斯普林格简报
网络安全系统和网络

Kwangjo Kim
Muhamad Erza Aminanto
Harry Chandra Tanuwidjaja

使用深度学习的网络入侵检测
一种特征学习方法

Springer

斯普林格网络安全系统与网络简报

主编

澳大利亚维多利亚州墨尔本霍桑斯威本科技大学数字研究与创新能力平台的杨翔

系列编辑

陈立群，英国萨里大学，吉尔福德
Kim-Kwang Raymond Choo，信息系统与网络安全系，美国德克萨斯州圣安东尼奥大学Sherman S. M. Chow，信息工程系，香港中文大学，香港

Robert H. Deng，信息系统学院，新加坡管理大学，新加坡

Dieter Gollmann，汉堡科技大学，德国汉堡
Javier Lopez，马拉加大学，西班牙马拉加
Kui Ren，布法罗大学，美国纽约州布法罗
周建英，新加坡科技与设计大学，新加坡，新加坡

该系列旨在发展和传播有关网络安全系统和网络相关研究和研究的创新、范式、技术和技术的理解。 它发表了关于网络安全的最新主题的全面而有凝聚力的概述，以及关于网络系统的原创研究报告和深入案例研究。 该系列还提供了一个单一的覆盖点，涵盖了先进和及时的新兴主题，以及可能尚未达到成熟水平的核心概念的论坛。 它解决了网络系统和网络的安全、隐私、可用性和可靠性问题，并欢迎与网络安全研究相关的人工智能、云计算、网络物理系统和大数据分析等新兴技术。 主要关注以下研究主题：

基础理论

- 密码学与网络安全
- 网络安全理论
- 可证明安全性

网络系统与网络

- 网络系统安全
- 网络安全
- 安全服务
- 社交网络安全与隐私
- 网络攻击与防御
- 数据驱动的网络安全
- 可信计算与系统

应用与其他

- 硬件和设备安全
- 网络应用安全
- 网络安全的人类和社会因素

有关该系列的更多信息，请访问http://www.springer.com/series/15797

Kwangjo Kim • Muhamad Erza Aminanto
Harry Chandra Tanuwidjaja

使用深度学习的网络入侵检测

一种特征学习方法

Springer

Kwangjo Kim计算机学
院韩国科学技术高级学院
大田，韩国（共和国）

Muhamad Erza Aminanto
计算机学院韩国科学技术
高级学院大田，韩国（共
和国）

Harry Chandra Tanuwidjaj
a计算机学院韩国科学技
术高级学院大田，韩国（
共和国）

ISSN 2522-5561 ISSN 2522-557X（电子版）
斯普林格网络安全系统与网络简报
ISBN 978-981-13-1443-8 ISBN 978-981-13-1444-5（电子书）
https://doi.org/10.1007/978-981-13-1444-5

图书馆国会控制号码：2018953758

© 作者（们），在Springer Nature Singapore Pte Ltd. 2018的独家许可下
本作品受版权保护。 出版商保留所有权利，无论是全部还是部分
材料，特别是翻译、重印、插图重用、背诵、
广播、微缩胶片复制或以任何其他实体方式复制、
信息传输或存储和检索、电子适应、计算机软件，或者通过类似或不同的方法
现在已知或今后开发。
在本出版物中使用一般描述性名称、注册名称、商标、服务标志等
并不意味着，即使在没有特定声明的情况下，这些名称免于相关的
保护法律和法规，并因此可以自由使用。
出版商、作者和编辑可以安全地假设本书中的建议和信息在出版日期时是真实准确的。 出版商
、作者或编辑对本书中所含材料不提供任何明示或暗示的保证，也不对可能存在的任何错误或
遗漏负责。 出版商在已发表的地图和机构隶属方面保持中立。

这本Springer印记由注册公司Springer Nature Singapore Pte Ltd.出版。注册公司地址是：15
2 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

感谢我们的家人给予的可爱支持。

前言

本专著介绍了使用深度学习模型进行入侵检测系统（IDS）的最新进展，这些模型在计算机视觉、自然语言处理和图像处理等领域取得了巨大成功。

本专著系统而有条理地概述了深度学习的最新发展，并对基于深度学习的IDS进行了比较。本专著还提供了深度学习应用于IDS的全面概述，包括新颖的深度特征提取和选择以及深度学习聚类方法。 本专著还介绍了进一步的挑战和研究方向。

这本专著为基于深度学习的入侵检测系统提供了丰富的概述，适合对深度学习和入侵检测感兴趣的学生、研究人员和从业者作为参考书。 对各种深度学习应用的全面比较有助于读者对机器学习有基本了解，并激发在入侵检测和其他网络安全领域的应用。

本专著的大纲如下：

第1章通过对计算机网络中的安全漏洞进行调查，描述了当今计算机网络中入侵检测的重要性。 重点强调了深度学习模型可以提高入侵检测系统的性能。 它还解释了调查基于深度学习的入侵检测系统的动机。

第2章提供了所有相关的入侵检测系统的定义。然后根据检测模块的放置位置和使用的方法解释了当前入侵检测系统的不同类型。 本章还提供了常见的性能指标和公开可用的基准数据集。

第3章对经典机器学习进行了简要的初步研究，包括有监督、无监督、半监督、弱监督、强化和对抗机器学习。 它简要调查了22篇使用机器学习技术进行入侵检测的论文。

第4章讨论了几种包含生成、判别和混合方法的深度学习模型。

第5章调查了利用深度学习模型的各种IDS，这些IDS分为四类：生成、判别、混合和深度强化学习。

第6章讨论了深度学习模型作为IDS研究中的特征学习（FL）方法的重要性。 我们进一步解释了两个模型，即深度特征提取和选择以及深度学习聚类。

第7章通过概述深度学习在IDS中的挑战和未来研究方向来总结本专著。 附录讨论了使用深度学习模型进行网络恶意软件检测的几篇论文。 由于恶意软件数量的增加和IDS类似的方法，恶意软件检测也是一个重要问题。

大田，韩国
2018年3月

Kwangjo Kim
Muhamad Erza Aminanto
Harry Chandra Tanuwidjaja

致谢

本专著部分得到了韩国政府（MSIT）资助的韩国信息与通信技术促进研究所（IITP）资助（2013-0-00396，基于生物启发算法的通信技术研究，以及2017-0-00555，基于格的可证明安全的多方认证密钥交换协议在量子世界中），以及韩国政府（MSIT）资助的韩国国家研究基金会（NRF）资助（编号：NRF-2015R1A-2A2A01006812）。

我们非常感谢KAIST电气工程学院的洪植洙教授，他为我们提供了将深度学习应用于安全无线网络入侵检测的绝佳机会。

我们还要感谢英国克兰菲尔德国防与安全学院的Paul D. Yoo教授和Taufiq Asyhari教授，在我们的研究工作期间给予我们启发性的讨论。

作者真诚感谢加密学和信息安全实验室（CAISLAB）的校友和现任成员，感谢韩国科学技术院（KAIST）信息安全研究生院、计算机学院的贡献。感谢 Khalid Huseynov、Dongsoo Lee、Kyungmin Kim、Hakju Kim、Rakyong Choi、Jeeun Lee、Soohyun Ahn、Joonjeong Park、Jeseoung Jung、Jina Hong、Sungsook Kim、Edwin Ayisi Opare、Hyeongcheol An、Seongho Han、Nakjun Choi、Nabi Lee 和 Dongyeon Hong。

我们衷心感谢本专著系列的编辑们对我们宝贵的意见，并感谢 Springer 出版社给予我们撰写本专著的机会。

最后，我们也非常感谢我们的家人对我们的坚强支持和无尽的爱。

目录

缩略语

ACA	蚂蚁聚类算法
ACC	蚁群聚类
AE	自编码器
AIS	人工免疫系统
ANN	人工神经网络
APT	高级持续性威胁
ATTA-C	自适应时变传输蚂蚁聚类
AWID	爱琴海Wi-Fi入侵数据集
BM	玻尔兹曼机
CAN	控制器局域网
CCN	内容中心网络
CFS	CfsSubsetEval
CNN	卷积神经网络
CoG	重心
Corr	相关性
CPS	网络物理系统
DAE	去噪自编码器
DBM	深度玻尔兹曼机
DBN	深度置信网络
DDoS	分布式拒绝服务
D-FES	深度特征提取和选择
DNN	深度神经网络
DoS	拒绝服务
DR	检测率
DT	决策树
ERL	进化强化学习
ESVDF	增强支持向量决策函数
FIS	模糊推理系统
FL	特征学习
FN	假阴性
FNR	假阴性率

FP	假阳性
FPR	假阳性率
FW	防火墙
GAN	生成对抗网络
GPU	图形处理单元
GRU	门控循环单元
HJI	哈密顿-雅可比-艾萨克
HIS	人类推理系统
ICV	完整性检查值
IDS	入侵检测系统
IG	信息增益
IoT	物联网
IPS	入侵预防系统
IV	初始化向量
JSON	JavaScript对象表示法
KL	库尔巴克-莱布勒
kNN	K最近邻算法
LoM	最大最大值
LSTM	长短期记忆
MDP	马尔可夫决策过程
METIS	移动和无线通信使能技术：面向二十一世纪信息社会MF
	隶属函数
MLP	多层感知器
MoM	最大值的平均
MSE	均方误差
神经网络	神经网络
粒子群优化	远程到本地
R2L	限制玻尔兹曼机
RBM	限制玻尔兹曼机
RL	强化学习
循环神经网络	循环神经网络
SAE	堆叠自编码器
SDAE	堆叠去噪自编码器
SDN	软件定义网络
SFL	监督特征学习
SGD	随机梯度下降
SNN	共享最近邻
自组织映射	自组织映射
自组织映射	最大的最小
SPN	和积网络
STL	自学习
支持向量机	支持向量机
SVM-RFE	支持向量机递归特征消除
TBM	建立模型的时间

传输控制协议/互联网协议	传输控制协议/互联网协议
TN	真阴性
TP	真阳性
TT	测试时间
U2R	用户到根
UFL	无监督特征学习

第一章
引言

摘要 本章讨论了IDS在计算机网络中的重要性，同时也提供了对无线网络中安全漏洞的调查。 许多方法已经被用来提高IDS的性能，其中最有希望的方法是部署机器学习。 然后，强调了最近的机器学习模型——深度学习——在提高IDS性能方面的用途，特别是作为一种特征学习（FL）方法。 我们还解释了调查基于深度学习的IDS的动机。

计算机网络和互联网与人类生活密不可分。 丰富的应用程序依赖于互联网，包括医疗保健和军事中的生命关键应用。 此外，每天都有大量的金融交易在互联网上进行。近年来，互联网的快速增长导致了无线网络流量的显著增加。 根据全球电信联盟的《二十二世纪信息社会的移动和无线通信促进者（METIS）》[1]，未来几十年将出现5G和Wi-Fi网络的大量增加。 他们认为，由于社会需求的发展，移动和无线流量的雪崩式增长将会发生。 诸如在线学习、网上银行和电子健康等应用将会传播并变得更加移动化。 到2020年，无线网络流量预计将占据总互联网流量的三分之二，其中66%的IP流量预计仅由Wi-Fi和移动设备产生。

随着物联网的广泛应用，网络攻击以惊人的速度增长[2]。

IBM [3] 在2016年报告了一起巨大的账户劫持事件，垃圾邮件的数量比前一年增加了四倍。 常见的攻击在同一领域中被注意到

1 Cisco视觉网络指数：2015-2020年预测和方法，发布于www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

© 作者（们），独家许可给Springer Nature Singapore Pte Ltd. 2018年K. Kim等人，使用深度学习进行网络入侵检测，SpringerBriefs on Cyber Security Systems and Networks，https://doi.org/10.1007/978-981-13-1444-5_1

报告中包括暴力破解、恶意广告、钓鱼、SQL注入、DDoS、恶意软件等。 大多数恶意软件是勒索软件（一年中85%的恶意软件是勒索软件）。 这些攻击可能泄露敏感数据或破坏正常运营，导致巨大的财务损失。 受安全事件影响最严重的公司是金融服务相关公司，其次是信息和通信、制造、零售和医疗保健[3]。 IEEE 802.11等无线网络已广泛部署，为用户提供高速本地区域连接的移动性和灵活性。 然而，隐私和安全等其他问题也引起了关注。 物联网设备的快速传播导致无线网络成为被动和主动攻击的目标，攻击数量大幅增长[2]。 这些攻击的例子包括冒充、洪水和注入攻击。 使用Wi-Fi网络的计算设备的广泛和快速传播导致了复杂、大规模和高维度的数据，这在捕获攻击属性时会产生混淆，并迫使我们加强系统的安全措施。

已经进行了全面的研究，以避免如前面提到的攻击。 入侵检测系统（IDS）是每个网络安全基础设施[4]，包括无线网络[5]中最常见的组件之一。 由于其无模型特性和可学习性[6]，机器学习技术已经被广泛采用作为IDS中的主要检测算法。 利用最近深度学习等机器学习技术的发展，可以预期在改进现有IDS特别是在大规模网络中检测冒充攻击方面带来显著的好处。 基于检测方法，IDS可以分为三种类型：误用型、异常型和规范型IDS。误用型IDS也被称为基于签名的IDS[8]，通过检查攻击特征是否与先前存储的攻击签名或模式匹配来检测任何攻击。 这种类型的IDS适用于检测已知攻击；然而，新的或未知的攻击很难检测到。

在学术界和工业界都进行了大量研究，应用机器学习方法来进行入侵检测系统（IDS）的研究。 然而，安全专家们仍在追求具有最高检测率（DR）和最低误报率的性能更好的IDS。 此外，整体威胁分析有望保护他们的网络[9]。 通过采用最新的机器学习技术——深度学习[6]，可以改进IDS的性能。 深度学习应用在模式识别和机器学习方面赢得了许多比赛[10]。

深度学习属于一类机器学习方法，它以层次化的方式使用连续的信息处理阶段进行模式分类和特征学习[11]。 根据[12]，深度学习近年来备受关注有三个重要原因。 首先，处理能力（例如GPU单元）大幅提升。 其次，计算硬件变得更加经济实惠，第三个原因是机器学习研究的最新突破。 浅层学习和深度学习的区别在于它们的信用分配路径的深度，这些路径是可能可学习的因果链接的链条，连接了动作和效果。 通常，深度学习在图像分类结果中起着重要作用。 此外，深度学习也常用于语言处理。

图形建模，模式识别，语音，音频，图像，视频，自然语言和信号处理[11]。 有许多深度学习方法，如深度置信网络（DBN），玻尔兹曼机（BM），受限玻尔兹曼机（RBM），深度玻尔兹曼机（DBM），深度神经网络（DNN），自动编码器，深度自动编码器（DAE），堆叠自动编码器（SAE），堆叠去噪自动编码器（SDAE），分布式表示和卷积神经网络（CNN）。 谷歌的AlphaGo [13]是其中一个著名的应用，它使用了CNN。AlphaGo最近在“围棋”比赛中击败了韩国世界冠军，展示了远程机器学习的超人能力。

学习算法的进步可能提高IDS的性能，
达到更高的DR和更低的误报率。

计算设备在互联网上的广泛和快速传播，特别是Wi-Fi网络，产生了复杂、大规模和高维数据，这在捕捉攻击特性时会产生不可避免的混淆。 FL作为改进机器学习模型学习过程的重要工具。 它包括特征构建、提取和选择。 特征构建扩展了原始特征以增强其表达能力，而特征提取将原始特征转化为新形式，特征选择则消除了不必要的特征[14]。 FL是提高现有基于机器学习的IDS性能的关键。

我们意识到在IDS应用中如何正确采用深度学习存在困惑，因为每个先前的方法都采用了不同的方法。一些研究仅部分使用深度学习方法，而其他研究仍然使用传统的神经网络。 深度学习方法的复杂性可能是其中之一的原因。 此外，深度学习方法需要大量时间来正确训练。 然而，我们发现一些研究人员在其网络中采用深度学习方法进行特征学习和分类，用于智能IDS。

我们对它们之间的IDS性能进行了比较。

Tran等人提供了一个关于如何在IDS中使用深度学习的例子。使用遗传编程生成了经典的机器学习算法Naive Bayes和C4.5，并辅以高级特征。 这种方法是在IDS中利用深度学习模型的常见方式，其中深度学习模型辅助任何具有高级特征的经典机器学习。 这种方法也被Aminanto等人采用，详细说明见第6章。 在本专著中，我们重点介绍了一些使用深度学习模型的IDS。Hamed等人对IDS研究中的几种预处理技术进行了调查，包括如何从真实世界和蜜罐中收集数据以及如何从原始输入数据构建数据集。 尽管大多数IDS使用深度学习模型作为其数据预处理技术，但本专著重点回顾了基于深度学习的IDS。

深度学习在IDS中非常有益，特别是对于FL。本专著通过分析这些深度学习方法及其优缺点，以更好地理解如何将深度学习应用于IDS。最后，我们提供了未来的挑战和方向，以相应地应用深度学习于IDS。

参考文献

1. A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, H. Tullberg, M. A. Uusitalo, B. Timus, and M. Fallgren, "Scenarios for 5G mobile and wireless communications: The vision of the metis project," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 26–35, May 2014.
2. C. Kolias, A. Stavrou, J. Voas, I. Bojanova, and R. Kuhn, "学习物联网安全" "实践"， *IEEE Security Privacy*, vol. 14, no. 1, pp. 37–46, 2016.
3. M. Alvarez, N. Bradley, P. Cobb, S. Craig, R. Iffert, L. Kessem, J. Kravitz, D. McMilen, and S. Moore, "IBM X-force威胁情报指数2017年,"*IBM*公司, pp. 1–30, 2017.
4. C. Kolias, G. Kambourakis, and M. Maragoudakis, "入侵检测中的群体智能：一项调查,"计算机与安全, vol. 30, no. 8, pp. 625–642, 2011.
5. A. G. Fragkiadakis, V. A. Siris, N. E. Petroulakis, and A. P. Traganitis, "基于异常的干扰攻击入侵检测：本地与协作检测的比较,"无线通信和移动计算, vol. 15, no. 2, pp. 276–294, 2015.
6. R. Sommer和V. Paxson，"在封闭世界之外：关于使用机器学习进行网络入侵检测的研究"，在*Proc. Symp. Security and Privacy, Berkeley, California*. IEEE, 2010,pp. 305–316.
7. G. Anthes，"深度学习的时代到来了"， *Communications of the ACM*, vol. 56, no. 6, pp. 13–15, 2013.
8. A. H. Farooqi和F. A. Khan，"无线传感器网络的入侵检测系统：一项调查研究"，在*Proc. Future Generation Information Technology Conference, Jeju Island, Korea*.Springer, 2009, pp. 234–241.
9. R. Zuech，T. M. Khoshgoftaar和R. Wald，"入侵检测和大型异构数据：一项调查研究"，*Journal of Big Data*, vol. 2, no. 1, p. 3, 2015.
10. J. Schmidhuber，"神经网络中的深度学习：概述"，神经网络，卷61，页85-117，2015年。
11. L. Deng，"深度学习的架构、算法和应用的教程调查"，*APSIPA*信号与信息处理交易，卷3，2014年。
12. L. Deng，D. Yu，等，"深度学习：方法和应用"，信号处理基础与趋势®，卷7，第3-4期，页197-387，2014年。
13. D. Silver，A. Huang，C. J. Maddison，A. Guez，L. Sifre，G. Van Den Driessche，J. Schrittwieser，I. Antonoglou，V. Panneershelvam，M. Lanctot，等，"用深度神经网络和树搜索掌握围棋"，自然，卷529，第7587期，页484-489，2016年。
14. H. Motoda 和 H. Liu，"特征选择、提取和构建"，*IIC*M（信息与计算机机构研究所）通信，台湾，第5卷，第67-72页，2002年。
15. B. Tran、S. Picek 和 B. Xue，"网络入侵检测的自动特征构建"，亚太模拟进化与学习会议，Springer，2017年，第569-580页。
16. M. E. Aminanto、R. Choi、H. C. Tanuwidjaja、P. D. Yoo 和 K. Kim，"用于Wi-Fi冒充检测的深度抽象和加权特征选择"，*I*EEE信息取证与安全交易，第13卷，第3期，第621-636页，2018年。
17. T. Hamed、J. B. Ernst 和 S. C. Kremer，"入侵检测系统的数据和预处理技术调查和分类"，计算机和网络安全要点，Springer，2018年，第113-134页。

第二章
入侵检测系统

摘要本章简要介绍入侵检测系统（IDS）的相关定义，然后根据检测模块的位置和采用的方法对当前IDS进行分类。 我们还解释并提供了一个在研究领域中常见的基于机器学习的IDS的示例。然后，我们讨论了使用生物启发式聚类方法的IDS的示例。

2.1 定义

IDS成为计算机网络中的标准安全措施。 与图2.1a中的防火墙（FW）不同，IDS通常位于网络内部，以监视所有内部流量，如图2.1b所示。可以考虑同时使用防火墙和IDS来有效保护网络。 IDS被定义为自主的入侵检测过程，其目的是在计算机网络中发现违反安全策略或标准安全实践的事件[1]。 除了识别安全事件，IDS还具有其他功能：记录现有威胁和威慑对手[1]。 IDS需要具备特定的属性，作为一种被动的对策，仅监视整个或部分网络，并旨在实现高攻击检测率和低误报率。

2.2 分类

我们可以根据放置和部署在网络中的方法来划分IDS。 通过IDS模块在网络中的位置，我们可以将IDS分为三类：基于网络的IDS，基于主机的IDS和混合IDS。第一个IDS，基于网络的IDS如图2.2所示，将IDS模块放置在整个网络中进行监控。 这个IDS通过检查网络中传输的所有数据包来检测恶意活动。另一方面，图2.3显示了

© 作者，独家许可给Springer Nature Singapore Pte Ltd. 2018年K. Kim等人，使用深度学习进行网络入侵检测，SpringerBriefs on Cyber Security Systems and Networks，https://doi.org/10.1007/978-981-13-1444-5_2

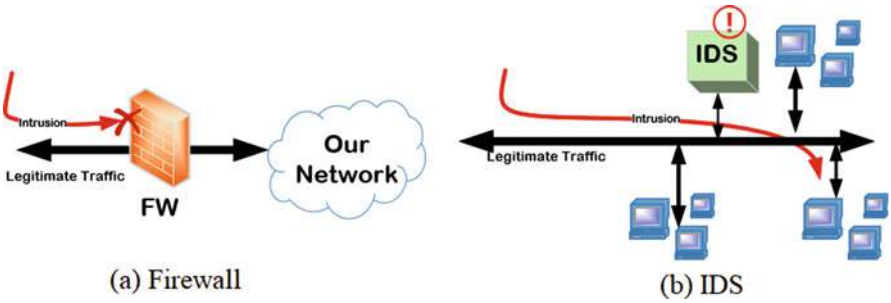


图**2.1** 典型网络使用(**a**)防火墙和(**b**)IDS

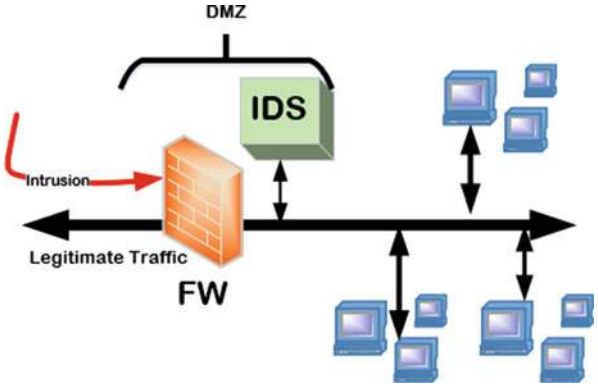


图**2.2** 基于网络的入侵检测系统

主机型入侵检测系统将入侵检测模块放置在网络的每个客户端上。 该模块检查相应客户端的所有入站和出站流量，从而实现对特定客户端的详细监控。 两种类型的入侵检测系统都有特定的缺点——基于网络的入侵检测系统可能会增加工作负荷，从而错过一些恶意活动，而基于主机的入侵检测系统则无法监控所有网络流量，工作负荷比基于网络的入侵检测系统要小。因此，如图2.4所示的混合型入侵检测系统将入侵检测模块放置在网络和客户端中，同时监控特定客户端和网络活动。

根据检测方法，入侵检测系统可以分为三种不同类型：误用型、异常型和规范型入侵检测系统。误用型入侵检测系统，也称为基于签名的入侵检测系统[2]，通过将已知攻击的签名或模式与监控的流量进行匹配来寻找任何恶意活动。 这种入侵检测系统适用于已知攻击的检测；然而，新的或未知的攻击（也称为零日攻击）很难被检测到。 异常型入侵检测系统通过对正常行为进行建模，如果有任何偏离，则触发警报。该入侵检测系统的优势在于其对未知攻击的检测能力。 与异常型入侵检测系统相比，误用型入侵检测系统通常对已知攻击的检测性能更高。规范型入侵检测系统手动定义一组规则和约束来表达正常操作。 任何违反规则和约束的行为都可能被视为潜在的入侵行为。

图**2.3** 基于主机的入侵检测系统

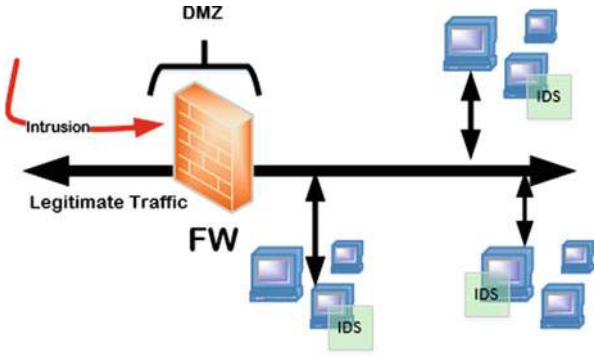


图**2.4** 混合入侵检测系统

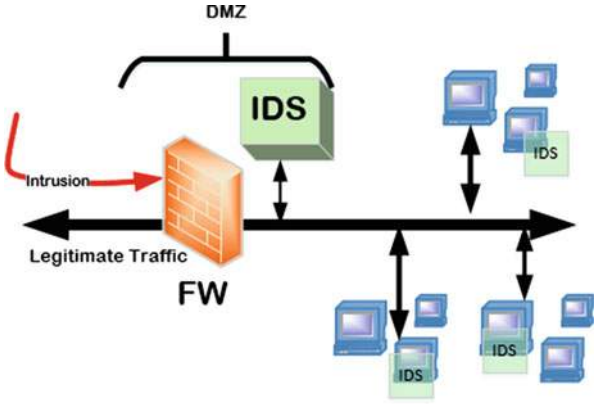


表**2.1** 基于方法论的入侵检测系统比较

	基于误用的	基于异常的	基于规范的
方法	识别已知的攻击模式	识别异常的活动模式	识别违反预定义规则的行为
DR	高	低	高
误报率	低	高	低
无法检测未知攻击		能够检测未知攻击	无法检测未知攻击
缺点	更新签名是繁重的	计算任何机器学习都很耗费资源	在定义规则时依赖专家知识是不可取的

在执行过程中受到限制的行为被标记为恶意[3]。 表2.1总结了基于方法论的入侵检测系统的比较。

我们进一步讨论了属于基于异常的入侵检测系统的基于机器学习的入侵检测系统[4]。 有两种类型的学习，即有监督学习和无监督学习。 无监督学习不需要标记的数据集进行训练，这对于最近的大规模网络流量至关重要，而有监督学习则需要标记的数据集。 无监督学习能力非常重要，因为它允许模型在不创建昂贵的标签或依赖的情况下检测新的攻击。

表**2.2** 监督学习和无监督学习的比较

	监督	无监督
定义	数据集带有预定义的类别标签	数据集没有预定义的类别标签
方法	分类	聚类
方法	支持向量机，决策树等	K均值聚类，蚁群聚类算法等
已知攻击检测	高	低
未知攻击检测	低	高

变量。 表2.2概述了监督学习和无监督学习的比较。

2.3基准

本节讨论IDS研究中的基准技术。 基准数据集和统一的性能指标对于评估和比较两个或多个模型至关重要。 通过公平比较，我们可以确定任何提出的方法的改进。

2.3.1 性能指标

可以通过采用常见的模型性能度量[5]来评估任何IDS的性能：准确率（Acc），DR，误报率（FAR），Mcc，精确率， F_1分数，构建模型的CPU时间（TBM）和测试的CPU时间（TT）。Acc显示算法的整体效果[6]。DR，也称为 $Recall$，指的是在测试数据集中检测到的模拟攻击实例数除以总模拟攻击实例数。 与 $Recall$不同，$Precision$计算的是被分类为攻击的实例数除以总实例数。 F_1分数衡量了$Precision$和 $Recall$的调和平均值。FAR是将正常实例分类为攻击的数量除以测试数据集中的正常实例总数，而FNR表示无法检测到的攻击实例数。

Mcc代表检测到的数据与观测数据之间的相关系数[7]。 直观上，目标是实现高准确率、召回率、精确率、相关系数和 F_1得分，并同时保持低误报率、漏报率和漏检率。上述指标可以通过公式(2.1)、(2.2)、(2.3)、(2.4)、(2.5)、(2.6)和(2.7)来定义：

$$\text{准确率} = \frac{\text{真正例} + \text{真负例}}{\text{真正例} + \text{真负例} + \text{假正例} + \text{假负例}}, \tag{2.1}$$

$$召回率 = \frac{真正例}{真正例 + 假负例,} \tag{2.2}$$

$$精确率 = \frac{真正例}{真正例 + 假正例,} \tag{2.3}$$

$$误报率 = \frac{FP}{真负例 + 假正例,} \tag{2.4}$$

$$漏报率 = \frac{FN}{假负例 + 真正例,} \tag{2.5}$$

$$F_1 = \frac{2TP}{2TP + FP + FN}, \tag{2.6}$$

$$Mcc = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{2.7}$$

其中，真正例（TP）是正确分类为攻击的入侵数量，真负例（TN）是正确分类为正常数据包的正常实例数量，假负例（FN）是错误分类为正常数据包的入侵数量，假正例（FP）是错误分类为攻击的正常实例数量。

2.3.2 公共数据集

基准数据集是评估IDS模型有效性的一个重要方面。KDD Cup'99数据集一直是评估异常检测方法的最流行数据集[8]。 该数据集基于DARPA'98 IDS评估计划中捕获的数据，包含大约4,900,000个单个连接实例。 表2.3显示了KDD Cup 99数据集的数据包分布[9]。 每个实例包含41个特征，并被标记为正常实例或攻击实例。 该数据集提供了以下四种不同的攻击类型：

1.探测攻击：攻击者试图收集有关计算机网络的信息，以绕过安全控制。 探测攻击的一个例子是端口扫描。

2.拒绝服务（**DoS**）攻击：攻击者阻止合法用户访问授权数据的攻击。 攻击者通过向网络发送不必要的数据包请求来消耗计算资源，使其无法处理合法请求。 拒绝服务攻击的一个例子是**S**YN洪水攻击。

3.用户到根（**U2R**）攻击：攻击者通过访问系统上的普通用户帐户开始攻击。 然后，攻击者利用漏洞获取系统的根访问权限。 用户到根攻击的一个例子是x**ter**m漏洞利用。

表**2.3** KDD Cup'99数据集的数据包分布

类型	数据包数量	比例（%）
正常	972,781	19.86
探测	41,102	0.84
拒绝服务	3,883,370	79.28
U2R	52	0.00
R2L	1,126	0.02
总计	4,898,431	100

4.远程到本地（**R2L**）攻击：这种攻击是由一个能够通过网络向机器发送数据包但没有该机器上账户的攻击者执行的。 攻击者利用一些漏洞远程获得该机器上的本地访问权限。 **R2L**攻击的一个例子是f **t**p_*write*利用。

尽管KDD Cup'99数据集非常有用，但该数据集存在一些统计缺陷，即冗余实例和训练和测试数据集中特定类别的不合理分布。 因此，NSL-KDD数据集被开发出来以改进原始的KDD Cup'99数据集的限制[10]。 NSL-KDD数据集也有与原始数据集相同的41个属性。

在Wi-Fi网络区域，有一个名为Aegean Wi-Fi入侵数据集（AWID）的数据集，由Kolias等人开发[11]。 AWID数据集有两种类型。 第一种类型称为“CLS”，有四个目标类别，而第二种类型称为“ATK”，有16个目标类别。 “ATK”数据集的16个类别属于“CLS”数据集中的四个攻击类别。 例如，在“ATK”数据集中列出的Caffe-Latte、*H*irte、*H*oneypot和*E*vilTwin攻击类型被归类为“CLS”数据集中的冒充攻击。 根据包含的数据实例的大小，AWID数据集包括完整版本和精简版本。

在减少的训练数据集中有1,795,595个实例，其中1,633,190个是正常实例，162,385个是攻击实例。 在减少的测试数据集中有575,643个实例，其中530,785个是正常实例，44,858个是攻击实例，如表6.1所示（见第6章）。

参考文献

1. K. Scarfone和P. Mell，“入侵检测和预防系统（*I*DPS）指南”，*NIST*特刊，第800卷，第2007号，2007年。
2. A. H. Farooqi和F. A. Khan，“无线传感器网络的入侵检测系统：一项调查”，在未来信息技术会议上，韩国济州岛，Springer出版社，2009年，第234-241页。
3. R. Mitchell和I. R. Chen，“基于行为规则规范的安全关键医疗网络物理系统入侵检测”，I*EEE*可靠安全计算期刊，第12卷，第1期，第16-30页，2015年1月。
4. I. Butun, S. D. Morgera, 和 R. Sankar, “无线传感器网络中入侵检测系统综述,”I*EEE* 通信调查与教程, vol. 16, no. 1, pp. 266–282, 2014.

5. O. Y. Al-Jarrah, O. Alhussein, P. D. Yoo, S. Muhaidat, K. Taha, 和 K. Kim, “用于僵尸网络入侵检测的数据随机化和基于簇的分区,”*IEEE* 交易网络安全, vol. 46, no. 8, pp. 1796–1806, 2015.
6. M. Sokolova, N. Japkowicz, 和 S. Szpakowicz, “超越准确率、F-分数和ROC：一族用于性能评估的判别度量方法,” in澳大利亚人工智能联合会议, 澳大利亚霍巴特. Springer, 2006, pp. 1015–1021.
7. P. D. Schloss和S. L. Westcott，“评估和改进用于16s rRNA基因序列分析的操作taxonomic unit-based方法”，应用和环境微生物学，第77卷，第10期，第3219-3226页，2011年。
8. M. Tavallaee，E. Bagheri，W. Lu和A.-A. Ghorbani，“对kdd cup99数据集的详细分析”，第二届*I*EEE计算智能安全与防御应用研讨会论文集，第53-58页，*2009*年。
9. H. M. Shirazi，“使用遗传算法和特征选择的智能入侵检测系统”，Ma*j*lesi电气工程杂志，第4卷，第1期，2010年。
10. S. Potluri和C. Diedrich，“用于增强入侵检测系统的加速深度神经网络”，在*E*mer*g*in*g* *T*echnolo*g*ies and *F*actory Automation *(ETF*A)，*2016 IEEE*第*2*1届国际会议上。*IEEE*，*2016*年，第1-8页。
11. C. Kolias, G. Kambourakis, A. Stavrou, 和 S. Gritzalis, “802.11网络中的入侵检测：威胁的实证评估和公共数据集”，I*EEE*通信调查教程，卷18，号1，页184-208，2015年。

第3章
经典机器学习及其在IDS中的应用

摘要本章提供了关于经典机器学习的简要初步研究，其中包括六种不同的模型：有监督、无监督、半监督、弱监督、强化和对抗机器学习。然后，对使用机器学习技术进行IDS的22篇论文进行了调研。

3.1 机器学习的分类

异常检测型IDS中使用了不同类型的机器学习模型。本章提供了关于经典机器学习模型的初步研究。 机器学习可以根据训练数据类型分为五种不同的模型：有监督、无监督、半监督、弱监督和强化学习（RL）。 以下子章节对每个模型进行了进一步的解释。 此外，我们还讨论了几种基于机器学习的IDS。

3.1.1 监督学习

在监督学习中，训练过程中需要目标类别数据。 网络会根据与标签数据的匹配正确性建立模型。 在监督学习中有许多机器学习模型，其中一些如下所示。

3.1.1.1 支持向量机

监督支持向量机通常用于分类或回归任务。 如果 n是输入特征的数量，则支持向量机将每个特征值绘制为坐标点。

© 作者，独家许可给 Springer Nature Singapore Pte Ltd. 2018

K. Kim 等人，使用深度学习进行网络入侵检测，
SpringerBriefs on Cyber Security Systems and Networks,
https://doi.org/10.1007/978-981-13-1444-5_3

n维空间。 随后，通过找到区分两个类别的超平面执行分类过程。 虽然支持向量机可以处理任意复杂度的非线性决策边界，但由于数据集的性质可以通过线性判别分类器进行研究，因此我们使用线性支持向量机。 线性支持向量机的决策边界是二维空间中的一条直线。 支持向量机的主要计算特性是支持向量，它们是离决策边界最近的数据点。 输入向量$\mathbf{x}$的决策函数 D($\mathbf{x}$) [1] 如公式 (3.1) 所示，严重依赖于支持向量。

$$D(\mathbf{x}) = \mathbf{w}\mathbf{x} + b \tag{3.1}$$

$$\mathbf{w} = \sum_{k} \alpha_k y_k \mathbf{x}_k \tag{3.2}$$

$$b = (y_k - \mathbf{w}\mathbf{x}_k), \tag{3.3}$$

其中，$\mathbf{w}$、y、α和b分别表示权重向量、类标签、边际支持向量和偏置值。k表示样本数量。
方程（3.2）和（3.3）展示了如何计算$\mathbf{w}$和b的值。

SVM-递归特征消除（RFE）是使用权重的大小进行排名聚类的一种应用[1]。 RFE对特征集进行排名，并消除对分类任务贡献较小的低排名特征[2]。

3.1.1.2 决策树

C4.5对噪声数据具有鲁棒性，并能够学习离散表达式[3]。 它具有k叉树结构，可以通过每个节点对输入数据的属性进行测试。 树的每个分支显示潜在选择的重要特征作为节点的值和不同的测试结果。 C4.5使用贪婪算法以自顶向下的递归分治方法构建树[3]。 该算法首先选择产生最佳分类结果的属性。 然后为相应的属性生成一个测试节点。 然后根据父节点中的测试属性的信息增益（IG）值将数据进行划分。 当所有数据分组为同一类别，或者根据预定义的阈值，添加额外分割产生类似的分类结果时，算法终止。

3.1.2 无监督学习

与监督学习不同，无监督学习在训练过程中不需要任何标签数据。 这是使用无监督学习的一个优点，因为构建一个全面的标记数据集通常很棘手。 由于没有提供目标类别，网络会寻找每个训练实例的相似属性，并创建一个包含这些相似实例的群组。 对于异常检测，可以将异常实例视为异常。 无监督学习的一些例子如下：

3.1.2.1 *K*-均值聚类

K-均值聚类算法将所有观测数据迭代地分组为 *k*个聚类，直到达到收敛。 最终，一个聚类包含相似的数据，因为每个数据都进入最近的聚类。*K*-均值算法将聚类成员的均值值作为聚类中心。 在每次迭代中，它计算观测数据到任何聚类中心的最短欧氏距离。

此外，通过迭代更新聚类中心，还可以最小化聚类内部的方差。 当达到收敛时，算法将终止，新的聚类与上一次迭代的聚类相同[4]。

3.1.2.2 蚂蚁聚类

蚂蚁聚类算法（ACA）在二维网格上模拟随机蚂蚁行走，其中对象随机放置[5]。 与输入数据的维度不同，每个数据实例都会随机投影到网格的一个单元格上。 一个网格单元格可以指示数据实例在二维网格中的相对位置。 ACA的一般思想是保持相似的项在其原始的N维空间中。

V*izi*ne等人[5]假设网格上的每个站点或单元格最多可以容纳一个对象，并且可能发生以下两种情况之一：（*i*）一个蚂蚁持有一个对象*i*，并评估将其放置在当前位置的概率；（*ii*）一个未装载的蚂蚁估计捡起一个对象的可能性。 随机选择一个蚂蚁，可以在其当前位置上捡起或放下一个对象[5]。

在周围区域中，拾取物体的概率增加了物体之间的差异，反之亦然。 相比之下，周围区域中物体之间的相似性越高，丢弃物体的概率就越高。V*izi*ne等人[5]在他们的N维空间中，将对象*i*和*j*之间的欧几里得距离定义为方程*(*3.4*)*中的d*(i,j)*。 对象*i*在特定网格位置的密度分布函数如下所示，根据方程*(*3.4*)*定义：

$$f(i) = \begin{cases} \frac{1}{s^{\wedge}2} \sum \text{当} (1 - d(i, j)/\alpha) f(i) > 0 \text{时}, \\ j \text{为} 1, \text{否则为} \quad \text{否则}, \end{cases} \tag{3.4}$$

其中，$s^{\wedge}2$是对象i周围区域中的单元格数，α是描述对象之间差异的常数。当周围区域的所有位置都被相似或相等的对象占据时，$f(i)$可能达到最大值。拾取和丢弃对象i的概率分别由方程(3.5)和(3.6)给出：

$$P_{\text{选择}}(i) = (k^{\star}_{p} \frac{p}{k^{\star}_{p} + f(i))}^{2}, \tag{3.5}$$

$$P_{\text{丢弃}}(i) = \begin{cases} 2^{\star} f(i) f(i) < k^{\star}_{d} \\ \text{否则}, \end{cases} \tag{3.6}$$

其中参数k_p和k_d是选择和丢弃对象的概率的阈值常数。一只负载的蚂蚁会考虑它本地区域中的第一个空单元格来丢弃对象，因为蚂蚁的当前位置可能被另一个对象占据[5]。

曾等人[6]定义了两个变量：簇内距离和簇间距离，以衡量ACA的性能。较大的簇内距离意味着更好的紧凑性。同时，较大的簇间距离意味着更好的分离性。一个好的ACA应该提供最小的簇内距离和最大的簇间距离，以展示数据模式中的内在结构和知识。

3.1.2.3（稀疏）自编码器

自动编码器（AE）是一种对称的神经网络（NN）模型，它使用无监督的方法来构建一个模型，该模型使用非标记的数据，如图3.1所示。AE通过仅通过隐藏层运行从输入中提取新特征的编码器-解码器范式来提取新特征。这种范式提高了计算性能，并验证了代码是否从数据中捕获到相关信息。编码器是一个将输入 x映射到隐藏表示的函数，如公式（3.7）所示。

$$y = s_f \left(W \cdot x + b_f \right), \tag{3.7}$$

其中 s_f是一个非线性激活函数，它是一个决策函数，用于确定任何特征的必要性。通常，由于其连续性和可微性，使用逻辑sigmoid函数$\text{sig}(t) = \frac{1}{1 + e^{-t}}$作为激活函数。

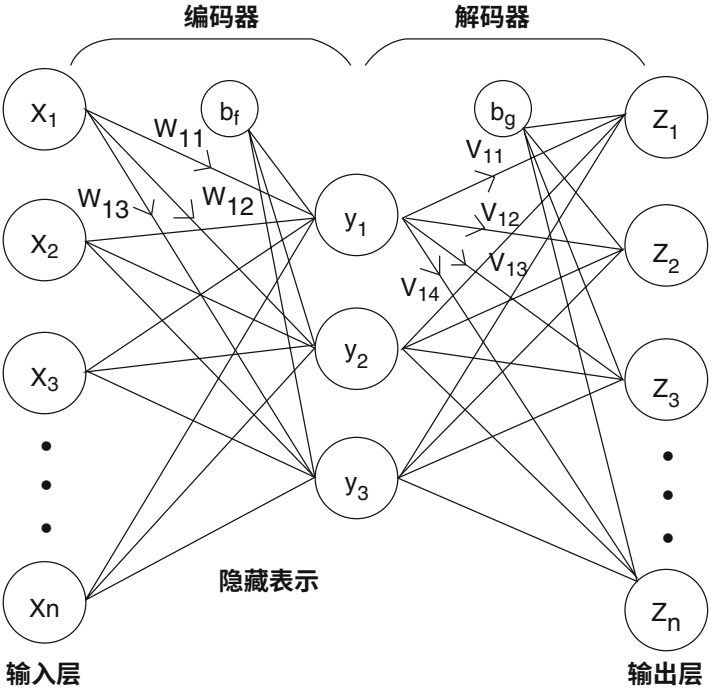


图**3.1**具有对称输入输出层和一个隐藏层中的三个神经元的AE网络

可靠性属性[7]。 解码器函数在公式（3.8）中表示，将隐藏表示映射回重构。

$$z = s_g\left(V \cdot y + b_g\right), \tag{3.8}$$

其中，解码器的激活函数常常使用恒等函数， $s_g(t) = t$，或者使用si_gmoid函数，如编码器。 我们使用 W和 V作为特征的权重矩阵。b_f和 b_g分别作为编码和解码的偏置向量。 其训练阶段找到最优参数$\theta = \{W, V, b_f, b_g\}$，使得输入数据和其在训练集上的重构输出之间的重构误差最小化。

我们进一步解释了一种改进的自编码器，即稀疏自编码器[8]。 这基于Esk in等人的实验[9]，其中异常通常在特征空间的分散区域中形成小的聚类。 此外，密集且大的聚类通常包含良性数据[10]。 对于自编码器的稀疏性，我们首先观察神经元 i的平均输出激活值，如公式（3.9）所示。

$$\hat{\rho}_i = \frac{1}{N}\sum_{j=1}^{N} s_f\left(w_i^T x_j + b_{f,i}\right), \tag{3.9}$$

其中 N是训练数据的总数， x_j是第 j个训练数据， $w_i{}^T$是权重矩阵 W的第i行， $b_{f,i}$是编码的偏置向量的第 i行b_f。 通过降低 $\hat{\rho}_i$的值，隐藏层中的神经元 i显示出在较少数量的训练数据中呈现的特定特征。

机器学习的任务是将模型拟合给定的训练数据。 然而，该模型通常只能拟合特定的训练数据，无法对其他数据进行分类，这就是过拟合问题。 在这种情况下，我们可以使用正则化技术来减少过拟合问题。 稀疏正则化 Ω稀疏度评估平均输出激活值 $\hat{\rho}_i$和期望值 ρ之间的接近程度，通常使用*K*ullback-*L*eibler (KL)散度来确定两个分布之间的差异，如公式(3.10)所示。

$$\begin{aligned}\Omega_{\text{稀疏度}} &= \sum_{i=1}^{h} KL(\rho \,\|\, \hat{\rho}_i) \\ &= \sum_{i=1}^{h} \left[\rho \log\left(\frac{\rho}{\hat{\rho}_i}\right) + (1-\rho)\log\left(\frac{1-\rho}{1-\hat{\rho}_i}\right)\right],\end{aligned} \tag{3.10}$$

在隐藏层中的神经元数量是多少?

我们可以增加权重矩阵W的条目值来减小稀疏正则化的值。 为了避免这种情况，我们还为权重矩阵添加了正则化，即L_2正则化，如公式（3.11）所述。

$$\Omega_{\text{权重}} = \frac{1}{2}\sum_{i=1}^{h}\sum_{j=1}^{N}\sum_{k=1}^{K}\left(w_{ji}\right)^2, \tag{3.11}$$

其中N和K分别是训练数据的数量和每个数据的变量数量。

稀疏自编码器的训练目标是找到最优参数$\theta=\{W, V, bf, bg\}$，以最小化公式（3.12）中的代价函数。

$$E = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K}(zkn\_xkn)^{\wedge 2} + \lambda\cdot\Omega_{\text{权重}} + \beta\cdot\Omega_{\text{稀疏度}}, \tag{3.12}$$

这是一个带有L_2正则化和稀疏正则化的调节均方误差（MSE）。 在训练自编码器时，可以指定L_2正则化项λ和稀疏正则化项β的系数。

3.1.3 半监督学习

半监督学习是一种可以被描述为“介于监督学习和无监督学习之间”的技术。在机器学习中使用半监督学习的背景是因为许多研究人员发现，结合未标记的数据和少量标记的数据可以提高学习的准确性。 我们知道未标记的数据很便宜。 另一方面，标记的数据可能很难获取，需要专门设备的专家，或者处理时间太长。 因此，我们可以说半监督学习的目标是使用标记和未标记的数据来构建比分开使用它们更好的学习器。 由于半监督学习利用未标记的数据，必须做出一些假设来推广它。有三种假设[11]：

1.**半监督平滑假设**

在这个假设中，我们考虑输入的密度。 如果两个点x和y在一个高密度区域中彼此接近，那么相应的输出x'和y'也处于相同的条件中。

2.**聚类假设**

在这个假设中，我们考虑每个输入的聚类。 如果两个点在同一个聚类中，那么它们很可能属于同一类别。

3.**流形假设**

在这个假设中，我们考虑输入的维度。 它表明高维数据位于低维流形上。

半监督学习有三种方法[11]：

1.**生成模型**

在使用生成模型时，我们涉及条件密度的估计。 数据的分布属于每个类别。 因此，问题结构的知识可以通过建模自然地集成。2.低密度分离

在使用低密度分离时，我们试图通过将决策边界远离未标记的点来实现它。 在这种方法中，我们需要利用最大间隔算法，如支持向量机（SVM）。 通过使用低密度分离，我们可以最小化熵。

3.**基于图的方法**

当使用基于图的方法时，我们通过使用图的节点来表示数据。我们用每个节点的边缘标记相邻节点之间的配对距离。
如果存在任何缺失的边缘，那意味着该边缘对应于无限距离。

3.1.4 弱监督学习

弱监督学习是一种机器学习框架，其中模型是使用部分注释或标记的示例进行训练的。 大多数现代计算机视觉系统都涉及从人类标记的图像示例中学习的模型。
因此，我们需要减少训练模型所需的人工干预量。 由于这个原因，弱监督学习模型试图利用仅部分标记的示例。

弱监督学习有三种类型[12]：

1.**不完全监督**
在这种方法中，只给出了一部分（通常很小）带有标签的训练数据，而其他数据则保持未标记。
2.**不精确监督**
在这种方法中，只给出了粗粒度的标签。 例如，让我们考虑一个图像分类任务。 在图像中，我们将进行图像级别的标签，而不是对象级别的标签。
3.**不准确的监督**
在这种方法中，给定的标签并不总是准确的真相。 当图像标注者粗心大意或某些图像由于其模糊性而无法分类时，这种情况可能发生。

3.1.5 强化学习

强化学习采用了一种与监督学习和无监督学习不同的方法。 在强化学习中，一个代理被指派为目标类的责任，在监督学习中为每个实例提供正确的对应类别。 代理负责决定执行哪个动作来完成其任务[13]。 由于没有训练数据参与，代理在强化学习训练过程中通过经验进行学习。 在训练过程中，学习过程采用试错方法来实现一个目标，即获得长期和最高的奖励。 强化学习训练通常被描述为玩家在一路上赢得游戏的方式，其中有一些小点，并在路上获得最终奖励。玩家（或强化学习代理）将探索达到最终奖励的方式。 有时，代理会因为将小点作为目标而陷入其中。因此，在强化学习中，需要探索新的方式，尽管已经达到了小点。 通过这种方式，代理可以在最后获得最终奖励。换句话说，代理和游戏环境形成了一个信息的循环路径，代理执行动作，环境根据相应的动作提供反馈。 达到最终奖励的过程可以使用马尔可夫决策过程（MDP）进行形式化，我们可以为每个状态建模转移概率。 现在，目标函数通过使用MDP得到了很好的表示。有两种标准解决方案可以实现这个目标函数，即Q-learning和策略学习。 前者基于动作值函数进行学习，

而后者则是通过策略函数进行学习，该函数是最佳行动与相应状态之间的映射。 更详细的解释可以在[13]中找到。

3.1.6 对抗机器学习

这种类型的机器学习与之前的所有类型都不同，其中对手机器学习利用和攻击现有机器学习模型的漏洞。 Laskov和Lippman在[14]中提到了一个例子，机器学习的能力基础是基于学习过程中训练数据的表达能力的假设。 然而，如果训练数据或测试数据的分布被有意修改以混淆学习过程，这个假设可能会被违反。 在[14]中还提供了在对抗性框架中发生的其他攻击的例子。

总的来说，机器学习面临两种威胁模型[15]。 第一种威胁是规避攻击，试图绕过学习结果。 对手试图逃避机器学习模型进行的模式匹配。 垃圾邮件过滤和入侵检测是对手试图规避的例子。

他们实现逃避攻击的基本思想是采用试错方法，直到给定的实例成功逃避模式匹配。 第二个威胁是污染攻击，试图影响原始训练数据以获得预期结果。 与第一个威胁模型不同，该模型将恶意数据注入到原始训练数据中，以获得对手期望的结果。 例如，对手在网络流量收集过程中秘密发送恶意数据包，以使学习将该数据包误分类为良性实例。 更详细的解释和示例可以在[16]和[17]中找到。

3.2 基于机器学习的入侵检测系统

通常使用两种典型方法的组合来构建IDS，例如学习或训练和分类，如图3.2所示。在第一阶段中，获取大量标记的网络连接记录以进行监督训练是困难且昂贵的。 FL可能成为第一选择的解决方案。 聚类分析最近已经成为一种异常检测方法[6]。 聚类是一种无监督的数据探索技术，将一组未标记的数据模式划分为组或簇，使得簇内的模式相似，但与其他簇的模式不相似[6]。 FL是改进机器学习算法学习过程的工具。 它通常包括特征构建、提取和选择。 特征构建扩展原始特征以增强其表达能力，而特征提取将原始特征转化为新形式，特征选择则消除不必要的特征。

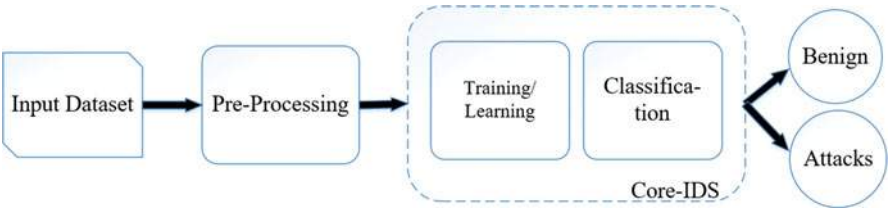


图**3.2** 典型的IDS方案

特征[18]。 分类任务是一种有监督的方法，根据提供的数据来区分良性和恶意流量，这些数据通常来自于图3.2所示的前一步骤。

如图3.2所示，在进入核心IDS模块之前，需要进行预处理步骤。 预处理模块通常包括归一化和平衡步骤。 数据归一化是一个过程，用于输出每个属性的相同值范围，这对于任何机器学习算法的正确学习至关重要[19]。 同时，在实践中，良性流量的频率要远大于恶意流量的频率。 这一特性可能会使核心IDS模块难以正确学习底层模式[20]。 因此，平衡过程是创建具有良性和恶意实例的相等比例数据集的先决条件步骤。 然而，我们应该在测试目的中使用原始比例，即不平衡的比例，以验证IDS能够在实际网络中实施。

机器学习为基础的入侵检测系统已经研究了几十年，特别是在异常检测系统上。Frag*k*iada*k*is等人[21]提出了一种基于异常的入侵检测系统，使用Dempster-Shafer规则进行测量。Shah等人[22]还开发了一种证据理论，用于结合基于异常和基于误用的入侵检测系统。Bostani等人[19]通过修改最优路径森林与*k*-means聚类相结合的方法，提出了一种基于异常的入侵检测系统。 Kolias等人[23]提出了一种分布式基于异常的入侵检测系统，称为*T*ermI D，它结合了蚁群优化和规则归纳，在减少数据操作和降低隐私风险方面实现了数据并行处理。

特征选择技术对于减少模型复杂性、实现更快的学习和实时处理非常有用。Kayacik等人[24]研究了KDD'99数据集中每个特征的相关性，并列出了每个类别标签的最相关特征列表，并详细讨论了信息增益的重要性。

他们的工作证实了特征选择在构建准确的IDS模型中的作用。Puthran等人[25]还在KDD'99数据集中研究了相关特征，并通过使用二进制和四分法改进了决策树（DT）。Almusallam等人[10]利用了基于过滤器的特征选择方法。 Zaman和Karray[26]根据传输控制协议/互联网协议（TCP/IP）网络模型使用了一种名为增强支持向量决策函数（ESVDF）的特征选择方法对IDS进行了分类。Louvieris等人[27]提出了一种基于效果的特征识别IDS，使用朴素贝叶斯作为特征选择方法。

Zhu等人[28]还提出了一种使用多目标方法的特征选择方法。

另一方面，Manekar和Waghmare[29]利用粒子群优化（PSO）和支持向量机（SVM）。PSO执行特征优化以获得优化的特征，然后SVM执行分类任务。 Saxena和Richariya[30]引入了类似的方法，尽管加权特征选择的概念是由Schaffernicht和Gross[31]引入的。

Guyon等人提出了一种利用基于SVM的算法作为特征选择方法的方法[1]。 该方法利用了支持向量学习过程中调整的权重，并对输入特征的重要性进行了排序。 Wang [32]提出了另一种相关方法，根据ANN学习到的权重对输入特征进行排序。这种方法展示了DNN在原始数据中找到有用特征的能力。Aljawarneh等人[33]提出了一种特征选择和分类器集成的混合模型，需要大量计算。

Venkatesan等人[34]强调了僵尸网络的新用途，即数据窃取。 僵尸网络需要具备隐蔽性和抗干扰性才能窃取数据。 这种类型的僵尸网络成为高级持续性威胁（APT）代理工具。 作者[34]在资源受限环境中提出了基于RL模型的蜜罐和僵尸网络入侵检测的组合方法。 蜜罐旨在检测入侵事件，而僵尸网络检测则分析僵尸程序的行为。 RL模型的开发旨在减少资源受限环境中隐蔽僵尸网络的生命周期。 RL代理并没有采取任务为中心的方法，而是学习了一种策略，以最大化被检测到的僵尸数量。

他们的强化学习模型可以在代理人连续两次决策期间进行扩展。
为了实验目的，在PeerSim软件中模拟了一个包含106台机器、98个客户端和8个服务器的企业网络，分布在4个子网中。 根据实验结果，强化学习模型成功地控制了僵尸网络的演化。

我们已经检查了几种用于入侵检测系统的特征选择方法。Huseynov等人[35]检查了蚁群聚类（ACC）方法，以找到僵尸网络流量的特征簇。 [35]中选择的特征与流量有效负载无关，代表了僵尸网络流量的通信模式。 然而，由于缺乏聚类阈值的控制机制，这种僵尸网络检测在大规模和嘈杂的数据集上无法扩展。Kim等人[36]测试了人工免疫系统（AIS）和基于群体智能的聚类方法来检测未知攻击。

此外，Aminanto等人[37]讨论了ACA和模糊推理系统（FIS）在入侵检测系统中的实用性。他们探索了几种常见的入侵检测系统，并结合了学习和分类，如表3.1所示。

表**3.1** 常见的IDS结合了学习和分类

出版物	学习	分类
AKKK17 [37]	ACA	FIS
HKY14 [35]	ATTA-C	ATTA-C + 标签
KKK15 [36]	ACA	AIS
KHKY16 [38]	ACA	DT, ANN

ACA是最流行的聚类方法之一，源自群体智能。 ACA是一种无监督学习算法，可以在不需要预定义的聚类数量的情况下找到近似最优的聚类解决方案[6]。
然而，ACA很少作为独立的方法用于入侵检测的分类。 相反，ACA与其他监督算法（如自组织映射（SOM）和支持向量机（SVM））结合使用，以提供更好的分类结果[39]。 在AKKK17 [37]中，提出了一种基于ACA和FIS的新型混合IDS方案。 作者[37]在训练阶段应用了ACA，在分类阶段应用了FIS。 然后，选择了FIS作为分类阶段，因为模糊方法可以减少误报，提高对入侵活动的可靠性[40]。 同时，KKK15 [36]和KHKY16 [38]也使用了相同的ACA和不同的分类器进行了研究，分别使用了人工AIS和DT以及人工神经网络（ANN）。 AIS是为计算系统设计的，受到人类推理系统（HIS）的启发。 AIS可以区分“自身”（系统拥有的细胞）和“非自身”（对系统来说是外来实体）。 ANN可以学习更复杂的某些未知攻击结构，这是ANN的特点。此外，还研究了一种改进的ACA，即自适应时变传输蚂蚁聚类（ATTA-C），该算法是在HKY14 [35]上进行了基准测试的少数几种算法之一，现在在GNU协议下公开可用[35]。

除了上述常见的IDS之外，还进一步研究了其他IDS模型，利用Hadoop框架[41]和软件定义网络（SDN）环境[42]的优势。Khalid等人[41]提出了一种利用Hadoop和行为流分析的方法。 这个框架在P2P流量分析的情况下特别有用，因为这种应用程序具有固有的流特性。同时，Lee等人[42]提出了一种新颖的IDS方案，用于进行轻量级入侵检测以进行详细的攻击分析。 在这个方案中，基于流的IDS检测入侵，但操作成本低。 当检测到攻击时，IDS请求将攻击流量转发到基于数据包的检测，以便安全专家稍后可以分析基于数据包的检测得到的详细结果。

参考文献

1. I. Guyon, J. Weston, S. Barnhill, 和 V. Vapnik, “使用支持向量机进行癌症分类的基因选择,”机器学习, vol. 46, no. 1–3, pp. 389–422, 2002.
2. X. Zeng, Y.-W. Chen, C. Tao, 和 D. van Alphen, “使用递归特征消除进行手写数字识别的特征选择,” in智能信息隐藏和多媒体信号处理 *(IIH-MSP)* 会议论文集, 京都, 日本. IEEE, 2009, pp. 1205–1208.
3. C. A. Ratanamahatana 和 D. Gunopulos, “扩展朴素贝叶斯分类器：使用决策树进行特征选择,” in数据清洗和预处理 *(*DCA*P)* 研讨会, *IEEE*国际数据挖掘大会 *(I*CD*M)*, 前桥, 日本. IEEE, Dec 2002.
4. C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, 和 L. Hanzo, “面向下一代无线网络的机器学习范式”，*IEEE* 无线通信，卷 24，号 2，页 98-105，2017年。

5. A. L. Vizine, L. N. de Castro, 和 E. Hrusch, “改进聚类蚂蚁：一种自适应蚂蚁聚类算法”，*In*formatica 杂志，卷 29，号 2，页 143-154，2005年。
6. C.-H. Tsang 和 S. Kwong, “蚂蚁群聚类和特征提取用于异常入侵检测”，数据挖掘中的群智能，页 101-123，2006年。
7. R. Rojas, “反向传播算法”，神经网络。 柏林，Springer，1996年，页 149-182。

8. B. A. Olshausen和D. J. Field，“使用过完备基函数的稀疏编码：*V*1所采用的策略？”视觉研究，第37卷，第23期，页码3311-3325，1997年。
9. E. Eskin，A. Arnold，M. Prerau，L. Portnoy和S. Stolfo，“一种用于无监督异常检测的几何框架”，计算机安全中的数据挖掘应用，第6卷，页码77-101，2002年。

10. N. Y. Almusallam，Z. Tari，P. Bertok和A. Y. Zomaya，“多数据流中入侵检测系统的降维方法：一项无监督特征选择方案的综述和提案”，新兴计算，第24卷，页码467-487，2017年。[在线].
可用：https://doi.org/10.1007/978-3-319-46376-6_22
11. X. 朱和A. B. 高伯格，“半监督学习导论”，合成讲座
人工智能和机器学习，第3卷，第1期，第1-130页，2009年。
12. Z.-H. 周，“弱监督学习简介”，国家科学评论，
2017年。
13. C. 奥拉，“人类的机器学习”，https://www.dropbox.com/s/e38nil1dnl7481q/machine_learning.pdf?dl=0，2017年，[在线；访问日期：2018年3月21日]。
14. P. 拉斯科夫和R. 利普曼，“对抗环境中的机器学习”，机器
学习，第81卷，第2期，第115-119页，2010年11月。[在线]. 可用：https://doi.org/10.1007/s10994-010-5207-6
15. S. J. Lewis，“入侵机器学习简介”，https://mascherari.press/introduction-to-adversarial-machine-learning/，2016年，[在线；于2018年3月27日访问]。
16. L. Huang，A. D. Joseph，B. Nelson，B. I. Rubinstein和J. Tygar，“对抗机器
学习”，在第*4*届*ACM*安全和人工智能研讨会的论文集中。
ACM，2011年，第43-58页。
17. I. J. Goodfellow，J. Shlens和C. Szegedy，“解释和利用对抗性示例”，
*arXiv*预印本*arXiv:1412.6572*，2014年。
18. H. Motoda和H. Liu，“特征选择、提取和构建”，台湾*IIC*M（信息与计算机机构）通信，第5卷，第67-72页，2002年。
19. H. Bostani和M. Sheikhan，“使用无监督学习和社交网络概念修改监督OPF基于入侵检测系统”，模式识别，卷62，第56-72页，2017年。

20. M. Sabhnani和G. Serpen，“在误用检测环境中应用机器学习算法到KDD入侵检测数据集。”在国际会议机器学习；模型，技术和应用（M*L*M*T*A）中，拉斯维加斯，美国，2003年，第209-215页。
21. A. G. Fragkiadakis，V. A. Siris，N. E. Petroulakis和A. P. Traganitis，“基于异常的干扰攻击入侵检测，本地与协作检测”，无线通信和移动计算，卷15，第2期，第276-294页，2015年。
22. V. Shah和A. Aggarwal，“使用证据理论提高入侵检测系统对kdd99数据集的性能”，《国际网络安全与数字取证杂志》，第5卷（第2期），第106-114页，2016年。

23. C. Kolias，V. Kolias和G. Kambourakis，“Termid：一种基于分布式群体智能的无线入侵检测方法”，《国际信息安全杂志》，第16卷，第4期，第401-416页，2017年。

24. H. G. Kayacik，A. N. Zincir-Heywood和M. I. Heywood，“选择用于入侵检测的特征：对KDD 99入侵检测数据集的特征相关性分析”，在《隐私、安全和信任会议》中，加拿大新不伦瑞克。*C*iteseer，2005年。
25. S. Puthran和K. Shah，“使用改进的决策树算法进行入侵检测，使用二进制和四分法”，在计算与通信安全的会议中。Springer，2016年，第427-438页。

26. S. Zaman和F. Karray，“基于特征选择和IDS分类方案的轻量级IDS”，在计算科学与工程的会议中。I*EEE*，2009年，第365-370页。
27. P. Louvieris，N. Clewley和X. Liu，“基于效果的特征识别用于网络入侵检测”，神经计算，第121卷，第265-273页，2013年。
28. Y. Zhu，J. Liang，J. Chen和Z. Ming，“用于入侵检测中的特征选择的改进NSGA-iii算法”，基于知识的系统，第116卷，第74-85页，2017年。
29. V. Manekar和K. Waghmare，“使用支持向量机（SVM）和粒子群优化（PSO）的入侵检测系统”，国际高级计算机研究杂志，第4卷，第3期，第808-812页，2014年。

30. H. Saxena和V. Richariya，“使用SVM-PSO和信息增益进行KDD99数据集的入侵检测”，国际计算机应用杂志，第98卷，第6期，2014年。

31. E. Schaffernicht和H.-M. Gross，“用于特征选择的加权互信息”，在人工神经网络会议上，芬兰埃斯波。斯普林格出版社，2011年，第181-188页。
32. Z. Wang，“深度学习在流量识别上的应用”，在黑帽会议上，美国拉斯维加斯。*UB*M，2015年。
33. S. Aljawarneh, M. Aldwairi, 和 M. B. Yassein, “通过特征选择分析和构建混合高效模型的基于异常的入侵检测系统,”*Journal ofComputational Science*, 2017年3月. [在线]. 可用：http://dx.doi.org/10.1016/j.jocs.2017.03.006

34. S. Venkatesan, M. Albanese, A. Shah, R. Ganesan, 和 S. Jajodia, “在资源受限环境中使用强化学习检测隐蔽僵尸网络,” in*Proceedings of the2017 Workshop on Moving Target Defense*. ACM, 2017, pp. 75–85.
35. K. Huseynov, K. Kim, 和 P. Yoo, “使用蚁群聚类的半监督僵尸网络检测,” in*Symp. Cryptography and Information Security (SCIS), Kagoshima, Japan*, 2014.
36. K. M. Kim, H. Kim, and K. Kim, “基于生物启发算法的未知攻击入侵检测系统设计”，计算机安全研讨会（*CSS*），日本长崎，2015年。

37. M. E. Aminanto, H. Kim, K. M. Kim, and K. Kim, “基于蚁群聚类算法的模糊异常检测系统”，*IEIC*E电子、通信和计算机科学基础交易，第100卷，第1期，第176-183页，2017年。
38. K. M. Kim, J. Hong, K. Kim, and P. Yoo, “基于ACA的未知攻击入侵检测系统评估”，密码学与信息安全研讨会（*SCIS*），日本熊本，2016年。

39. C. Kolias, G. Kambourakis, and M. Maragoudakis, “入侵检测中的群体智能：一项调查”，计算机与安全，第30卷，第8期，第625-642页，2011年。
40. A. Karami和M. Guerrero-Zapata，“基于混合PSO-Kmeans算法的模糊异常检测系统在内容为中心的网络中”，*Neurocomputing*，卷149，页1253-1269，2015年。

41. K. Huseynov，P. D. Yoo和K. Kim，“在Hadoop框架中使用阈值设置的可扩展P2P僵尸网络检测”，韩国信息安全与密码学研究所杂志，卷25，号4，页807-816，2015年。

42. D. S. Lee，“改进SDN中基于流的IDS的检测能力”，*KAIST*，硕士论文，2015年。

第4章
深度学习

摘要本章简要介绍了深度学习的历史和定义。 由于深度学习中存在多种模型，我们将深度学习模型分为三个分支：生成模型、判别模型和混合模型。 在每个模型中，我们展示了一些学习模型的例子，以便看到三个模型之间的区别。

4.1 分类

深度学习最初来自于神经网络算法的进展。 为了克服神经网络中仅有一个隐藏层的局限性，已经应用了各种方法。这些方法使用连续的隐藏层进行层级级联。 由于深度学习中存在多种模型，Aminanto等人[1]根据Deng [2, 3]的方法对几种深度学习模型进行了分类，将深度学习分为生成式、判别式和混合式三个子类。 分类是基于架构和技术的意图，例如合成/生成或识别/分类。深度学习方法的分类如图4.1所示。

4.2 生成式（无监督学习）

无监督学习或所谓的生成模型使用无标签数据。 将生成式架构应用于模式识别的主要概念是无监督学习或预训练[2]。 由于学习后续网络的较低层级很困难，需要使用深度生成结构。 因此，从有限的训练数据中，逐层学习每个较低层级而不依赖于所有上层是必要的。生成模型还旨在学习给定数据的联合统计分布[3]。这些模型计算

© 作者（们），在Springer Nature Singapore Pte Ltd. 2018独家许可下

K. Kim等人，使用深度学习进行网络入侵检测，
关于网络安全系统和网络的SpringerBriefs，
https://doi.org/10.1007/978-981-13-1444-5_4

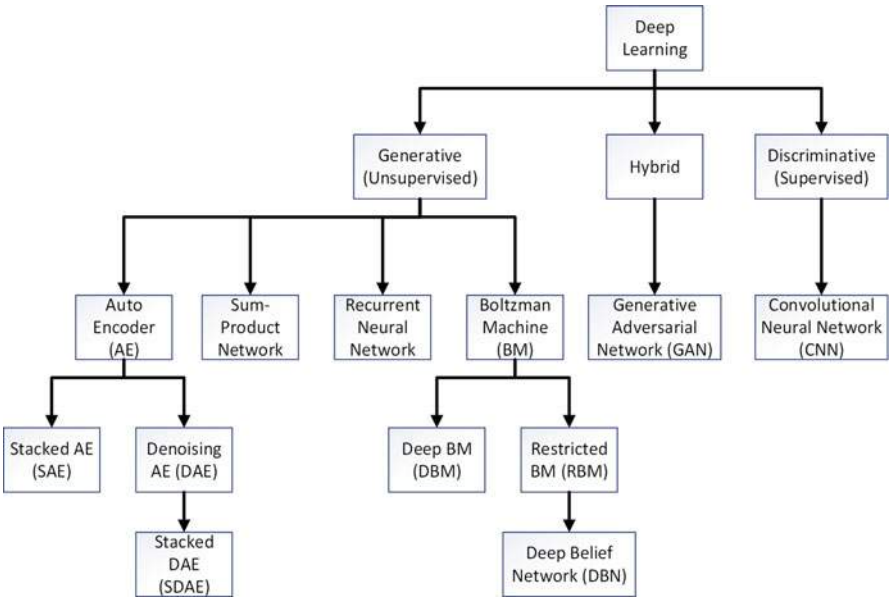


图**4.1** 深度学习方法的分类

给定输入的联合概率，并选择具有最高概率的类别标签[4]。 有许多被归类为无监督学习的方法。

4.2.1 堆叠（稀疏）自编码器

（稀疏）自编码器可以通过无监督的贪婪逐层预训练算法（SAE）作为深度学习技术来使用。在这里，预训练是指使用单个隐藏层对单个自编码器进行训练。 每个自编码器在级联之前单独进行训练。 这个预训练阶段是构建堆叠自编码器所必需的。在这个算法中，除了输出层之外的所有层都在多层神经网络中进行初始化。然后，每一层都以无监督的方式作为自编码器进行训练，构建输入的新表示。

SAE使用相同的神经元数量来训练输入和输出层。
同时，隐藏层中的节点数量少于输入层，这代表了一个新的较少特征集。这种架构可以在复杂计算后重构数据，从而带来新的能力。 AE旨在高效地学习一组紧凑的数据，并可以堆叠以构建深度网络。 每个隐藏层的训练结果被级联，可以通过不同的深度提供新的转换特征。 为了更精确地训练，可以附加一个带有类标签的额外分类器层。 此外，还可以训练一个去噪自编码器（DAE）来重构输入中被噪声破坏的精确修正输入。

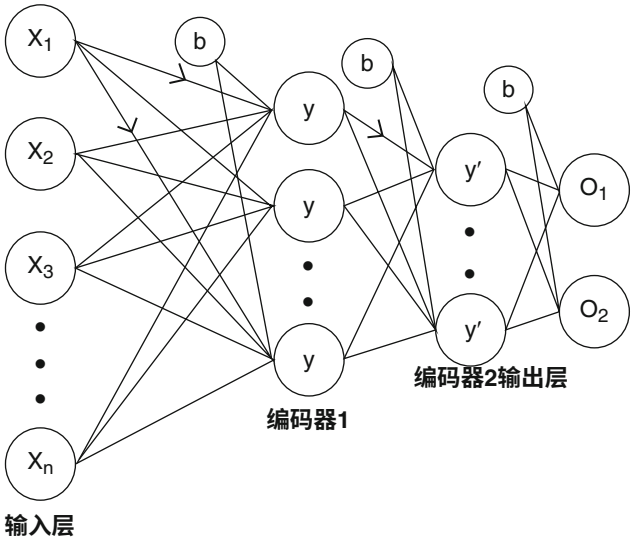


图**4.2** SAE网络，具有两个隐藏层和两个目标类别

通过噪声输入进行精确修正输入的一种去噪自编码器[6]。 DAE也可以堆叠以构建深度网络。

无监督的贪婪逐层预训练算法的性能可以比有监督的算法更准确。这是因为贪婪的有监督过程可能过于贪婪，提取的信息较少并且只考虑一层 [7, 8]。 只包含一个隐藏层的神经网络可能会丢弃一些关于输入数据的信息，因为通过组合额外的隐藏层可以利用更多的信息。 预训练阶段的特征，即贪婪逐层预训练，可以用作标准有监督机器学习算法的输入，也可以用作深度有监督神经网络的初始化。

图4.2显示了具有两个隐藏层和两个目标类的SAE网络。
最后一层在DNN中实现了softmax函数进行分类。
这个softmax函数使SAE既是无监督学习又是有监督学习。
softmax函数是对数几率函数的广义术语，将K维向量$\mathbf{v}\in\mathbb{R}K$压缩为K维向量$\mathbf{v}^{*\in}(0,1)K$，总和为1。 在这个函数中，T和C分别定义为训练实例的数量和类的数量。 softmax层最小化损失函数，可以是交叉熵函数（如公式（4.1））或均方误差。

$$E = \frac{1}{T}\sum_{j=1}^{T}\sum_{i=1}^{C}\left[z_{ij}\log y_{ij} + (1 - z_{ij})\log\left(1 - y_{ij}\right)\right], \tag{4.1}$$

4.2.2 伯努利机

BM是一个由对称配对的二进制单元组成的网络[9]，这意味着所有输入节点都连接到所有隐藏节点。 BM是一个只有一个隐藏层的浅层模型。 BM具有神经元单元的结构，可以对是否激活进行随机决策[3]。 如果一个BM的输出级联到多个BM中，则称为深度BM（DBM）。 与此同时，RBM是一个没有隐藏节点和输入节点之间连接的定制BM[9]。 RBM由可见变量和隐藏变量组成，可以推断出它们之间的关系。这里的可见变量指的是训练数据中的输入神经元。 如果堆叠多层RBM，则称为深度信念网络（DBN）的逐层方案。 当使用未标记的数据集和反向传播时，DBN可以用作降维的特征提取方法（即无监督训练）。 相反，当使用适当标记的带有特征向量的数据集时，DBN用于分类（即有监督训练）[10]。

4.2.3 和-积网络

另一个深度生成模型是Sum-Product Networks (SPN)，它是一个有向无环图，变量作为叶子节点，求和和乘积操作作为内部节点，带有加权边[11]。 求和节点提供混合模型，而乘积节点表示特征层次结构[3]。 因此，我们可以将SPN视为混合模型和特征层次结构的组合。

4.2.4 循环神经网络

循环神经网络（RNN）是神经网络的扩展，具有循环链接以处理序列信息。 这些循环链接放置在较高层和较低层神经元之间，使RNN能够将数据从先前事件传播到当前事件。 这个特性使得RNN具有时间序列事件的记忆[12]。 图4.3显示了RNN的一个单循环，左侧与右侧拓扑结构相似，当循环被打破时。

RNN的一个优点是能够将先前的信息连接到当前的任务；然而，它无法达到“远”之前的记忆。 这个问题通常被称为长期依赖性。 长短期记忆（LSTM）网络是由Hochreiter和Schmidhuber [14]引入的，以克服这个问题。 LSTM是RNN的扩展，具有四个神经网络在一个单层中，而RNN只有一个，如图4.4所示。

LSTM的主要优势是存在状态单元，该线路通过每个层的顶部传递。 该单元负责将信息从上一层传播到下一层。然后，LSTM中的“门”会

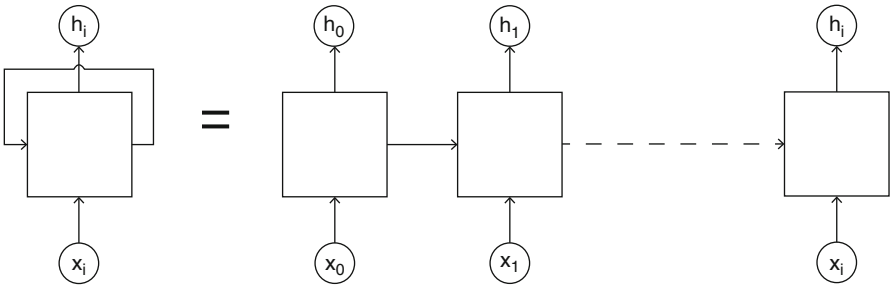


图**4.3** 展开拓扑的RNN [13]

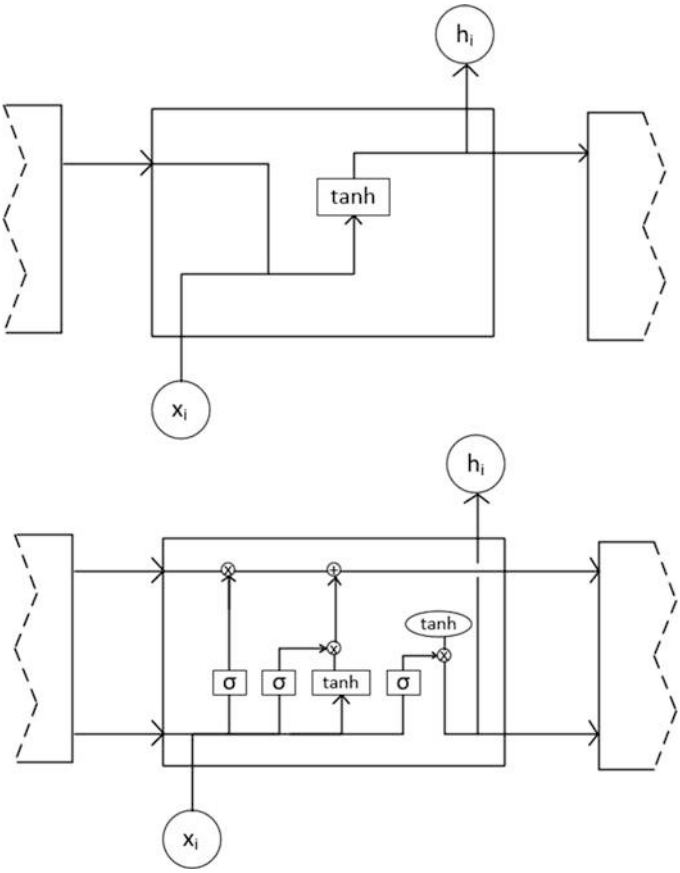


图**4.4** RNN拓扑结构（上）与LSTM拓扑结构（下）[13]

管理哪些信息将被传递或丢弃。 有三个门来控制信息流动，分别是输入门、遗忘门和输出门[15]。 这些门由一个sigmoid神经网络和一个运算符组成，如图4.4所示。

4.3 判别式

监督学习或判别模型旨在使用标记数据对某些数据部分进行模式分类[2]。 判别式架构的一个例子是CNN，它采用了一种特殊的架构，特别适用于图像识别。 CNN的主要优势是不需要手工特征提取。 CNN可以使用梯度下降训练多层网络，从大量的数据中学习复杂的高维非线性映射[16]。 CNN使用三个基本概念：局部感受野，共享权重和池化[17]。 成功使用CNN部署的一个广泛研究是Google的AlphaGo [18]。 判别模型的其他例子包括线性和逻辑回归[4]。

当RNN的输出被用作输入的标签序列时，RNN也可以被视为一种判别模型[3]。 Graves [19]提出了这种网络的一个例子，他利用RNN构建了一个概率序列转换系统，可以将任何输入序列转换为任何有限的离散输出序列。

4.4 混合

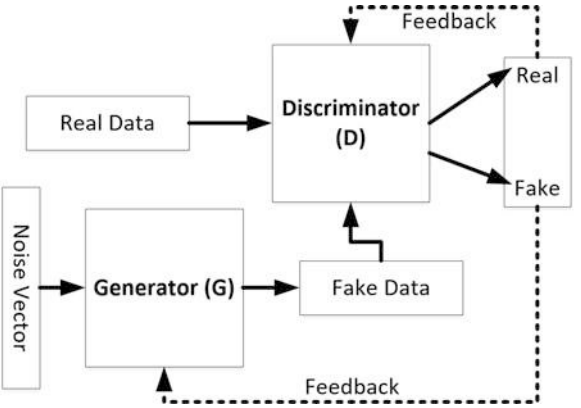
混合深度架构结合了生成模型和判别模型的特点。 混合结构旨在区分数据以及判别方法。 然而，在早期阶段，它在生成模型的结果方面起到了重要的辅助作用。 混合架构的一个例子是深度神经网络（DNN）。 然而，DNN和DBN之间存在一些混淆的术语。

在开放文献中，DBN也使用反向传播判别训练作为“微调”。这个DBN的概念与DNN类似[2]。 根据Deng [3]的说法，DNN被定义为具有级联全连接隐藏层的多层网络，在预训练阶段使用堆叠的RBM。 当分类任务添加了类标签时，许多其他生成模型可以被视为判别模型或混合模型。

4.4.1 生成对抗网络（*GAN*）

*G*oodfellow [20]引入了一种新的框架，同时训练生成模型和判别模型，其中生成模型*G*捕捉数据分布，判别模型*D*区分原始输入数据和来自模型*G*的数据。 这是一个零和游戏，涉及到生成模型*G*和判别模型*D* [4]，其中生成模型*G*旨在伪造原始输入数据，而判别模型*D*旨在区分原始输入和生成模型*G*的输出。 根据*D*imokranitou的说法

图**4.5**显示了GAN的典型架构



[4]，*G*AN的优点是在达到平衡后保持一致性，不需要近似推理或马尔可夫链，并且可以使用缺失或有限的数据进行训练。另一方面，应用*G*AN的缺点是找到生成模型*G*和判别模型*D*之间的平衡。*G*AN的典型架构如图4.5所示。

参考文献

1. M. E. Aminanto和K. Kim，“入侵检测系统中的深度学习：概述”，*2016*年国际工程与技术研究会议，印度尼西亚巴厘岛，*2016*年*6*月*28*日至*30*日。

2. L. Deng，“深度学习的架构、算法和应用的教程调查”，A*PSIP*A信号与信息处理交易，第3卷，2014年。
3. L. Deng，D. Yu等，“深度学习：方法和应用”，*F*ound*ati*ons *a*nd *T*rends◯R信号处理，第7卷，第3-4期，pp. 197-387，2014年。
4. A. Dimokranitou，“用于图像异常事件检测的对抗自编码器”，博士论文，普渡大学，2017年。
5. Z. Wang，“深度学习在交通识别上的应用”，在黑帽会议上，拉斯维加斯，美国。*UB*M，2015年。
6. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, “堆叠去噪自编码器：使用局部去噪准则在深度网络中学习有用的表示”，《机器学习研究杂志》，第11卷，第12期，2010年，页码3371-3408。
7. Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle,*et al.*，“贪婪逐层训练深度网络”，《神经信息处理系统进展》，第19卷，2007年9月，页码153-160。
8. Y. Bengio, Y. LeCun,*et al.*，“将学习算法扩展到人工智能”，《大规模核机器》，第34卷，第5期，2007年，页码1-41。
9. R. Salakhutdinov和G. Hinton，“深度Boltzmann机器”，人工智能和统计学, 2009年，第448-455页。
10. M. Salama，H. Eid，R. Ramadan，A. Darwish和A. Hassanien，“混合智能入侵检测方案”，软计算在工业应用中, 2011年，第293-303页。
11. H. Poon和P. Domingos，“和积网络：一种新的深度架构”，在计算机视觉研讨会（*ICCV W*orkshops），*20*11年*IEEE*国际会议. *IEEE*, *20*11年，第689-690页。

12. R. C. Staudemeyer，“将长短期记忆递归神经网络应用于入侵检测”，南非计算机杂志，第56卷，第1期，第136-154页，2015年。
13. C. Olah，“理解LSTM网络”，http://colah.github.io/posts/2015-08-Understanding-LSTMs/，2015年，[在线；访问日期：2018年2月20日]。
14. S. Hochreiter和J. Schmidhuber，“长短期记忆”，神经计算，卷9，第8期，页1735-1780，1997年。
15. J. Kim，J. Kim，H. L. T. Thu和H. Kim，“用于入侵检测的长短期记忆递归神经网络分类器”，在平台技术和服务（*PlatCon*）上，*2016*年国际会议上。 IEEE，2016年，页1-5。
16. Y. LeCun，L. Bottou，Y. Bengio和P. Haffner，“基于梯度的学习应用于文档识别”， *IEEE*会议论文集 ，卷86，第11期，页2278-2324，1998年。
17. M. A. Nielsen，“神经网络和深度学习”，2015年。
18. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser,I. Antonoglou, V. Panneershelvam, M. Lanctot,*et al.*，“使用深度神经网络和树搜索掌握围棋”，《自然》杂志，第529卷，第7587期，页码：484-489，2016年。
19. A. Graves， “使用递归神经网络进行序列转导 ”， *arXiv*预印本 *arXiv:1211.3711*，2012年。
20. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, 和Y. Bengio，“生成对抗网络”，收录于神经信息处理进展系统，2014年，页码：2672-2680。

第五章
基于深度学习的入侵检测系统

摘要本章回顾了在2016年和2017年发表的利用深度学习模型作为方法的最新入侵检测系统。 将讨论每个出版物的问题领域、方法论、数据集和实验结果等关键问题。 这些出版物可以根据第4章中的深度学习分类分为三个不同的类别，即生成式、判别式和混合式。 生成式模型组包括仅使用深度学习模型进行特征提取，并使用浅层方法进行分类任务的入侵检测系统。 判别式模型组包括使用单一深度学习方法进行特征提取和分类任务的入侵检测系统。 混合式模型组包括使用多个深度学习方法进行生成式和判别式目的的入侵检测系统。 通过比较所有入侵检测系统，概述了深度学习在入侵检测研究中的进展。

5.1 生成

本小节将使用深度学习进行特征提取的IDS进行分组，并使用浅层方法进行分类任务。

5.1.1 深度神经网络

Roy等人[1]提出了一种利用深度学习模型的IDS，并验证了深度学习方法可以提高IDS性能。 选择DNN，包括具有400个隐藏层的多层前馈NN。 输出层使用浅层模型，使用修正线性单元和softmax激活函数。 前馈神经网络的两个优点是直接从输入值中提供复杂多变量非线性函数的精确逼近，并为大类提供稳健建模。 此外，作者声称DNN

© 作者，独家许可给Springer Nature Singapore Pte Ltd. 2018年K. Kim等人，使用深度学习进行网络入侵检测，Cyber Security Systems and Networks的SpringerBriefs，https://doi.org/10.1007/978-981-13-1444-5_5

由于通过对类别的后验分布进行表征，DBN的判别能力优于其他方法，适用于模式分类[1]。

为了验证，使用了KDD Cup'99数据集。 该数据集有41个特征作为网络的输入。 作者将所有的训练数据分为75%用于训练和25%用于验证。 他们还比较了浅层分类器SVM的性能。根据实验结果，DNN的准确率达到了99.994%，而SVM只有84.635%。 这个结果表明了DNN在入侵检测方面的有效性。

5.1.2 加速深度神经网络

Potluri和Diedrich在2016年提出了另一种不同架构的DNN。 本文主要关注于通过使用多核CPU和GPU来改进DNN的实现，这是重要的，因为DNN需要进行大量的计算来进行训练[3]。 他们回顾了一些利用硬件增强的IDS，包括GPU、多核CPU、内存管理和FPGA。还讨论了负载平衡（和分割）或并行处理的方法。 在这项工作中，选择了深度学习模型SAE来构建DNN。 该网络的架构包括来自NSL-KDD数据集的41个输入特征，第一个AE的第一隐藏层有20个神经元，第二个AE的第二隐藏层有10个神经元，输出层有5个神经元，并采用softmax激活函数。 在训练阶段，每个AE都是分别训练的，但是按顺序进行，因为第一个AE的隐藏层成为第二个AE的输入。有两次微调过程，第一次通过softmax激活函数进行，第二次通过整个网络的反向传播进行。

选择NSL-KDD数据集来测试这种方法。 这个数据集是KDD Cup'99数据集的修订版。 它具有相同的特征数量，即41个，但分布更合理，没有KDD Cup'99数据集中存在的冗余实例。 作者首先测试了网络对来自两个类到四个类的不同攻击组合。 较少数量的攻击类别比较多数量的攻击类别表现更好，这是预期的，因为不平衡的类别分布会导致较少的攻击类型有良好的结果。

为了加速，作者使用了两个不同的CPU和一个GPU。他们还尝试了串行和并行CPU。实验结果显示，使用并行CPU进行训练比使用串行CPU快三倍。使用GPU进行训练的性能与并行CPU相似。 有趣的是，第二个CPU的并行训练比GPU更快。他们解释说，这种情况是由于当前CPU的时钟速度过高导致的。 不幸的是，作者没有提供关于检测准确率或误报率的性能比较。

5.1.3 自学习

自学习（STL）是由Niyaz等人提出的一种用于IDS的深度学习模型[4]。 作者提到了开发高效IDS的两个挑战。第一个挑战是选择特征，因为特定攻击的选定特征可能与其他攻击类型不同。 第二个挑战是处理用于训练目的的有限标记数据集。 因此，选择了一种生成式深度学习模型来处理这个无标记数据集。 提出的STL包括两个阶段，无监督特征学习（UFL）和有监督特征学习（SFL）。 对于UFL，作者利用了稀疏自编码器，而对于SFL，使用了softmax回归。图5.1显示了本文中使用的STL的两阶段过程。 UFL负责使用无标记数据集进行特征提取，而SFL负责使用标记数据进行分类任务。

作者使用NSL-KDD数据集验证了他们的方法。 在训练过程之前，作者为数据集定义了一个预处理步骤，其中包括1到N的编码和最小-最大归一化。在1到N编码过程之后，121个特征准备好进行归一化步骤，并作为UFL的输入特征。十折交叉验证和来自NSL-KDD数据集的测试数据集被选用进行训练和测试。 作者还评估了STL在三种不同的攻击组合中的性能，即2类、5类和23类。 总体而言，在训练阶段，他们的STL对于所有组合的分类准确率都高于98%。 在测试阶段，STL对于2类和5类分类分别达到了88.39%和79.10%的准确率。 他们提到未来的工作是基于原始网络流量开发使用深度学习模型的实时入侵检测系统。

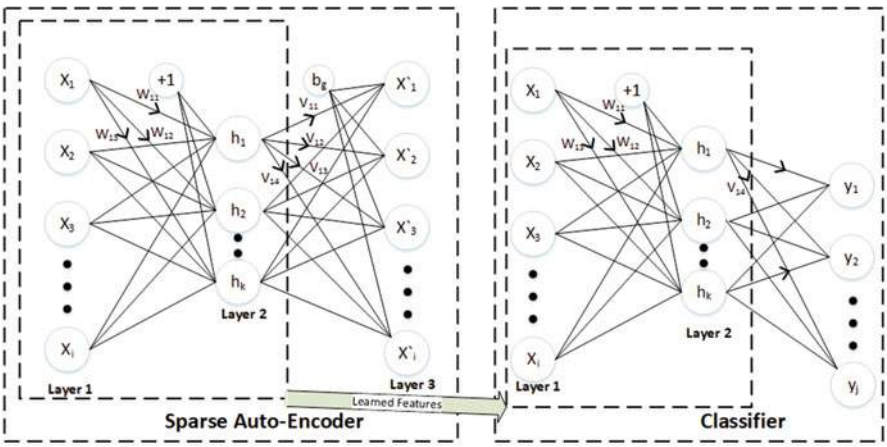


图**5.1** STL的两个阶段[4]

5.1.4 堆叠去噪自编码器

Yu等人[5]提出了一种基于深度学习架构的基于会话的IDS。
他们发现常见的IDS缺点：高FP和误报。
常见数据集中的大多数攻击特征都具有结构化且具有特定专家知识的特殊语义，并且手工制作的数据集与特定攻击类别密切相关。 因此，利用深度学习模型，无监督深度学习可以自动从大数据中学习关键特征。 所提出的方法包括从原始数据中提取特征并应用无监督的SDAE。会话数据从原始网络数据中提取，其中恶意和僵尸网络实例分别来自UNB ISCX 2012和CTU-13。 由于数据是从原始数据中提取的，因此需要进行预处理步骤。 数据预处理过程包括会话构建、记录构建和归一化。 会话构建区分三种不同的会话，即TCP、UDP和ICMP。记录构建从数据包头中提取前17个特征，其余983个特征从有效负载中提取。 最后，归一化会话使用最小-最大函数。 SDAE本身包含两个隐藏层和一个softmax回归层。 为了去噪，作者随机将输入特征的10%、20%、30%设置为零。 作者提到使用SDAE的优点是可以自动从未标记的实例中学习重要特征、通过去噪策略使输入具有鲁棒性以及在隐藏层非线性时具有良好的维度约简能力。

他们以性能指标为准，测量了准确率、精确率、召回率、F值和ROC曲线。 二元和多类分类与43%的数据集和整个数据集的组合一起使用，以验证SDAE的性能。SDAE还与其他深度学习模型进行了比较，包括SAE、DBN和AE-CNN模型。 总体而言，SDA在使用整个数据集进行多类分类时取得了最佳性能，准确率达到了98.11%。

5.1.5 长短期记忆循环神经网络

Kim等人采用了LSTM-RNN的生成方法用于入侵检测系统。
他们将softmax回归层用作输出层。 其他超参数分别为批量大小、时间步长和迭代次数，分别为50、100和500。 此外，随机梯度下降（SGD）和均方误差（MSE）被用作优化器和损失函数。从KDD Cup'99数据集中提取了41个输入特征。
实验结果表明，最佳学习率为0.01，隐藏层大小为80，DR为98.88%，误报率为10.04%。 刘等人也提出了类似的网络拓扑结构[7]，但超参数不同：时间步长、批量大小和年龄分别为50、100和500。 使用KDD Cup'99数据集，实现了98.3%的DR和5.58%的误报率。

5.2 判别式

本小节将使用单一深度学习方法进行特征提取和分类任务的IDS进行分组。

5.2.1 软件定义网络中的深度神经网络

SDN是一种新兴的网络技术，由控制平面和数据平面构建而成。 控制平面将网络控制和转发功能解耦。 控制平面的集中式方法使得SDN控制器适用于IDS功能，因为控制器可以捕获整个网络。 不幸的是，由于控制平面和数据平面的分离，会导致一些重要的威胁。唐等人提出了一种用于SDN环境中IDS的DNN方法[8]。 DNN架构为6-12-6-3-2，即6个输入特征，隐藏层分别为12、6和3个神经元，输出为2个类别，如图5.2所示。

他们使用NSL-KDD数据集来验证他们的方法。 由于数据集具有41个特征，作者根据他们的专业知识选择了SDN中的6个基本特征。 他们使用准确率、精确率、召回率、F值和ROC曲线作为性能指标。 从实验结果来看，学习率为0.001的效果最好，因为学习率为0.0001的情况下会过拟合。 然后，将提出的方法与利用各种机器学习模型的先前工作进行了比较。 DNN的准确率达到了75.75%，低于使用全部41个特征的其他方法，但高于仅使用6个特征的其他方法。 基于这个事实，作者声称

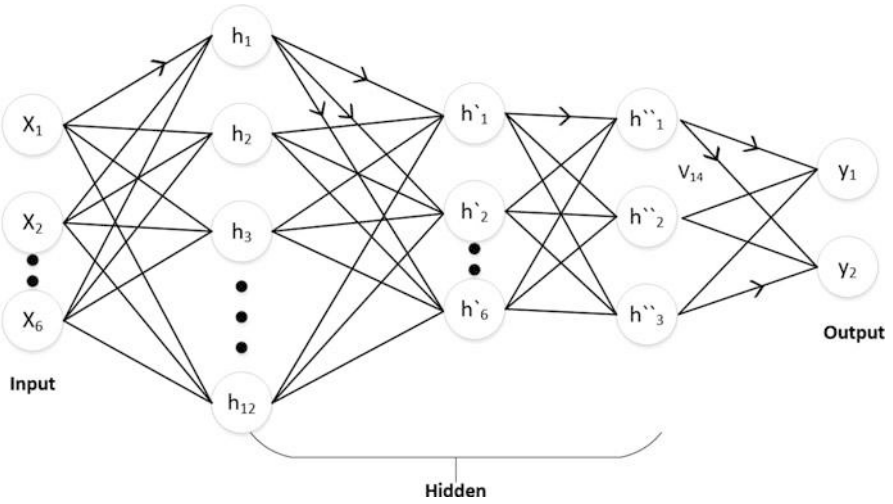


图**5.2**[8]中使用的DNN架构

提出的DNN能够仅通过有限的特征来概括和抽象网络流量的特征。

5.2.2 递归神经网络

Yin等人[9]强调了传统机器学习方法的缺点，即无法高效地解决大规模入侵数据分类问题。 他们利用RNN在IDS上下文中的实现优势。 RNN包含前向传播和反向传播，后者是计算前向传播残差的相同神经网络。

提出的RNN-IDS从数据预处理步骤开始，包括数值化和归一化。 特征准备好的数据被传递到RNN的训练步骤。训练的输出模型用于应用于测试步骤，使用测试数据集。

为了实验目的，他们使用了NSL-KDD数据集进行训练和测试。 原始特征有41个，但在数值化后变为122个特征，将字符串映射为二进制。 测试了两种类型的分类，即二进制和多类分类。 根据实验结果，二进制分类的最佳超参数是学习率为0.1，迭代次数为100，隐藏节点为80，准确率为83.28%（使用KDDTest^{+}）。 而对于多类分类，最佳超参数是学习率为0.5，隐藏节点为80，准确率为81.29%。RNN-IDS在二进制和多类分类方面优于作者测试的其他机器学习方法。

5.2.3 卷积神经网络

Li等人[10]在IDS中使用CNN进行特征提取和分类器的实验。CNN在与图像相关的分类任务中取得了许多成功的实现；然而，对于文本分类来说，这仍然是一个巨大的挑战。 因此，在IDS环境中实现CNN的主要挑战是图像转换步骤，这是由Li等人[10]提出的。 NSL-KDD数据集被用于实验目的。 图像转换步骤首先将41个原始特征映射为464个二进制向量。 映射步骤包括两种类型的映射，一种是用于符号特征的独热编码器，另一种是用于连续特征的独热编码器，使用十个二进制向量。 图像转换步骤继续将464个向量转换为8×8像素的图像。 这些图像可以用于CNN的训练输入。作者决定尝试使用已学习的CNN模型，ResNet 50和GoogLeNet。在KDDTest^{+}上的实验结果显示，ResNet 50和GoogLeNet的准确率分别为79.14%和77.14%。 尽管这个结果没有改进IDS的最新技术水平，但这项工作展示了如何在IDS环境中应用CNN和图像转换。

5.2.4 长短期记忆循环神经网络

由于在各个研究领域中的成功应用，LSTM-RNN变得更加流行。 从先前事件中自学习的能力可以应用于入侵检测系统，这意味着可以从先前的攻击行为中进行学习。
下面描述了一些实现了LSTM-RNN的入侵检测系统。

5.2.4.1 LSTM-RNN Staudemeyer

Staudemeyer [11]通过将LSTM-RNN应用于网络流量建模作为时间序列进行了实验。 训练数据来自KDD Cup'99数据集。 作者还使用决策树算法选择了一些显著特征的子集，并将其与整个特征集和子集特征在实验中进行了比较。 他们的实验使用了不同的LSTM-RNN参数和结构，例如内存块的数量和每个内存块的单元数，学习率以及对数据的遍历次数。 此外，还使用了一层隐藏神经元、窥视孔连接、遗忘门和LSTM快捷连接进行了实验。 根据实验结果，最佳性能是由包含两个单元的四个内存块实现的，带有遗忘门和快捷连接，学习率为0.1，最多进行1,000个周期的训练。 总体准确率为93.82%。他们还在结论中提到，LSTM-RNN适用于对具有大量记录的攻击进行分类，但对于数量有限的攻击实例效果较差。

5.2.4.2 LSTM-RNN用于集体异常检测

Bontemps等人[12]在IDS中利用LSTM-RNN实现了两个目标：时间序列异常检测器和通过提出循环数组的集体异常检测器。 集体异常本身是关于整个数据集的相关异常数据实例的集合[12]。 他们在实验中使用了KDD Cup'99数据集，并解释了从KDD Cup'99数据集构建时间序列数据集所需的预处理步骤。

5.2.4.3 GRU在物联网中的应用

Putchala [13]在物联网环境中实现了一种简化形式的LSTM，称为门控循环单元(GRU)。 由于其简单性，GRU非常适合物联网，可以减少网络中的门数量。 GRU将遗忘门和输入门合并为更新门，并将隐藏状态和细胞状态组合成一个简单的结构，如图5.3所示。

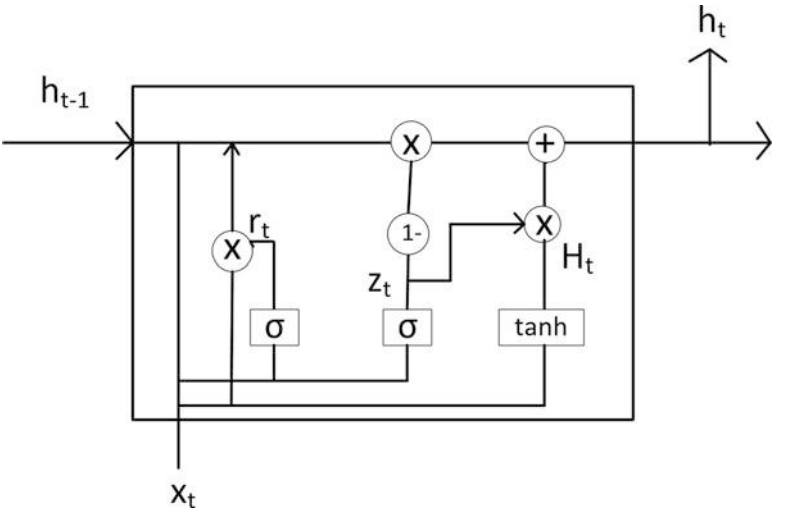


图**5.3** GRU单元[13]

然后作者采用了多层GRU，即在RNN的每个隐藏层中使用GRU单元，并且还通过使用随机森林算法进行特征选择。
实验使用了KDD Cup'99数据集，分别达到了98.91%的准确率和0.76%的误报率。

5.2.4.4 用于DDoS的LSTM-RNN

Bediako [14]提出了一种使用LSTM-RNN的分布式拒绝服务(DDoS)检测器。作者使用CPU和GPU检查了LSTM-RNN的性能。实验使用了NSL-KDD数据集。 显著的检测准确率为99.968%。

5.3 混合

本小节包括一个使用多个深度学习模型进行生成和判别的IDS。

5.3.1 对抗网络

Dimokranitou等人[15]提出了一种使用对抗网络在图像中检测异常事件的方法。 尽管该检测器不是IDS，但其目标是检测异常。 他们实现了一个结合了对抗自编码器的方法

AEs和GAN。网络试图将AE的隐藏代码向量的聚合后验与任意先验分布匹配。 学习到的AE的重构误差在正常事件中较低，在异常事件中较高。

5.4 深度强化学习

Choi和Cho [16]强调了数据库应用中入侵检测的两个主要问题：在真实环境中，良性数据的数量远大于恶意数据，并且内部入侵更难以检测。

所提出的方法是一种自适应数据库应用IDS，使用进化强化学习（ERL）结合进化学习和强化学习来进行种群和个体的学习过程。 该方法包括两个MLP，一个是行为网络，一个是评估网络。 前者网络旨在通过根据网络的权重执行误差反向传播来检测异常查询，而后者网络提供学习率作为反馈，以提高行为网络的检测率。 由于用于进化学习的评估网络，网络会演化以探索最优模型。 实验使用了一个特定的场景，称为TPC-E，它是一个经纪公司的在线事务处理工作负载[16]。 经过25代后，达到了90%的分类准确率。

冯和徐[17]关注于在网络物理系统（CPS）中检测未知攻击。 然后提出了一种新颖的基于深度强化学习的最优策略[17]。
这种创新体现在明确的网络状态相关动态和解决Hamiltonian-Jacobi-Isaac（HJI）方程的零和博弈模型上。 开发了一种具有博弈论参与者批评者神经网络结构的深度强化学习算法，用于解决HJI方程。 深度强化学习网络由三个多层神经网络组成；第一个用于批评者部分，第二个用于近似可能的最坏攻击策略，最后一个用于实时估计最优防御策略。

5.5 比较

我们在本章前面提到的所有先前工作进行了比较和总结。 所有使用KDD Cup '99和NSL-KDD数据集的讨论模型总结在表5.1和表5.2中。

IDS在KDD Cup'99上的整体性能是令人满意的，预期的准确率超过90%。表5.1中的四个IDS使用LSTM-RNN方法，这意味着时间序列分析适用于区分网络流量中的良性和异常。 此外，GRU [13]证明了轻量级深度学习模型在物联网环境中的实施非常有用，这在当今非常重要。

表**5.1** KDD Cup'99数据集上的模型比较

模型	特征提取器	分类器	准确率 (%)
DNN [1]	FF-NN	Softmax	99.994
LSTM-RNN-K [6]	LSTM-RNN	Softmax	96.930
LSTM-RNN-L [7]	LSTM-RNN	Softmax	98.110
LSTM-RNN-S [11]	LSTM-RNN	LSTM-RNN	93.820
GRU [13]	GRU	GRU	98.920

表**5.2** NSL KDD数据集上的模型比较

模型	特征提取器	分类器	准确率 (%)
STL [4]	AE	Softmax	79.10
DNN-SDN [8]	NN	NN	75.75
RNN [9]	RNN	RNN	81.29
CNN [10]	CNN	CNN	79.14

如表5.2所示，当使用NSL-KDD数据集时，仍有改进的空间。最准确的模型是RNN [9]，准确率为81.29%。
再次，这个事实表明时间序列分析可能会提高IDS的性能。
尽管使用CNN的IDS尚未达到最佳性能，但通过应用适当的文本到图像转换，我们可以获得CNN的好处，而CNN在图像识别方面已经被证明是最好的。

参考文献

1. S. S. Roy, A. Mallik, R. Gulati, M. S. Obaidat, 和 P. Krishna, “一种基于深度学习的人工神经网络入侵检测方法,” in国际数学
与计算会议. Springer, 2017, pp. 44–53.
2. S. Potluri 和 C. Diedrich, “用于增强入侵检测系统的加速深度神经网络,” in新兴技术和工厂自动化 *(ETF*A*)*, *2016 IEEE* 第*21*届国际会议. IEEE, 2016, pp. 1–8.

3. H. Larochelle, Y. Bengio, J. Louradour, 和 P. Lamblin, “探索训练深度神经网络的策略,”机器学习研究, vol. 10, no. Jan, pp. 1–40, 2009.
4. A. Javaid, Q. Niyaz, W. Sun, and M. Alam，“一种用于网络入侵检测系统的深度学习方法”，在第*9*届*EAI*国际生物启发式信息与通信技术会议（前身为*BIONETICS*）的论文集中。 *ICST*（计算机科学、社会信息学和电信工程研究所），2016年，第21-26页。

5. Y. Yu, J. Long, and Z. Cai，“基于会话的网络入侵检测：一种深度学习架构”，在“人工智能建模决策”中。Springer，2017年，第144-155页。
6. J. Kim, J. Kim, H. L. T. Thu, and H. Kim，“用于入侵检测的长短期记忆递归神经网络分类器”，在“平台技术与服务（*P*lat*C*on）”上，*201*6年国际会议上。*I*EEE，*201*6年，第*1*-5页。
7. Y. LIU, S. LIU, 和 Y. WANG, “基于长短期记忆循环神经网络的路由入侵检测,”*DE*Stech 计算机科学与工程交易, no.cii, 2017.

8. T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, 和 M. Ghogho, “软件定义网络中的网络入侵检测的深度学习方法,” 在无线网络和移动通信（*WINCOM*）, *2016*国际会议. IEEE, 2016, pp. 258–263.

9. C. Yin, Y. Zhu, J. Fei, 和 X. He, “一种使用循环神经网络进行入侵检测的深度学习方法,”*IEEE Access*, vol. 5, pp. 21 954–21 961, 2017.
10. Z. Li, Z. Qin, K. Huang, X. Yang, 和 S. Ye, “使用卷积神经网络进行表示学习的入侵检测,” 在神经信息处理国际会议. Springer, 2017, pp. 858–866.

11. R. C. Staudemeyer, “将长短期记忆循环神经网络应用于入侵检测,”南非计算机杂志, vol. 56, no. 1, pp. 136–154, 2015.
12. L. Bontemps, J. McDermott, N.-A. Le-Khac,*et al.*, “基于长短期记忆循环神经网络的集体异常检测,” 在未来数据和安全工程国际会议. Springer, 2016, pp. 141–152.

13. M. K. Putchala, “使用门控循环神经网络进行物联网中入侵检测系统 (IDS) 的深度学习方法,” 博士论文, WrightState University, 2017.

14. P. K. Bediako，“使用TensorFlow实现的长短期记忆递归神经网络用于检测DDoS洪水攻击。”2017年。
15. A. Dimokranitou，“用于图像异常事件检测的对抗自编码器，”博士学位论文，普渡大学，2017年。
16. S.-G. Choi和S.-B. Cho，“使用进化强化学习的自适应数据库入侵检测，在国际联合会议S*OCO*’17-C*ISIS*’17-*ICEUTE*’17 *Le* n,西班牙，*2017*年9月6日至*8*日，会议。Springer，*2017*年，pp. 547–556.
17. M. Feng和H. Xu，“基于深度强化学习的未知网络攻击下的网络物理系统最优防御，在计算智能（*SSCI*），*2017*年*IEEE*会议系列。*IEEE*，*2017*年，pp. 1–8.

第6章
深度特征学习

摘要FL是一种仅对数据的子集属性进行建模的技术。 它还有效地展示了检测性能与流量模型质量之间的相关性（Palmieri等人，Concurrency Comput Pract Exp26(5)：1113-1129，2014年）。 然而，特征提取和特征选择是不同的。 特征提取算法从原始特征中派生出新特征，以(i)减少特征测量成本，(ii)提高分类器效率，和(iii)提高分类准确性，而特征选择算法从总共 M个输入特征中选择不超过 m个特征，其中 m小于 M。因此，新生成的特征仅仅是从原始特征中选择出来的，没有任何转换。 然而，他们的目标是导出或选择一个具有较低维度的特征向量，用于分类任务。

深度学习模型的一个优点是可以处理输入数据的底层信息，适用于特征学习任务。 因此，我们讨论了深度学习在入侵检测系统中的关键作用，即深度特征提取和选择（D-FES）以及深度学习用于聚类。

6.1 深度特征提取和选择

移动技术的最新进展导致物联网设备在我们的日常生活中变得更加普遍和集成。 需要克服的安全挑战主要源于无线媒介（如Wi-Fi网络）的开放性质。 冒充攻击是指对系统或通信协议中的合法方进行伪装的攻击。 连接设备普遍存在，产生大规模的高维数据，这给同时检测带来了复杂性。 然而，特征学习可以规避网络数据大量性可能引起的潜在问题。Aminanto等人[2]提出了一种新颖的D-FES方法，它结合了堆叠特征提取和加权特征选择。 堆叠自编码能够通过重构相关信息来提供更有意义的表示。

© 作者（们），在Springer Nature Singapore Pte Ltd.的独家许可下。2018年

K. Kim等人，使用深度学习进行网络入侵检测，
关于网络安全系统和网络的SpringerBriefs简报，
https://doi.org/10.1007/978-981-13-1444-5_6

从原始输入中提取信息。 然后将这些表示与现有的浅层结构机器学习器中的修改加权特征选择相结合。 展示了精简特征集合减少机器学习模型的偏差以及计算复杂性的有用性。

6.1.1 方法论

可以采用D-FES中的特征提取和选择。图6.1展示了D-FES在两个目标类别下的逐步过程。 预处理过程包括归一化和平衡步骤是必要的。 该过程在第6.1.2节中详细解释。 如算法1所示，D-FES首先通过构建基于SAE的特征提取器，其中包括两个连续的隐藏层，来优化学习能力和执行时间[3]。 SAE输出50个提取的特征，然后与AWID数据集中存在的154个原始特征相结合[4]。 然后使用SVM、ANN和C4.5等经过良好引用的机器学习器利用加权特征选择方法构建候选模型，分别为D-FES-SVM、D-FES-ANN和D-FES-C4.5。 SVM使用支持向量（超平面）分离类别。

然后，ANN优化与隐藏层相关的参数，以最小化与训练数据相关的分类错误，而C4.5采用分层决策方案（如树）来区分每个特征[5]。 检测任务的最后一步是仅使用12-22个训练特征来学习ANN分类器。

图6.1中的监督特征选择模块由三种不同的特征选择技术组成。 这些技术类似，它们考虑其产生的权重来选择关键特征的子集。

ANN被用作加权特征选择方法之一。 ANN仅使用两个目标类（正常和冒充攻击类）进行训练。

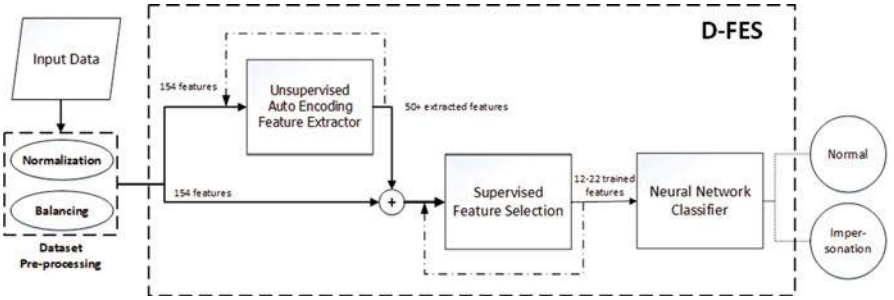


图**6.1**D-FES的逐步过程，包括两个目标类别：正常和冒充攻击

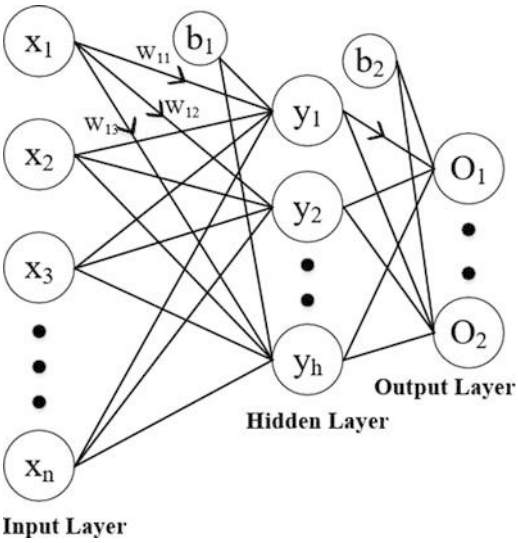
算法**1**D-FES的伪代码

```
1: 过程 D-FES
2:     函数 DATASET PRE-PROCESSING（原始数据集）
3:         函数（数据集归一化）原始数据集4：
               返回归一化数据集
5:         结束函数
6:         函数（数据集平衡）归一化数据集7：
               返回平衡数据集
8:         结束函数
9:         返回输入数据集
10:    结束函数
11:    函数DEEP ABSTRACTION（输入数据集）12
：         对于i等于1到h的循环                                  h等于2，隐藏层的数量13：
               对于每个数据实例的循
环14：             计算y_i（公式（3.7
））15：            计算 zi (方程式 (3.8))
16:                最小化 Ei (方程式 (3.12))
17:                θ_i = {W_i, V_i, b_fi, b_gi}
18:            结束循环
19:            W ← W_2                                      第二层，提取50个特征
20:        结束循环
21:        输入特征 ← W + 输入数据集
22:        返回输入特征
23:    结束函数
24:    函数特征选择 (输入特征)
25:        开关 D-FES执行
26:            情况 D-FES-ANN(输入特征)
27:                返回选择的特征
28:            情况 D-FES-SVM(输入特征)
29:                返回选择的特征
30:            情况D-FES-C4.5(输入特征)
31:                返回选择的特征
32:    结束函数
33:    过程分类(选择的特征)
34:        训练ANN
35:        最小化 E(公式(4.1))
36:    结束过程
37: 结束过程
```

图6.2显示了一个只有一个隐藏层的ANN网络，其中 b_1和 b_2 分别表示相应隐藏层和输出层的偏置值。

为了选择关键特征，考虑了第一和第二层之间的权重值。 权重表示输入特征对第一隐藏层的贡献。 接近零的 w_{ij}值意味着相应的输入特征 x_j对进一步传播没有意义，因此对于这个特定任务来说，只有一个隐藏层就足够了。每个输入特征的重要值如公式(6.1)所示。

图**6.2**只有一个隐藏层的ANN网络



算法**2**D-FES-ANN函数

函数 D-FES-ANN(输入特征)
训练ANN
W_{ij}
对于每个输入特征执行
计算 V_j(公式(6.1))
结束循环
降序排序
$SelectedFeatures \leftarrow V_j >$ 阈值
返回*SelectedFeatures*
：结束函数

$$V_j = \sum_{i=1}^{h} |w_{ij}|, \tag{6.1}$$

其中， h 是第一隐藏层中的神经元数量。 如算法2所述，特征选择过程涉及在输入特征按照 V_j值降序排序后，选择 V_j值高于阈值的特征。

在加权特征选择之后，ANN也被用作分类器。 在使用ANN进行学习时，执行最小全局误差函数。 它有两种学习方法，有监督和无监督。 本研究使用有监督方法，因为知道类别标签可能会提高分类器的性能。 此外，使用适用于大规模问题的缩放共轭梯度优化器。

在D-FES-SVM中使用SVM-RFE，使用线性情况[8]中描述的算法3。 输入是训练实例和类标签。首先，进行特征

算法3 D-FES-SVM函数

```
1: 函数 D-FES-SVM(输入特征)
2:     训练SVM
3:     计算 w (方程式(3.2))
4:     计算排名准则
5:     c_i = w_i^2
6:     找到最小的排名准则
7:     f = argmin(c)
8:     更新特征排名列表
9:     r = [s(f), r]
10:    消除最小的排名准则
11:    s = s(1:f-1, f+1:length(s))
12:    SelectedFeatures ← s
13:    返回 SelectedFeatures
14: 结束函数
```

初始化排名列表，其中包含一组重要特征的子集，用于选择训练实例。 该方案继续通过训练分类器并计算维度长度的权重向量。 在获得权重向量的值之后，计算排名准则并找到具有最小排名准则的特征。 使用该特征更新特征排名列表，并消除具有最小排名准则的特征。 最终创建了一个特征排名列表作为输出。

最后的特征选择方法是使用DT C4.5。特征选择过程从选择前三个层级节点开始，如算法4所述。 然后，它会删除相同的节点并更新所选特征的列表。

算法4：D-FES-C4.5函数

```
1: 函数 D-FES-C4.5(输入特征)
2:     训练 C4.5
3:     SelectedFeatures ← 前三个层级节点
4:     对于 i 从 1 到 n 执行                                  n = 选定特征的大小
5:         对于 j 从 1 到 n 执行
6:             if SelectedFeatures [i]=SelectedFeatures [j] 然后 删除 Se
   lectedFeatures[ j]7:
               结束如果
8:         结束循环
9:     结束循环
10:    返回 SelectedFeatures
11: 结束函数
```

6.1.2 评估

进行了一系列实验来评估所提出的 D-FES 方法在Wi-Fi冒充检测中的性能。选择适当的数据集是IDS研究领域中的重要步骤[9]。 本研究使用了AWID数据集[4]，该数据集包含了从真实网络环境中收集的Wi-Fi网络数据。 通过在与[4]相同的测试集上进行实验，实现了公平的模型比较和评估。 该方法使用MATLAB R2016a和从WEKA软件包[10]中提取和修改的Java代码在Intel Xeon E-3-1230v3 CPU @3.30 GHz和32 GB RAM上运行。

6.1.2.1 数据集预处理

AWID数据集中的数据具有多样性，包括离散、连续和符号，具有灵活的值范围。 这些数据特征可能使得分类器难以正确学习底层模式[11]。 因此，预处理阶段包括根据规范化步骤和数据集平衡过程将符号值属性映射为数值，具体描述见算法5。 目标类别被映射为这些整数值类别之一：正常实例为1，冒充为2，洪水攻击为3，注入攻击为4。 同时，接收器、目的地、发射器和源地址等符号属性被映射为整数值，最小值为1，最大值为所有符号的数量。 一些数据集属性，如WEP初始化向量（IV）和完整性检查值（ICV），是十六进制数据，也需要转换为整数值。 连续数据，如时间戳，也保留用于规范化步骤。 一些属性有问号，?，表示缺失值。

算法5：数据集预处理函数

```
1：函数DATASET PRE-PROCESSING（原始数据集）
2:      函数DATASET NORMALIZATION（原始数据集）
3:          对于每个数据实例，执行以下操作
4:              转换为整数值
5:              归一化（公式（6.2））
6:              归一化数据集
7:          结束循环
8:      结束函数
9:      函数DATASET BALANCING（归一化数据集）
10:         随机选择10%的正常实例
11:         平衡数据集
12:     结束函数
13:     输入数据集 ← 平衡数据集
14:     返回输入数据集
15：结束函数
```

表**6.1**：平衡和不平衡数据集中每个类的分布

类别		训练	测试
正常	不平衡	1,633,190	530,785
	平衡	163,319	53,078
攻击	冒充	48,522	20,079
	洪水	48,484	8,097
	注入	65,379	16,682
	总计	162,385	44,858

AWID数据集模拟了正常和攻击实例之间的自然不平衡网络分布。“平衡”意味着在正常实例（163,319个）和总攻击实例之间实现平等分布。

(162,385)。15%的训练数据被撤回用于验证数据。

不可用的值。 选择了一种替代方案，其中问号被赋予一个常数零值[12]。 最后，数据被转换为数值，需要进行属性归一化[13]。 数据归一化是一个过程；因此，每个属性的值范围都是相等的。采用了均值范围方法[14]，其中每个数据项在零和一之间进行线性归一化，以避免不同尺度的不当影响[12]。方程(6.2)显示了归一化公式：

$$z_i = \frac{x_i - min(x)}{max(x) - min(x)}, \tag{6.2}$$

其中 z_i表示归一化值， x_i指的是相应的属性值，而 $min(x)$和 $max(x)$分别是属性的最小值和最大值。

减少的“CLS”数据是真实网络的一个很好的表示，其中正常实例明显多于攻击实例。 正常和攻击实例之间的比例为10:1，适用于不平衡的训练和测试数据集如表6.1所示。这个特性可能对训练模型有偏见并影响模型性能[15, 16]。 为了缓解这个问题，通过随机选择10%的正常实例来平衡数据集。 然而，为了可重复性，设置了一个特定的值作为随机数生成器的种子。 正常和攻击实例之间的比例变为1:1，这是训练阶段的适当比例[16]。 使用平衡的数据集对D-FES进行训练，然后在不平衡的数据集上进行验证。

6.1.2.2 实验结果

提出的D-FES在一组实验中进行了评估。 首先，实现和验证了不同的特征提取器架构，如SAE。其次，采用了两种特征选择方法：基于过滤器和包装器的方法。

表**6.2** SAE方案的评估

SAE方案	*DR* (%)	*FAR* (%)	*Acc* (%)	F_1 (%)
Imbalance_40 (154:40:10:4)	64.65	**1.03**	96.30	73.13
Imbalance_100 (154:100:50:4)	85.30	1.98	**97.03**	81.75
Balance_40 (154:40:10:4)	72.58	18.87	77.21	74.48
Balance_100 (154:100:50:4)	**95.35**	18.90	87.63	**87.59**

验证。 最后，对一个真实的不平衡测试数据集验证了D-FES的有用性和实用性。

SAE架构被改变以优化SAE的实现，采用了两个隐藏层。 从第一个编码器层生成的特征被用作第二个编码器层的训练数据。 同时，每个隐藏层的大小相应减小，以便第二个编码器层学习输入数据的更小表示。 然后，在最后一步中实现了具有softmax激活函数的回归层。

对四种方案进行了研究，以确定SAE的学习特性。 第一个方案Imbalance_40有两个隐藏层，每层分别有40个和10个隐藏神经元。 第二个方案Imbalance_100也有两个隐藏层，但每层分别有100个和50个隐藏神经元。 虽然没有确定隐藏神经元数量的严格规则，但我们考虑了一个常见的经验法则[17]，即从输入中占70%到90%的范围。 第三个和第四个方案分别命名为Balance_40和Balance_100，与第一个和第二个方案具有相同的隐藏层结构；然而，在这种情况下，使用了平衡的数据集，因为常见的假设是高度不平衡的数据分布构建的分类器模型在少数类检测上表现不佳[18]。 为了测试目的，使用了AWID数据集中包含的所有四个类别。

表6.2显示了SAE方案的评估结果。 每个模型都使用平衡或不平衡的数据进行SAE算法，具体参数如下：输入特征，第一隐藏层中的特征数量，第二隐藏层中的特征数量和目标类别。 具有$_{100}$个隐藏神经元的SAE架构比具有40个隐藏神经元的架构具有更高的*DR*。 另一方面，具有40个隐藏神经元的SAE架构比具有$_{100}$个隐藏神经元的架构具有更低的*FAR*。 为了得出正确的结论，需要考虑其他整个类别的性能指标，因为*DR*仅检查攻击类别，而*FAR*仅测量正常类别。 Acc指标会受到数据分布的影响，不同的平衡和不平衡分布可能导致错误的结论。 如果我们仅考虑Acc指标，如图6.3所示，我们可能会错误地选择97.03%准确率的Imbalance_100，而Balance_100只能达到87.63%的准确率。 实际上，由于正常类别与攻击类别的不平衡比例，Imbalance_100实现了最高的准确率。 通过检查F_1分数，Balance_100在所有方案中实现了最高的F_1分数，为87.59%。因此，选择了拓扑结构为$_{154:100:50:4}$的SAE架构。

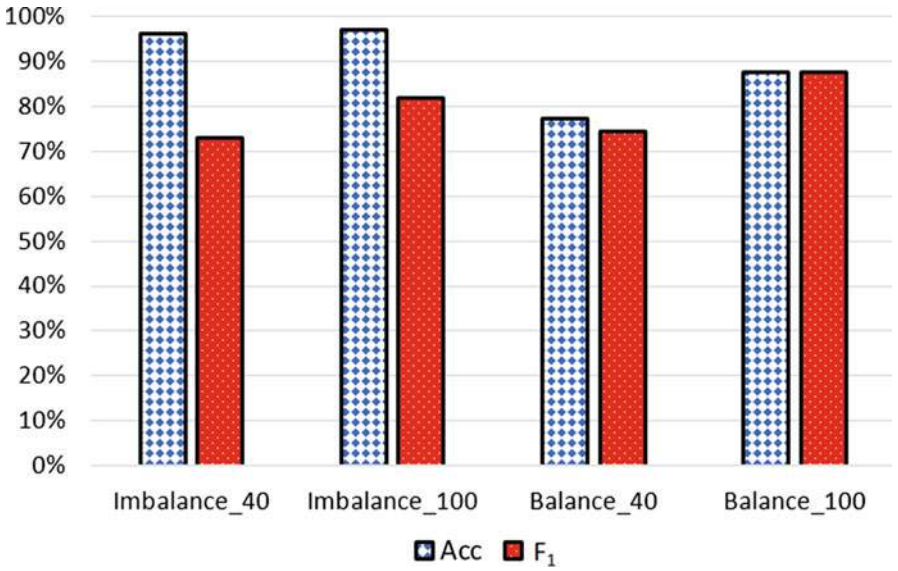


图**6.3**对SAE方案的评估Acc和F_1得分。 红色柱状图表示F_1得分，蓝色柱状图表示准确率。

为了展示D-FES的有效性，我们比较了以下特征选择方法：

- CfsSubsetEval [19]（CFS）考虑了每个特征的预测能力以及它们之间的冗余程度，以评估特征子集的重要性。 该方法选择与类别高度相关且相互关联性低的特征子集。
- 相关性（Corr）衡量特征与类别之间的相关性，以评估特征子集的重要性。
- 训练好的ANN模型的权重模拟了相应输入的重要性。 通过仅选择重要特征，训练过程比以前更轻量化和更快速[20]。
- SVM根据SVM分类结果的权重来衡量每个特征的重要性。
- C4.5是决策树方法之一。 它可以选择一组不高度相关的特征。 相关特征应该在同一分割中；因此，属于不同分割的特征不高度相关[21]。

基于过滤器的方法通常在不执行学习算法的情况下测量每个属性的相关性和冗余性。 因此，基于过滤器的方法通常是轻量级和快速的。 另一方面，基于包装器的方法检查任何输出特征子集的学习算法的结果[22]。

CFS和Corr属于基于过滤器的技术，而ANN、SVM和C4.5属于基于包装器的方法。

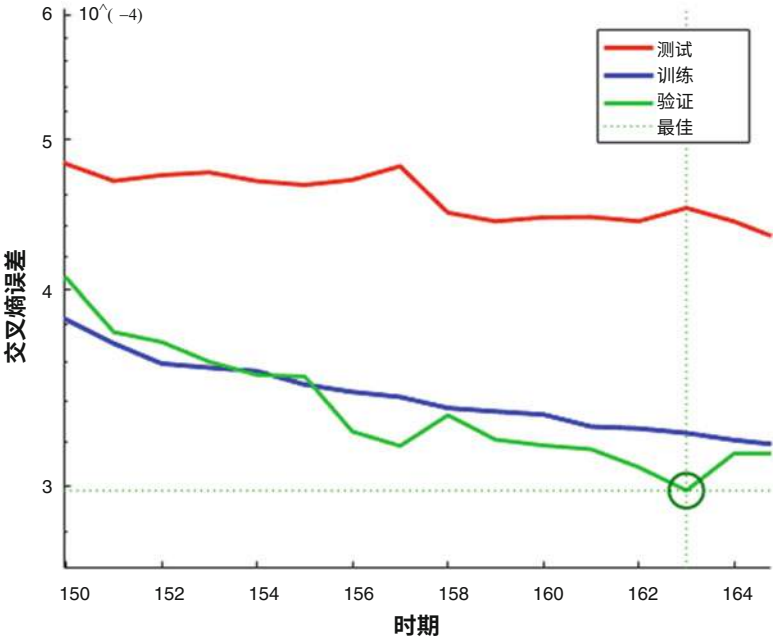


图**6.4**ANN的交叉熵误差。在第163个时期达到了最佳验证性能。

使用基于包装器的方法选择了一组特征来考虑每个特征的权重。 对于ANN，定义了一个阈值权重值，如果一个特征的权重高于阈值，则选择该特征。 SVM属性选择函数根据它们的权重值对特征进行排名。

然后选择具有高于预定义阈值的权重值的特征子集。 同样，C4.5生成一棵深度二叉树。 选择属于树的前三个级别的特征。 CFS生成一个固定数量的选定特征，Corr提供相关特征列表。

在特征选择和分类的ANN训练过程中，使用单独的验证数据集对训练模型进行优化；即将数据集分为三个部分，训练数据、验证数据和测试数据，比例分别为70%、15%和15%。 在训练过程中，将训练数据作为ANN的输入，并根据其分类错误调整神经元的权重。 验证数据用于测量模型的泛化能力，提供有关何时终止训练过程的有用信息。 测试数据用于训练后对模型性能进行独立测量。 当模型在验证数据集上达到最小平均平方误差时，可以说模型已经优化。 图6.4显示了ANN训练过程中交叉熵误差的性能示例。 在第163个时期，交叉熵误差，一种基于对数的误差

表**6.3** 特征集比较：特征选择与D-FES

方法	选定的特征	D-FES
CFS	5, 38, 70, 71, 154	38, 71, 154, 197
Corr	47, 50, 51, 67, 68, 71, 73, 82	71, 155, 156, 159, 161, 165, 166, 179, 181, 191, 193, 197
ANN	4, 7, 38, 77, 82, 94, 107, 118	4, 7, 38, 67, 73, 82, 94, 107, 108, 111, 112, 122, 138, 140, 142, 154, 161, 166, 192, 193, 201, 204
SVM	47, 64, 82, 94, 107, 108, 122, 154	4, 7, 47, 64, 68, 70, 73, 78, 82, 90, 94, 98, 107, 108, 111, 112, 122, 130, 141, 154, 159
C4.5	11, 38, 61, 66, 68, 71, 76, 77, 107, 119, 140	61, 76, 77, 82, 107, 108, 109, 111, 112, 119, 158, 160

在输出值和期望值之间进行比较的测量表明，在第163个时期，模型已经优化。 尽管训练数据在第163个时期之后输出的错误值逐渐减少，但模型的性能不再继续改善，因为逐渐减少的交叉熵误差可能表明过拟合的可能性。

表6.3包含从各种特征选择方法中选择的所有特征列表。 某些特征对于检测冒充攻击是必要的。 这些特征是第4个和第7个，由ANN和SVM选择，以及第71个，由CFS和Corr选择。所选特征的特性如图6.5a-b所示。蓝线表示正常实例，红线表示冒充攻击的特征。 根据数据实例的属性值，可以区分正常和攻击实例。

例如，一旦数据实例在第166个特征中具有属性值0.33，该数据实例被分类为攻击的概率就很高。 这也可以应用于第38个和其他特征。

表6.4列出了每个算法在所选特征集上的性能。
SVM实现了最高的检测率(99.86%)和 *Mcc*(99.07%)。然而，构建模型需要10,789秒的CPU时间，是观察到的模型中最长的时间。
正如预期的那样，基于过滤器的方法（CFS和Corr）快速构建了它们的模型；然而，CFS的 *Mcc*最低（89.67%）。

表6.5比较了候选模型在由D-FES生成的特征集上的性能。SVM再次实现了最高的检测率 (99.92%)和 *Mcc*(99.92%)。它还以仅为0.01%的值实现了最高的 *FAR*。同样，Corr实现了最低的 *Mcc*（95.05%）。这表明，基于包装器的特征选择优于基于过滤器的特征选择。 由于SVM表现最佳，所选特征的属性在表6.6中描述。

从表6.4和6.5中观察到以下模式：只有两种方法（Corr和C4.5）在没有D-FES的情况下显示出较低的*FAR*，这是预期的，以最小化所提出的IDS的*FAR*值。这种现象可能存在，因为原始特征和提取的特征之间没有相关性，因为Corr和C4.5测量每个特征之间的相关性。 基于过滤器的特征选择

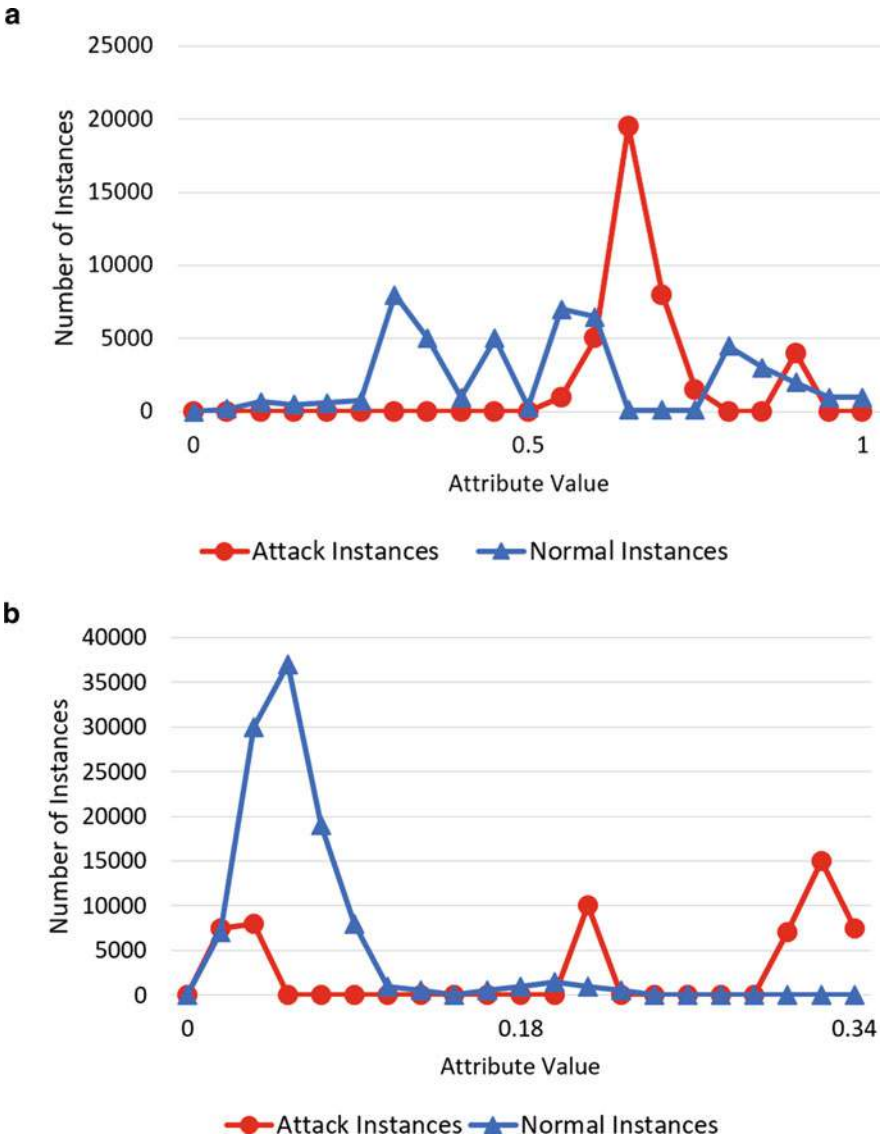


图**6.5**显示了第38个特征（**a**）和第**166**个特征（**b**）的特点。 蓝线表示正常实例，红线表示攻击实例

与D-FES所需的CPU时间相比，这些方法需要较短的CPU时间。然而，D-FES显著提高了基于过滤器的特征选择的性能。

图6.6a-c捕捉到了类似的模式，分别以准确率、精确率和F_1得分的性能来描述不同模型的表现。 D-*F*ES-SVM实现了最高的准确率、精确率和F_1得分，分别为99.97%、99.96%、

表**6.4** 选定特征的模型比较

模型	*DR* (%)	*FAR* (%)	*Acc* (%)	F_1 (%)	*Mcc* (%)	TBM (s)
CFS	94.85	3.31	96.27	92.04	89.67	80
Corr	92.08	0.39	97.88	95.22	93.96	**2**
ANN	99.79	0.47	97.88	99.10	98.84	150
SVM	**99.86**	0.39	**99.67**	99.28	99.07	10,789
C4.5	99.43	**0.23**	99.61	**99.33**	**99.13**	1,294

表**6.5** D-FES特征集的模型比较

模型	*DR* (%)	*FAR* (%)	*Acc* (%)	F_1 (%)	*Mcc* (%)	TBM (s)
CFS	96.34	0.46	98.80	97.37	96.61	1,343
Corr	95.91	1.04	98.26	96.17	95.05	**1,264**
ANN	99.88	0.02	99.95	99.90	99.87	1,444
SVM	**99.92**	**0.01**	**99.97**	**99.94**	**99.92**	12,073
C4.5	99.55	0.38	99.60	99.12	98.86	2,595

表**6.6** D-FES-SVM选择的特征集

索引	特征名称	描述
47	radiotap.datarate	数据速率 (Mb/s)
64	wlan.fc.type_subtype	类型或子类型
82	wlan.seq	序列号
94	wlan_mgt.fixed.capabilities.preamble	短前导码
107	wlan_mgt.fixed.timestamp	时间戳
108	wlan_mgt.fixed.beacon	信标间隔
122	wlan_mgt.tim.dtim_period	DTIM周期
154	data.len	长度

分别为99.92%和99.94%。 通过D-*F*ES，所有方法的准确率均超过96%，这表明D-*F*ES可以减少将正常实例错误分类为攻击的数量。 此外，D-*F*ES显著提高了基于过滤器的特征选择的准确率。 除了C4.5之外，所有特征选择方法在使用D-*F*ES时都改善了准确率和F_1得分。我们将D-*F*ES-SVM与随机选择的特征进行比较，以了解训练过程中涉及的特征数量，如图6.7所示。D-*F*ES-SVM的时间比随机方法长，并且显著增加了DR。 然而，随机方法甚至无法对单个冒充攻击进行分类。 这使得提出的D-*F*ES成为入侵检测器的一个很好的候选。

6.2 深度学习用于聚类

IDS已成为任何网络中的重要措施，尤其是Wi-Fi网络。
由于大量微型设备通过Wi-Fi网络连接，Wi-Fi网络的增长是不可否认的。 遗憾的是，对手可能利用这一点发动冒充攻击，这是一种典型的无线网络攻击。任何IDS

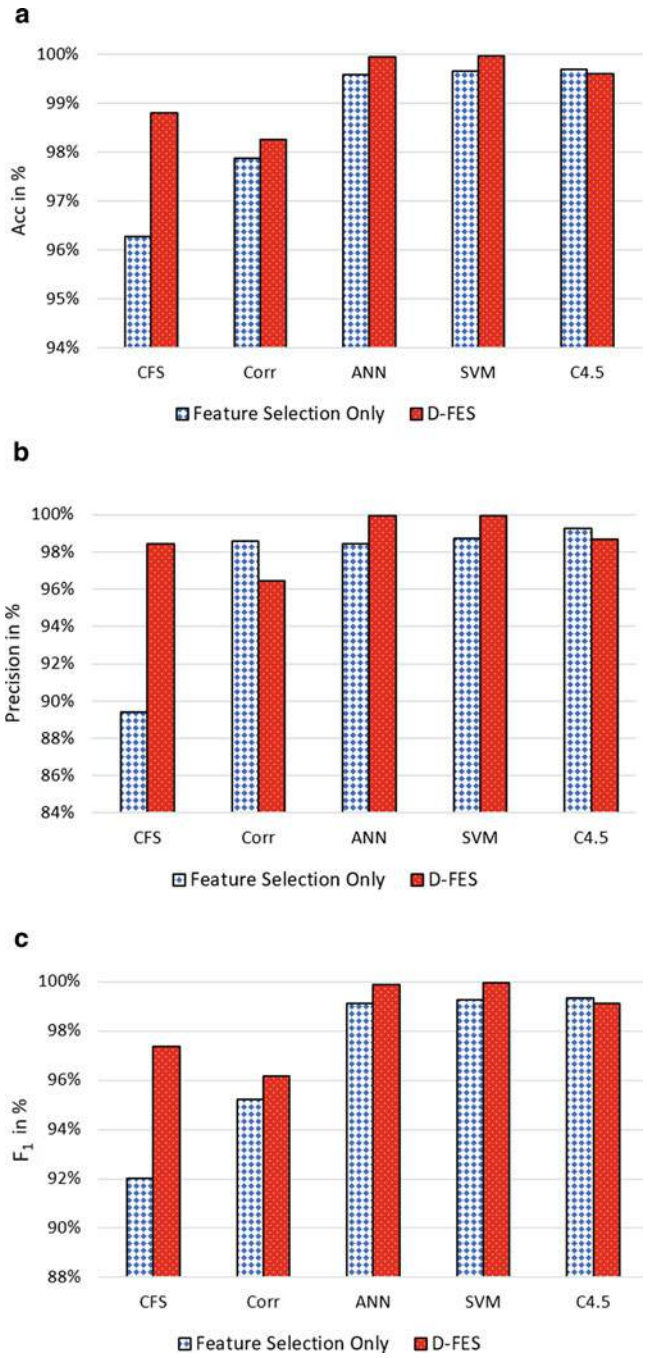


图**6.6** 模型性能比较，以(**a**)准确率，(**b**)精确率和(**c**) F_1得分为指标。蓝色柱代表仅使用特征选择的性能，而红色柱代表使用D-FES的性能

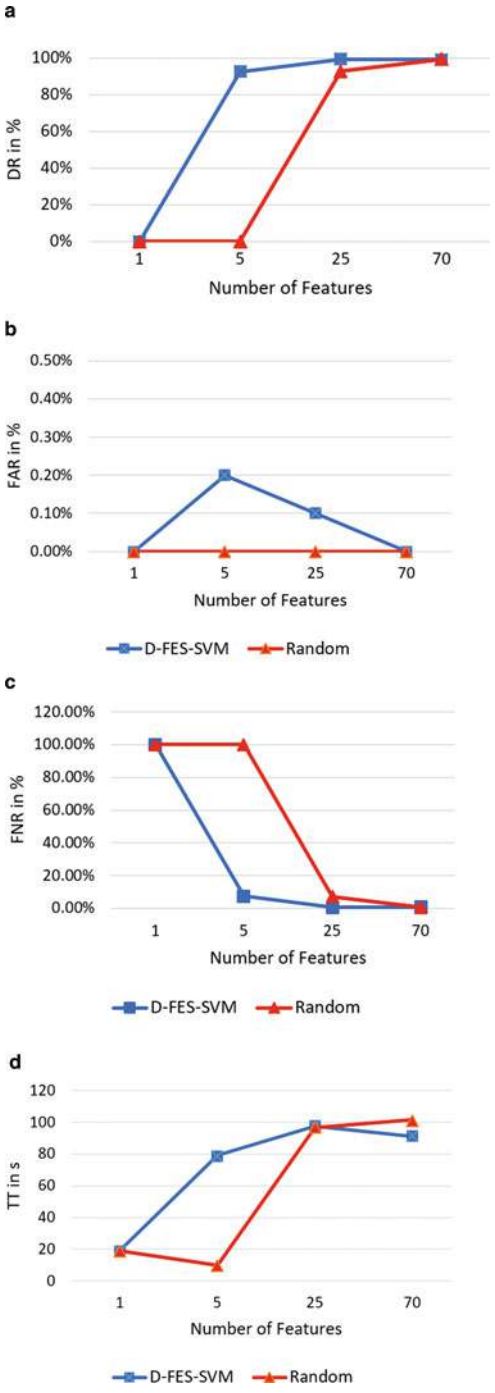


图**6.7** D-**F**ES与随机方法在(**a**)检测率，(**b**)误报率，(**c**)漏报率和(**d**) TT方面的模型性能比较

通常依赖于机器学习的分类能力，其中有监督学习方法在区分良性和恶意数据方面表现最佳。 然而，由于大量的流量，很难在Wi-Fi网络中收集带标签的数据。 因此，Aminanto和Kim [23]提出了一种新颖的完全无监督方法，可以在没有数据标签的先验信息的情况下检测攻击。 该方法配备了一个无监督的SAE用于提取特征，以及一个*k*-means聚类算法用于聚类任务。

6.2.1 方法论

在本节中，将解释一种新颖的完全无监督的基于深度学习的IDS，用于检测冒充攻击。 有两个主要任务，特征提取和
聚类任务。 图6.8显示了包含两个主要函数的方案
级联。 使用真实的Wi-Fi网络跟踪数据集AWID [4]，其中包含
154个原始特征。 在方案开始之前，应进行归一化和平衡处理
以实现最佳的训练性能。 算法6详细解释了
方案的过程。

该方案从两个级联编码器开始，然后将第二层的输出特征传递给聚类算法。 第一个编码器具有100个神经元作为第一隐藏层，而第二个编码器只有50个神经元。 选择隐藏层神经元数量的标准规则是使用前一层的70%到90%。 在本文中，$k=2$被定义为只考虑两个类别。 该方案通过*k*均值聚类算法形成了两个簇。 这些簇代表良性和恶意数据。

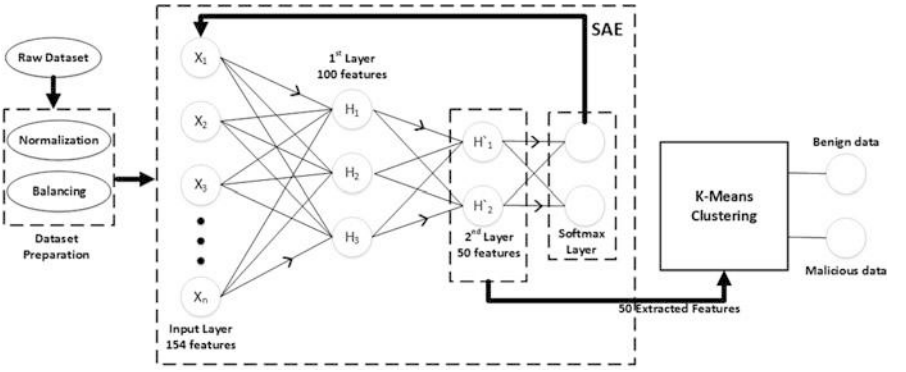


图**6.8**我们提出的方案包含特征提取和聚类任务

算法**6**完全无监督深度学习的伪代码

过程 开始
  函数 D数据集P准备 (原始数据集)
    对于每个数据实例，执行以下操作
      转换为整数值
      归一化 $z_i = \frac{x_i - min(x)}{max(x) - min(x)}$
    结束循环
    平衡归一化的数据集
    返回输入数据集
  结束函数
  函数 SAE(输入数据集)
    对于i等于1到h的循环 ▷ h=2；隐藏层的数量
      对于每个数据实例d执行
        计算 $H = s_f\left(W\,X + b_f\right)$
        计算 $X' = s_g\left(V\,H + b_g\right)$
        最小化 $E = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K}\left(X'_{kn} - X_{kn}\right)^2 + \lambda \cdot \Omega_{\text{权重}} + \beta \cdot \Omega_{\text{稀疏性}}$
        $\theta_i = \{W_i,\, V_i,\, b_{f_i},\, b_{g_i}\}$
      结束循环
      输入特征 $\leftarrow W_2$ ▷第二层，提取了50个特征
    结束循环
    返回输入特征
  结束函数
  初始化聚类和k=2 ▷两个聚类：良性和恶意
  函数 k-MEANS CLUSTERING (输入特征)
    返回聚类
  结束函数
  绘制聚类和目标类之间的混淆
结束过程

6.2.2 评估

SAE网络中有两个隐藏层，分别有100个和50个神经元 第二层的编码器由第一层编码器生成的特征输入 在SAE的最后阶段实现了softmax激活函数来优化SAE的训练 从SAE中提取的50个特征作为输入传递给k-means聚类算法 对于k-means聚类算法使用了随机初始化 然而，为了可重现性，必须定义一个特定的随机数种子值 将聚类结果与三个输入进行比较：原始数据，来自SAE第一隐藏层的特征和来自SAE第二隐藏层的特征，如表6.7所示

观察到传统的k-means算法的局限性，即无法对AWID数据集的复杂和高维数据进行聚类，仅准确率为55.93%。 尽管来自第一个隐藏层的100个特征达到了100%的检测率，但误报率仍然不可接受，为57.48%。

表**6.7** 我们提出的方案的评估

输入	*DR*(%)	*FAR*(%)	*Acc*(%)	精确率（%）	F_1(%)
原始数据	100.00	57.17	55.93	34.20	50.97
第一个隐藏层	100.00	57.48	55.68	34.08	50.83
第二个隐藏层	92.18	4.40	94.81	86.15	89.06

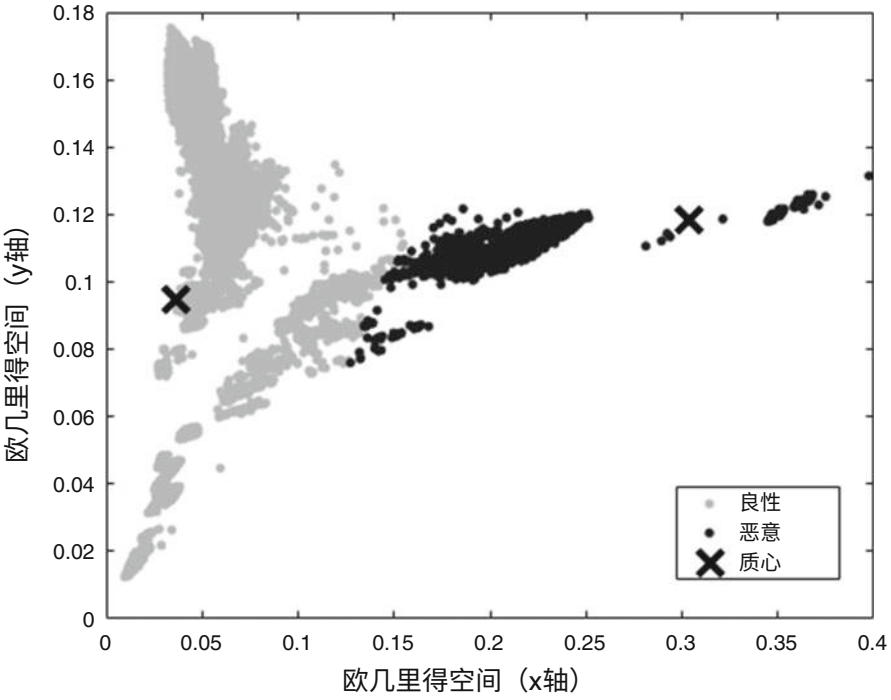


图**6.9** 我们提出的方案在欧几里得空间中的聚类结果

从第二个隐藏层的50个特征输入的*k*-means算法在所有算法中表现最好，显示出最高的F_1得分（89.06%）和准确率（94.81%），同时最低的误报率（4.40%）。尽管检测率稍低，但该方案在整体上改进了传统的*k*-means算法，F_1得分和准确率几乎提高了一倍。

图6.9展示了使用该方案在欧几里得空间中的聚类分配结果。
黑点代表攻击实例，而灰点代表良性实例。
每个聚类的质心位置用 *X*标记表示。

该方案的性能还与Kolias等人[4]和Aminanto和Kim [24]的两项先前相关工作进行了比较，如表6.8所示。该方案可以将冒充攻击实例分类为92.18%的DR，同时保持低 *FAR*，即4.40%。Kolias等人[4]在AWID数据集上测试了各种分类算法，如随机树、随机森林、J48、朴素贝叶斯等。其中

表**6.8** 与先前工作的比较

方法	*DR*(%)	*FAR*(%)	*Acc*(%)	精确率（%）	F_1(%)
Kolias等人[4]	22.01	0.02	97.14	97.57	35.92
Aminanto和Kim [24]	65.18	0.14	98.59	94.53	77.16
我们提出的方案[23]	92.18	4.40	94.81	86.15	89.06

表**6.9** 利用SAE的IDSs

出版物	SAE的作用	结合
AK16a [24]	分类器	ANN
AK16b [28]	特征提取器	Softmax回归
AK17 [23]	聚类	*K*均值聚类
ACTYK17 [2]	特征提取器	SVM, DT, ANN

在所有方法中，朴素贝叶斯算法在正确分类20,079个冒充实例中表现最好，正确分类了4,419个。 它仅达到了约22%的检测率，这是不令人满意的。 Aminanto和Kim [25]提出了另一种冒充检测器，将ANN与SAE结合起来，成功改进了用于冒充攻击检测任务的*IDS*模型，其检测率达到了65.18%，误报率为0.14%。在这项研究中，SAE被用于辅助提取特征的传统*k*均值聚类。 尽管该方案导致了较高的误报率，从而对*IDS* [26]产生了严重影响，但是由于采用了完全无监督的方法，这个误报率值约为4%，是可以接受的。 参数可以调整并降低误报率，但是较低的误报率或较高的检测率仍然是一种权衡，需要在将来进行研究。 观察到SAE的优势在于将复杂和高维数据抽象化，以辅助传统聚类算法，这通过方案实现了可靠的检测率和F_1得分。

6.3比较

深度学习方法的目标是从较低级别到较高级别的特征层次学习 [27]。 该技术可以独立地学习多个抽象级别的特征，从而直接从原始数据中发现将输入与输出之间的复杂函数映射，而不依赖于专家定制的特征。 在更高级的抽象中，人们通常无法从原始感官输入中看到关系和连接。 因此，随着数据量的急剧增加，必须强调学习复杂特征（也称为特征提取）的能力 [27]。 SAE是特征提取器的一个很好的例子。 因此，讨论了几个以SAE作为特征提取器和IDS模块中其他角色的先前工作，如表6.9所示。

表**6.10** 模拟检测比较

方法	DR (%)	误报率（%）
AK16a [24]	65.178	0.143
AK16b [28]	92.674	2.500
AK17 [23]	92.180	4.400
ACTYK17 [2]	99.918	0.012
KKSG15 [4]	22.008	0.021

通过SAE进行特征提取可以降低数据集的原始特征复杂性。 然而，除了作为特征提取器外，SAE还可以用于分类和聚类任务，如表6.9所示。AK16b [28]使用半监督方法进行IDS，其中包含特征提取器（无监督学习）和分类器（监督学习）。 SAE被用于特征提取和具有softmax激活函数的回归层进行分类器。 SAE作为特征提取器也用于ACTYK17 [2]，但是ANN、DT和SVM被用于特征选择。 换句话说，它结合了堆叠特征提取和加权特征选择。 根据实验结果 [2]，D-FES通过结合堆叠特征提取和加权特征选择改进了特征学习过程。 SAE的特征提取能够通过重构其输入并提供一种检查数据中是否捕获到相关信息的方式，将原始特征转化为更有意义的表示。

SAE可以高效地用于复杂数据集的无监督学习。

与之前的两种方法不同，AK16a [24] 和 AK17 [23] 在其他角色中使用了SAE，即分类和聚类方法。 ANN被采用作为特征选择，因为训练模型的权重模拟了相应输入的重要性 [24]。 通过仅选择重要特征，训练过程变得比以前更轻、更快。

AK16a [24] 将SAE用作分类器，因为它以分层方式使用连续的处理阶段进行模式分类和特征或表示学习。 另一方面，AK17 [23] 提出了一种新颖的完全无监督方法，可以在没有数据标签的先验信息的情况下检测攻击。 该方案配备了一个无监督的SAE用于提取特征，以及一个*k*-means聚类算法用于聚类任务。

Kolias等人[4]以启发式的方式测试了数据集上的许多现有机器学习模型。特别是在冒充攻击中观察到最低的DR，仅达到22%的准确率。 因此，改进冒充检测是具有挑战性的，因此在冒充检测方面的先前方法的比较总结在表6.10中。DR指的是在测试数据集中检测到的攻击实例数除以攻击实例的总数，而FAR是将正常实例分类为攻击实例的正常实例总数除以测试数据集中的正常实例总数。

从表6.10可以看出，与KKSG15 [4]相比，SAE可以提高IDS的性能。 验证了SAE对复杂和庞大的Wi-Fi网络数据的高级抽象能力。 SAE的模型无关性和

对复杂和大规模数据的可学习性适应了Wi-Fi网络的开放性质。 在所有IDS中，使用SAE作为分类器的IDS仅达到65.178%的冒充攻击DR。 这表明SAE可以作为分类器，但不如原始角色的SAE出色，原始角色是特征提取器。 AK16b [28]和ACTYK17 [2]验证了SAE作为特征提取器的可用性，它们实现了最高的DR。此外，通过SAE提取器和加权选择[2]的组合，实现了最佳的DR和FAR性能。 此外，有趣的是，SAE可以辅助*k*-means聚类算法实现更好的性能，DR为92.180% [23]。 然而，需要进一步分析以减少FAR，因为较高的FAR对于实际的IDS是不可取的。

参考文献

1. F. Palmieri, U. Fiore, 和 A. Castiglione，“基于独立成分分析的分布式网络异常检测方法”，并发计算与实践，卷26，第5期，页1113-1129，2014年。

2. M. E. Aminanto, R. Choi, H. C. Tanuwidjaja, P. D. Yoo, 和 K. Kim，“用于Wi-Fi冒充检测的深度抽象和加权特征选择”，*I*EEE信息取证与安全交易，卷13，第3期，页621-636，2018年。
3. Q. Xu, C. Zhang, L. Zhang, 和 Y. Song，“不同隐藏层堆叠自编码器的学习效果”，在第二届国际智能人机系统与控制论文集中，浙江，中国，卷02，*I*EEE，2016年8月，页148-151。
4. C. Kolias, G. Kambourakis, A. Stavrou, 和 S. Gritzalis, “802.11网络中的入侵检测：威胁的实证评估和公共数据集，” I*EEE* Commun. Surve*y*s *T*uts.，vol. 18, no. 1, pp. 184–208, 2015.
5. H. Shafri 和 F. Ramle, “使用兰卡威岛卫星数据的支持向量机和决策树分类的比较，” *I*nformation *T*echnology *J*ournal，vol. 8, no. 1, pp. 64–70, 2009.

6. L. Guerra, L. M. McGarry, V. Robles, C. Bielza, P. Larra aga, 和 R. Yuste, “神经元细胞类型的监督和无监督分类比较：一个案例研究，” *D*evelopmental neurobiology，vol. 71, no. 1, pp. 71–82, 2011.
7. M. F. Møller，“用于快速监督学习的缩放共轭梯度算法，”神经网络，第6卷，第4期，页525-533，1993年。
8. I. Guyon，J. Weston，S. Barnhill和V. Vapnik，“使用支持向量机进行癌症分类的基因选择，”机器学习，第46卷，第1-3期，页389-422，2002年。
9. A. zg r和H. Erdem，“2010年至2015年入侵检测和机器学习中KDD99数据集的使用综述，”*PeerJ PrePrints*，第4卷，第e1954v1页，2016年。
10. M. Hall，E. Frank，G. Holmes，B. Pfahringer，P. Reutemann和I. H. Witten，“weka数据挖掘软件的更新，”*ACM SIGKDD Explorations Newsletter*，第11卷，第1期，页10-18，2009年。
11. M. Sabhnani和G. Serpen，“在误用检测环境中应用机器学习算法于KDD入侵检测数据集。”在*Proc. Int. Conf. Machine Learning;Models, Technologies and Applications (MLMTA), Lax Vegas, USA*, 2003, pp. 209–215.
12. D. T. Larose,在数据中发现知识：数据挖掘简介. John Wiley & Sons, 2014.
13. H. Bostani和M. Sheikhan，“使用无监督学习和社交网络概念修改监督OPF基础入侵检测系统”，*Pattern Recognition*, vol. 62,pp. 56–72, 2017.

14. W. Wang, X. Zhang, S. Gombault和S. J. Knapskog，“网络入侵检测中的属性归一化”，在 *Proc. Int. Symp. Pervasive Systems, Algorithms, and Networks (ISPAN),Kaohsiung, Taiwan*. IEEE, Dec. 2009, pp. 448–453.
15. N. Y. Almusallam, Z. Tari, P. Bertok, and A. Y. Zomaya, “多数据流中入侵检测系统的维度约简 - 一项无监督特征选择方案的综述和提案”，*E*merg*e*nt *C*omputation，第24卷，第467-487页，2017年。[在线].
可用：https://doi.org/10.1007/978-3-319-46376-6_22
16. Q. Wei和R. L. Dunbrack Jr，“在生物信息学中用于二元分类器的平衡训练和测试数据集的作用”，*P*ublic Library of *S*cience *(PLoS)* one，第8卷，第7期，第1-12页，2013年。

17. Z. Boger和H. Guterman，“从人工神经网络模型中提取知识”，于*P*roc. *I*nt. *C*onf. *Sy*stems, *M*an, and *Cyb*ernetics, *O*rlando, *USA*，第4卷，*I*EEE，1997年，第3030-3035页。

18. G. M. Weiss和F. Provost，类别分布对分类器学习的影响：一项经验研究。新泽西州罗格斯大学计算机科学系，技术报告ML-*T*R-44，2001年。

19. M. A. Hall和L. A. Smith，“机器学习的实用特征子集选择”，在澳大利亚计算机科学会议（A*CSC*）上。珀斯，澳大利亚。S*p*ringer，1998年，第181-191页。

20. Z. Wang，“深度学习在流量识别上的应用”，在黑帽会议上。拉斯维加斯，美国。*UB*M，2015年。
21. C. A. Ratanamahatana和D. Gunopulos，“扩展朴素贝叶斯分类器：使用决策树进行特征选择”，在*IEEE*国际数据挖掘会议（*I*CD*M*）的数据清理和预处理研讨会上。前桥，日本。*IEEE*，2002年12月。
22. R. Kohavi和G. H. John，“用于特征子集选择的包装器”，人工智能，卷。97，第1期，第273-324页，1997年。
23. M. E. Aminanto和K. Kim，“通过完全无监督的深度学习改进Wi-Fi冒充检测”，信息安全应用：第*18*届国际研讨会，WI*SA2017*，20*17*年。

24. ——，“使用深度学习方法检测Wi-Fi网络中的冒充攻击”，信息安全应用：第*17*届国际研讨会，*WISA 2016*，2016年。
25. ——，“使用深度学习方法检测Wi-Fi网络中的冒充攻击”，在韩国济州岛信息安全应用研讨会（*WISA*）论文集中。Springer，2016年，第136-147页。
26. R. Sommer和V. Paxson，“超越封闭世界：使用机器学习进行网络入侵检测”，在加利福尼亚大学伯克利分校的安全与隐私研讨会上。IEEE，2010年，第305-316页。

27. Y. Bengio等人，“学习人工智能的深度架构”，*F*ound*at*ions *a*nd *tr*ends® *in Machine Learning*，卷2，号1，页1-127，2009年。
28. M. E. Aminanto和K. Kim，“通过半监督深度学习检测Wi-Fi网络中的主动攻击”，*Con*fer ence on *In*formation *S*ecurity and *C*ryptograp*hy* 2*017* Winter，2*016*年。

第7章
总结和进一步挑战

摘要本章作为本专著的结尾，提供了使用深度学习模型进行IDS目的的优势的总结声明，并解释了为什么这些模型可以提高IDS性能。 此外，还提出了深度学习在IDS的应用中的挑战和未来研究方向的概述。

总之，深度学习是机器学习模型的一个衍生物，它利用分层结构的数据处理阶段来进行UFL和模式分类。 深度学习的原理是处理所提供输入数据的分层特征，其中高层特征由低层特征组成。 此外，深度学习模型可以将特征提取器和分类器集成到一个框架中，从未标记的数据中自动学习特征表示，因此安全专家不需要手动设计所需的特征[1]。 基本上，深度学习方法可以从抽象的方面发现复杂的潜在结构/特征。 深度学习的这种抽象能力使得在所提供的数据中抽象出良性或恶意特征成为可能[2]。

深度学习建模的目标是学习和输出特征表示，使得这些模型更适合特征工程。 这里的特征工程包括特征/表示学习和特征选择[3]。

从最具特征的原始输入内部动态建模流量行为的能力对于展示异常检测性能与流量模型质量之间的相关性至关重要[4]。

未来还有进一步的挑战需要改进入侵检测系统。 基于我们之前的工作，我们建议未来入侵检测系统研究的方向，但不限于以下几点：

1. 深度学习方法的训练负载通常很大。 应该将DNN与异步多线程搜索相结合，在CPU上执行模拟，并在GPU上并行计算策略和值网络。因此，

© 作者，独家许可给 Springer Nature Singapore Pte Ltd. 2018

K. Kim 等人，使用深度学习进行网络入侵检测，
SpringerBriefs on Cyber Security Systems and Networks,
https://doi.org/10.1007/978-981-13-1444-5_7

如何在计算受限设备上应用这个深度学习模型是一个非常具有挑战性的任务。 我们应该使其更轻便，以适应物联网环境，例如无人驾驶车辆使用的控制器区域网络（CAN）。

2. 将深度学习模型作为实时分类器整合进来将是具有挑战性的。
   在以前的大多数利用深度学习方法的入侵检测系统环境中，它们执行特征提取或降低特征维度。
   而且，获得带有类标签的完整数据集并不容易。然而，深度学习模型仍然是分析大量数据的合适方法。
3. 改进无监督方法，因为很难获得大量标记数据。
   因此，一种利用无监督方法的入侵检测系统是可取的。
4. 构建一个能够以高检测率和低误报率检测零日攻击的入侵检测系统。

5. 未来需要一个综合性的措施，不仅包括检测，还包括预防。 因此，构建一个具有检测和预防能力的入侵检测系统（例如入侵预防系统（IPS））是期望的。
6. 通过使用LSTM网络进行时间序列分析，可以提供良好的异常检测器。 然而，再次强调，实时分析的训练工作量仍然很大。
   因此，这种网络的轻量级模型是可取的，如[5]所示。
7. CNN在许多研究领域，特别是图像识别领域取得了杰出的成果。 然而，在IDS研究中，并没有多少作品受益于使用CNN。我们期望通过应用适当的文本到图像转换，能够充分发挥CNN的潜力，就像在图像识别研究中已经展示的那样。

参考文献

1. Y. Wang, W.-d. Cai, and P.-c. Wei, “用于检测恶意JavaScript代码的深度学习方法”，《安全与通信网络》，第9卷，第11期，第1520-1534页，2016年。
2. W. Jung, S. Kim, and S. Choi, “海报：用于零日闪存恶意软件检测的深度学习”，在第*36*届*IEEE*安全与隐私研讨会上，2015年。
3. P. Louvieris, N. Clewley, and X. Liu, “基于效果的特征识别用于网络入侵检测,”神经计算, vol. 121, pp. 265–273, 2013.
4. F. Palmieri, U. Fiore, and A. Castiglione, “基于独立分量分析的网络异常检测的分布式方法,” 并行与计算实践, vol. 26, no. 5, pp. 1113–1129, 2014.

5. M. K. Putchala, “使用门控循环神经网络（GRU）的深度学习方法进行物联网（IoT）网络入侵检测系统（IDS）”，博士论文，WrightState University, 2017.

附录A
关于深度学习中的恶意软件检测的调查

本附录讨论了关于深度学习中的恶意软件检测的调查。
由于恶意软件和类似IDS的增加，恶意软件检测也是一个重要问题。

A.1 使用机器学习自动分析恶意软件行为

最近，计算机安全面临着安全挑战的增加。 由于静态分析容易受到混淆和逃避攻击的影响，Rieck等人尝试开发动态恶意软件分析。 使用动态恶意软件分析的主要挑战是执行分析所需的时间。 此外，随着恶意软件的数量和多样性增加，生成检测模式所需的时间也更长。因此，Rieck等人提出了一种基于行为分析的恶意软件检测方法，以提高恶意软件检测器的性能。

在这个实验中，使用了Malheur数据集。 这些数据集是由反恶意软件供应商Sunbelt Software的恶意软件二进制行为报告自行创建的。每个样本都在CW Sandbox的分析环境中执行和监控，并生成了3,131个行为报告。

在这个实验中，Rieck等人[1]使用了四个主要步骤。 首先，在沙盒环境中执行和监控恶意软件二进制文件。 它会输出系统调用和参数。 然后，在第二步中，基于行为模式将前一步产生的顺序报告嵌入到高维向量空间中。 通过这样做，可以几何地分析向量表示，以设计聚类和分类方法。 在第三步中，应用机器学习技术进行聚类和分类，以识别恶意软件的类别。 最后，对恶意软件的行为进行增量分析。

© 作者，独家许可给Springer Nature Singapore Pte Ltd. 2018

K. Kim等人，使用深度学习进行网络入侵检测，
SpringerBriefs on Cyber Security Systems and Networks,
https://doi.org/10.1007/978-981-13-1444-5

通过在聚类和分类步骤之间交替进行，结果表明该方法成功地减少了运行时间和内存需求，通过以块的方式处理行为报告。 结果显示，所提出的方法通过以块的方式处理行为报告，成功地减少了运行时间和内存需求。 增量分析处理数据需要25分钟，而常规聚类需要100分钟。此外，常规聚类在计算过程中需要5千兆字节的内存，而增量分析只需要不到300兆字节。因此，可以得出结论：基于行为的增量分析技术在时间和内存需求方面比常规聚类具有更好的性能。

A.2 恶意软件系统调用序列的深度学习分类

如今，恶意软件的数量和种类不断增加。 因此，恶意软件的检测和分类需要改进以进行安全预防。
本文旨在对恶意软件系统调用序列进行建模，并使用深度学习进行分类。 在这个实验中，利用机器学习的主要目的是在大型数据集中找到基本模式。 他们使用了从Virus Share、Maltrieve和私人收藏中收集的恶意软件样本数据集。 本文的主要贡献有三个。 首先，Kolosnjaji等人构建了深度神经网络，并将其应用于系统调用序列的检测。 然后，为了优化恶意软件分类过程，结合了卷积神经网络和循环神经网络。 最后，通过检查神经元的激活模式来分析他们提出的方法的性能。

在恶意软件分类过程中，Kolosnjaji等人[2]使用数据集中的恶意软件收集作为Cuckoo Sandbox的输入。然后，沙箱会生成数值特征向量作为输出。 之后，他们使用TensorFlow和Theano框架构建和训练神经网络。 神经网络的输出是恶意软件家族的列表。神经网络由卷积部分和循环部分组成。 卷积部分包括卷积层和池化层。 首先，卷积层捕捉相邻输入向量之间的相关性，并生成新的特征，从而得到特征向量。 然后，卷积层的结果被转发到循环层的输入。 在循环层中，使用LSTM单元对结果序列进行建模，并根据平均池化对重要性进行排序。 最后，使用dropout和softmax层来防止输出中的过拟合。 实验结果显示，卷积网络和LSTM的组合相比前馈网络和卷积网络，具有更高的准确性（89.4%对比79.8%和89.2%）。

A.3 使用深度神经网络进行恶意软件检测 基于进程行为

本文的背景问题是基于数据流量检测计算机是否存在恶意软件感染。 通常需要专家知识来进行分析，而且所需时间不少。 因此，本文的目的是利用机器学习通过使用流量数据来检测恶意软件感染。Tobiyama等人利用循环神经网络（RNN）进行特征提取，并使用卷积神经网络（CNN）进行分类。 本文之所以好，是因为在使用RNN进行训练阶段时，他们使用了LSTM。RNN由于其顺序结构而被称为错误消失问题。 其输出取决于先前的输入。 结果是，当先前的输入随时间增长时，会发生错误。 LSTM通过选择未来输出所需的信息来避免错误问题，以减少数据量。

在这篇论文中，Tobiyama等人[3]使用了81个恶意软件日志文件和69个良性进程日志文件进行训练和验证。 该数据集是通过在模拟环境中运行恶意软件文件来生成的，使用了Cuckoo Sandbox。 然后，他们追踪恶意软件的进程行为，以确定生成和注入的进程。
本文的方法如下：（1）在行为过程监控期间生成日志文件；（2）使用RNN进行特征提取，基于第1步的日志文件；（3）将提取的特征转换为图像特征；（4）使用图像特征训练CNN；（5）使用训练好的模型评估验证过程。
结果显示，所提出的模型实现了92%的检测准确率。 缺点是数据集太小，他们进行了5分钟的日志记录10次，因此该系统未经过大规模数据的测试。

基于深度学习的网络行为的高效动态恶意软件分析

现今，恶意软件检测方法可以分为三类：静态分析、主机行为分析和网络行为分析。 静态分析方法可以通过打包技术来规避。 主机行为分析可以通过代码注入来欺骗。 因此，网络行为分析成为焦点，因为它没有这些漏洞，并且攻击者和受感染主机之间的通信需求使得这种方法有效。 阻碍网络行为分析在恶意软件检测中使用的一个主要挑战是分析时间。 由于用户不知道恶意软件何时开始活动，需要收集和分析各种恶意软件样本很长一段时间。 本文作者Shibahara等人提出的主要思想针对恶意软件通信的两个特征，即通信目的的变化和常见的潜在功能。 对于数据集，首先从VirusTotal收集了被检测为恶意软件的样本。

通过杀毒程序。 然后，他们使用在VirusTotal中具有与先前收集的恶意软件样本不同的sha1哈希的恶意软件样本。 他们总共使用了29,562个恶意软件样本进行训练、验证和分类。

他们的方法包括三个主要步骤：特征提取、神经网络构建以及训练和分类标签。首先，Shibahara等人[4]从通信中提取特征，这些特征是通过动态恶意软件分析收集的。 然后，这些特征被用作递归神经网络（RNN）的输入。 然后，在神经网络构建阶段，捕捉到了通信目的的变化。 在训练和分类过程中，根节点的特征向量是根据VirusTotal计算的。然后，基于这些向量，给出了分类结果。 在实验中，比较了提出的方法和常规连续方法之间的分析时间和时间减少。

结果显示，所提出的方法将分析时间减少了67.1%，并且与完整分析方法相比，覆盖URL范围保持在97.9%。

A.5 自动恶意软件分类和新恶意软件检测使用机器学习

恶意软件的发展正在迅速进行。 各种恶意软件不断出现，增加了恶意软件家族的多样性。 因此，传统的基于静态的恶意软件分析无法检测到这些新型恶意软件。 因此，刘等人提出了一种基于机器学习的恶意软件分析系统。

为了支持这项研究，使用ESET NOD32、VX Heavens和Threat Trace Security从他们的校园网络收集了大量的恶意软件信息。 收集了大约21,740个恶意软件样本，属于9个家族，包括病毒、蠕虫、木马、后门等。 总共使用了19,740个样本进行训练，使用了2,000个样本进行测试。

该方法包括三个主要模块：数据处理、决策和恶意软件检测。 数据处理模块包括灰度图像、Opcode n-gram和导入函数，用于提取恶意软件特征。

决策模块根据数据处理模块提取的特征进行分类和恶意软件识别。 它包括分类器1、分类器2、分类器3，直到分类器N。最后，恶意软件检测模块中的聚类过程使用共享最近邻（SNN）聚类算法来发现新的恶意软件。 聚类的结果可能是明确的，也可能是模棱两可的。 对于模棱两可的结果，使用SNN进行识别。在实验中，使用随机森林、K最近邻（kNN）、梯度提升、朴素贝叶斯、逻辑回归、支持向量机（SVM）和决策树等多个分类器来测试所提出方法的性能。 实验结果显示，所提出方法的平均准确率为91.4%，而使用随机森林分类器时的最佳准确率为96.5%。

A.6 DeepSign：用于自动恶意软件签名生成和分类的深度学习

最近，各种恶意软件不断增长，包括一种新的变种恶意软件，这些软件无法被防病毒软件检测到。 已经提出了几种方法来克服这个问题，例如使用基于特定漏洞、载荷和诱饵的签名。 然而，所有这些方法都有一个很大的问题；它们针对恶意软件的特定方面。 因此，如果对手修改其恶意软件的一小部分，它们将无法检测到。 基于这个背景，David等人提出了一种不依赖于恶意软件特定部分的签名生成方法，因此它可以抵抗代码修改器。 在这项研究中，David等人使用了一个包含六类恶意软件的数据集，如Zeus、Carberp、Spy-Eye、Cidox、Andromeda和DarkComet。他们总共使用了1800个样本，每个类别300个样本。

该方法论包括四个主要部分。 首先，在模拟环境Cuckoo Sandbox中运行恶意软件程序。 沙箱的输出是沙箱日志文件。 然后，将日志文件转换为二进制位串。 位串作为深度置信神经网络（DBN）的输入，生成30个大小的向量作为输出层。 通过这一步骤生成了恶意软件签名。 问题是他们如何将沙箱文件转换为固定大小的输入。 为了做到这一点，David等人从数据集中提取了每个沙箱文件的所有单字。 然后，对于每个单字，计算它出现的文件数。 之后，他们选择了出现频率最高的前20,000个单字，并最终将每个沙箱文件转换为一个20,000位的位串。 在实验中，他们使用了1,200个样本进行训练（每个恶意软件类别200个样本），并使用了600个样本进行测试（每个类别100个样本）。 生成的特征数量为30，准确率为98.6%。输入噪声设置为0.2，学习率设置为0.001。

A.7 选择用于分类恶意软件的特征

自几年前机器学习的发展以来，有各种想法将机器学习作为恶意软件检测软件中的引擎实现。 然而，由于恶意软件着陆在用户系统上和签名生成过程之间存在时间延迟，这可能对用户造成伤害。 因此，Raman等人利用数据挖掘来识别Microsoft PE文件格式中的七个关键特征，这些特征可以用作分类器的输入。 这七个特征将用于机器学习算法进行恶意软件分类。 Raman等人从PE文件中生成他们的数据集。 首先，他们编写了一个解析器来从PE文件中提取特征。 他们利用自己在恶意软件分析方面的经验从最初的645个特征中选择了一组100个特征。 最后，他们创建了一个包含5,193个恶意文件和3,722个干净文件的数据集来评估这100个特征。

该方法的重点是特征提取和特征选择，在特征选择过程中结合了直观方法和机器学习方法。首先，Raman等人[7]利用他们的知识将特征数量从645个减少到100个。 然后，利用随机森林算法选择了13个特征。 最后，使用四个分类器（J48Graft、PART、IBk和J48）来检查每个特征的准确性，并选择最高的七个特征。 这些特征包括调试大小（表示调试目录表的大小）、图像版本（表示文件的版本）、调试RVA（表示导入地址表的相对虚拟地址）、导出大小（表示导出表的大小）、资源大小（表示资源部分的大小）、虚拟大小2（表示第二部分的大小）和节的数量（表示节的数量）。

A.8 分析基于行为的恶意软件检测中使用的机器学习技术

本文的主要问题是恶意软件种类的增加，导致易受攻击的手动启发式恶意软件检测方法。 为了解决这个问题，Firdausi等人提出了一种利用机器学习的自动行为-based恶意软件检测方法。 在这项研究中，他们使用了Windows可移植可执行文件格式的数据集。该数据集包括从Windows XP 32位SP2的System 32收集的250个良性实例。他们还从各种资源收集了220个恶意软件样本。 对于监控过程，Firdausi等人使用了Anubis，一个免费的在线自动动态分析服务，来监控恶意软件和良性样本。 然后，使用他们提出的方法比较了五个分类器的性能。 这五个分类器分别是kNN、朴素贝叶斯、J48决策树、支持向量机和多层感知器（MLP）。

对于研究的方法论，首先Firdausi等人从社区和virology.info进行了数据采集。然后，在模拟的沙盒环境中分析了每个恶意软件的行为。 选择了Anubis沙盒进行API挂钩和系统调用监视。 之后，报告将被处理成稀疏向量模型。报告以xml格式生成。 在下一步中，基于该XML文件进行了XML文件解析、特征选择和特征模型创建。 最后一步是基于该模型进行分类。

Firdausi等人应用了机器学习工具，进行了参数调整，最后测试了他们的方案。 这个实验的结果表明，特征提取将属性从5,191个减少到116个。 通过进行特征选择，训练和构建模型所需的时间变得更短。 J48分类器取得了最佳综合性能，真正阳性率为94.2%，误报率为9.2%，精确度为89.0%，准确度为92.3%。

A.9 使用基于机器学习的虚拟内存访问模式分析进行恶意软件检测

多年来，人们知道传统的恶意软件检测可以分为静态和动态方法。 这些方法通常在防病毒软件中实现。 静态方法使用签名数据库来检测恶意软件；另一方面，动态方法运行可疑程序来检查其行为，判断其是否为恶意软件。然而，这里存在一个问题。 在感染过程中，软件容易受到恶意软件的攻击，并且可以被恶意软件禁用。 因此，在本文中，徐等人提出了一种硬件辅助检测机制的想法，该机制不容易受到禁用问题的影响。 然而，这个想法依赖于可执行二进制文件及其内存布局的专家知识。 因此，作为解决方案，他们决定使用机器学习来检测恶意软件的恶意行为。

本文的主要目的是为每个应用程序学习一个模型，将感染恶意软件的执行与合法执行分开。此外，徐等人将根据内存访问模式进行分类。

为了监控内存访问，徐等人[9]进行了基于时期的监控。
一种监控方法将程序执行分为时期。 然后，每个时期通过在内存流中插入一个标记来分隔。 他们发现，对于大多数恶意行为，决定性特征是内存访问的位置和频率，而不是它们的顺序。 然后，在监控完成后，使用时期的摘要直方图对分类器进行训练。 在训练过程中，执行程序，并将每个直方图标记为恶意或良性。

在定义了训练模型之后，验证了二进制签名，并将模型加载到硬件分类器中。 最后，最后一步是硬件执行监控。 如果检测到恶意软件，将自动启动验证处理程序。 在实验中，他们使用了三个分类器（SVM、随机森林和逻辑回归）并比较了它们的性能。 表现最好的分类器是随机森林，真正阳性率为99%，假阳性率小于1%。

A.10 零日恶意软件检测

最近，恶意软件的种类不断增加，威胁着我们计算机系统的安全。 人们不能再依赖传统的基于签名的防病毒软件了。 零日恶意软件很容易绕过常规的防病毒软件，因为它们的签名尚未包含在防病毒数据库中。为了解决这个问题，Gandotra等人提出了一种结合静态和动态恶意软件分析的机器学习算法，用于恶意软件的检测和分类。 然而，这种方案存在一些问题。 首先，这种方案的误报率和漏报率很高。 其次，由于数据集很大，建立分类模型需要时间。 因此，这种方案无法实现早期的恶意软件检测。

在了解了这些问题之后，Gandotra等人得出结论，挑战在于选择相关的特征集，以减少建模时间并提高准确性。 本实验使用的数据集来自VirusShare。 使用了约3130个可执行文件，其中包括1720个恶意文件和1410个清洁文件。 所有文件都在沙箱中执行，以获取它们的属性，然后使用WEKA构建分类模型。方法包括六个主要步骤。 第一步是数据获取。

在这一步中，他们从VirusShare数据库中收集了针对Windows操作系统的恶意软件样本。 他们还手动从Windows系统目录中收集了干净的文件。第二步是自动化恶意软件分析。 在这个阶段，他们使用修改过的Cuckoo Sandbox执行样本并生成结果，结果以JavaScript对象表示法（JSON）文件的形式呈现。 第三步是特征提取。 在这一步中，通过解析Cuckoo Sandbox生成的JSON报告，获取了各种恶意软件特征。 结果是一个包含18个恶意软件属性的特征集，可以用于构建分类模型。 下一步是特征选择。

他们使用信息增益方法选择了七个顶级特征。 信息增益方法是一种基于熵的特征评估方法，在机器学习中广泛使用。
最后一步是分类。 他们使用选定的七个特征，在WEKA库中使用机器学习算法构建了分类模型。 所使用的七个分类器是IB1、朴素贝叶斯、J48、随机森林、装袋、决策表和多层感知器。 他们实验的结果显示，随机森林具有最高的准确率，达到99.97%，建模时间为0.09秒。

参考文献

1. K. Rieck, P. Trinius, C. Willems, and T. Holz, “使用机器学习自动分析恶意软件行为,”《计算机安全杂志》, vol. 19, no. 4, pp. 639–668, 2011.
2. B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, “使用深度学习对恶意软件系统调用序列进行分类,” in《澳大利亚人工智能联合会议》.Springer, 2016, pp. 137–149.
3. S. Tobiyama, Y. Yamaguchi, H. Shimada, T. Ikuse, and T. Yagi, “使用深度神经网络对进程行为进行恶意软件检测,” in《计算机软件和应用会议》*(COMPSAC), 2016* I*EEE*第*40*届, vol. 2. IEEE, 2016, pp. 577–582.
4. T. Shibahara, T. Yagi, M. Akiyama, D. Chiba, and T. Yada, “基于深度学习的网络行为的高效动态恶意软件分析”，在*2016*年*I*EEE全球通信大会*(GLOB*EC*OM)*上。 *I*EEE, *2016*, pp. *1*–7.
5. L. Liu, B.-s. Wang, B. Yu, and Q.-x. Zhong, “使用机器学习进行自动恶意软件分类和新恶意软件检测”，《信息技术与电子工程前沿》, vol. 18, no. 9, pp. 1336–1347, 2017.
6. O. E. David 和 N. S. Netanyahu, “Deepsign: 用于自动恶意软件签名生成和分类的深度学习”，在*2015*年国际联合神经网络会议*(IJC*NN*)*上。 *I*EEE, *2015*, pp. *1*–8.
7. K. Raman等，"选择用于分类恶意软件的特征"，*I*nfoS*e*c Sou*th*w*est*，卷2012，2012年。

8. I. Firdausi，A. Erwin，A. S. Nugroho等，"行为基础恶意软件检测中使用的机器学习技术分析"，在计算、控制和电信技术进展（A*CT*）中，*2010*年第二届国际会议上。 IEEE，*2010*年，页码*201-203*。
9. Z. Xu, S. Ray, P. Subramanyan, 和 S. Malik, “基于机器学习的虚拟内存访问模式分析的恶意软件检测,” 在欧洲设计、自动化和测试会议的论文集中。 欧洲设计和自动化协会，2017年，第169-174页。

10. E. Gandotra, D. Bansal, 和 S. Sofat, “零日恶意软件检测,” 在嵌入式计算和系统设计 *(I*SED*)*，*2016*年第六届国际研讨会中。*I*EEE，*2016*年，第171-175页。