摘 要

. 网络信息系统需要采取主动的防御措施。入侵检测技术是近 20 年来出现的一种主动保护系统、免受黑客攻击的新型网络安全技术。传统的入侵检测算法是基于监督学习的，检测率较高，误报率较低，但无法检测到未知攻击，且要求将数据正确地标记为正常或异常。网络环境中存在大量的未标记数据，要正确地标记这些数据，几乎是不可行的。如果将非监督学习方法应用到入侵检测中，基于聚类的入侵检测算法能够检测未知攻击，检测率较高，但误报率也较高。由此本文提出基于半监督学习的入侵检测算法。

半监督学习是机器学习领域中一个新的研究热点，通过标记数据和未标记数据的联合概率分布，来改进分类器的性能。根据网络数据的特点，本文提出了基于半监督聚类的入侵检测算法，利用少量的标记数据，生成用于初始化算法的种子聚类，然后辅助聚类过程，检测已知和未知攻击。在网络环境中，标记数据是有限的，为了充分利用监督信息，用户需要主动查询标记数据的约束，而不是随机选择约束，这样即使少量的约束也能大大改进算法的性能。

本文系统地研究入侵检测系统的基本理论，介绍了入侵检测的定义，分析了入侵检测的模型、研究现状和当前存在的问题。针对基于聚类的入侵检测算法误报率高的问题，提出了基于半监督聚类的入侵检测算法 ACKID。论文将主动学习策略应用于半监督聚类过程中，主动学习策略查询网络中未标记数据与标记数据的约束关系，采用 FarthestFirst 对未标记数据进行标记。

KDD Cup99 数据集是用于评估入侵检测算法的标准数据集，结合 KDD Cup99 数据集，分析了 ACKID 入侵检测算法的评估过程，确定 ROC 曲线为 ACKID 算法的评估指标，分析网络数据的属性特征并对数据预处理，最后评估 ACKID 算法的性能。

实验结果表明，ACKID 算法能够检测出未知攻击，证实利用标记数据和约束可以提高算法的检测率，降低误报率，并且表明采用主动学习策略能够获取最有用的监督信息以检测未知攻击。

关键词：入侵检测；半监督聚类；主动学习；ROC 曲线

Abstract

Information system needs active protection measures. During these two decades, intrusion detection which protects system actively from hacker's attacks is a new technique. The traditional algorithms for intrusion detection based on supervised learning can't detect unknown attacks and request that data are correctly labeled as normal or anomaly, which detection rates are higher and false positive rates are lower. There are lots of data in network environment, especially for labeling new unknown attacks correctly is hardly possible. If the methods of unsupervised learning are applied to intrusion detection, the intrusion detection algorithms based on clustering can detect unknown attacks, which detection rates are higher whereas false negatives rates are also higher. Consequently, the paper proposes the algorithm for intrusion detection based on semi-supervised clustering.

Semi-supervised learning is one of new research of many hot topics, which attains joint probability distribution of labeled data and unlabeled data to improve classifier's performance. The paper proposes the algorithm for intrusion detection based on semi-supervised clustering which uses a few limited labeled data to generate seed clusters initiating the algorithm and then aids clustering process to detect known and unknown attacks. There are a few labeled data in network environment. In order to maximize the utility of the limited supervised data available in a semi-supervised setting, constrains of labeled data should be selected as maximally informative ones actively rather than chosen at random, if possible. In that case, fewer constraints will be required to improve the clustering accuracy significantly.

Systematically, the paper investigates the basic theory of intrusion detection system, introduces the definition of intrusion detection, and analyses the models of intrusion detection and research state-of-art and existing problems nowadays. Aiming at the problems of intrusion detection algorithm based on clustering, the paper proposes the algorithm for intrusion detection based on semi-supervised clustering, namely ACKID algorithm. The paper applies active learning strategy to semi-clustering process. Active learning queries constrains on labeled data and unlabeled data, which uses FarthestFirst to label the unlabeled data.

KDD Cup99 datasets are standard datasets used to evaluate the algorithms for intrusion detection. The paper uses KDD Cup99 datasets to analyze the evaluation process of ACKID algorithm, confirming ROC curve as evaluation standard of ACKID

algorithm, analyzing the attribute features of network data, preprocessing data and analyzing results.

The experimental results demonstrate that ACKID algorithm which has the capability of generalizing unknown intrusion can detect unknown attacks, approve that ACKID algorithm using labeled data and constrains can improve the detection rates and low the false positive rates of the algorithm, and confirm that ACKID algorithm adopting active learning can acquire the most useful supervised information to detect unknown attacks.

Key word: intrusion detection; semi-supervised clustering; active learning; ROC curve

关于学位论文独立完成和内容创新的声明

本人向河南大学提出硕士学位申请。本人郑重声明：所呈交的学位论文是本人在导师的指导下独立完成的，对所研究的课题有新的见解。据我所知，除文中特别加以说明、标注和致谢的地方外，论文中不包括其他人已经发表或撰写过的研究成果，也不包括其他人为获得任何教育、科研机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位申请人（学位论文作者）签名：李雯睿

2007年6月19日

关于学位论文著作权使用授权书

本人经河南大学审核批准授予硕士学位。作为学位论文的作者，本人完全了解并同意河南大学有关保留、使用学位论文的要求，即河南大学有权向国家图书馆、科研信息机构、数据收集机构和本校图书馆等提供学位论文（纸质文本和电子文本）以供公众检索、查阅。本人授权河南大学出于宣扬、展览学校学术发展和进行学术交流等目的，可以采取影印、缩印、扫描和拷贝等复制手段保存、汇编学位论文（纸质文本和电子文本）。

（涉及保密内容的学位论文在解密后适用本授权书）

学位获得者（学位论文作者）签名：李雯睿

2007年6月19日

学位论文指导教师签名：

2007年6月19日

第 1 章 绪论

入侵检测是一种主动保护系统、免受黑客攻击的新型网络安全技术，提供对网络内部攻击、外部攻击和误操作的实时保护。本章重点阐述选题背景、入侵检测研究现状和本文所做的工作。

1.1 选题背景

二十年来，Internet 改变了人类的生活方式。然而，随着越来越多的人使用 Internet，计算机系统本身存在的漏洞逐渐暴露出来，使得恶意入侵有机可乘，像计算机病毒、窃取数据、黑客攻击等。

根据数据统计，99%的大公司都发生过大的入侵事件，如世界著名的商业网站 Yahoo、Buy、Amazon 等都曾被黑客入侵，造成巨大的经济损失，甚至连专门从事网络安全的 RSA 网站也受到黑客的攻击。

我国同样存在计算机安全问题，利用计算机网络进行的各类违法行为以每年 30%的速度递增。我国 85%的与 Internet 相连的网络管理中心都遭受过境内外黑客的攻击或侵入，其中银行、金融和证券等机构是黑客攻击的重点，这些金融机构因黑客犯罪案件而损失的金额已高达数亿元，同时针对其他行业的黑客犯罪也时有发生。

因此，计算机网络的安全问题已成为一个亟待解决的国际化问题。确保计算机系统、网络系统及整个信息基础设施的安全对于我国的经济建设和国防安全具有重要的意义。

Russel 与 Gangemi(1991)提出计算机安全是建立在系统的机密性、完整性及可用性的要求之上的[1]。机密性（Confidentiality）指只有授权用户才能获取信息；完整性（Integrity）指系统中的数据要保持一致性和正确性，不会被偶然或恶意修改；可用性（Availability）指当授权用户需要系统资源时，系统会一直提供资源，而不会拒绝授权用户的访问。

Kumar(1995)对计算机安全系统的定义是：能够保护数据、资源免于未经授权的访问、篡改数据和拒绝服务的系统[2]。在他所提出的框架中，数据机密性对商业和国家安全是很重要的；数据一致性允许医院维护病人的看病记录，为的是在关键时刻做出决策；数据可用性允许实时地在线交易。

随着计算机和的普及和网络的快速发展，未经授权的访问、篡改数据和拒绝服务攻击的现象日趋严重。日益增长的网络连接不仅方便了获取大量的数据，而且也提供了数据的访问路径。网络入侵者根据网络所提供的信息，在理解系统是如何工作后，利用系统的漏洞获取权限来完成他的目的。入侵者利用入侵模式掩饰他的活动轨迹，使得系统无法识别他就是入侵者。

为了使得网络安全人员尽量发现和察觉入侵行为和入侵企图，需要采取有效的措施来堵塞漏洞和修复系统。入侵检测技术是近 20 年来出现的一种主动保护系统、免受黑客攻击的新型网络安全技术。入侵检测被认为是防火墙之后的第二道安全闸门，它在不影响网络性能的情况下对网络进行监测，从而提供对内部攻击、外部攻击和误操作的实时保护。

入侵检测通过分析计算机网络或系统中的审计数据，从中发现网络或系统中是否存在入侵行为或入侵企图。入侵检测的本质是一个模式分类问题，就是将网络数据正确地分为正常类和异常类，因此各种机器学习技术越来越多的应用到入侵检测领域中[3]。

传统的入侵检测算法是基于监督学习的[4]，检测率较高，误报率较低，但是无法有效地检测到未知攻击，且要求训练集中的数据被正确地标记为正常或异常。然而，在网络环境中存在大量的数据，尤其是对未知攻击正确地标记几乎是不可能的。因此，非监督学习方法被应用到入侵检测中[5]，基于聚类的入侵检测算法不用对网络数据进行标记，就可以检测到未知的入侵行为，所以，基于非监督学习的入侵检测算法的检测率较高，然而，一旦有入侵行为被作为正常数据包含在训练集中，就会导致该类的入侵行为及其变种都被视为正常数据，所以误报率也较高。

在网络环境中，为了解决监督学习和非监督学习应用于入侵检测中的问题，本文引入了半监督学习。半监督学习技术[6]是机器学习领域中一个新的研究热点，它通过标记数据和未标记数据的联合概率分布来改进分类器的性能。在网络环境中，获得少量的标记是可行的。因此，本文提出基于半监督聚类的入侵检测算法 ACKID，利用少量的标记数据辅助聚类过程，采用主动学习策略查询标记数据与未标记数据的约束关系，也就是利用标记和未标记数据的联合分布来检测入侵行为。

通过查阅大量的文献，仅有一篇文献提出自标记启发式算法标记网络数据[7]，利用标记数据检测未知攻击。与文献[7]不同的是，本文采用主动学习策略查询标记数据的约束，可以充分利用标记数据的监督信息以指导聚类结构的形成。

1.2 研究现状

随着入侵检测技术的发展，目前已经出现很多入侵检测系统，不同的入侵检测系统具有不同的特征。根据不同的分类标准，入侵检测系统可分为不同的类别。对于入侵检测系统，要考虑的因素主要有：信息源、响应机制、分析算法、同步技术、控制策略等[8]。图 1-1 从不同角度对入侵检测系统进行分类。

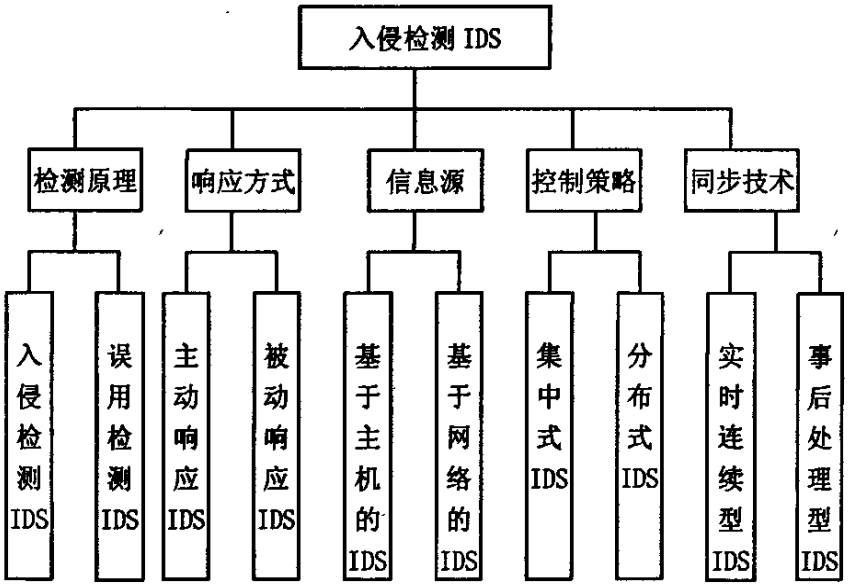

图 1-1 入侵检测系统的分类

根据入侵检测系统的检测原理的不同可分为：误用检测和入侵检测。误用检测首先抽取入侵特征，并构建入侵特征库，通过模式匹配的方式来检测已知类型攻击及其变种。入侵检测是对正常的网络和用户的行为构建模型，那么偏离模型的行为都被认为是异常。

根据信息源的不同，入侵检测系统可以分为：基于主机的入侵检测系统和基于网络的入侵检测系统。基于主机的 IDS 一般用于监视主机信息，其数据源通常包括操作系统的审计记录、系统日志、基于应用的审计信息、基于目标的对象信息等。基于网络的 IDS 主要用来实时监控某一网段，数据源是网络上所有分组采集的数据包。

IDS 对检测到的入侵行为可采取不同的反应方式：采取某种行动的 IDS 为主动响应，如断开网络连接、增加安全日志、杀死可疑进程等；若只是产生一些警告通知，则称为被动响应。根据系统监控到事件和对事件进行分析处理之间的间

隔，可将其分为实时的和事后处理两类。“实时”是指不间断持续运行的检测过程，表明 IDS 对入侵反应足够快；而系统在收集到信息之后要隔一段时间才对其进行处理，称为事后处理 IDS。

按照体系结构，IDS 可分为集中式和分布式，这与计算机系统的发展趋势是一致的。传统的 IDS 是集中式的，可能有多个分布于不同主机的审计程序收集到的数据交由一个中央入侵检测服务器进行分析处理；而分布式 IDS 由多个基于主机的 IDS 组成，这些 IDS 不分等级的执行自己的监控任务，各 IDS 之间通过消息或其他机制进行交互。

1.3　本文工作

当入侵模式和网络行为特征改变时，传统的入侵检测系统就无能为力了，用聚类算法可以检测到新的未知的入侵行为。然而，聚类算法处理、描述网络行为的特征有局限性，一旦攻击被当作正常数据包含在训练集中，就无法检测到这类攻击及其变种。因此，目前入侵检测算法无法准确检测未知攻击、误报率等问题。为了改进入侵检测算法，提高检测率，降低误报率，本文提出基于半监督聚类的入侵检测算法。

本文研究的主要内容是，设计实现基于半监督聚类的入侵检测算法，能够检测到已知攻击的变体和未知攻击，具有高的检测率和低的误报率。论文主要创新点和贡献如下：

(1) 系统地研究入侵检测系统的基本理论，分析了入侵检测模型的研究现状和当前存在的问题。讨论如何将聚类算法应用于入侵检测中，并分析基于聚类的入侵检测算法存在的问题。

(2) 针对基于划分聚类的入侵检测算法误报率高的问题，提出基于半监督聚类的入侵检测 ACKID（Active Constrained K-means Intrusion Detection)算法，分析网络中标记数据的监督信息是如何并入到 K-means 算法中的，并将主动学习策略应用于半监督聚类过程中。通过查询网络中未标记数据与标记数据的约束关系，采用 FarthestFirst 对未标记数据进行标记，这样可以检测出未知攻击，即使少量的标记和约束也能大大改进算法的性能。

(3) KDD Cup99 数据集是用于评估入侵检测算法的标准数据集，结合 KDD Cup99 数据集，分析 ACKID 入侵检测算法的评估过程。主要包括：确定 ROC 曲线为 ACKID 算法的评估指标，分析网络数据的属性特征，数据预处理中训练集过滤和对离散型数据做归一化处理，分析结果。

(4) 在实验中，通过分析比较 ACKID、K-means 和 SVM 三种算法，表明 ACKID 算法具备对未知入侵行为的推广能力，并且能发现未知入侵行为。证实利用标记数据和约束可以提高算法的检测率，降低误报率，并且表明采用主动学习策略能够获取最有用的监督信息。

第 2 章　入侵检测概述

随着网络环境越来越复杂，仅依赖防火墙技术不能阻止来自内部的攻击，而防火墙本身有各种漏洞和后门，且不能提供实时的入侵检测能力。为了增强计算机系统或网络系统的安全，需要采用更强大的主动策略和方案，其中一个有效的解决途径就是入侵检测。自从 Denning 提出入侵检测模型[9]，人们对入侵检测产生了极大的研究兴趣，提出许多关于入侵检测系统原型。本章通过对现有入侵检测模型及其实现技术的分类，说明入侵检测系统的主要特征和实现技术的优缺点，从整体上把握入侵检测的研究与发展。

2.1　入侵检测的定义

入侵行为是采用未经授权的行为，通过扫描系统漏洞，获得用户帐号，篡改用户文件，这样的行为就是入侵行为。根据 CIDF(Common Intrusion Detection Framework)标准[10]，IDS（Intrusion Detection System）就是对计算机网络或系统中的数据进行自动分析，从中发现网络或系统最终是否存在违反安全策略的行为或遭到攻击迹象的网络安全技术。

入侵检测系统应具有以下 6 个方面的特性：

(1) 监视、分析用户及系统活动，查找非法用户和合法用户的非授权操作。

(2) 检测系统配置的正确性和安全漏洞，并提示管理者修补漏洞。

(3) 识别已知攻击类型并向网络管理者报警。

(4) 分析异常行为模式。

(5) 操作系统的审计跟踪管理，并识别用户违反安全策略的行为。

(6) 评估重要系统和数据文件的完整性。

2.2　入侵检测的模型

如何从大量的网络数据中区分网络和用户的正常行为或入侵行为，这是建立入侵检测模型的关键。从 Denning 提出的用统计算法建立网络和用户正常行为的入侵检测模型至今，入侵检测技术已经经历了十多年的发展历程。下面从误用检测和入侵检测两方面概述目前入侵检测模型的研究状况。

2.2.1 误用检测

误用检测是利用特征库中良好定义的入侵模式，通过与审计数据的匹配来检测入侵。误用检测系统首先对标记的入侵行为模式进行编码，建立入侵模式库，然后分析网络数据，检测是否与入侵行为匹配。误用检测系统面临的问题是如何描述一个攻击及其变种的特征模型，而该模型又不能与非入侵行为匹配。

误用检测能够准确地检测入侵模式库中已有的入侵行为，有较低的误报率。但当出现新的攻击时，需要将新的攻击特征模式手工添加到入侵模式库中，这就意味着它需要不断升级、更新，才能保证系统检测能力的完备性。另外，因为误用检测系统对目标系统的依赖性很强，所以系统移植性不好；由于不能检测到未知的入侵行为，所以检测率低。

1. 专家系统

最初的误用检测系统是基于规则的专家系统[11]。它是将已知的入侵编码为一个规则集，其中规则具有 if-then 结构，条件部分为入侵特征，then 部分为系统防范措施。当规则的条件部分得到满足时，就执行 then 部分的动作。专家系统的建立依赖于知识库的完备性，知识库的完备性又取决于数据的完备性与实时性。

运用专家系统时可以把它看成一个自治的黑盒子，用户不需要干涉专家系统内部的推理过程。它的缺点主要有：提取入侵特征难度较大，处理海量数据存在效率问题，速度难于满足实时性要求；由于更改规则时必须考虑规则库中不同规则间的依赖性，所以维护规则库很困难。

2. 状态转换

状态转换是一种用于误用检测的分析算法，它使用系统状态和状态转换表达式来描述和检测已知的入侵行为。实现入侵状态转换主要有两种模型：状态转移分析和有色 Petri-Net。

状态转移分析使用状态转移图来表示已知入侵行为，状态图由系统状态变化的初始状态、中间状态以及结束状态的一个序列组成的[12]。系统状态通过系统属性或用户权限加以描述，状态转换是由系统事件驱动的，状态转换引擎保存着一份状态转移表，每一事件发生时则对此表进行刷新。

状态转移算法的优点是，状态转移规则比较容易创建和更新，并且转移规则容易理解；只需分析引发状态转移的事件，提供与数据独立的入侵行为描述。其缺点是状态声明和动作事件的列表需要手工编码，不能充分表达较为复杂的入侵模式，系统也难以检测入侵行为的简单变体，运行效率低下。

另一种采用状态转移技术的模型是有色 Petri 网（Colored Petri Nets，CPNs），该模型是由 Purdue 大学的 Kumar 和 Spafford 提出用来优化误用检测系统，具体的实现是 IDIOT 系统[13]。它使用 CPN 来表示和检测入侵模式，每一个入侵模式表示为一个 CPN。CPN 中令牌的颜色表示事件的属性，令牌的移动表示入侵过程的进展，当令牌从 CPN 的初始状态移动到结束状态时，则表示入侵过程成功完成。

基于着色 Petri 网状态转移的误用检测系统的优点是，检测效率高，能自动响应，另外模式匹配独立于数据格式，因而具有较好的移植性；再者入侵模式中事件的前后相关性和排列顺序可以直接体现出来。缺点是由于其检测基础是误用检测，所以不能检测出未知入侵。

2.2.2 异常检测

异常检测是根据系统行为和资源的使用状况是否偏离正常情况来判断入侵是否发生。它通过对数据的训练学习，从中发现正常行为模式，以定量方式描述可接受的行为特征，并由测试数据和正常行为模式的偏差捕获异常，以区分非正常的，潜在的入侵行为。偏差超出给定的阈值时，就会报警发现入侵行为。

异常检测与系统相对无关，不需要系统的先验知识，通用性较强，能够检测到一些未知的入侵，但不可能对系统的所有用户的正常行为建立统计模型。另外，每个用户行为是不断变化的，所以只要偏离正常行为模型的行为都会被认为异常。然而其中有一些并不是攻击，所以它的误报率很高，这是异常检测中需要解决的问题。

异常检测的正常行为模型的建立完全依赖于对训练数据集中正常数据的训练，所以要保证数据集的纯净性，对建立一个实用的异常检测系统这是很重要的。而实际上，由于行为模式的统计数据不断更新，收集一个纯净的数据集不太容易。入侵者可以通过恶意训练的方式，使得检测系统缓慢地更改行为模型，最初被认为是异常行为经过一段时间的训练就会被认为是正常的，也就是说，一旦有入侵行为被作为正常数据包含在训练集中，那么会导致该类的入侵行为及其变种都被视为正常数据，这是异常检测面临的困难之一。

Denning 入侵检测模型是一个通用入侵检测模型，它独立于具体系统、应用环境和攻击类型，为后来研究的检测模型和系统提供了借鉴价值。如 IDES/NIDES[14]都是在 Dennying 模型的基础上扩展的。

1. 统计分析

统计分析用于异常检测。它通过设置阈值的算法，将检测数据与已有的正常

行为模式加以比较，如果偏差超出阈值，则认为是入侵行为。常用的异常检测统计分析模型包括：

(1) 操作模型：该模型假设异常行为可通过观测结果超过一定的指标来判断，主要针对系统中的事件计算观测值。例如，在短时间内，多次失败登陆很有可能是口令尝试攻击。

(2) 均值与标准方差模型：针对数据均值和标准方差的特征提取算法。计算参数的方差，设定其置信区间，当测量值超过置信区间的范围时，可能是异常。该算法适用于事件计数、内部定时以及资源使用状况等统计范畴。

(3) 多元模型：操作模型的扩展，通过同时对两个或多个系统变量之间的相关性分析来检测异常。例如，同时考虑处理器时间和资源利用情况，或登录频率和会话消耗时间。

(4) 马尔柯夫过程模型：将每种不同类型的事件定义为系统状态，使用状态转移矩阵表示系统状态的变化。检测过程中使用正常情况下的状态转移矩阵，针对每一次系统的实际状态变化计算其发生的概率，如果该转移的概率较小则可能是异常事件。

(5) 时间序列分析：将事件计数与资源耗用根据时间排成序列，如果一个新事件在该时间发生的概率较低，则该事件可能是入侵。

统计算法的最大优点是它可以"学习"用户的使用习惯，从而具有较高检测率与可用性。但是它的"学习"能力也给入侵者机会，通过逐步"训练"使入侵行为符合正常行为的统计规律，从而越过入侵检测系统。

2. 模式预测

模式预测是一种基于异常检测的入侵检测算法，其前提假设是系统中事件的发生序列不是随机的而是遵循可辨识的模式，该算法的特点是考虑事件间的相互关系。Teng 和 Cheng 给出基于时间推理的模式预测算法，应用时间规则识别用户正常行为的特征[15]。规则通过归纳学习动态产生，并能实时在线调整使之具有较高的预测性、准确性和可信性。如果规则的大部分是准确的，而且可以成功预测到所观察到的数据，则规则具有较高的可信性（如果规则 1 比规则 2 成功地预测更多的事件，则规则 1 比规则 2 更具有预测性）。系统在自学习过程中，只将良好的用户行为（信息熵较低的规则）保留。

模式预测首先在入侵检测系统 TIM(Time-Based Inductive Machine)中实现,TIM 系统有效地实现了 Denning 所提出的 Markov 状态转移概率模型，它是

Polycenter 入侵检测产品的基础。TIM 系统和其他入侵检测系统的区别在于：TIM 从事件序列的角度，而不是单个的事件来检查系统或用户的行为模式是否偏离正常行为模式。模式预测的主要优点如下：

(1)能较好地处理用户变化多样的行为事件，并且具有很强的时序性；

(2)入侵检测时能够集中考察仅与安全相关的事件序列而非整个会话过程；

(3)该算法没有"弱化敏感"的问题，"弱化敏感"是一个与入侵检测相关的失败策略，攻击者随着时间推移逐渐改变自己的行为模式，直到系统将其当作正常行为来接受。"弱化敏感"的消除是因为把语义直接融入于检测规则当中。该算法的缺点是误报率高。这是异常检测有待解决的问题。

2.2.3 其他检测模型

下面的检测模型不能简单地归类为误用检测或是异常检测，它们提供更具有普遍意义的分析技术，在两类检测中都有应用。

1. 神经网络

基于神经网络的模型[16]首先从训练数据集得出正常行为模式，然后使用自学习技术来提取异常行为的特征。自学习可以在线或离线进行。神经网络建模分为两个阶段：训练阶段和检测阶段。训练集中的数据来自正常的网络数据，经数据信息预处理模块的处理后，作为神经网络的输入向量；然后，使用神经网络对输入向量处理，从中提取用户正常行为的模式特征，并创建用户的正常行为特征轮廓；最后，当网络接收输入的事件时，从中找出偏离特征轮廓的用户行为。

对于入侵检测，神经网络的优点是，不需要获取描述用户行为特征的特征集以及用户行为特征测度的统计分布，避开选择统计特征的困难问题；具备相当强的攻击模式分析能力，能够较好地处理带噪声数据，并且分析速度快，可用于实时分析。它的主要缺点是，不能解释或说明任何异常行为，这妨碍了用户获取入侵行为的详细信息，因而难以满足安全管理的需要；其次，神经网络的拓扑结构及权值的调整需要对大量的数据进行训练，因此建模代价高。

2. 遗传算法

遗传算法是基于自然选择和基因遗传学原理的搜索算法，在搜索过程中自动获取和积累有关搜索空间的知识，并控制搜索过程，从而得到最优解或次优解。遗传算法应用到入侵检测中[17]，是利用若干基因串序列来定义用于分析检测的指令组，识别正常或者异常行为的这些指令在初始训练阶段不断进化，提高分析能

力。

遗传算法的优点是自学习、自适应能力较强，能够通过基因串的不断复制和重组，产生性能良好的检测器；在学习的过程中，淘汰不良检测器。通过反复的学习和淘汰，系统不仅能够检测出已知入侵行为，并且能准确检测出其相应变体及未知的入侵。遗传算法的缺点是，由于网络行为的复杂性，很难用基因串完整表达检测向量，而过于复杂的基因串会使系统性能逐渐恶化。另外，适应函数的选取需要做多次试验，加以验证才能选取。

3．数据挖掘

随着网络规模的扩大，系统产生大量的数据。数据挖掘通过分析这些数据，试图从中找出偏离正常行为模式的异常行为，这是一个自动的过程，不需要人工分析和编码入侵模式。将数据挖掘应用于入侵检测中，具有代表性的成果是Columbia大学的 Wenke Lee 研究小组设计开发的入侵检测系统 MADAN ID[18]，主要技术是分类、关联分析和序列规则分析，其中序列规则分析的 frequent episodes 算法测试结果比较理想。

基于数据挖掘算法的入侵检测的优点是，系统能够自动发现未知的入侵行为，从而实现自学习、自适应功能。它的缺点是实时性问题，由于通过数据挖掘产生未知入侵模式是一个较为缓慢的过程，所以对数据的检测过程只能是进行事后分析。

4．免疫系统

计算机系统的保护机制与免疫系统非常相似，免疫系统中重要的能力是识别“自我/非自我”。根据免疫系统算法，New Mexico 大学的 Stephanie Forrest、Hofmeyr 和 Somayagi 提出将生物免疫机制引入计算机系统的安全保护框架中，利用程序运行过程中产生的系统调用短序列来定义正常行为模式，用来识别攻击行为[19]。系统调用短序列是系统调用序列中一定长度、相当稳定的片段，能够识别“自我”。Forrest 小组提出短序列匹配算法，用于计算机系统调用序列与正常序列模式的相似度，该算法只考虑系统调用在时间上的次序，并没有考虑调用的参数。对三种异常的行为模式(成功的入侵、不成功的入侵和错误条件)进行实验研究后，取得了令人满意的结果。

免疫算法主要特征在于分层保护、分布式检测，能够检测新的未知类型攻击行为。免疫算法的不足之处是，不涉及系统特权进程使用的攻击行为，往往无法检测到，如条件竞争、身份伪装、违背安全策略等攻击。

2.3 存在问题

可以从三个方面来评价入侵检测模系统的优劣：有效性、适应性和可扩展性。有效性是指 IDS 具有高的检测率和低误报率。适应性是指通过编码快速更新入侵模式，能够检测到已知攻击的变体和未知攻击。可扩展性指根据网络配置，系统能够并入检测模块，或者定制其他服务。

总的来说，目前入侵检测系统主要有以下几个缺陷：

(1) 缺乏有效性：专家设计安全系统需要手工编码规则和模式，由于网络系统的复杂性，专家知识通常是不完备和不精确的，因此造成了检测率低和误报率高的问题。

(2) 缺乏适应性：专家通常只是分析当前的入侵检测模型和系统漏洞。所以，基于专家知识的误用检测模型不能检测到新的未知攻击，而添加新的规则需要不断地更新模型，因此很难适应目前层出不穷的攻击手段。入侵检测虽然具备检测未知攻击，但需要为系统收集一个纯净的数据集，而在现实的网络环境中，这几乎是不可能的，因为训练集中一旦有入侵数据被认为是正常数据，那么该类入侵行为及其变种都被视为正常数据，因此导致较高的误报率。

(3) 缺乏可扩展性：滥用检测的入侵模式和入侵检测的统计测度是基于特定的环境和系统的，不具有通用性，因此，在新的网络环境中重用和定制已经建立起来的入侵检测系统难度很大。

由于当前网络环境复杂多变，审计记录日益庞大，攻击方式不断变化，需要一个更加系统化、自动化的算法来构造入侵检测模型。

2.4 本章小结

本章首先描述入侵检测的定义，然后从检测原理的不同，对当前入侵检测模型进行详细的阐述，并分析各自的优劣，这有助于从整体上把握入侵检测领域的研究和发展方向，为今后的研究指明方向。最后分析当前入侵检测模型的主要问题，这是亟待解决的问题。

本论文的主要研究目标是设计一种入侵检测算法，使其满足当前入侵检测发展的要求，即有效性、自适应性和可扩展性。其主要思想就是利用半监督聚类技术来设计入侵检测算法，其中涉及到的相关理论基础将在后面章节具体介绍。通过半监督聚类算法，从网络数据中获取相应的入侵检测模型。

第 3 章 基于聚类的入侵检测算法

“物以类聚，人以群分”，聚类是人类对事物内在规律的认识。聚类是按照相似度的大小，将事物划分成类，使类间的相似性尽可能小，类内的相似性尽可能大。

在网络环境中，根据网络数据的特征属性，可将网络数据分为正常行为和异常行为，即每一个网络数据可以被识别为正常或攻击类型。传统的入侵检测算法是基于监督学习算法的，需要足够的训练数据，以生成具有良好的泛化性能的检测模型。当入侵行为和网络数据的属性特征改变时，为所有的数据作标记是非常繁琐、耗时的过程，并且容易出错，基于监督学习的入侵检测算法就无能为力了。因此，可将非监督学习算法应用到入侵检测中[20]。聚类是一个聚类过程，将聚类技术应用于入侵检测中，克服了监督学习算法要求训练集中标记数据纯净的问题，并且可以检测到未知的入侵行为。

3.1 聚类概述

聚类的基本思想是在数据之间定义距离，距离代表数据之间的相似性度量，按相似程度的大小，将数据逐一归类，直到所有的数据都聚集完毕。

3.1.1 数据间的相似性度量

距离可以用来度量数据间的相似性[21]。设有n个数据的多元观测属性：

$$x_i=\left(x_{i1},x_{i2},\cdots,x_{ip}\right)^{\mathrm{T}},\quad i=1,2,\cdots,n. \tag{3-1}$$

这时，每个数据可看成p元空间的一个点，n个数据组成p元空间的n个点。设$d\left(x_i,x_j\right)$是数据x_i与x_j之间的距离，一般应满足以下要求：

(1) 相似性度量应为非负值，即$d\left(x_i,x_j\right)\geq 0$。

(2) 数据内之间的相似性度量应为最大。

(3) 相似性度量应满足对称性，即$d\left(x_i,x_j\right)=d\left(x_j,x_i\right)$。

下面介绍几种聚类分析中常用的距离：

1. 欧氏距离

在D维欧几里得空间中，

$$d(x_i,x_j)=[\sum_{k=1}^{p}(x_{ik}-x_{jk})^2]^{\frac{1}{2}} \tag{3-2}$$

令 $d_{ij}=d(x_i,x_j)$，$D=(d_{ij})_{p\times p}$ 形成一个距离矩阵

$$\begin{pmatrix} 0 & d_{12} & \cdots & d_{1n} \\ d_{21} & 0 & \cdots & \\ \vdots & \vdots & & \vdots \\ d_{n1} & d_{n2} & \cdots & 0 \end{pmatrix} \tag{3-3}$$

其中 $d_{ij}=d_{ji}$。

2. 绝对距离

$$d(x_i,x_j)=\sum_{k=1}^{p}|x_{ik}-x_{jk}| \tag{3-4}$$

3. Minkowski 距离

$$d(x_i,x_j)=[\sum_{k=1}^{p}(x_{ik}-x_{jk})^m]^{\frac{1}{m}} \tag{3-5}$$

其中 $m\geq 1$。当 $m=2,1$ 时分别是欧氏距离、绝对距离。

4. Chebyshev 距离

$$d(x_i,x_j)=\max_{1\leq k\leq p}|x_{ik}-x_{jk}| \tag{3-6}$$

Chebyshev 距离是 Minkowski 距离当 $m\to+\infty$ 时的极限。

以上距离与各属性指标的量纲有关，为消除量纲的影响，有时应先对数据进行标准化，然后用标准化数据计算距离。标准化数据

$$x_{ik}^{*}=\frac{x_{ik}-\bar{x}_k}{s_k} \tag{3-7}$$

$$s_k^2=\frac{1}{n-1}\sum_{i=1}^{n}(x_{ik}-\bar{x}_k) \tag{3-8}$$

$$\bar{x}_k=\frac{1}{n}\sum_{i=1}^{n}x_{ik},\quad i=1,2,\cdots,n;k=1,2,\cdots,p \tag{3-9}$$

3.1.2 类间距离

G_p 与 G_q 分别表示两个类，设它们分别包含 n_p，n_q 个数据。如果 G_r 是由类 G_p

与类G_q合并而成，则G_r包含$n_r=n_p+n_q$个数据。要解决的问题是G_p、G_q与其他类$G_k(k\neq p,q)$的距离，也就是计算G_r与$G_k(k\neq p,q)$的类间距离的递推公式。下面介绍主要使用的类间距离的递推公式。

1. 最短距离

$$D_{rk}=\min_{i\in G_r,j\in G_k} d_{ij}$$

$$=\min\{\min_{i\in G_p,j\in G_k} d_{ij},\min_{i\in G_q,j\in G_k} d_{ij}\} \tag{3-10}$$

2. 重心距离

$$D_{rk}^2=\frac{n_p}{n_r}D_{pk}^2+\frac{n_q}{n_r}D_{qk}^2-\frac{n_p}{n_r}\frac{n_q}{n_r}D_{pq}^2 \tag{3-11}$$

在一定条件下，以上介绍的类间距离的递推公式可以构成统一的形式。假定数据之间的距离都采用欧氏距离，其平方距离皆采用欧氏平方距离，即

$$d_{ij}^2=d^2(x_i,x_j)=(x_i-x_j)^{\mathrm{T}}(x_i-x_j)=\left\|x_i-x_j\right\|^2 \tag{3-12}$$

则类间距离递推公式有统一的形式：

$$D_{rk}^2=\alpha_p D_{kp}^2+\alpha_q D_{kq}^2+\beta D_{pq}^2+\gamma\left|D_{kp}^2-D_{kq}^2\right| \tag{3-13}$$

统一递推公式（α,β,λ为类间距离参数）体现了各种距离的共性，这对计算机统一编程提供了方便。

3.2 主要聚类算法的分类

1. 划分算法

划分聚类[22]是先将数据粗略地分类，然后按照某种原则进行修正，直到分类比较合理为止。它的基本思想是：对于给定n个数据的数据集，根据经验先设定类的个数为k，按照某种规定生成k个聚类中心，然后依次计算每个数据与中心的距离，选取距离为最小值的中心，将数据归入这个类中，得到一个聚类。

基于划分的聚类有两个代表性算法：K-means 算法[23]和 K-medoid[24]算法。在 K-means 算法中，每个类用每个类中数据的均值来表示。在 K-medoid 算法中，每个类用接近聚类中心的一个数据来表示。

划分算法的特点是，运行速度较快，能够有效地处理大规模数据集；但k必须事先确定，而且中心选取得好坏对聚类结构有很大影响。

2. 层次算法

层次聚类[25]的基本思想是：先将n个数据分成n类，然后规定类与类之间的距离，选择距离最小的一对合并成一个新类，此时为$n-1$个类，计算新类与其他类的距离，再把距离最近的两类合并，这样每次减少一个类，直到所有的子类都聚集成一类。

CURE(Cluster Using REprentatives)是一种自底向上的层次聚类算法[26]。在CURE 中，基于中心点的算法和所有点的距离计算算法都不适用于非球形或任意形状的聚类。所以 CURE 采用固定数目的点表示一个聚类，从而提高算法挖掘任意形状聚类的能力。对于低维数据的情况，CURE 算法的复杂度为$O(n^2)$，n为数据的数目。在处理大量数据时，算法必须基于抽样、划分等技术。

聚类过程构造一棵生成树，其中包含类的层次信息以及所有类内和类间的相似度，它生成层次化的嵌套类，且能够用于任何特征类型，精确度高。但由于每次合并时，需要全局比较所有类之间的相似度，并从中选择距离最小的两个类，因此运行速度较慢，不适合用于大规模数据集。

3. 基于模型的算法

基于模型的聚类[27]试图假定一个模型，寻找数据与给定模型的最佳拟合。这个假定是：目标数据集是由一系列的概率分布所决定的。那么，可以在空间中寻找诸如概率密度函数这样的模型来实现聚类。

COBWEB 是一种流行的简单增量概念聚类算法[28]，它的输入数据用分类属性－值对来描述。COBWEB 是以一个分类树的形式创建层次聚类的。COBWEB 有若干局限，首先，它基于每个属性上的概率分布是彼此独立的，然而这样的假设并不总是成立的。另外，聚类的概率分布表示使得更新和存储聚类结果相当昂贵。

4. 基于密度的算法

基于密度的聚类可以发现任意形状的聚类，可用来过滤孤立点数据。其主要思想[29]是：只要给定半径的邻域中数据点的数目超过某个阈值，也可以说，对于一个聚类中的每一个数据，在给定半径的邻域中包含的数据大于给定的阈值，然后对具有密度连接特性的数据进行聚类，于是一个聚类能被其中的任意一个核心数据所确定。

DBSCAN(Density-based Spatial Clustering of Application with Noise)是一种基于密度的聚类算法[30]，它根据一个密度阈值来控制类的增长。它能从含有噪声的

数据中发现任意形状的聚类，该算法的时间复杂性为 $O(n^2)$。它的特点是对数据输入顺序不敏感。

3.3 划分算法

划分算法一般是通过优化一个评价函数把数据集划分成 k 个类，主要有两种算法：K-means 算法和 K-medoid 法。

3.3.1 K-means 算法

K-means 算法以误差平方和准则为基础[31]。为了得到最优的结果，首先对数据集进行初始划分，数据均值作为簇的中心，经过反复迭代，逐次降低一个误差目标函数值，直到目标函数不再变化，得到分类的结果。K-means 算法描述如下：

算法：基于类中数据均值的划分的聚类算法。

输入：数据集 $X=\{x_i\}_{i=1}^{n}, x_i \in \mathrm{R}^d$，簇数目 k。

输出：满足平方误差准则最小的 k 个不相交簇 $\{X_h\}_{h=1}^{k}$。

算法：

1) 选取 k 个初始簇中心 $\{\mu_h^{(0)}\}_{h=1}^{k}$；

2) 重复到(3)到(5)，迭代直到目标函数收敛；

3) 计算各簇中数据均值，将每个数据 x 重新赋给最类似的类 $h^*, h^* = \arg\min_h \left\| x - \mu_h^{(t)} \right\|^2$；

4) 重新计算类的均值 $\mu_h^{(t+1)} \leftarrow \frac{1}{X_h^{(t+1)}} \sum_{x \in X_h^{(t+1)}} x$，赋给每个类；

5) $t \leftarrow (t+1)$。

K-means 算法的时间复杂度为 $O(tkn)$，其中 n 是数据的总数，k 是簇的个数，t 是算法循环的次数，通常有 $k,t \ll n$，所以算法效率很高。

它的特点是：计算中止于一个局部最优解；由于欧氏距离的局限性，它只能处理数值属性；对异常点敏感；聚类结果可能是不平衡的，有的簇甚至是空的；对数据输入顺序敏感。这个算法的主要问题是必须先确定 k 的值，并且种子选择的好坏对聚类结构有较大影响。

3.3.2 K-medoid 算法

K-medoid 没有采用类中数据均值作为参照点，而是选用类中最中心的位置的

数据(中心点)作为簇中心，从而根据各数据与各中心点之间的距离之和最小化原则，将剩余数据分配给最近的一个类，然后反复用非中心点数据代替中心点，以改进聚类质量。

与 K-means 算法一样，K-medoid 算法也是基于是聚类性能指标最小化原则的，不过使用数据作为中心点。K-medoid 算法描述如下：

算法：基于中心点的划分聚类算法。

输入：数据集 $X=\{x_i\}_{i=1}^{n}, x_i \in \mathrm{R}^d$，簇数目 k。

输出：k 个类，使得所有数据与最近中心点的相异度总和最小。

算法：

1) 任意选择 k 个数据作为初始的中心点；
2) 重复(3)到(6)，直到中心点不再变化；
3) 指派每个剩余的数据给离它最近的中心点所代表的类；
4) 随机地选择一个非中心点数据 O_{random}；
5) 计算用 O_{random} 代替 O_j 的总代价 S；
6) if $S<0$, then O_{random} 替换 O_j，形成新的 k 个中心点的集合。

与 K-means 算法相比，K-medoid 算法在处理噪声数据和孤立点时更具有鲁棒性，因为中心点不会像均值容易受噪声数据影响，但是 K-medoid 算法比 K-means 执行代价高。

3.4 孤立点分析

孤立点定义为数据集中存在这样一些数据，在某种度量下，它们与数据的一般模型不一致[32]。

孤立点可能是度量或执行的错误所导致。例如，一个人的身高为 4 米可能是程序对未记录的身高的缺省设置所产生的。另外，孤立点也可能是固有的数据变异性的结果。例如，一个公司的首席执行官的工资远远高于公司其他雇员的工资，成为一个孤立点。

许多数据挖掘算法试图使孤立点的影响最小化，或者排除它们。但是由于一个人的噪声可能是另一个人的信号，这可能导致重要的隐藏信息的丢失。换句话说，孤立点本身是非常重要的。例如，入侵检测中，孤立点可能预示着入侵行为。这样，孤立点检测和分析是一个有趣的数据挖掘任务，被称为孤立点挖掘。

孤立点挖掘可描述如下：给定一个 n 个数据，以及预期的孤立点的数目 k，

发现与剩余的数据相比是显著相异的、异常的和不一致的头 k 个数据。孤立点挖掘问题可以被看作两个子问题：

(1) 给定的数据集中定义什么样的数据可以被认为是不一致的；

(2) 找到一个有效的算法来挖掘这样的孤立点。

在入侵检测中，入侵行为可以看作是孤立点。对孤立点的假定是，正常行为远远多于异常行为（孤立点）。这样一个孤立点包含网络数据的异常行为的有用信息，通过检测孤立点可以发现入侵行为。检测孤立点的一般步骤是：首先为正常行为建立模型，然后用此模型检测异常，与模型不匹配的数据则被认为是孤立点。

目前基于孤立点分析的入侵检测算法可以分为三类：基于统计学算法，基于距离的算法和基于模型的算法。在本论文中，主要使用的是基于模型的算法为每个簇假定了一个模型，寻找数据对给定模型的最佳拟合，通过构建反映数据点空间分布的密度函数来定位簇，落在簇集合之外的点被认为是孤立点。

3.5 传统聚类算法在入侵检测中存在的问题

网络数据的属性特征具有数据量大，维数高的特点，所以传统聚类算法在对数据划分时存在以下问题：

(1) 描述网络数据的属性特征不仅包括数值类型，还包括字符型数据，如连接时间（duration）为数值类型，服务（service）是字符类型。K-means 算法和 K-medoids 处理数据只限于处理数值类型，不能处理字符类型。通常的解决算法是把字符型转换为数值型，也就是把多个值的字符属性转换成二元属性（用 1 表示属于，用 0 表示不属于）。在入侵检测中，网络数据的属性值具有成百上千个离散值，如果将这些离散值都转换成二元属性，算法复杂度会大大增加。

(2) 在网络数据集中，无法预知未知攻击类型，也就无法预知未知攻击类型的种类数目。不同类型的入侵属于不同的簇，K-means 和 K-medoids 算法都需要用户提供簇的数目 k，而 k 是凭经验确定的，所以如果簇数目未知，那么聚类结果的稳定性就较差。通常的解决算法是通过大量的实验选择相对适当的 k 值，而主观设定 k 值可能会降低聚类性能，破坏最后划分结果的可靠性。

(3) 网络数据集数量庞大，数据的属性维数也很高（仅 TCP 连接记录的属性有 27 个），所以要求聚类算法的执行速度足够快。

K-medoids 算法虽然不受噪音数据的影响，但是执行效率低。虽然 K-means 算法能够处理大数据集，但其算法本质是迭代的，执行过程中需要反复扫描整个数据集，不断改变簇中心和数据所属的簇，直到所有的簇中心不再改变。由此聚

类所需时间会随着数据集规模的增大而迅速膨胀，使计算量很大，所以对庞大的网络数据集进行划分也是很困难的。

3.6 本章小结

本章首先介绍了聚类中用到数学知识，然后对聚类算法进行分类，并说明各个算法的特点，接着引入可用于入侵检测的划分聚类算法，并结合孤立点分析算法，阐述传统聚类算法在入侵检测中存在的问题。

第 4 章 基于半监督聚类的入侵检测算法

传统的入侵检测算法是基于监督学习的，检测率高，误报率低，但无法检测到未知攻击，且要求训练集中的数据被正确地标记为正常或异常，然而在网络环境中存在着大量的数据，尤其是对未知的入侵行为正确地标记几乎是不可能的。而非监督学习方法应用到入侵检测中，检测率高，误报率也高，只能对未标记数据建模，不能利用标记数据的类别信息。

为了解决监督学习和非监督学习方法应用于入侵检测中的问题，引入了半监督学习。在网络环境中，获得少量的标记是可行的。标记数据在一定程度上反映了真实网络数据的分布情况，因此本文提出基于半监督聚类的入侵检测算法，利用少量的标记数据辅助聚类过程，也就是通过少量的标记检测未标记数据是否存在入侵行为。

4.1 半监督学习

半监督学习是机器学习领域中一个新的研究热点，它通过标记数据和未标记数据的联合概率分布来改进分类器的性能。半监督学习可分为半监督分类和半监督聚类。

半监督分类算法利用大量非标记数据辅助监督学习过程，改善分类结果[33]。这些算法包括 co-training(Blum & Mitchell,1998)[34]，transductive SVM(Joachims, 1999)[35]。在EM算法中，将未标记数据并入到训练过程中(Nigam, McCallum, Thrun, & Mitchell, 2000)[36]，用未标记数据学习好的测度(Hastie & Tibshirani, 1996)[37]等。

半监督聚类算法利用少量类别标记或一些数据的约束辅助聚类过程，得到研究者的广泛关注(Basu, Banerjee, & Mooney, 2002; Klein, Kamvar, & Manning,2002; Wagstaff, Cardie, Rogers, & Schroedl, 2001; Xing, Ng, Jordan, & Russell, 2003)[38,39,40,41]。如果监督数据是以类别标记的形式提供的，并且这些标记数据代表所有相关的类别，那么半监督聚类和半监督分类算法都可以用于数据分类。然而在许多领域中，相关的类别标记信息是不完全的，与半监督分类不同的是，半监督聚类可以用初始的标记数据作为类别聚集数据，并扩展、修改已有的类别标记以反映数据的其他规律。

4.2 半监督聚类

半监督聚类利用一些数据上的类别标记或约束来辅助非监督的聚类过程。由于许多情况下，数据的类别信息是不完全的，半监督聚类可用一些标记数据的类别信息或约束对数据进行聚集，扩展并且修改最初的类别标记。

半监督聚类算法分为两类：

(1) 基于标记数据和约束的算法。该算法利用标记数据和约束使聚类的结果满足目标函数。标记数据指明一个实例所属的类别，而约束指明两个实例应属于相同的簇 must-link 或属于不同的簇 cannot-link。具体的算法有：修改目标函数满足指定的约束[42]；在聚类过程中人为地添加约束；根据从标记数据中得出的近邻，初始化簇和推理簇的约束；基于图模型的约束聚类算法[43]；其他类型的约束[44]等。

(2) 基于距离的算法。该算法使用满足标记或约束的距离测度函数实现聚类过程。具体的算法有：基于凸的优化算法而得到的马氏距离[45]；基于最短路径算法而得到的欧氏距离；利用梯度下降而得到的 Jensen-Shannon 离散量[46]；用 EM 算法改进编辑距离（string-edit distance）[47]等。

本文主要讨论基于标记和约束的半监督聚类算法。

4.2.1 问题描述

K-means 算法只能对未标记数据建模，而不能利用标记数据的监督信息。下面解释在半监督聚类算法中，标记数据的监督信息是如何并入到 K-means 算法中的。首先用标记数据生成的种子簇初始化聚类算法，然后利用标记数据的约束，指导对未标记数据的聚类过程。选择恰当的种子可以避免局部最优，并且产生和标记相似的簇[48]。

给定一个数据集 X，K-means 算法可以产生关于 X 的 k 个划分 $\{X_h\}_{h=1}^{k}$，这时的目标函数是局部最优的。假设种子集 $S \subseteq X$，可以通过以下步骤得到种子集 S：

(1) 首先，对于每个种子 $x_i \in S$，用户提供 x_i 应属于的类别 X_h。

(2) 对应于数据集 X 的每个划分 X_h，X_h 中至少有一个种子 $x_i \in S$。

(3) 于是得到种子的不相交的 k 个划分 $\{S_h\}_{h=1}^{k}$，即种子簇。

半监聚类算法是通过种子簇指导 K-means 算法得到目标聚类。

基于标记半监督聚类算法有两种：Seeded K-means 算法和 Constrained K-means 算法。

4.2.2 Seeded K-means 算法

在 Seeded K-means 算法中，首先用种子簇初始化 k 个中心，这里中心不是像 K-means 算法中随机的 k 个均值，而是选择种子集 S 的第 h 个划分 S_h 的中心作为第 h 个簇的中心。Seeded K-means 算法描述如下：

算法：Seeded K-means 算法。

输入：数据集 $X=\{x_i\}_{i=1}^{n}, x_i \in \mathrm{R}^d$，簇数目 k，初始种子集 $S=\bigcup_{h=1}^{k} S_h$。

输出：数据集 X 的不相交的 k 个划分 $\{X_h\}_{h=1}^{k}$，使得 Seeded K-means 目标函数最优。

算法：

1) 种子簇初始化 k 个中心 $\mu_h^{(0)} \leftarrow \frac{1}{|S_h|}\sum_{x \in S_h} x, h=1,\cdots,k; t \leftarrow 0$;

2) 重复(3)到(5)，迭代直到目标函数收敛；

3) 计算各簇中数据均值，将每个数据 x 重新赋给最相似的类 h^*（集合 $X_{h^*}^{(t+1)}$），$h^* = \arg\min_{h \in \{1,\cdots,k\}} \left\| x - \mu_h^{(t)} \right\|^2$；

4) 重新计算类的均值 $\mu_h^{(t+1)} \leftarrow \frac{1}{X_h^{(t+1)}} \sum_{x \in X_h^{(t+1)}} x$，赋给每个类；

5) $t \leftarrow (t+1)$。

4.2.3 Constrained K-means 算法

在 Constrained K-means 算法执行过程中，种子簇的类别标记是保持不变的，只对非种子数据重新计算均值，也就是非种子数据的标记可能改变，适用于种子中无噪声或不需要改变种子标记的情况。而 Seeded K-means 算法适用于种子中有噪声的情况，由于在聚类过程中种子标记可以改变，所以在初始化 k 个种子中心后，就可以去除掉噪声种子。Constrained K-means 算法描述如下：

算法：Constrained K-means 算法。

输入：数据集 $X=\{x_i\}_{i=1}^{n}, x_i \in \mathrm{R}^d$，簇数目 k，初始种子集 $S=\bigcup_{h=1}^{k} S_h$。

输出：数据集 X 的不相交的 k 个划分 $\{X_h\}_{h=1}^{k}$，使 Constrained K-means 目标函数最优。

算法：

1) 种子簇初始化 k 个中心 $\mu_h^{(0)} \leftarrow \frac{1}{|S_h|}\sum_{x\in S_h} x, h=1,\cdots,k; t\leftarrow 0$;

2) 重复(3)到(5)，迭代直到目标函数收敛；

3) 计算各簇中数据均值，当 $x\in S$ 时，如果 $x\in S_h$ 将数据 x 赋给类 h（集合 $X_h^{(t+1)}$）；当 $x\notin S$，则将数据 x 赋给类 h^*（集合 $X_{h^*}^{(t+1)}$），$h^* = \arg\min_{h\in\{1,\cdots,k\}} \left\| x-\mu_h^{(t)} \right\|^2$；

4) 重新计算类的均值 $\mu_h^{(t+1)} \leftarrow \frac{1}{X_h^{(t+1)}}\sum_{x\in X_h^{(t+1)}} x$，赋给每个类；

5) $t\leftarrow (t+1)$。

4.3 半监督聚类与 EM 算法

4.3.1 模型假设

K-means 是 EM（Expectation Maximization）算法的特例，可以用高斯混合模型来求解。该模型基于以下假设：所有数据为独立同分布，且同类别数据服从高斯概率分布，则所有数据构成高斯混合分布。

高斯混合模型(Gaussian mixture model，描述混合密度分布的模型)[49]是统计学习中最经典、最完善的建模算法之一。该模型假设数据来自于不同数据源，每个数据源可用确定的数学形式进行建模。假设混合分量的数目为 k，混合模型可表达为：

$$P(x|\Theta) = \sum_{h=1}^{k} \pi_h p_h(x|\theta_h) \tag{4-1}$$

其中，$\pi_h = P(h|\theta)$ 为混合分量 h 的先验概率，$\sum_{h=1}^{k}\pi_h = 1$，$\Theta = (\pi_1,\cdots,\pi_k,\theta_1,\cdots\theta_k)$，$p_h$ 为各混合分量密度函数。学习的目标即为根据来自各分量加权混合分布的标记与未标记数据，估计该混合分布的产生，即估计混合分量的类条件概率密度的参数 θ_h 与混合参数 π_h。

高斯密度函数容易计算且适用于大多数场合，d 维高斯分布的概率密度函数表达式为：

$$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{4-2}$$

式中 μ 为均值，Σ 为协方差阵，所以将高斯分布的参数表示为：$\theta = (\mu,\ \Sigma)$。

4.3.2 EM 算法

当存在不完全数据时，EM 算法可以通过迭代方式解决模型的参数极大似然估计问题。“不完全数据”一般指两种情况:一种是由于观测过程本身的限制或错误,造成观测数据成为有错误的“不完全数据”；另一种是直接优化参数的似然函数十分困难,而引入额外的参数(隐含的或丢失的)后就比较容易优化,所以定义原始观测数据加上额外的参数组成“完全数据”。实际上，在机器学习及其相关领域中，后一种情况更为常见。

本文中把未标记数据的类别看作是不完全数据。假设 X 表示观测数据，Z 表示不完全数据，$Y=\{X,Z\}$ 为“完全数据”，Θ 表示未知参数[31]。未观测到的 Z 可被看作一个随机变量，它的概率分布依赖于未知参数 Θ 和已知数据 X。与此类似，Y 是一个随机变量，因为它是由随机变量 Z 来定义的。在本节后续部分，使用 h 表示参数 Θ 的假设值，而 h' 表示在 EM 算法的每次迭代中修改的假设。

EM 算法通过搜寻使 $E[\ln P(Y|h')]$ 最大的 h' 来寻找极大似然假设 h'。此期望值是在 Y 所遵循的概率分布上计算，此分布由未知参数 Θ 确定。$P(Y|h')$ 是给定假设 h' 下全部数据 Y 的似然度。其合理性在于要寻找一个 h' 使该量的某函数值最大化。其次，使该量的对数 $\ln P(Y|h')$ 最大化也就是使 $P(Y|h')$ 最大化。第三，引入期望值 $E[\ln P(Y|h')]$ 是因为 Y 本身也是一个随机变量。已知完全数据 Y 是观测到的 X 和未观测到的 Z 的合并，必须在未观测到的 Z 的可能值上取平均并以相应的概率为权值。换言之，要在随机变量 Y 遵循的概率分布上取期望值 $E[\ln P(Y|h')]$。该分布由已知的 X 加上 Z 服从的分布来确定。

一般不知道 Y 服从的概率分布，因为它是由待估参数 Θ 确定的。然而，EM 算法使用当前假设 h 代替实际参数 Θ，以估计 Y 的分布。现定义一个函数 $Q(h'|h)$，它将 $E[\ln P(Y|h')]$ 作为 h' 的一个函数给出，在 $\Theta=h$ 和完全数据 Y 的观测到的部分 X 的假定之下。

$$Q(h'|h)=E[\ln p(Y|h')|h,X] \tag{4-3}$$

将 Q 函数写成 $Q(h'|h)$ 是为了表示其定义是在当前假设 h 等于 Θ 的假定下。EM 算法通过下面两步的反复迭代直至收敛：

(1) E-步:使用当前假设 h 和观测到的数据 X 来估计 Y 上的概率分布以计算 $Q(h'|h)$。

$$Q(h'|h)\leftarrow E[\ln p(Y|h')|h,X] \tag{4-4}$$

(2) M-步：将假设 h 替换为使 Q 函数最大化的假设 h'：

$$h \leftarrow \arg\max_{h'} Q(h' \mid h) \tag{4-5}$$

当函数 Q 连续时，EM 算法收敛到似然函数 $P(Y \mid h')$ 的一个不动点。若此似然函数有单个的最大值时，EM 算法可以收敛到这个对 h' 的全局的极大似然估计。否则，它只保证收敛到一个局部最大值。

4.3.3 EM 框架下的半监督 K-means 算法

Seeded K-means 算法和 Constrained K-means 算法从本质上讲是在某种假设的情况下，EM 算法的 k 个高斯混合模型。在半监督 K-means 算法中,假设所有类别的全部数据服从的分布是一个“较大”的高斯混合分布,则高斯数就是类别数,而对每一个高斯分布而言,它又可以被分解为“较小”的 k 个高斯分布；而这 k 个“较小”的高斯分布混合形成的分布就是每个类别的高斯混合分布[50]。

K-means 算法的聚类过程是：根据 k 个高斯混合项的先验概率分布，选择一个高斯分布，接下来选择的数据都符合这个高斯分布。对于数据集 $X=\{x_i \in \mathrm{R}^d \mid i=1,\cdots,n\}$，不完全数据 $Z=\{1,\cdots,k\}$，假设高斯混合项的先验概率分布 $\pi_h=1/k, \forall h$，参数集 $\Theta=\{\mu_h\}_{h=1}^k$，且每个高斯分布有相同的协方差，可以得出：

$$\begin{aligned} & E_{Z|X,\Theta}\left[\log p(X,Z \mid \Theta)\right] \\ &= \sum_{h=1}^{k}\sum_{i=1}^{n} \log\left(\pi_h \cdot \frac{1}{(2\pi)^{d/2}} e^{-\|x_i-\mu_h\|^2}\right) p(z_h \mid x_i,\Theta) \\ &= -\sum_{h=1}^{k}\sum_{i=1}^{n} \|x_i-\mu_h\|^2 p(z_h \mid x_i,\Theta) + c \end{aligned} \tag{4-6}$$

其中 c 为常数。做进一步假设，并取代上公式：

$$p(z_h \mid x_i,\Theta) = \begin{cases} 1 & \text{当} h' = \arg\min_h \|x_i-\mu_h\|^2, \\ 0 & \text{其他情况,} \end{cases} \tag{4-7}$$

在上述假设成立时，可以得出完全数据的似然函数的极大值等于使 K-means 算法的目标函数取极小值。K-means 算法的目标函数写为：

$$\sum_{h=1}^{k}\sum_{i=1}^{n} \|x_i-\mu_h\|^2 p(z_h \mid x_i,\mu_h) \tag{4-8}$$

K-means 算法中要解决的不完全数据的问题是，不完全数据的类别的条件分布是 $p(z_h \mid x_i,\mu_h)$，理论上可以解决这个问题，而实际上这是无法计算的。在半监

督聚类中，用户提供一些数据类别的条件分布 $p(z_h|x_i,\mu_h)$ 和标记数据的约束。例如，两个数据 x_i 和 x_j 之间的约束为 must-link，那么 $p(z_h|x_i,\mu_h)$ 和 $p(z_h|x_j,\mu_h)$ 的分布是相同的。实际上，在 must-link 约束上的一组相关集合的传递闭包中的所有数据都服从同一分布，因此半监督聚类提供的监督信息是数据类别的条件分布 $p(z_h|x_i,\mu_h)$。

Constrained K-means 算法的目标函数为:

$$E_{Z|X,\Theta}\left[\log p(X,Z|\Theta)\right]=\sum_{h=1}^{k}\sum_{i=1}^{n}x_i^{T}\mu_h p(z_h|x_i,\Theta)+c \qquad (4\text{-}9)$$

使 Constrained K-means 算法的目标函数值最大，也就是在 EM 算法的 E 步时，使得完全数据的对数似然最大。

4.3.4 Seeded K-means 与 Constrained K-means 的比较

正如前面所讨论到的，不完全数据的问题就是要求出给定数据的标记的条件分布 $p(z_h|x_i,\mu_h)$。在半监督聚类中，用户已提供一些数据的标记，也就是需要确定了标记的条件分布 $p(z_h|x_i,\mu_h)$。

标准 K-means 算法没有提供任何监督信息，在 E-步时 k 均值是随机选取的，随后的 M-步时数据划分给最近的均值。而在半监督聚类中数据集中的每个数据 x_i 对应 k 个均值，并有相应的 k 个条件分布。在 E-步时将数据 x_i 随机地赋给一个簇，相当于从 k 个条件分布中选取数据相应的一个条件分布。

Seeded K-means 算法的监督信息是，给出种子数据 $x_i \in S$ 的条件分布 $p(z_h|x_i,\mu_h)$。算法只在 E-步时用到种子的条件分布，在 M-步时对于所有的数据包括种子数据，重新计算 $p(z_h|x_i,\mu_h)$，适合于种子中有噪声的情况。

E-步时，Constrained K-means 算法与 Seeded K-means 算法相同，给出种子数据 $x_i \in S$ 的条件分布 $p(z_h|x_i,\mu_h)$。与 Seeded K-means 算法不同的是，Constrained K-means 算法执行过程中，种子标记相应的条件分布 $p(z_h|x_i,\mu_h)$ 始终保持不变，而非种子数据的条件分布在 M-步需要迭代计算，适用于不需要改变种子标记的情况。

4.4 基于半监督聚类的入侵检测算法

目前已有的入侵检测基本上都是基于监督学习。但基于监督学习的入侵检测

算法要求手工标记训练数据，对网络数据除去噪声，这都是极其困难的。因此，算法只能检测到已知攻击，不能检测到新的未知攻击，算法可扩展性差。

为了解决上述问题，Eskin 提出非监督的入侵检测算法[51]，此算法的策略是将隐藏于正常数据中的入侵行为看作孤立点，根据得到的时序变化和特征的距离，判断网络行为是否异常。实验结果表明，与基于监督学习的入侵检测算法相比，这种算法在降低误报率和漏报率上并不突出，但其优势是明显的：对数据进行预分类不需要构建带标记的训练集，也不需要未知攻击的先验知识。

非监督学习是对未标记数据的分布进行建模，但无法利用已有的标记数据。在半监督学习中，假定数据 $x(x \in X)$ 与标记数据 $z(z \in Z = \{1,\cdots,k\})$ 的联合概率分布模型 $P(x,z)$，标记数据集 $D_l = \{(x_i,z_i)|i=1,\cdots,n\}$，其中 (x_i,z_i) 是独立采样于联合分布 $P(x,z)$，未标记数据集 $D_u = \{x_i|i=n+1,\cdots,N+m\}$，其中 x_i 独立采样于边缘分布 $P(x)=\sum_{z=1}^{k}(x,z)$。半监督学习的目的是根据标记与未标记数据 $D=(D_l,D_u)$ 预测类别标记 z。

网络数据集中存在大量的未标记数据，而根据 $P(z|x)$ 对 $P(x)$ 的数据进行标记而得到的训练数据很少，所以在网络环境中面临的问题是，学习目标与监督学习相同，但获得的数据更符合非监督学习。因此，本文提出半监督聚类的入侵检测（Active Constrained K-means Intrusion Detection，ACKID)算法，利用少量的标记数据及其约束辅助聚类过程，也就是基于标记与未标记数据联合分布检测异常。

4.4.1 未知攻击检测

网络中存在与已知类型数据具有不同分布的未知类型的数据，在聚类之后应被划分到不同的簇中。另外，由于标记数据很少，未知类型数据所在的簇也有可能未被标记。所以，在利用半监督聚类算法对数据集进行聚集并标记后，仍会存在未知类型的数据。因此，本文采用主动学习策略分析已有数据的标记情况和未标记数据的分布，通过对标记数据的查询来引导采样过程。

主动学习策略使用尽可能少的标记数据来提高分类器的性能，从而有效地减少人工标记数据的代价。目前主动学习有相当多的研究，其基本原理是用少量的标记数据建立初始分类器，每次学习过程中分类器可以主动在未标记的数据集中选择最有利于分类器性能的数据，并将这些数据以一定的方式加入到训练集中，来进一步训练分类器。Freund 等[52]从理论上证明，在一定假设条件下，采用合理的主动采样策略达到相同的学习效果(即相同泛化误差，generalization error)，所

需训练样本可减少到任意采样情况下的对数倍，而假设用户查询总是可以得到高的期望信息，主动学习的泛化误差将以所查询样本数量指数倍的速度下降[53,54,55,56]（文献标注中是主动学习策略的代表性论文）。

半监督聚类算法通过少量的标记数据和约束来辅助聚类过程。为了最大限度利用有限的标记数据，本文主动选择包含信息量最大的训练数据，而不是随机选择。

ACKID 算法是通过专家或用户查询选择网络数据的约束。主动学习策略的目标是为了在查询过程中得到数据尽可能多的信息，寻找某种途径选择对检测过程最有用的数据，同时利用所得数据尽快终止搜索过程。所谓最有用数据是指能够最大可能改善当前所得分类器性能的数据，以减少标记数据的数目，同时最小化查询学习所需的迭代次数以加快学习过程，以及提高预测的准确性。即使只有少量的标记数据，约束的主动查询策略也很大程度上提高了检测未知攻击类型的精确性。

4.4.2 ACKID 算法描述

在半监督聚类中，标记数据是有限的，为了充分利用其监督信息，主动查询数据的约束，而不是随机地选择约束，即使少量的约束也能大大改进算法的性能。

在网络环境中，ACKID 算法对标记数据和未标记数据进行分析，用户通过查询约束来引导采样过程。学习的初始阶段，多数标记数据的约束具有较高的有用性；随着迭代过程的进行，分类器的预测能力得到提高，此时仅有少量数据的约束就可以提升分类器的性能，通过对约束的合理选择，可以减少用户标记数据的工作量。

对于数据集 $X=\{x_i\}_{i=1}^{n}$，使用 K-means 算法可以得到 k 个不相交的 $\{X_h\}_{h=1}^{k}$，每个划分的中心点为 μ_h，好的初始中心点是 K-means 算法运行的关键。这样的原则对半监督聚类同样适用，半监督聚类中要求选择好的种子约束。

Constrained K-means 算法中引入约束 must-link 和 cannot-link。假定约束集 $C=(c_{12},c_{13},\cdots,c_{n-1,n})$，$c_{ij}$ 取值(-1, 0, 1)，$c_{ij}=1$ 表示 $(x_i,x_j)\in C_{ML}$，C_{ML} 是 must-link 约束集，对于 $(x_i,x_j)\in C_{ML}$ 是指 x_i 和 x_j 必须在同一个簇，另外属于 must-link 约束集的数据就不能作为其他簇的中心；$c_{ij}=-1$ 表示 $(x_i,x_j)\in C_{CL}$，C_{CL} 是 cannot-link 约束集，对于 $(x_i,x_j)\in C_{CL}$ 是指 x_i 和 x_j 不能在同一个簇中；$c_{ij}=0$ 表示 x_i 和 x_j 没有

约束关系。这里需要注意的是，C_{ML} 和 C_{CL} 中的数据是无序的，即 $(x_i,x_j)\in C_{CL}\Rightarrow(x_j,x_i)\in C_{CL}$，$C_{ML}$ 中的数据同样如此。

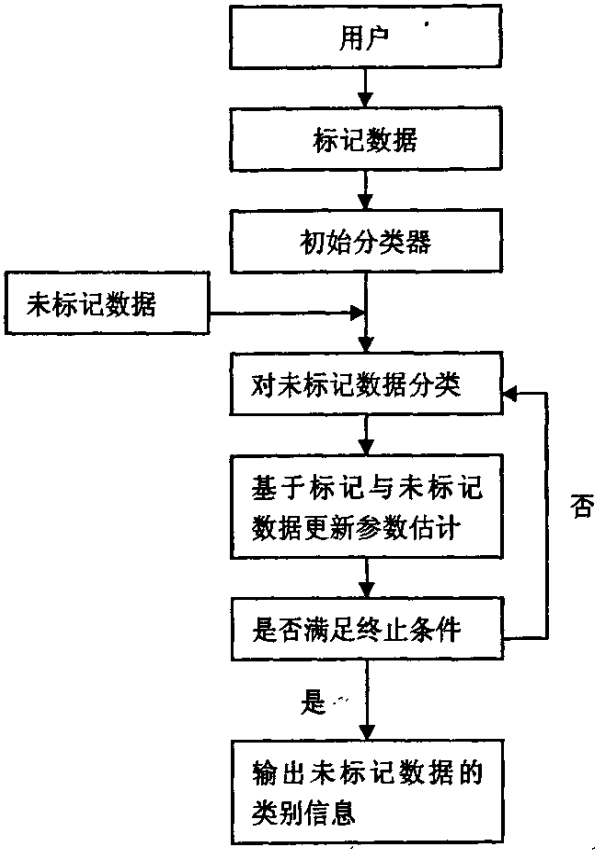

图 4-1　ACKID 算法

ACKID 算法通过有限的查询，可知数据 (x_i,x_j) 约束的类型(must-link 或 cannot-link)。算法目标是，获得约束的最少查询次数，得到比随机选择约束算法更好的聚类结构。从标记数据中初始化 k 个簇中心作为聚类的种子，种子选取的好坏对聚类结构有很大影响。图 4-1 给出了 ACKID 算法的流程。ACKID 算法描述如下：

算法：Active Constrained K-means Intrusion Detection 算法。

输入：网络数据集 $X=\{x_i\}_{i=1}^{n}, x_i\in \mathrm{R}^d$，must-link 约束集 C_{ML}，cannot-link 约束集 C_{CL}，簇数目 k，查询总数 Q。

输出：k 个不相交近邻集的数据数目按降序排列。网络数据 $x\in D_u$ 的数据类型（攻击或正常）。

算法：

1) 初始化簇：设置近邻集的数目为 λ；
2) 随机选择第一个标记数据 x，并加入到 N_1，$\lambda \leftarrow 1$；
3) While 允许查询 and $\lambda < k$

将离已有的近邻集 N 中标记数据最远的数据赋给 x

if while 查询 x 与已有的近邻集中的数据的关系，

如果 x 与所有的近邻中的标记数据是 cannot-link 约束关系

则 $\lambda \leftarrow \lambda+1$，一个包含 x 的新近邻集 N_λ 生成

else

将 x 赋给与它是 must-link 约束关系的近邻集。这时得到 λ 个不相交的近邻集 $\{N_p\}_{p=1}^{\lambda}$，其中 $\lambda \le k$；

4) 计算每个近邻集的中心 $\{\mu_h\}_{h=1}^{k}$；
5) while 允许查询

5a 随机选择不在已有近邻集中的未标记数据 x

5b 将距离 $\|x-\mu_h\|^2$ 按升序排列

5c for $h=1$ to k

按距离的排序查询 x 与每个近邻集的约束关系，直到获得 must-link 约束，并把 x 加入到这个近邻集中。

在 ACKID 算法中，对标记数据用“FarthestFirst”遍历策略，可以快速获得 λ 个不相交的非空近邻集，每个近邻集代表簇标记，即使每个近邻集中只有一个数据，近邻集的标记也可以正确地反映簇标记。

FarthestFirst 策略的基本思想是：寻找数据集中 λ 个彼此远离的数据集。首先，初始化时的第一个标记数据是随机选择的；再将离这个数据最远的数据加入到近邻集中，随后选择离这个近邻集最远的点（一个数据离近邻集标准的距离：$d(x,S)=\min_{x' \in S} d(x,x')$），考察它与已有的近邻集的约束关系(must-link 或 cannot-link)。FarthestFirst 是一个快速、简单，以 λ 均值为模型的近似的聚类器[57]。

在 ACKID 中，即使允许查询，也并不总能找到 k 个不相交近邻集。例如，未标记数据 x 是离已有近邻集最远的数据，通过查询 x 与已有近邻集中的数据的约束关系，判断 x 是否可以创建新的近邻集。如果 x 与所有近邻集中的数据是 cannot-link 约束，那么一个包含 x 的新的近邻集生成。如果 x 与某个近邻中的数据是 must-link 约束，则把 x 加入到这个近邻集中。算法迭代执行，直到所有的查询结束，或者是找到 k 个不相交近邻集。

这时 ACKID 算法运行到步骤 4 时，已找到k个不相交的近邻集，计算每个近邻集的中心$\{\mu_h\}_{h=1}^{k}$。如果查询还没有结束，那么其余查询用于形成聚类结构，此时每个近邻集中至少有一个数据。查询未标记数据x（不在任何已有近邻集中的数据）与每个近邻集$N=\{N_p\}_{p=1}^{k}$中的一个成员的约束关系，至多需要$k-1$次查询就可以找到x应属于的近邻集。也就是说，步骤 5 至多查询$k-1$次，就可以获得x正确的簇标记。

步骤 5 的主要思想是,当得到k个不相交的近邻集，且每个近邻集中至少有一个数据，对于所有的簇都可以获得其相应的一个标记，任何未标记数据x通过至多$k-1$次查询可以确定其应属于的近邻集，即可获得其正确的簇标记。查询过程是，从每个近邻集中轮流选取一个数据x_j，并且查询(x_i,x_j)的约束，直到获得 must-link 约束。获得(x_i,x_j)的约束关系，至多需要$k-1$次查询获得 must-link 约束，此时可以推出x_i与第k个近邻集是 must-link 约束。这里需要注意的是，未标记数据x离均值的距离按升序排列，首先查询x与距离最近的近邻集中数据的约束关系，这样在查询过程中，很快可以得到x应属于的正确的 must-link 近邻集。

对于聚类算法，适当的簇数目k值是不知道的，同样对于主动的半监督聚类算法，k值也是未知的。ACKID 算法中步骤 1-3，只要允许查询，就能很快的发现新的簇。然而对于网络数据集中的每个数据x，使用 FarthestFirst 策略总是可以找到与x有 must-link 约束关系的近邻集。因此找到所有的簇时，开始合并簇。如果k值已知，就进入步骤 4-5，根据潜在的数据分布，随机选择数据，这样能够产生适当的中心点，而在步骤 1-3 使用 FarthestFirst 策略选择的数据不具有这样的特性。

4.5　本章小结

本章主要介绍了本文设计的一种新的半监督聚类入侵检测算法，利用少量的标记数据辅助聚类过程。同时阐述了算法的基本理论：针对传统的基于监督学习的入侵检测算法在标记数据不足的情况下的无法检测未知攻击，而基于非监督学习的入侵检测算法无需标记样本，可以检测出未知攻击，但是它误报率高，本文提出 ACKID 算法采用主动学习策略，选择对检测过程最有用的标记数据，查询未标记数据与标记数据的约束关系，改进了以往聚类算法分析数据类型不足的缺点。

第 5 章 ACKID 入侵检测算法的评估

目前，需要建立一个相对客观、完备、准确、规范的测试评估体系，这将有助于用户选择适合自己的 IDS，同时帮助研究人员评估自己研发的 IDS 性能。因此，研究并选择一种好的 IDS 评估算法是目前研究的一个重要课题。本章引入测试评估 ACKID 算法性能的指标 ROC 曲线（Receiver Operating Characteristic Curves）；分析 KDD Cup99 入侵检测数据集中网络数据的属性特征，并对数据进行预处理；给出实验仿真环境，最后使用本文介绍的 ROC 曲线评估算法性能，实验结果证实了 ACKID 算法的有效性。

5.1 算法性能评估指标

在 IDS 出现的早期，用户常用检测率（Detection Rate，DR）和误报率（False Positive Rate，FPR）作为评价标准。检测率是指被正确检测的攻击数据占总的攻击数据的比例；误报率表示正常数据被检测为攻击数据占总的正常数据的比例。检测率提高，误报率也会提高；同样误报率降低，检测率也会降低。好的入侵检测算法要求尽可能多的检测出攻击，而被误检测为攻击行为的正常数据要尽量少。

在检测率和误报率之间寻找一个合适的折衷点，是机器学习中的一个重要问题。通过比较这些值，可以评价标记数据类别时的一些偏差。这在入侵检测问题上非常重要，因为现实中正常行为的数据和入侵数据的比一般是 100:1，如果只按照传统的正确率来判断入侵检测算法的优劣，那么可能就会出现一个总是把数据分类为正常的系统，它的准确率可能会达到 99%。这种入侵检测算法没有任何价值。因此在评价检测算法的性能时必须要综合考虑检测率和误报率[58]。这二者的关系取决于 IDS 阈值，通过改变阈值，可以获得检测率和误报率的 ROC 曲线来综合评价算法性能。

5.1.1 ROC 曲线分析

ROC 是受试者工作特征(Receiver Operating Characteristic)或相对工作特征(Relative Operating Characteristic)的缩写。ROC 分析技术不仅是一种通用图形化性能的算法，而且 ROC 曲线的独特属性使它在类别分布未知的领域和代价敏感学习中变得越来越重要。

ROC 分析五十年代起源于统计决策理论，用来说明分类器命中率和误报率之

间的关系，最早在第二次世界大战中应用于雷达信号观察能力的评价。1960 年，Lusted 首次提出了 ROC 分析可用于医学决策评价，六十年代中期大量成功地用于实验心理学和心理物理学研究。自从八十年代起该算法广泛用于医疗诊断性能的评价，如用于诊断放射学实验室医疗癌症的筛选和精神病的诊断,尤其为医疗影像诊断做出准确性的评价。目前，大量学者采用 ROC 曲线对入侵检测算法的性能进行评估[59,60,61,62,63]。

不同的入侵检测算法具备不同的 ROC 曲线。同一 ROC 曲线上不同的点代表同一检测算法当阈值不同时的检测率和误报率。要找到同时具备理想的检测率及误报率的适当的检测算法，就是找到在 ROC 曲线中具有理想斜率的点，使检测率与误报率之和最低，即准确性最高[64].

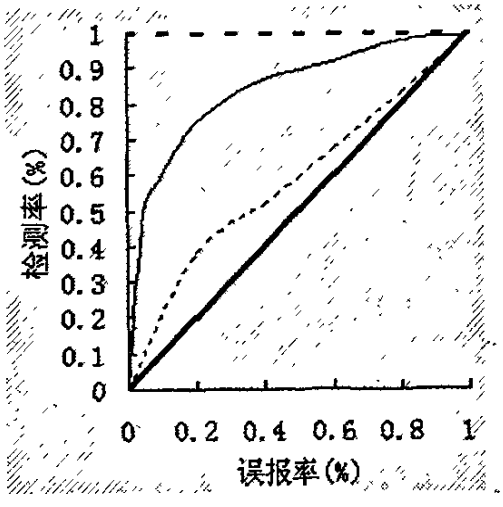

图 5-1　ROC 曲线图

如图 5-1 所示，X 轴表示误报率，Y 轴表示检测率，ROC 曲线是在一个二维坐标系中表示出检测率和误报率之间的关系的。入侵检测算法可能的工作状态可以是 ROC 曲线上的任何一点，以此来调整入侵检测算法的工作状态，使其处于所希望的最佳工作状态。粗实线是一条从原点到右上角的直线 DR=FPR，表示随机接受的检测，所以是毫无价值的。一般 ROC 曲线位于正方形的上三角。粗虚线表示理想情况下总能 100%的拒绝误报。另外两条细线由两个性能不同的入侵检测算法产生，检测率比误报率要高，性能比随机接受好，但比理想情况差。其中，实线对应的假设检验又比虚线的好，因为当误报率相同情况下，它的检测率更高。

5.1.2 ROC 曲线构建

由前面的分析可以看出，通过改变阈值获得多组检测率和误报率的值，可以绘制 ROC 曲线评估算法性能。计算所有可能的阈值，ROC 曲线能显示出检测率和误报率之间的变化关系。在入侵检测评估中，ROC 曲线上的各个工作点表示在给定的一个阈值下检测率和误报率的折衷。用直线连接各相邻两点构建 ROC 曲线。

为了用 ROC 曲线比较几个入侵检测算法，需要生成一个 ROC 曲线凸壳。首先将 ROC 空间中所有的点，包括点(0,0)和(1,1)，自左向右连成一条曲线。然后在保证曲线连接的前提下删除曲线中所有的凹陷处的点。因为在 ROC 曲线中的一个凹陷，表示该算法在某分布条件下并不是最优的分类器。或者说，从左向右连接的过程中，在连接一个新点之前要检查连接线段的斜率。ROC 曲线凸壳应该拥有一个单调递减的斜率。如果新的连接线段的斜率大于前面线段的斜率，那么放弃连接前面的 ROC 点。重复这个步骤，直到曲线只剩下一个点。这样就可保证斜率的单调并使 ROC 曲线凸起。所以在得到正常数据和入侵数据的分布比例和错误代价信息以后，在 ROC 曲线凸壳上的算法将拥有最佳性能。

通过对 ROC 曲线的分析，可以得到构建 ROC 曲线的难点。具体有以下几点：

(1) 比较不同入侵检测算法的性能

仅仅对比不同检测算法的检测率和误报率往往很难比较这些算法的性能优劣。例如，在 ROC 曲线的左上部分的点显示高的检测率和高的误报率，在实际应用中这些组合是没有价值的。

入侵行为与非入侵行为概率分布重叠的数量决定检测算法的识别能力，这种重叠也决定 ROC 曲线的形态及位置。如果入侵行为与非入侵行为的概率分布是相同的，也即它们完全重叠，检测率和误报率在任何阈值下都相等，这种检测没有识别能力也就没有价值，这种检测的 ROC 曲线是从点（0,0）到点（1,1）的一条直线。此入侵检测算法以 0.5 的概率随机猜测正常数据，那么将有一半的正常数据和入侵数据分类正确。在另一方面，一个理想的检测在分布上没有重叠，ROC 曲线有最佳作业点(即检测率为 100%，误报率为 0%)，相当于 ROC 曲线图的点（1,0），在这 ROC 曲线下的区域为 1.0(全部区域为 100%)。

ROC 曲线下的区域是检测的精确性的量度，常用于对检测算法的评估。比较 ROC 曲线下区域面积的大小，这种评估算法的主要优势是不依赖于诊断标准，这样可以消除检测率和误报率的评估对阈值的影响。

(2) 最佳阈值

ROC 曲线的另一个重要作用是找到检测的最佳阈值。在不实际的高阈值下，所有入侵都被认为是正常，导致检测率与误报率同时为 0，这与 ROC 曲线左下角的作业点（0,0）是一致的。降低阈值既增加误报率又增加检测率。对于可能最低的阈值，误报率和检测率均达到 100%，与 ROC 曲线右上角的工作点（1,1）相一致。ROC 曲线包含在所有可能的阈值上所有检测率和误报率的折衷。这为实践提供了找到最佳阈值的途径。对于一个特定的入侵检测算法来说，曲线中最接近左上角理想工作点（1,0）的转折点为最佳的工作点。该点的阈值即为该曲线表示的检侧算法的最佳阈值。

5.2 数据集

5.2.1 数据集描述

由于对入侵检测的研究空前高涨，出现了大量的研究模型和系统，因此对研究成果的测试和评估是一件很重要的事。1998 年，美国麻省理工学院林肯实验室在美国国防部高级研究项目局的资助下进行研究 DARPA 入侵检测评估计划，研究人员在林肯实验室内按照美国空军局域网结构建造一个实验网，模仿正常的网络行为，并模仿了一些网络攻击。

研究人员通过 TCPDump 软件监听了 9 周的网络流量数据。原始 DARPA 数据压缩后有 4GByte,其中前 7 周的数据作为训练集大约包含 5,000,000 个记录，后两周的数据为测试集大约包含 2,000,000 个记录。DARPA 入侵检测评估计划为入侵检测系统提供了可评估、可分析、可模拟、可应用的环境，极大促进了入侵检测系统的发展。

由于采集到的网络数据无法明显区分正常行为数据和入侵行为数据，Wenke Lee 提出一种数据特征属性抽取技术对网络数据进行处理[65]，形成 KDD Cup99 数据集。KDD Cup99 训练集中包含有大量的正常网络连接及 land 攻击、ping of death 攻击、猜测口令、端口扫描等 22 种 4 大类攻击，如表 5-1；测试集中则包含正常网络连接和 37 种攻击方式，其中 17 种相对训练数据是全新的攻击方式，如表 5-2。

表 5-1　KDD Cup99 训练集中包含的网络攻击类型

back，land，neptune，pod，smurf，teardrop，
guess_passwd，ftp_write，imap，multihop，phf，spy，warezclient，warezmaster，
buffer_overflow，loadmodule，perl，rootkit，
ipsweep，nmap，portsweep，satan

表 5-2 KDD Cup99 测试集中包含的新的未知网络攻击类型
snmpgetattack，named，xlock，xsnoop，sendmail，saint，apache2，udpstorm，xterm，mscan，processtable，ps，httptunnel，worm，mailbomb，sqlattack，snmpguess

入侵数据可分为 4 大类。四大类是：

(1) DOS(Denial of Service)：拒绝服务攻击。DoS 攻击是利用非法的服务请求占用过多的资源，从而使合法用户无法获得服务。如 back、land、neptune、pod、smurf、teardrop 等。

(2) R2L(Remote to Local)：远程权限获取。R2L 攻击指未经授权的本地用户试图访问远程没有用户账号的机器，利用系统漏洞来获取这台机器的存取权限。如 guess_passwd、ftp_write、imap、multihop、phf、spy、warezclient、warezmaster 等。

(3) U2R(User to Root)：各种权限提升。U2R 攻击指的是只有普通权限的用户通过系统的漏洞来获得系统的根权限的行为。如 buffer_overflow、loadmodule、perl、rootkit 等。

(4) PROBE 攻击：各种端口扫描和漏洞扫描。PROBE 攻击指攻击者通过扫描网络上的计算机，发现系统漏洞，如 ipsweep、nmap、portsweep、satan 等。

5.2.2 网络数据的属性特征分析

此处分析网络数据集属性特征，在此基础上提出本文实验过程中数据集预处理算法。网络中的每个数据表示两台主机间的一条连接，包括 TCP、UDP 和 ICMP 连接记录。每条连接数据有 41 个属性（包括连续和离散属性），其中 7 个属性是离散型变量，其余为连续性数字变量。可以分为以下四类属性集：

(1) 基本属性集，如连接的持续时间、协议、服务、发送字节数、接受字节数等。

(2) 内容属性集，即利用领域知识扩展的一些属性，如连接中 hot 标志的个数、连接失败登陆的次数、是否成功登陆等。

(3) 流量属性集，即基于时间的与网络流量相关的属性。这类属性又分为两种集合：一种为 Same Host 属性集，即在过去两秒内与当前连接具有相同目标主机的连接中，有关协议行为、服务等的一些统计信息，如此类连接的数目、此类连接中存在 SYN 错误的连接所占的百分比等；另外一种为 Same Service 属性集，即在过去两秒内与当前连接具有相同服务的连接中的一些统计信息，如此类连接中有 SYN 错误的连接所占的百分比、有 REJ 错误所占百分比等。

(4) 主机流量属性集，即基于主机的与网络流量相关的属性，这类属性是为了发现慢速扫描而设的属性，获取的办法是统计在过去的 100 个连接中的一些统计特性，如过去 100 个连接中与当前连接具有相同目的主机的连接数、与当前连接具有相同服务的连接所占的百分比等。

在现实网络环境中，可以对原始数据包进行处理，提取以上特征。约束的表式如表 5-3 和表 5-4 所示，这里并没有完全列出每种攻击类型的约束，而是举例说明了 4 大类攻击类型的典型的约束关系。由于相同的攻击具有相同的属性特征，所以相同的攻击数据之间的约束关系为 must-link，而不同的攻击数据之间的约束关系为 cannot-link。

表 5-3　DoS 攻击和 Probe 攻击

约束关系	含义
smurf: count≥5, srv_count≥5, service = ecr_i.	如果服务是 ICMP echo request，在过去的 2 秒内，向同一目的主机发出的连接数≥5，且具有相同服务的连接数≥5，就认为是 smurf 攻击（DOS 攻击的一种）。
satan: rerror_rate≥83%, diff_srv_rate≥87%.	过去 2 秒对同一目的主机发出连接，“rejected”错误连接率百分比≥83%，且不同服务的连接所占百分比≥87%，那么认为是 satan 攻击（PROBE 攻击的一种）。

表 5-4　R2L 攻击和 U2R 攻击

约束关系	含义
guess: failed_login≥4.	错误连接请求个数≥4，那么这个 telnet 连接就是 guess 猜口令攻击（R2L 攻击中的一种）。
overflow: hot≥3, compromised≥2, root_shell = 1.	hot 指示个数≥3，compromised 条件数目≥2，且启动了管理员 shell，那么认为是 buffer_overflow 攻击（U2R 攻击的一种）。

5.3 数据预处理

5.3.1 训练集过滤

KDD Cup99 数据集中，正常数据约占 20%，DOS 攻击约占选取数据集的 80%，并且大部分是 Neptune 攻击和 Smurf 攻击，而 R2L 和 U2R 类攻击发生概率比较小。

数据集中含有大量的攻击数据，为了模拟现实的网络环境，需要预处理数据集中的攻击数据，将攻击数据的比例控制在 1%到 1.5%，正常数据控制在 98.5%到 99%。

从 KDD Cup99 数据集 10%的数据中选取 10，658 条数据作为实验的数据集，其中 106 条攻击数据，10552 条正常数据，这满足聚类假设——正常数据远远多于入侵数据。实验中用到的数据集的类型分布如表 5-5。

表 5-5 数据集的类型分布

类型	数目	百分比（%）
DOS	49	0.4600
U2R	12	0.1126
R2L	23	0.2185
PROBE	22	0.2064
NORMAL	10552	99

观察数据集可以发现每种攻击的属性特征并不完全相同，但是有很多共性，并且需要采用不同的属性集检测不同的攻击才有效。表 5-6 为不同类型攻击需要的检测属性集。

表 5-6 检测不同攻击类型所需要的属性集

攻击类型	检测该类攻击的属性集
DOS	基本属性集+流量属性集
U2R	基本属性集+内容属性集
R2L	基本属性集+内容属性集
PROBE	基本属性集+流量属性集+主机流量属性

5.3.2 归一化处理

对于连续型属性特征，不同的属性有不同的度量标准，所以要考虑数据集作

归一化处理。对于给定的训练集，特征向量的均值 $avg\_vector[j]$ 与标准方差 $std\_vector[j]$，其中 $vector[j]$ 是特征向量的第 j 个属性。

$$avg\_vector[j]=\frac{1}{n}\sum_{i=1}^{n}instance_i[j] \tag{5-1}$$

$$std\_vector[j]=(\frac{1}{n-1}\sum_{i=1}^{n}(instance_i[j]-avg\_vector[j])^2)^{\frac{1}{2}} \tag{5-2}$$

$$new\_instance[j]=\frac{instance[j]-avg\_vector[j]}{std\_vector} \tag{5-3}$$

通过计算每个特征值与平均值之间的标准方差，数据集中的每个数据都可以转换为归一化空间的新数值。这是从数据集提取的统计信息的基础上，将连接数据从其初始空间转换到归一化空间中的新值。

用聚类方法进行入侵检测的一个重要前提是，同类型数据不管是正常数据或是攻击，都将在特定的度量空间上聚集在一起，根据网络数据的特点，本文采用前面介绍的标准的欧几里德距离度量来计算特征向量间的距离。

5.4 实验仿真

5.4.1 仿真环境

为了验证 ACKID 算法的有效性，本文进行了仿真实验，仿真环境如下：

(1) CPU: Pentium IV 1.8GHz;

(2) 内存: 512M RAM;

(3) 操作系统: Windows 2000 Professional 操作系统;

(4) 编程语言：JAVA 语言。

5.4.2 仿真结果和分析

本文的评估过程在独立主机上采用离线测试，如图 5-2 所示。

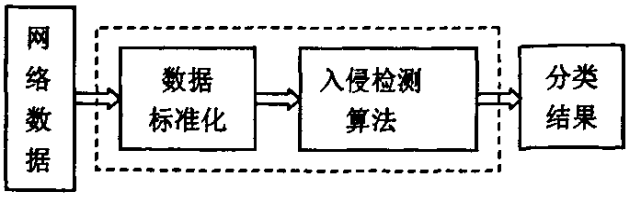

图 5-2　入侵检测算法评估过程

1．实验一

实验时，把数据集中 10，658 条数据划分为训练集和测试集。首先把 DOS 攻击与其他攻击分离开，并按比例混入正常数据作为训练集，其余的数据作为测试集，测试集包含相对于训练集中未知攻击类型（U2R，R2L，PROBE）。对网络数据进行聚类的过程中，找到数目最大的簇，并将其标记为正常类别，把这个簇中心记为 μ_0；然后，其余簇中心与 μ_0 的距离按升序排列，并且簇内的每个网络数据与 μ_0 的距离也按升序排列；接下来选择 $N_1 = \beta N$ 个网络数据（β 是正常数据比），并将它们标记为正常；最后，将其他数据都标记为入侵。通过改变参数 β，可由多组检测率和误报率画出 ROC 曲线。设置 ACKID 算法 k 值为 100。表 5-7 显示 β 取不同值时各组的实验结果，并由此得出 ACKID 算法的 ROC 曲线。

表 5-7 β 取不同值时各组的实验结果

正常数据比 β (%)	检测率(%)	误报率(%)
5	90	8.9
10	85	6.2
15	82	4.5
20	80	1.8
25	75	1.0
30	68	0.4

从表 5-7 可知，各组的检测率和误报率是随着正常数据比 β 的增加而降低的，这与估计的情况是一致的。因为生成的簇是按其中数据的多少排序，大体上正常数据首先标记，入侵数据稍后标记。β 取值越小，则表示入侵数据越多，则检测率增加，同时误报率也增加。理想情况下，聚类结果形成的每个簇只包含同种类型的数据，要么是正常数据，要么是入侵数据。而实际情况下，这是不可能的，各个簇都将不可避免地含有被错误划分的数据，即正常簇中包含入侵数据，入侵簇中也包含正常数据。评估 ACKID 算法所得的多组检测率和误报率可由 ROC 曲线表示，如图 5-3 所示。

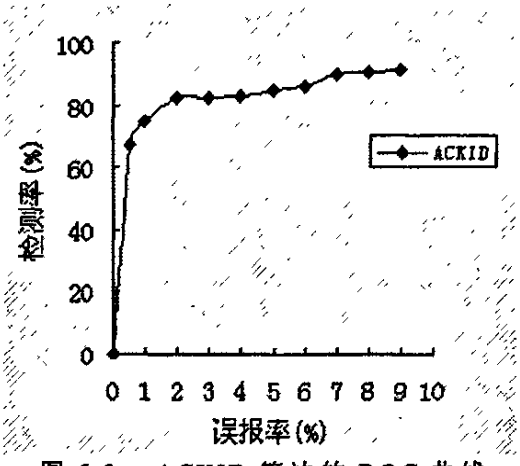

图 5-3　ACKID 算法的 ROC 曲线

2. 实验二

比较 ACKID 算法和文献[7]中 Combined 算法检测已知和未知攻击的性能。算法在训练集上得到数据模型，然后检验测试集中未知攻击的能力。设置 ACKID 算法 k 值为 100，表 5-8 给出了 ACKID 算法的检测结果，并与 Combined 算法检测结果进行了比较。

表 5-8　入侵检测结果

攻击类型	ACKID 算法		Combined 算法	
	检测率(%)	误报率(%)	检测率(%)	误报率(%)
已知攻击	76.27	5	75.76	5
未知攻击	93.06	10	92.53	10

由表 5-8 可以看出，当误报率相同时，ACKID 算法对已知攻击和未知攻击的检测率比 Combined 算法的略高，这是因为 ACKID 算法通过用户主动查询标记数据的约束，充分利用标记数据的监督信息，辅助聚类结构的形成，能够有效地检测出已知和未知攻击。

3. 实验三

比较 ACKID 算法与 K-means、SVM 算法检测攻击的性能，也就是比较了基于半监督学习的入侵检测算法与基于非监督学习和监督学习的入侵检测算法检测性能。K-means、SVM 是来自 weka 机器学习软件[66]算法。实验参数设置：ACKID 算法和 K-means 算法的 $k=100$，SVM 算法参数按缺省设置。ACKID、K-means、

SVM 三种算法的 ROC 曲线如图 5-4 所示。

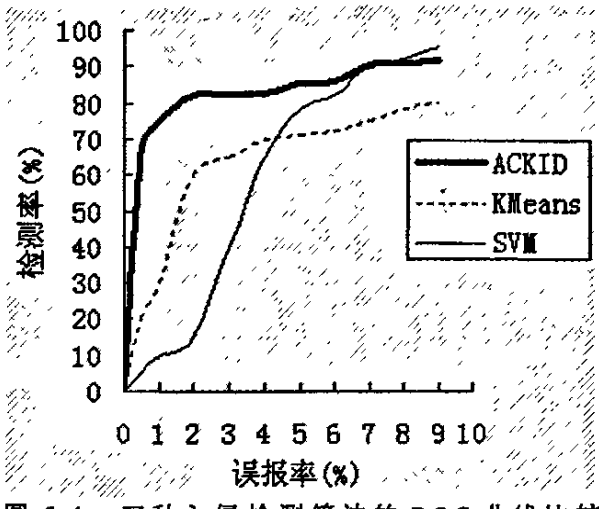

图 5-4 三种入侵检测算法的 ROC 曲线比较

比较 ACKID、K-means、SVM 三种算法的 ROC 曲线，可以看出，ACKID 算法的检测率明显高于非监督的 K-means 算法，因为 ACKID 算法采用了少量的标记数据来指导聚类过程，提高了检测未知攻击的精确性。当误报率小于 7%时，ACKID 算法比 SVM 具有更高的检测率，这是因为 ACKID 算法在半监督聚类的基础上，采用主动学习策略查询未知攻击的类别标记，解决了基于 SVM 的入侵检测算法检测未知攻击类别标记不足的问题，降低了误报率。

针对 4 类攻击的检测和算法总体检测性能，固定误报率为 4%和 8%，结果如表 5-9 所示。

表 5-9 入侵检测结果

		检测率（%）				
		DOS	U2R	R2L	PROBE	总体
误报率=4%	ACKID	76.18	55.24	25.71	98.62	84.15
	K-means	68.92	50.67	10.15	97.53	71.32
	SVM	93.26	0	35.63	72.35	68.63
误报率=8%	ACKID	76.35	96.72	92.16	99.08	90.57
	K-means	69.56	96.48	91.59	98.67	77.54
	SVM	94.02	0	30.58	73.47	92.46

由表 5-9 分析可知，SVM 对已知攻击 DOS 有较高的检测率，而对未知攻击

U2R 的检测率为 0，这表明 U2R 攻击的属性特征与 DOS 攻击的属性特征完全不同；对于 R2L 攻击类型，在误报率=4%时 SVM 算法的检测率更高，而在误报率等于 8%时，ACKID 算法的检测率更高。这是因为随着数据的增多，ACKID 算法对未知攻击有更好的学习能力。检测 PROBE 攻击，ACKID 算法比 SVM 检测率高，这是由于在给定的时间内 PROBE 攻击向同一主机发送大量的连接，相对来说变化较小，所以 ACKID 对 PROBE 攻击的检测率较高。

虽然 ACKID 算法解决了网络环境中难以完全标记样本问题，但还是无法达到有监督的检测算法的检测率。

5.5 本章小结

本章引入评估入侵检测算法性能指标 ROC 曲线，并且从理论和实践上介绍算法性能的评估，用 ROC 曲线从检测率和误报率两方面对 ACKID 算法、K-means、SVM 三种算法进行分析评估。实验结果表明，基于半监督聚类的入侵检测算法的有效性，能够满足对检测精度的要求。

结 语

论文总结

入侵检测是采用主动策略的网络安全技术，能够检测网络数据中的入侵行为。虽然传统的基于监督学习的入侵检测算法检测精度高，但无法检测出未知的入侵行为；而基于非监督学习的入侵检测算法能够检测到未知的入侵行为，但误报率高。因此，提高入侵检测算法的检测率，同时降低误报率，是当前需要迫切研究的一个课题。本论文的主要研究工作如下：

(1) 系统地研究入侵检测系统的基本理论，介绍入侵检测的定义，分析了入侵检测模型的研究现状和当前存在的问题。针对基于聚类的入侵检测算法误报率高的问题，提出基于半监督聚类的入侵检测算法 ACKID。

(2) 根据网络数据的特点，将主动学习策略应用于半监督聚类过程中。主动学习策略查询网络中未标记数据与标记数据的约束关系，采用 FarthestFirst 对未标记数据进行标记，即使少量的标记和约束也能大大改进算法的性能。

(3) 结合 KDD Cup99 数据集，分析基于半监督聚类的入侵检测算法的评估过程。主要包括：确定 ROC 曲线为基于半监督聚类的入侵检测算法的评估指标、分析网络数据的属性特征、训练集过滤和对连续型数据做归一化处理。

在实验中，比较 ACKID、K-means 和 SVM 三种算法，证实 ACKID 算法具备对已知入侵行为的推广能力，并且能发现未知的入侵行为。

工作展望

本文的工作是基于半监督聚类入侵检测算法的研究，由于能力及时间所限，不论在软件实现上还是在理论上都有很多值得进一步研究的地方，还需要继续研究的工作如下：

(1) 半监督聚类算法在识别未知的入侵行为具有很多优点，然而假设 ACKID 算法的约束是没有噪声的，今后要考虑如何处理网络数据的噪声约束。

(2) 采用主动学习策略的半监督聚类对网络数据集中的孤立点敏感。FarthestFirst 选择没有提供有关潜在的聚类结构信息的孤立点，这只会浪费主动学习阶段的查询。因此，将来可以采用修改的 FarthestFirst，通过选择远离网络数据空间中密集区域的数据点来处理孤立点的敏感性问题。

(3) 本文使用 KDD Cup 数据集测试算法的性能。今后要考虑的是，要将算法用于实际的网络环境中，怎样得到收集的网络数据包的更多的统计信息来有效地检测入侵行为。

(4) 本文是在离线的状态下对算法进行测试，要将算法用于实际的网络环境中，还要考虑算法的实时性和检测模型更新的问题，以能够实时地检测各种攻击，及早采取措施阻止攻击，减少危害。

虽然 ACKID 算法解决了网络环境中训练样本不足的问题，但还是无法达到有监督的检测算法的检测率。如何改进半监督聚类算法，将其更好地应用于入侵检测系统中，仍然是本文要研究的问题。

附录：KDD Cup99 数据集属性特征分类

在 KDD Cup99 数据集中，提供了 41 个属性特征，分为 4 类：基本属性集、内容属性集、流量属性集、主机流量属性集。附表如下：

附表 1 基本属性集

序 号	属 性 名	描 述 信 息	类 型
1	duration	连接时间的长短	continuous
2	protocol_type	协议类型，如tcp, udp等	discrete
3	service	目的端网络服务，如http, telnet等	discrete
4	flag	网络连接的状态	discrete
5	src_bytes	从源端传向目的端的字节数	continuous
6	dst_bytes	从目的端传向源端的字节数	continuous
7	land	源和目的主机端口相同否(同为1，不同为0)	discrete
8	wrong_fragment	错误分片的数目	continuous
9	urgent	紧急包的数目	continuous

“discrete”表示离散型数据，“continuous”表示连续型数据。以下同。

附表 2 内容属性集

序 号	属 性 名	描 述 信 息	类 型
10	hot	“hot”指示的个数	continuous
11	num_failed_login	错误连接请求的个数	continuous
12	logged_in	是否登录进系统(1/0)	discrete
13	num_compromised	“compromised”条件的数目	continuous
14	root_shell	是否启动了管理员shell(1/0)	discrete
15	su_attempted	是否尝试su root命令行(1/0)	discrete
16	num_root	利用管理员账户登录的次数	continuous
17	num_file_creations	创建文件操作的数目	continuous
18	num_shell	shell提示符的数目	continuous

附表 2　内容属性集

19	num_access_files	对存取控制文件操作的次数	continuous
20	num_outbound_cmds	在ftp中往外发送的命令数	continuous
21	is_hot_login	登录是否属于“hot”列表(1/0)	discrete
22	is_guest_login	是否以“guest”登录(1/0)	discrete

附表 3　流量属性集

序 号	属 性 名	描 述 信 息	类 型
23	count	两秒内对同一主机发出的连接数目	continuous
24	srv_count	两秒内与当前连接同样服务的连接数	continuous
25	serror_rate	带有“SYN”错误的连接所占百分比	continuous
26	srv_serror_rate	含有“SYN”错误的连接所占百分比	continuous
27	rerror_rate	带有“REJ”错误的连接所占百分比	continuous
28	src_rerror_rate	含有“REJ”错误的连接所占百分比	continuous
29	same_srv_rate	对于同一服务的连接所占百分比	continuous
30	diff_srv_rate	不同服务的连接所占百分比	continuous
31	srv_diff_host_rate	对不同主机的连接所占百分比	continuous

附表 4　主机流量属性集

序 号	属 性 名	描 述 信 息	类 型
32	dst_host_count	过去两秒对同一主机发出的连接数	continuous
33	dst_host_srv_count	两秒内与当前连接同一主机同一的服务的连接数	continuous
34	dst_host_same_srv_rate	到同一主机的服务相同的连接所占的百分比	continuous
35	dst_host_diff_srv_rate	到同一主机的服务不同的连接所占的百分比	continuous

附表 4 主机流量属性集

序 号	属 性 名	描 述 信 息	类 型
36	dst_host_same_src_port_rate	到同一主机的具有同一源端口号的连接的所占百分比	continuous
37	dst_host_srv_diff_host_rate	到同一主机的同一服务但来自不同主机的连接所占百分比	continuous
38	dst_host_serror_rate	带有“SYN”错误的连接所占百分比	continuous
39	dst_host_srv_serror_rate	带有“REJ”错误的连接所占百分比	continuous
40	dst_host_error_rate	含有“SYN”错误的连接所占百分比	continuous
41	dst_host_srv_rerror_rate	含有“REJ”错误的连接所占百分比	continuous

参考文献

[1] Deborah Russel G T, Gangemi Sr. Computer Security Basics [M]. O'Reilly & Associates, Inc., Sebastopol, CA, 1991: 12-14.

[2] Sandeep Kumar. Classification and detection of computer intrusions [D]. Purdue University, West Lafayette, USA, 1995 5-6.

[3] Barbara D, Couto J, Jajodia S, Wu N. ADAM:A Testbed for Exploring the Use of Data Mining in Intrusion Detection [C]. In SIGMOD Record, 2001,30(4):15-24.

[4] Horeis T. Intrusion Detection with Neural Networks Combination of Self-Organizing Maps and Radial Basis Function Networks for Human Expert Integration [DB/EL]. http://ieee-nns.org/edu/research/reports-2003/hortis.pdf.2003.

[5] Eskin E, Arnold A, Prerau M,et al. A geometric framework for unsupervised anomaly detection:detecting intrusions in unlabeled data [C]. Applications of Data Mining in Computer Security.Kluwer Academic Publisher, Boston, 2002:77-102.

[6] Zhu X.J.. Semi-Supervised Learning Literature Survey [R]. Computer Sciences TR 1530. University of Wisconsin–Madison Last modified on December 9, 2006.

[7] Shi Zhong, Taghi M. Khoshgoftaar, and Naeem Seliya. Clustering-based Network Intrusion Detection. In International Journal of Reliability, Quality, and Safety Engineering (IJRQSE), 2005.

[8] 罗守山.入侵检测 [M].北京：北京邮电大学出版社，2004：24-25.

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

[10] CIDF Working Group. Communication in the Common Intrusion Detection Framework[EB/OL]. http://www.isi.edu/gost/cidf/drafts/communication.txt. 1998.

[11] Sebring M M, Sheelhouse E, Hanna M E. Expert systems in intrusion detection[C]. In Proceedings of the 11th National Computer Security Conference, 1988:74-81.

[12] Koral I, Richard A K, Phillip A P. State transition analysis:a rule-based intrusion detection approach[J]. IEEE Transactions on Software Engineering,1995,21(3):181-199.

[13] Kumar S, Spafford E. A Pattern Matching Model for Misuse Intrusion Detection [C]. Proceedings of the 17th National Security Conference, 1994:11-21.

[14] Javitz H S, Valdes A. The SRI IDES Statistical Anomaly Detector. Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, California, USA, 1991: 316-326.

[15] TENG H S, CHEN K, LU S C. Adaptive real-time anomaly detection using inductively generated sequential patterns[C]. Proceedings of the IEEE Symposium on Research in Security and Privacy. Oakland CA, 1990,12(4):278-284.

[16] Mukkamala S, Janoski G, Sung A. Intrusion detection using neural networks and support vector machines[C]. Proceedings of the International Joint Conference on Neural Networks. New Jersey: IEEE Computer Society Press, 2002.1702-1707.

[17] 张凤斌，杨永田，江子扬.遗传算法在基于网络异常的入侵检测中的应用[J].电子学报,2004, 32(5):875-877.

[18] Wenke Lee , Salvatore Stolfo. Data mining approaches for intrusion detection[C].In Proceedings of the 7th USENIX Security Symposium, San Antonio, 1998.

[19] Hofmeyr, Forrest. Architecture for an artificial immune system [J]. Evolutionary Computation, 2000, 8(4): 443-473.

[20] Portnoy L, Eskin E,Stolfo S J.Intrusion Detection with Unlabeled Data Using Clustering. Philadelphia,PA:In Proceedings of ACM CSS Workshop on Data Mining Applied to Security(DMSA,2001),2001.

[21] 范金城,梅长林. 数据分析 [M].北京：科学出版社，2002:57-64.

[22] Bock H H. Probability models and hypotheses testing in partitioning cluster analysis [J]. In Clustering and Classification. Riverside, CA: World Scientific, 1996: 377–453.

[23] Bin Zhang. Generalized K-harmonic means-boosting in. unsupervised learning [R]. Hewlett-Packard Laboratories, 2000.

[24] ESTER M, KRIEGEL H-P, XU X. A database interface for clustering in large spatial databases[C]. In Proceedings of the 1st ACM SIGKDD, Montreal, Canada, 1995: 94-99.

[25] KAUFMAN L, ROUSSEEUW P. Finding Groups in Data: An Introduction to Cluster Analysis [M]. John Wiley and Sons, New York, NY, 1990:12-17.

[26] Guha S, Rastogi R, Shim K. CURE: an Efficient Clustering Algorithm for Large Databases[C]. Proceedings of the ACM SIGMOD InternationalConference on Management of Data. USA:ACM Press, 1998:73-84.

[27] CADEZ I, SMYTH P, MANNILA H. Probabilistic modeling of transactional data with applications to profiling, Visualization, and Prediction[C]. In Proceedings of the 7th ACM SIGKDD, San Francisco, CA, 2001:37-46.

[28] Fisher D. Knowledge Acquisition Via Incremental Conceptual Clustering[J]. Machine Learning,1987,2(2):461-465.

[29] Han J, Kamber M , Tung A K H. Spatial clustering methods in data mining: A survey. Geographic Data Mining and Knowledge Discovery, London: Taylor and Francis, 2001:188-217.

[30] Ester M, Kriegel H P, Sander J, Xu X. A density based algorithm for discovering clusters in large spatial databases with noise [C]. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining. Portland: AAAI Press, 1996: 226-231.

[31] Mitchell TM.机器学习[M].曾华军,张银奎译.北京:机械工业出版社,2003: 118-122, 136-140

[32] Han J, Kamber M, Fan Ming, Meng Xiao-Feng et al. Translated. Data Mining: Concepts and Techniques[M]. Beijing: China Mechine Press, 2001 (in Chinese)(Han J,Kamber M.范明,孟小峰等译.数据挖掘:概念与技术.北京:机械工业出版社,2001:232-235.

[33] Seeger M. Learning with labeled and unlabeled data[R]. Institute for ANC, Edinburgh, UK. 2000.http://www.dai.ed.ac.uk/~seeger/papers.html.

[34] Blum A, Mitchell T. Combining labeled and unlabeled data with co-training[C]. In Proceedings of the 11th Annual Conference on Computational Learning Theory, Madison, WI,

1998: 92–100.

[35] Joachims T. Transductive inference for text classification using support vector machines[C]. In Proceedings of the Sixteenth International Conference on Machine Learning (ICML-99), Bled, Slovenia, 1999: 200–209.

[36] Nigam K, McCallum A K, Thrun S, Mitchell T. Text classification from labeled and unlabeled documents using EM [J]. Machine Learning, 2000,39: 103-134.

[37] Hastie T, Tibshirani R. Discriminant adaptive nearest-neighbor classification[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996,18(6): 607–617.

[38] Basu S, Banerjee A, Mooney R J. Semi-supervised clustering by seeding[C]. In Proceedings of 19th International Conference on Machine Learning (ICML-2002), Sydney, Australia 2002:19-26.

[39] Klein D, Kamvar S D, Manning C. From instance-level constraints to space level constraints: Making the most of prior knowledge in data clustering[C]. In Proceedings of the The Nineteenth International Conference on Machine Learning (ICML-2002), Sydney, Australia,2002: 307–314.

[40] Wagstaff K, Cardie C, Rogers S, Schroedl S. Constrained K-Means clustering with background knowledge[C]. In Proceedings of 18th International Conference on Machine Learning (ICML-2001), 2001: 577-584.

[41] Xing E P, Ng A Y, Jordan M I, Russell S. Distance metric learning, with application to clustering with side-information[J]. In Advances in Neural Information Processing Systems, Cambridge, MA. MIT Press,2003,15: 505–512.

[42] Demiriz A, Bennett K P, Embrechts M J. Semi-supervised clustering using genetic algorithms[J]. In Artificial Neural Networks in Engineering (ANNIE-99), 1999: 809–814.

[43] Hiu M, Law C, Topchy A, Jain A K. Model-based clustering with probabilistic constraints [C]. In Proceedings of the 2005 SIAM International Conference on Data Mining (SDM-05),2005.

[44] Davidson I, Ravi S. Clustering with constraints: Feasibility issues and the k-means algorithm [C]. In Proceedings of the 2005 SIAM International Conference on Data Mining (SDM-05),2005.

[45] Bar-Hillel A, Hertz T, Shental N, Weinshall D. Learning distance functions using equivalence relations [C]. In Proceedings of 20th International Conference on Machine Learning (ICML-2003),2003: 11–18.

[46] Cohn D, Caruana R, McCallum A. Semi-supervised clustering with user Feedback [R].TR2003-1892, Cornell University, 2003.

[47] Bilenko M, Mooney R J. Adaptive duplicate detection using learnable string similarity measures [C]. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2003), Washington, D.C, 2003: 39–48.

[48] Basu S, Bilenko M, Mooney R J. (2004). A probabilistic framework for semi-supervised clustering [C]. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Seattle, WA, 2004: 59–68.

[49] 房祥忠,陈家鼎. EM算法在假设检验中的应用[J].中国科学(A 辑),2003,33(2):180～184.

[50] 孙广玲，唐降龙. 基于分层高斯混合模型的半监督学习算法[J]。计算机研究与发展，2004，

41(1): 156-161.

[51] ESKIN E, ARNOLD A, PRERAU M, et al. A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled data [M]. Kluwer: Data Mining for Security Applications, 2002: 77-102.

[52] Freund Y, Seung H, Shamir E, et al. Selective sampling using the query by committee algorithm[J]. Machine Learning, 1997, 28: 133-168.

[53] Lewis D, Gale W. A sequential algorithm for training text classifiers[C]. Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Ireland, 1994:3-12.

[54] Argamon Engelson S, Dagan I. Commitee-based sample selection for probabilistic classifiers[J]. Journal of Artificial Intelligence Research, 1999, 11:335–360.

[55] McCallum A, Nigam K. Employing EM and pool-based active learning for text classification [C]. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML-98) Madison, WI. Morgan Kaufmann,1998: 398-406.

[56] Basu S, Banerjee A, Mooney R J. Active semi-supervision for pairwise constrained clustering[C]. In Proceedings of the 2004 SIAM International Conference on Data Mining (SDM-04),2004 :333–344.

[57] Hochbaum D, Shmoys D.A best possible heuristic for the k-center problem. Mathematics of Operations Research, 1985,10(2):180–184.

[58] 罗敏，王丽娜，张焕国．基于非监督聚类的入侵检测方法[J]. 电子学报,2003,31(11):1713-1716.

[59] Liu Z, Bridges S, Vaughn R . Classification of Anomalous Traces of Privileged and Parallel Programs by Neural Networks [C]. Proceedings of FUZZIEEE 2003,St. Louis Missouri: IEEE press,2003: 1225-1230.

[60] Zhong S, Khoshgoftaar T, Seliya N. Evaluating clustering techniques for network intrusion detection[C]. Proceedings of 10th ISSAT lat. Conf. on Reliability and Quality Design. Las Vegas: FL press, 2004: 98-105.

[61] Mahoney M. Learning rules for anomaly detection of hostile network traffic [C]. Proceedings of ICDM 2003, Washington D.C.: IEEE press, 2003: 28-37.

[62] Lippmann R, Cunningham R. Improving Intrusion Detection performance using key words election and neural networks[C].Proceedings of Recent Advances in Intrusion Detection 99 Conference. New York: CA press, 1999: 34-47.

[63] Gonzalez F,Dasgupta D. Neuro-immune and self-organizing map approaches to anomaly detection: a comparison [C]. Proceedings of the 1st International Conference on Artificial Immune Systems. Canterbury: IEEE press, 2003: 203-211.

[64] 姚羽.基于神经网络的入侵检测方法研究[D]。东北大学,2004:116-118.

[65] Wenke Lee,Sal Stolfo,Kui Mok. A Data Mining Framework for Building Intrusion Detection Models [C]. In Proceedings of the 1999 IEEE Symposium on Security and Privacy, Oakland, CA, 1999.

[66] Witten I H, Frank E. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations[M] Morgan Kaufmann, San Francisco，2005.

致　谢

在此论文完稿之际，首先要向三年来给予我悉心指导、精心培养的导师申石磊教授致以最衷心的感谢！在三年的学习生活中，导师以其严谨求实的治学态度、精深的学术造诣和忘我的工作精神，给我留下了深刻的印象，这必将对我以后的学习、生活和工作产生积极而深远的影响。

感谢计算机学院领导提供良好而宽松的学术氛围和研究条件，同时感谢魏丹老师在生活上给予了无微不至的关怀和照顾。

感谢三年来相处愉快的母君臣、许宏云、刘骥宇、刘秀磊、于素萍、刘明理、马晓同学，感谢他们给予的热情支持和无私帮助！感谢 2004 级所有的同学一起度过的令人难忘的三年！

感谢我的父母，他们对我的支持和鼓励使我顺利完成学业，我的成长凝聚着他们的心血，同时也要感谢男友张鹏程对我的帮助和慰勉，使我有了更大的进步和提高！