 2019, Penang, Malaysia: Springer Singapore, 2020: 195-205

[62] K. SIMONYAN and A. ZISSERMAN. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014, arXiv preprint arXiv:1409.1556

[63] C. SZEGEDY, V. VANHOUCKE, S. IOFFE, et al. Rethinking the Inception Architecture for Computer Vision. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016: 2818-2826

[64] F. CHOLLET. Xception: Deep Learning with Depthwise Separable Convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017: 1251-1258

[65] I. SHARAFALDIN, A. H. LASHKARI, A. A. GHORBANI. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. ICISSp, 2018, 1: 108-116

[66] L. YANG, A. MOUBAYED, I. HAMIEH, et al. Tree-based Intelligent Intrusion Detection System

in Internet of Vehicles. 2019 IEEE Global Communications Conference (GLOBECOM). Hawaii, USA: IEEE, 2019: 1-6

[67] L. YANG, A. MOUBAYED, A. SHAMI. MTH-IDS: A Multitiered Hybrid Intrusion Detection System for Internet of Vehicles. IEEE Internet of Things Journal, 2021, 9(1): 616-632

[68] A. ROSAY, F. CARLIER, P. LEROUX. Feed-Forward Neural Network for Network Intrusion Detection. 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). Antwerp, Belgium: IEEE, 2020: 1-6

[69] W. ELMASRY, A. AKBULUT, A. H. ZAIM. Evolving Deep Learning Architectures for network Intrusion Detection Using a Double PSO Metaheuristic. Computer Networks, 2020, 168: 107042

[70] E. MUSHTAQ, A. ZAMEER, M. UMER, et al. A Two-Stage Intrusion Detection System with Auto-Encoder and LSTMs. Applied Soft Computing, 2022, 121: 108768

[71] D. ČEPONIS, N. GORANIN. Investigation of Dual-Flow Deep Learning Models LSTM-FCN and GRU-FCN Efficiency Against Single-Flow CNN Models for the Host-Based Intrusion and Malware Detection Task on Univariate Times Series Data. Applied Sciences, 2020, 10(7): 2373

[72] W. G. C. BANDARA, V. M. PATEL. A Transformer-Based Siamese Network for Change Detection. IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium. Kuala Lumpur, Malaysia: IEEE, 2022: 207-210

[73] H. HUANG, G. LIU, Y. ZHANG, et al. Ensemble Siamese Networks for Object Tracking. Neural Computing and Applications, 2022, 34(10): 8173-8191

[74] B. HUANG, A. ALHUDHAIF, F. ALENEZI, et al. Balance Label Correction Using Contrastive Loss. Information Sciences, 2022, 607: 1061-1073

[75] 张思聪,谢晓尧,徐洋.基于 dCNN 的入侵检测方法.清华大学学报(自然科学版), 2019, 59(01): 46-54