

# 文本复制检测报告单(全文对照)

№:ADBD2017R\_2017110718233920171217220451401494757791

检测时间:2017-12-17 22:04:51

检测文献: 80022497643220302\_基于动态手势的智能终端身份识别技术研究与实践

作者: 基于动态手势的智能终端身份识别技术研究与实践

检测范围: 中国学术期刊网络出版总库

中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库

中国重要会议论文全文数据库

中国重要报纸全文数据库

中国专利全文数据库

互联网资源(包含贴吧等论坛资源)

英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)

港澳台学术文献库

优先出版文献库

互联网文档资源

图书资源

CNKI大成编客-原创作品库

学术论文联合比对库

个人比对库

时间范围: 1900-01-01至2017-12-17

## 检测结果

总文字复制比: 19.8%

跨语言检测结果: 0%

去除引用文献复制比: 19.8%

去除本人已发表文献复制比: 19.8%

单篇最大文字复制比: 5.9% ( Hadoop YARN资源分配与调度的研究 )

重复字数: [21370]

总段落数: [19]

总字数: [107691]

疑似段落数: [15]

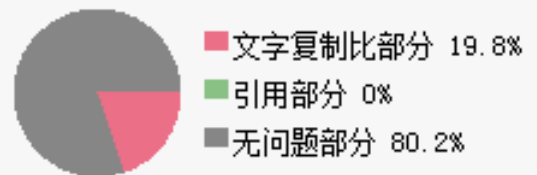
单篇最大重复字数: [6328]

前部重合字数: [973]

疑似段落最大重合字数: [5548]

后部重合字数: [20397]

疑似段落最小重合字数: [29]



指标: ☒ 疑似剽窃观点 ☒ 疑似剽窃文字表述 ☐ 疑似自我剽窃 ☐ 疑似整体剽窃 ☐ 过度引用

表格: 0 脚注与尾注: 0

4% ( 227 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第1部分 ( 总5655字 )
9.5% ( 746 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第2部分 ( 总7816字 )
0% ( 0 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第3部分 ( 总10828字 )
0% ( 0 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第4部分 ( 总4046字 )
0% ( 0 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第5部分 ( 总2749字 )
3.5% ( 29 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第6部分 ( 总837字 )
7.1% ( 651 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第7部分 ( 总9196字 )
22.2% ( 2090 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第8部分 ( 总9421字 )
14.3% ( 405 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第9部分 ( 总2829字 )
31.3% ( 1443 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第10部分 ( 总4607字 )
75.3% ( 5548 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第11部分 ( 总7366字 )
8.5% ( 209 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第12部分 ( 总2448字 )
2.5% ( 31 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第13部分 ( 总1229字 )
43.6% ( 3749 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第14部分 ( 总8600字 )
57.8% ( 5132 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第15部分 ( 总8875字 )
4.7% ( 468 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第16部分 ( 总9907字 )
0% ( 0 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第17部分 ( 总2377字 )
13.1% ( 154 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第18部分 ( 总1174字 )
6.3% ( 488 )	80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第19部分 ( 总7731字 )



( 注释： 无问题部分 文字复制比部分 引用部分 )

疑似剽窃观点 (5)
80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第7部分
1. 实验结果表明，采用本文的算法能有效地提高系统资源利用率，缩短集群的作业执行时间。
80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第11部分
1. 当AMA向RM中的Scheduler组件申请Container时，假设当前轮询的位置的节点Node2且Node2中可分配资源满足AM A所需资源量，Scheduler将Node2中的资源分配给AM A，然而从图4.1可以看出，相对于Node2，Node3的资源负载更小CPU处理速度更快，故将Node3中资源分配给AM A更合理。
2. 最终，根据本章算法编写相应的 Hadoop 资源调度器，并进行实验验证，实验表明本章调度器能够有效的提高集群资源分配的合理化，达到缩短作业执行时间的目的。
80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第14部分
1. 综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。
80022497643220302_基于动态手势的智能终端身份识别技术研究与应用_第15部分
1. 还是动态分析，都需要借助恶意代码分析技术和监控技术获得恶意代码的基本属性和执行信息，以便深入理解恶意代码的功能，进一步实现恶意代码的检测和抑制。

1. 80022497643220302\_基于动态手势的智能终端身份识别技术研究与应用\_第1部分 总字数：5655

相似文献列表 文字复制比：4%(227) 疑似剽窃观点：(0)

1	城市轨道交通对沿线住宅价格的影响研究 李林芸(导师：任波) - 《重庆大学硕士论文》 - 2011-10-01	1.4% ( 81 ) 是否引证：否
2	基于ZigBee的智能家居网关的研究与应用 张英会(导师：张乃通) - 《哈尔滨工业大学硕士论文》 - 2012-12-01	1.3% ( 72 ) 是否引证：否
3	016-220100902230-刘泉影 刘泉影 - 《学术论文联合比对库》 - 2013-04-02	1.2% ( 69 ) 是否引证：否
4	移动IPv6切换技术的研究 刘希伟(导师：代红) - 《辽宁科技大学硕士论文》 - 2011-12-05	1.2% ( 67 ) 是否引证：否
5	CNU1543011026 - 《学术论文联合比对库》 - 2014-04-21	1.2% ( 66 ) 是否引证：否
6	一种基于AF4iOS框架的移动软件设计方法 陈贝(导师：倪友聪) - 《福建师范大学硕士论文》 - 2014-04-06	1.0% ( 59 ) 是否引证：否

7	基于小波神经网络的人脸图像识别研究 倪洽凯(导师：杨静) - 《太原理工大学硕士论文》 - 2012-05-01	0.6% ( 33 ) 是否引证：否
8	船舶装备物联网服务平台的规划研究 周志凤(导师：朱岩) - 《南京理工大学硕士论文》 - 2013-05-01	0.6% ( 32 ) 是否引证：否
9	基于SVM和深度学习的情感分类算法研究 黄志勇(导师：杨富平) - 《重庆邮电大学硕士论文》 - 2016-05-30	0.5% ( 31 ) 是否引证：否
10	制导炮弹可控滚转执行机构关键技术研究 张皎(导师：姚晓先) - 《北京理工大学博士论文》 - 2015-06-01	0.5% ( 31 ) 是否引证：否
11	一种精确估计区域北斗接收机硬件延迟的方法 李昕;郭际明;周吕;覃发超; - 《测绘学报》 - 2016-08-15	0.5% ( 30 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 31 字相似</p> <p>忘导致用户自身无法正常的访问手机。指纹如果被复制就完全失去了对自己的手机的控制，导致无法挽回的损失。</p> <p>为了提高智能终端</p> <p>用户的安全性，人们开始研究基于生物特征的身份识别方法，比如利用</p> <p>智能终端自带的摄像头进行人脸识别和虹膜识别，然而这种方式受周围光线环境影响很大，如果光线很暗或者很亮，或者背景中有其他头</p>	<p>016-220100902230-刘泉影 刘泉影 - 《学术论文联合比对库》 - 2013-04-02 ( 是否引证：否 )</p> <p>1.用，研究其用户扩展性非常重要。由于脑电信号本身特性的限制，脑电只适合于一定样本容量的身份识别。本实验用于研究脑电识别随着用户人数扩展的性能变化，从而确定基于脑电的身份识别的应用范围。第四章结果与分析4.1 实验—4.1.1 脑电采集时间长度与分类正确率图4.1分类正确率随脑电采集时间</p>
2	<p>此处有 31 字相似</p> <p>手势中某一时刻的传感器数值还包括一段时间中所产生的轨迹或者运动特征，比如滑动的速度和距离等。动态手势按照数据采集的方式又</p> <p>分为基于视觉的动态手势和基于数据“手套”的动态手势。</p> <p>基于视觉的</p> <p>动态手势指的是利用手机自带的摄像头对用户的手势进行采集，然后利用采集的数据进行特征提取，最后用相应的算法进行身份识别。此</p>	<p>基于SVM和深度学习的情感分类算法研究 黄志勇 - 《重庆邮电大学硕士论文》 - 2016-05-30 ( 是否引证：否 )</p> <p>1.处理，过滤了生僻字和一些情感倾向模糊的词语，补充了新的情感词。目前依据情感词计算情感强度有多种思路，一般分为基于语料库的方法[16]和基于词语的方法。基于语料库的方法，需要人工来标注语料库，在此基础上，利用词语之间的共现关系、语义关系或者统计特征等，衡量词语的情感极性。这类方法简</p>
3	<p>此处有 35 字相似</p> <p>以此公式作为判决真伪用户的依据。同时还为新的预测样本增加了回收机制，对于判决的结果与真实情况不符的样本进行加权回收，加大</p> <p>模型学习遗漏的用户行为样本的学习的力度，使模型能够得到不断的完善。</p> <p>3、基于动态手势的身份识别平台的系统实现。主要实现了以下几个模块：1) 客户端 ( Android系统 )，主要承担用户交互界面</p>	<p>城市轨道交通对沿线住宅价格的影响研究 李林芸 - 《重庆大学硕士论文》 - 2011-10-01 ( 是否引证：否 )</p> <p>1.价格水平与预测值间的比较，对模型估计效果进行检验。最后，将新建楼盘加入模?52 型并更新模型，随着样本数量的增加，模型也能够得到不断完善。另一方面，特征价格模型对同一城市的不同轨道交通项目，或者其他具有类似经济、环境特征的轨道交通项目沿线楼盘</p>
4	<p>此处有 99 字相似</p> <p>验证明本文方法具有较低的FAR和FRR，能在不影响用户正常使用的条件下有效的监测智能终端是否具有非法的访问。</p> <p>1.4</p> <p>论文结构</p> <p>本文共分为6个章节，具体各个章节的安排如下：</p>	<p>基于ZigBee的智能家居网关的研究与实现 张英会 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否 )</p> <p>1. ZigBee 网络的平台，从而最终实现智能手机终端无线访问和控制家居内部传感器、家电和照明等节点功能。本文的结构安排如下：第一章：绪论。在本章中介绍了课题的研究背景和意义，并对国内外智能家居发展现状进行了研究，最后安排了工作的主要内容和章节。第二章：智能家居系统总体方案设计。在本章中主要分析</p>

<p>第一章绪论，首先介绍了本文的研究背景和研究的意义，接下来介绍了目前国内外在身份识别领域的研究现状，紧接着介绍了本文的主要研究内容和创新点，最后对本文的整体章节分布进行了介绍。</p> <p>第二章动态手势身份识别相关技术研究，首先研究了手势识别的方法，为后续手势识别奠定</p>	<p>了智能家居网关的整体设计架构。分析了家居内部网络平台的搭建，和网</p> <p>移动IPv6切换技术的研究 刘希伟 - 《辽宁科技大学硕士论文》 - 2011-12-05 (是否引证：否)</p> <p>1.它几种改进方案进行分析论证，证明该方案减少了80%以上的切换延迟。 1.4 论文框架 本文主要安排如下：第一章 引言。介绍了本文的研究背景，当前国内外的研究现状，提出了本文重点解决的问题和主要工作。第二章 移动 IPv6 概述。介绍了移动 IP 的基础知识，主要包括移动 IPv4 和移动 IPv6</p> <p>CNU1543011026 - 《学术论文联合比对库》 - 2014-04-21 (是否引证：否)</p> <p>1.前先用平面对影像进行纠正，减少影像之间的差异，然后再进行转点，看能否取得更好的结果。1.4 论文的结构框架论文总共分为五个章节，具体的结构框架如下：第一张为绪论。本章主要介绍了本课题的研究背景和意义、国内外研究现状以及本文的主要内容。第二章主要介绍了基于POS数据的多角度影像</p> <p>2.且快速有效。1.4 论文组织第一章绪论。主要介绍了论文研究的目的及意义，阐述了正射影像生产处理流程及镶嵌线的国内外研究现状，并提出了论文的主要研究内容。第二章研究了正射影像生成。针对影像正射纠正计算密集的特点，本文提出一种基于OpenMP的影像正射纠正方法，具体包括两个关</p> <p>一种基于AF4iOS框架的移动软件设计方法 陈贝 - 《福建师范大学硕士论文》 - 2014-04-06 (是否引证：否)</p> <p>1.进而形成一种移动软件设计方法。5福建师范大学陈贝硕士学位论文 1.3本文组织结构文章的组织结构如下：第一章为绪论。主要介绍本课题的研究背景和意义，以及国内外的研究现状。第二章介绍软件框架。首先介绍软件框架的相关概念，分析框架较类库而言的优势；然后根据框架的应用范围，分为通用框架和应用框架，并</p> <p>城市轨道交通对沿线住宅价格的影响研究 李林芸 - 《重庆大学硕士论文》 - 2011-10-01 (是否引证：否)</p> <p>1.本文以重庆市轨道交通建设为背景，围绕城市轨道交通对沿线住宅价格的影响展开讨论，主要包括以下内容：① 绪论。阐述了论文的研究背景、研究目的和意义以及研究方法，并对国内外相关领域的研究现状进行了简要综述。② 相关理论概述。主要对城市轨道交通进行了概念界定及特点阐述，分析了住宅价格的影响因素，并</p> <p>016-220100902230-刘泉影 刘泉影 - 《学术论文联合比对库》 - 2013-04-02 (是否引证：否)</p> <p>1.扩展测试，用以全面地评估脑电信号作为独立的生物模态进行个人身份识别的性能，以求提高现有生物特征识别系统的实际价值。本文的内容共分为五章，各章节的安排和其具体内容如下：第一章为绪论，介绍了基于脑电的身份识别技术的研究背景和意义、研究现状，并粗略介绍了本课题的主要研究内容和本文结构安排；第</p>
--	---



		二章介绍了本研究的具体内
		基于小波神经网络的人脸图像识别研究 倪洽凯 - 《太原理工大学硕士论文》 - 2012-05-01 ( 是否引证：否 )
		1.提到的算法都是在MATLAB上进行仿真实验的。本文中使用的数据库是目前很多研究者使用的ORL人脸数据库。论文的主要章节安排如下：第一章：详细介绍了本文研究人脸图像识别这一课题的意义；并且还对课题的研究现状做简要说明，主要介绍了目前有那些主要算法；对课题未来的发展趋势做
		船舶装备物联网服务平台的规划研究 周志凤 - 《南京理工大学硕士论文》 - 2013-05-01 ( 是否引证：否 )
		1.导及基本规约，也可以为其分领域、分系统的项目实施、系统集成、持续演化等提供有效的框架支撑。本论文的结构安排如下：第一章：绪论，介绍了本文的项目背景，深入分析了某系列海上科研试验船队先进装备系统所存在的断层化风险；采用了以构建船舶装备物联网服务平台的方式从根本上
		制导炮弹可控滚转执行机构关键技术研究 张皎 - 《北京理工大学博士论文》 - 2015-06-01 ( 是否引证：否 )
		1.合抗扰能力的鲁棒性框架内解决仍然是一个具有迫切性和挑战性的研究课题。1.4本文的主要工作和内容论文共分为七章,主要内容及安排如下:第一章——绪论。介绍论文研究的背景及意义,以及舰炮制导弹药的国内外研究现状,并对相关理论及关键技术进行了概述,包括鸭式布局双旋弹体的动力学建模与控制相关技术的研究
5	此处有 31 字相似 统实现 第五章实验结果与分析，首先介绍了本文的实验环境，然后介绍了实验数据的来源和实验结果的评估标准，最后对K值的选择进行了验证，验证了本文提出的方法的可行性，同时对比了其他方法，证明了本文方法优于其他方法。 第六章总结了本文所做的工作，并对本文的不足进行了分析，最后对未来进行展望。	一种精确估计区域北斗接收机硬件延迟的方法 李昕;郭际明;周吕;覃发超; - 《测绘学报》 - 2016-08-15 ( 是否引证：否 ) 1.用GEO实测双频数据扣除估计的接收机DCB和IGS获取的卫星DCB,计算对应穿刺点一天的VTEC量并和GIM格网内插结果进行比对分析,验证了本文提出的方法的可行性。1方法介绍1.1站间单差电离层延迟与卫星-接收机几何距离关系北斗目前发布了3个频率信号,主要对应载波和伪距观测值。双频无几何组合观测值

指 标		
疑似剽窃文字表述		
1. 论文结构		
本文共分为6个章节，具体各个章节的安排如下：		
第一章绪论，首先介绍了本文的研究背景和研究的意义，接下来介绍了目前国内外在身份识别领域的研究现状，紧接着介绍了本文的主要研究内容和创新点，		
2. 80022497643220302_基于动态手势的智能终端身份识别技术研究与实现_第2部分 总字数：7816		
相似文献列表 文字复制比：9.5%(746) 疑似剽窃观点：(0)		
1	基于Kubernetes的资源动态调度的研究与实现 杨鹏飞(导师：黄忠东) - 《浙江大学硕士论文》 - 2017-01-15	3.5% ( 270 ) 是否引证：否
2	智能移动平台融合定位技术综述 裴凌;刘东辉;龚正;赵毅; - 《导航定位与授时》 - 2017/9/18 8:	2.9% ( 224 ) 是否引证：否

3	基于人工神经网络的上肢关节刚度估算 宋伟任;崔泽;杨洪鑫;韩汪洋;王玉梅;邱国文;鄢旋; - 《计量与测试技术》 - 2017-08-30	2.6% ( 200 ) 是否引证 : 否
4	机器学习常见算法分类汇总 -- 程序猿 -- 传送门 - 《网络 ( <a href="http://chuansong.me/">http://chuansong.me/</a> ) 》 - 2015	2.4% ( 188 ) 是否引证 : 否
5	[转载]机器学习的认知和算法总结【仅供参考】_笨三乐 - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2015	2.4% ( 188 ) 是否引证 : 否
6	机器学习常见算法分类汇总 - 《网络 ( <a href="http://www.aiweibang">http://www.aiweibang</a> ) 》 - 2016	2.4% ( 188 ) 是否引证 : 否
7	机器学习算法汇总：人工神经网络、深度学习及其它_TW_mathematica - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2017	2.4% ( 188 ) 是否引证 : 否
8	机器学习常见算法分类汇总 - 爱橙子的OK绷的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	2.4% ( 188 ) 是否引证 : 否
9	神经网络使用情景 - 杨航的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	2.4% ( 188 ) 是否引证 : 否
10	机器学习常见算法分类 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2016	2.4% ( 188 ) 是否引证 : 否
11	机器学习常用算法总结 - 竭尽全力的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	2.3% ( 177 ) 是否引证 : 否
12	基于感知机优化的BP神经网络邮件分类算法研究 马秋明(导师：秦志光) - 《电子科技大学硕士论文》 - 2011-03-01	2.3% ( 176 ) 是否引证 : 否
13	基于免疫遗传算法的模糊神经网络研究 赵红(导师：史震) - 《哈尔滨工程大学硕士论文》 - 2007-12-01	2.1% ( 166 ) 是否引证 : 否
14	胡慧君_带钢缺陷图像分类中的智能算法研究 胡慧君 - 《学术论文联合比对库》 - 2015-10-17	1.7% ( 131 ) 是否引证 : 否
15	枣虫害图像自动识别关键技术研究 王东(导师：周桂红) - 《河北农业大学硕士论文》 - 2011-06-02	1.6% ( 123 ) 是否引证 : 否
16	含腐蚀缺陷海洋立管剩余强度预测 王超(导师：俞树荣) - 《兰州理工大学硕士论文》 - 2014-04-21	1.5% ( 118 ) 是否引证 : 否
17	泵送混凝土可泵性评价 李帅(导师：侯子义) - 《河北工业大学硕士论文》 - 2015-05-01	1.5% ( 115 ) 是否引证 : 否
18	基于神经网络的高效智能入侵检测系统 撒书良,蒋巍川,张世永 - 《计算机工程》 - 2004-05-20	1.4% ( 113 ) 是否引证 : 否
19	天然气水合物晶体结构判别及转换方法研究 钟煜(导师：刘武) - 《西南石油大学硕士论文》 - 2017-05-01	1.3% ( 105 ) 是否引证 : 否
20	从AlphaGo到智能汽车 AI将改变世界 - 《网络 ( <a href="http://auto.163.com/">http://auto.163.com/</a> ) 》 - 2017	1.3% ( 101 ) 是否引证 : 否
21	小波分解法在冻土路基下土体温度预测中的应用 文斌;吴青柏;刘永智; - 《公路交通科技(应用技术版)》 - 2012-09-15	1.0% ( 79 ) 是否引证 : 否
22	基于人工神经网络的遥操作预测仿真 王明明;李世其;朱文革;卢亚夫; - 《载人航天》 - 2012-09-25	0.9% ( 68 ) 是否引证 : 否
23	安卓平台下基于传感器的手势识别技术应用研究 李正山(导师：王海婴) - 《北京邮电大学硕士论文》 - 2014-01-08	0.9% ( 68 ) 是否引证 : 否
24	基于表面肌电多特征的下肢行走关键模态识别研究 刘亚伟(导师：万柏坤) - 《天津大学硕士论文》 - 2009-05-01	0.7% ( 58 ) 是否引证 : 否
25	浅析人工智能的现状与发展趋势 聂志伟; - 《数码世界》 - 2017-05-01	0.7% ( 56 ) 是否引证 : 否
26	12212780吴余 - 《学术论文联合比对库》 - 2014-04-23	0.6% ( 45 ) 是否引证 : 否
27	K-Means算法改进及其在森林健康评价中的应用 谭浩(导师：李建军) - 《中南林业科技大学硕士论文》 - 2015-06-30	0.4% ( 35 ) 是否引证 : 否
28	在线社会网络中基于属性的重叠社区发现算法研究与应用 王宇欢(导师：易秀双) - 《东北大学硕士论文》 - 2014-06-01	0.4% ( 35 ) 是否引证 : 否
29	Apriori关联规则挖掘算法在高校教学管理系统中的应用研究 杨超(导师：陆鑫;李文联) - 《电子科技大学硕士论文》 - 2013-03-25	0.4% ( 34 ) 是否引证 : 否
30	基于“合作—参与”计算认知模型的半监督学习算法研究与应用	0.4% ( 31 )

	原文内容	相似内容来源
1	<p>此处有 37 字相似</p> <p>仪[13]、手指滑屏的速度和距离、手指按压的力度和面积等)来判断当前访问手机的用户是合法用户还是冒名的假用户。手势识别的过程一般涉及数据采集、数据预处理、特征筛选、手势行为建模、手势识别等过程。具体的过程如图2-1所示。数据采集主要是采集用户在使用手机的过程中产生的各种传感器的数据,将数据存储到本地的数据库,</p>	<p>Apriori关联规则挖掘算法在高校教学管理系统中的应用研究 杨超 - 《电子科技大学硕士论文》 - 2013-03-25 (是否引证:否)</p> <p>1.Apriori 算法在教学管理系统中的应用。第四章详细介绍了 Apriori 算法的教学管理中的应用实现,对应用过程中的数据采集、数据筛选和清洗、数据分析、结果呈现等四大部分分别进行了详细描述,并在本章最后给出了设计源代码。7第五章详细介绍了 Apriori 算法在教</p>
2	<p>此处有 474 字相似</p> <p>目前常用的时间匹配算法是离散小波变换[15] ( Discrete Wavelet Transformation ) 算法。</p> <p>2.人工神经网络</p> <p>人工神经网络[16][17] ( Artificial Neural Network , ANN ) 简称神经网络(NN),是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系统对复杂信息的处理机制的一种数学模型。该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征,将信息的加工和存储结合在一起,以其独特的知识表示方式和智能化的自适应学习能力,引起各学科领域的关注。它实际上是一个有大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统。重要的人工神经网络算法包括:感知器神经网络[18] ( Perceptron Neural Network ),反向传递神经网络[19] ( Back Propagation ), Hopfield网络,自组织映射[20] ( Self-Organizing Map, SOM ) 和学习矢量量化[21] ( Learning Vector Quantization , LVQ ) 等。</p> <p>3.机器学习</p> <p>[22]</p> <p>模型的建立与数据类型是分不开的,不同的数据类型会导致建模方式也有所差异。根据学习方式的不同,将机器学习的算法</p>	<p>基于Kubernetes的资源动态调度的研究与实现 杨鹏飞 - 《浙江大学硕士论文》 - 2017-01-15 (是否引证:否)</p> <p>1.使用量序列进行预测,最后得出预测误差。如果预测误差在可承受范围内,则可以运用该模型进行预测。</p> <p>3.2.2 RBF神经网络模型人工神经网络 ( Artificial Neural Network , ANN), 简称神经网络 ( NN), 是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系统对复杂信息的处理机制的一种数学模型。该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征,将信息的加工和存储结合在一起,以其独特的知识表示方式和智能化的自适应学习能力,引起各学科领域的关注。它实际上是一个有大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[3G]。■径向基函数</p> <p>智能移动平台融合定位技术综述 裴凌;刘东辉;龚正;赵毅; - 《导航定位与授时》 - 2017/9/18 8: (是否引证:否)</p> <p>1.如高维的非线性、非高斯动态系统的状态递推估计或概率推理问题;因为粒子滤波的算法性能在理论上对这些都不敏感[30]。3.6人工神经网络人工神经网络(Artificial Neural Network,ANN)是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑</p> <p>2.上对这些都不敏感[30]。3.6人工神经网络人工神经网络(Artificial Neural Network,ANN)是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系统对复杂信息的处理机制的一种数学模型。该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征。它实际上是一个由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[31</p> <p>3.模拟人脑的神经系统对复杂信息的处理机制的一种数学模型。该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征。它实际上是一个由大量简</p>

	<p>单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[31]。</p> <p>人工神经网络的工作过程包括离线学习和在线判断两部分。学习过程中各神经元进行规则学习,权参数调整,非线性映射关系拟合以达到训练精度;</p>
	<p>基于人工神经网络的上肢关节刚度估算 宋伟任;崔泽;杨洪鑫;韩汪洋;王玉梅;邱国文;鄢旋; - 《计量与测试技术》 - 2017-08-30 ( 是否引证 : 否 )</p>
	<p>1.绍人工神经网络模型,第三节讲述训练样本的采集方法,第四节描述本文实验,最后一节总结全文。2 人工神经网络2.1 人工神经网络基础人工神经网络(Artificial Neural Network,ANN)简称神经网络(NN),是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系统对复杂信息的处理机制的一种测量模型。人工神经元结构如图1所示:人工神经元模型可以看成是有三种基本元素组成:(1)一组连接,连接强度由各连接上的值表示,权</p> <p>2.绍人工神经网络模型,第三节讲述训练样本的采集方法,第四节描述本文实验,最后一节总结全文。2 人工神经网络2.1 人工神经网络基础人工神经网络(Artificial Neural Network,ANN)简称神经网络(NN),是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系</p>
	<p>机器学习常见算法分类汇总 -- 程序猿 -- 传送门 - 《网络 ( <a href="http://chuansong.me/">http://chuansong.me/</a> ) 》 - ( 是否引证 : 否 )</p>
	<p>1.归问题。人工神经网络是机器学习的一个庞大的分支,有几百种不同的算法。(其中深度学习就是其中的一类算法,我们会单独讨论),重要的人工神经网络算法包括:感知器神经网络 ( Perceptron Neural Network ),反向传递 ( Back Propagation ), Hopfield网络,自组织映射 ( Self-Organizing Map,SOM )。学习矢量量化 ( Learning Vector Quantization , LVQ ) 深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注,特别是百度也开始发力深度学习后,更是在国内引起了很多关注</p>
	<p>[转载]机器学习的认知和算法总结【仅供参考】 笨三乐 - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证 : 否 )</p>
	<p>1.归问题。人工神经网络是机器学习的一个庞大的分支,有几百种不同的算法。(其中深度学习就是其中的一类算法,我们会单独讨论),重要的人工神经网络算法包括:感知器神经网络(Perceptron Neural Network),反向传递(Back Propagation), Hopfield网络,自组织映射(Self-Organizing Map,SOM)。学习矢量量化(Learning Vector Quantization , LVQ) 深度学习 ImgID_13# 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注,特别是 百度也开始发力深度学习后,更</p>
	<p>机器学习常见算法分类汇总 - 《网络</p>



	<div data-bbox="837 40 1544 73" data-label="Text"> <p>( <a href="http://www.aiweibang">http://www.aiweibang</a> ) 》 - ( 是否引证：否 )</p> </div> <div data-bbox="837 73 1544 546" data-label="Text"> <p>1.归问题。人工神经网络是机器学习的一个庞大的分支，有几百种不同的算法。(其中深度学习就是其中的一类算法，我们会单独讨论)，重要的人工神经网络算法包括：感知器神经网络 ( Perceptron Neural Network ),反向传递 ( Back Propagation )，Hopfield网络，自组织映射 ( Self-Organizing Map,SOM )。学习矢量量化 ( Learning Vector Quantization，LVQ ) 深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注，特别是百度也开始发力深度学习后，更是在国内引起了很多关注</p> </div> <div data-bbox="837 546 1544 636" data-label="Text"> <p>机器学习算法汇总：人工神经网络、深度学习及其它 TW_mathematica - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证：否 )</p> </div> <div data-bbox="837 636 1544 1061" data-label="Text"> <p>1.问题。人工神经网络是机器学习的一个庞大的分支，有几百种不同的算 法。(其中深度学习就是其中的一类算法，我们会单独讨论)，重要的人工神经网络算法包括：感知器神经网络 ( Perceptron Neural Network ),反向传递 ( Back Propagation )，Hopfield网络，自组织映射 ( Self-Organizing Map,SOM )。学习矢量量化 ( Learning Vector Quantization，LVQ ) 深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注，特别是百度也开始发力深度学习后，更是在国内引起了很多关</p> </div> <div data-bbox="837 1061 1544 1151" data-label="Text"> <p>机器学习常见算法分类汇总 - 爱橙子的OK绷的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )</p> </div> <div data-bbox="837 1151 1544 1621" data-label="Text"> <p>1.归问题。人工神经网络是机器学习的一个庞大的分支，有几百种不同的算法。(其中深度学习就是其中的一类算法，我们会单独讨论)，重要的人工神经网络算法包括：感知器神经网络 ( Perceptron Neural Network ),反向传递 ( Back Propagation )，Hopfield网络，自组织映射 ( Self-Organizing Map,SOM )。学习矢量量化 ( Learning Vector Quantization，LVQ )。深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注，特别是百度也开始发力深度学习后，更是在国内引起了很多关注</p> </div> <div data-bbox="837 1621 1544 1688" data-label="Text"> <p>神经网络使用情景 - 杨航的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )</p> </div> <div data-bbox="837 1688 1544 2145" data-label="Text"> <p>1.问题。人工神经网络是机器学习的一个庞大的分支，有几百种不同 的算法。(其中深度学习就是其中的一类算法，我们会单独讨论)，重要的人工神经网络算法包括：感知器神经网络 ( Perceptron Neural Network ),反向传递 ( Back Propagation )，Hopfield网络，自组织映射 ( Self-Organizing Map,SOM )。学习矢量量化 ( Learning Vector Quantization，LVQ ) 深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注，特别是百度也开始发力深度学习后，更是在国内引起了很多关</p> </div>
--	---

	<div data-bbox="837 40 1543 129"> <p>机器学习常见算法分类 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2016/9/20 22:10:27 ( 是否引证 : 否 )</p> </div> <div data-bbox="837 129 1543 600"> <p>1.归问题。人工神经网络是机器学习的一个庞大的分支 , 有几百种不同的算法。( 其中深度学习就是其中的一类算法 , 我们会单独讨论 ) , 重要的人工神经网络算法包括 : 感知器神经网络 ( Perceptron Neural Network ) ,反向传递 ( Back Propagation ) , Hopfield网络 , 自组织映射 ( Self-Organizing Map,SOM ) 。学习矢量量化 ( Learning Vector Quantization , LVQ ) 深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注 , 特别是百度也开始发力深度学习后 , 更是在国内引起了很多关注</p> </div> <div data-bbox="837 600 1543 689"> <p>机器学习常用算法总结 - 竭尽全力的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证 : 否 )</p> </div> <div data-bbox="837 689 1543 1120"> <p>1.归问题。人工神经网络是机器学习的一个庞大的分支 , 有几百种不同的算法。( 其中深度学习就是其中的一类算法 , 我们会单独讨论 ) , 重要的人工神经网络算法包括 : 感知器神经网络 ( Perceptron Neural Network ) ,反向传递 ( Back Propagation ) , Hopfield网络 , 自组织映射 ( Self-Organizing Map,SOM ) 。学习矢量量化 ( Learning Vector Quantization , LVQ ) 。 10、深度学习 深度学习算法是对人工神经网络的发展。在近期赢得了很多关注 , 特别是百度也开始发力深度学习后 , 更是在</p> </div> <div data-bbox="837 1120 1543 1187"> <p>基于感知机优化的BP神经网络邮件分类算法研究 马秋明 - 《电子科技大学硕士论文》 - 2011-03-01 ( 是否引证 : 否 )</p> </div> <div data-bbox="837 1187 1543 1657"> <p>1.方法等。 2.2.3 神经网络技术研究 2.2.3.1 神经网络概述 人工神经网络 ( 简称 ANN ) 是基于生物学中神经网络的基本原理 , 在理解和抽象了人脑结构和外界刺激响应机制后 , 以网络拓扑知识为理论基础 , 模拟人脑的神经系统对复杂信息的处理机制的一种数学模型[27]。该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征 , 将信息的加工和存储结合在一起 , 以其独特的知识表示方式和智能化的自适应学习能力 , 引起各学科领域的关注 , 拥有广阔的应用前景。神经网络的发展有悠久的历史。从 1943 年 McCulloch 和 Pitts 一起建</p> </div> <div data-bbox="837 1657 1543 1724"> <p>胡慧君 带钢缺陷图像分类中的智能算法研究 胡慧君 - 《学术论文联合比对库》 - 2015-10-17 ( 是否引证 : 否 )</p> </div> <div data-bbox="837 1724 1543 2105"> <p>1.物神经网络 , 是一类模式匹配算法。通常用于解决分类和回归问题。人工神经网络是机器学习的一个庞大的分支 , 有几百种不同的算法。重要的人工神经网络算法包括 : 感知器神经网络(Perceptron Neural Network) , 反向传递(Back Propagation) , Hopfield网络 , 自组织映射(Self-Organizing Map , SOM)。鉴于本论文讨论的对象——带钢缺陷图像的相关特性 , 本论文只分析研究两种分类效果较好、应用广泛且有着良好理论基础的分类算法—</p> </div> <div data-bbox="837 2105 1543 2152"> <p>枣虫害图像自动识别关键技术研究 王东 - 《河北农业大学</p> </div>
--	---

	<div data-bbox="837 38 1543 705"> <p>硕士论文》- 2011-06-02 ( 是否引证：否 )</p> <p>1.很好的发展应用前景。基于上述分析，本文采用神经网络方法设计和实现中国枣虫害识别的分类器。5.2 人工神经网络 人工神经网络(Artificial Neural Network,简称 ANN)是在人类对其大脑神经网络认识理解的基础上人工构造的能够实现某种功能的神经网络。它是理论化的人脑神经网络</p> <p>2.功能的神经网络。它是理论化的人脑神经网络的数学模型，是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。它实际上是由大量简单元件相互连接而成的复杂网络，具有高度的非线性，能够进行复杂的逻辑操作和非线性关系实现的系统。人工神经网络虽然反映了人脑功能的若干基本特性，但不是人脑生物神经网络的真实写照，而只是对它的简化、抽象与模拟。因此，人工神</p> </div> <div data-bbox="837 705 1543 1400"> <p>基于神经网络的高效智能入侵检测系统 撒书良,蒋巍川,张世永 - 《计算机工程》- 2004-05-20 ( 是否引证：否 )</p> <p>1.统做进一步的处理。这种结构可以通过减少系统的开销和IDS误报率来提高检测系统的效用。本文对这种结构进行了详细的讨论。1 人工神经网络人工神经网络(Artificial Neural Network, ANN)是在人类对其大脑神经网络认识理解的基础上人工构造的能够实现某种功能的神经网络。它是理论化的人脑神经网络的数学模</p> <p>2.够实现某种功能的神经网络。它是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统。神经网络采用样本学习的方式,直接从过程的输入输出关系中提取信息,并反映到神经元之间相互作用的权值上。神经网络采用的是分布式存储方</p> </div> <div data-bbox="837 1400 1543 2094"> <p>泵送混凝土可泵性评价 李帅 - 《河北工业大学硕士论文》- 2015-05-01 ( 是否引证：否 )</p> <p>1.因变量之间的线性回归方程,对于今后的对这类因素与指标间的变化关系有了更加便捷的研究途径,在实际工程中应用十分广泛。5.1人工神经网络人工神经网络(Artificial Neural Network,简称ANN)是在人类对其大脑神经网络认识理解的基础上人工构造的能够实现某种功能的神经网络。它是理论化的人脑神经网络的数学</p> <p>2.够实现某种功能的神经网络。它是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。它实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[30]。人工神经网络吸取了生物神经网络的许多优点,它以人的大脑工作模式为基础,研究自适应的、非程序的信息处理本质。这种工作机制的主要优点是,</p> </div> <div data-bbox="837 2094 1543 2159"> <p>基于免疫遗传算法的模糊神经网络研究 赵红 - 《哈尔滨工程大学硕士论文》- 2007-12-01 ( 是否引证：否 )</p> </div>
--	--

		<p>1.强,干扰和参数变化对控制效果的影响被大大减弱,尤其适合于非线性、时变及纯滞后系统的控制。</p> <p>.2.4人工神经网络 人工神经网络(Artificial Neural Network, 简称ANN)正是在人类对其大脑神经网络认为理解的基础上人工构造的能够实现某种功能的神经网络。它是理论</p> <p>2.网络。它是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息系统。它实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统。它涉及生物电子、计算机、数学、物理等学科,有着十分广泛的应用背景和前景。简单的说,神经网络就是用</p> <p>含腐蚀缺陷海洋立管剩余强度预测 王超 - 《兰州理工大学硕士论文》 - 2014-04-21 (是否引证:否)</p> <p>1.可以克服各自的局限性,到达令人满意的效果。因此,本文将遗传算法和人工神经网络相结合来解决实际问题。2.2人工神经网络人工神经网络(Artificial Neural Network, 简称ANN)是在人类对其大脑神经网络认识理解的基础上人工构造的能够实现某种功能的神经网络[34_38]。它是理论</p> <p>2.34_38]。它是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。它实际上是一个由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统。2.2.1人工神经元模型人工神经元是对人脑生物神经元的模仿,是人工神经网络的基本组成部分,是一个多输入单输出的信息处理单元</p> <p>天然气水合物晶体结构判别及转换方法研究 钟煜 - 《西南石油大学硕士论文》 - 2017-05-01 (是否引证:否)</p> <p>1.进行预测。3.1人工神经网络人工神经网络(Artificial Neural Network, ANN),是基于生物学中神经网络的基本原理,在理解和抽象了人脑结构和外界刺激响应机制后,以网络拓扑知识为理论基础,模拟人脑的神经系统对复杂信息的处理机制的一种数学模型,具有自适应学习能力、自组织、高容错性、并行处理等优点。3.1.1人工神经元如图3-1所示,所有的神经元都对应不同的权重(%,,),在每一个神经元中,加权的输入值(</p> <p>从AlphaGo到智能汽车 AI将改变世界 - 《网络(<a href="http://auto.163.com/">http://auto.163.com/</a>)》 - (是否引证:否)</p> <p>1.神经网络算法模拟生物神经网络,是一类模式匹配算法。通常用于解决分类和回归问题。重要的人工神经网络算法包括:感知器神经网络(Perceptron Neural Network),反向传递(Back Propagation),Hopfield网络,自组织映射(Self-Organizing Map,SOM)。深度学习算法对人工神经网络进一步发展。目前,在短短的不到10年时间里,深度学习带来了在视觉、语音等领域革命</p>
--	--	--



		<p>小波分解法在冻土路基下土体温度预测中的应用 文斌;吴青柏;刘永智;-《公路交通科技(应用技术版)》-2012-09-15 (是否引证:否)</p> <p>1.qBq1.3 BP神经网络模型人工神经网络是在人类对其大脑神经网络认识理解的基础上人工构造的能够实现某种功能的神经网络。它实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统。BP神经网络是人工神经网络应用最广泛的一种。经过小波分解后得到的逼近信号,是对原始数据进行平滑后得到的时间序列。然而在较低的分解水平下,其不能达</p> <p>基于人工神经网络的遥操作预测仿真 王明明;李世其;朱文革;卢亚夫;-《载人航天》-2012-09-25 (是否引证:否)</p> <p>1.神经网络预测模型人工神经网络是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。它实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[3]。人工神经网络处理单元大体可以分为3类:输入单元、输出单元和隐单元。输入单元接受外部环境的信号和其他系统模型处理的数据,输出单元将系统</p> <p>基于表面肌电多特征的下肢行走关键模态识别研究 刘亚伟-《天津大学硕士论文》-2009-05-01 (是否引证:否)</p> <p>1.。它是理论化的人脑神经网络的数学模型,是基于模仿大脑神经网络结构和功能而建立的一种信息处理系统。它实际上是由大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统[42][47]。人工神经网络吸收了生物神经网络的许多优点,因而有其固有的特点: 1. 高度</p> <p>浅析人工智能的现状与发展趋势 聂志伟;-《数码世界》-2017-05-01 (是否引证:否)</p> <p>1.至超越了人类专家的水平。4.4人工神经网络人工神经网络,作为未来人工智能应用的新领域,也被简称为类神经网络或神经网络,它实际上是一个有大量简单元件相互连接而成的复杂网络,具有高度的非线性,能够进行复杂的逻辑操作和非线性关系实现的系统,具有并行分布的处理能力、高容错性、智能化和自学习等能力。5改进策略为了促进人工智能的快速发展,我们应该以新角度、新眼光来</p>
3	<p>此处有 35 字相似</p> <p>3.机器学习[22]</p> <p>模型的建立与数据类型是分不开的,不同的数据类型会导致建模方式也有所差异。根据学习方式的不同,将机器学习的算法分为监督式学习算法与非监督式学习算法。监督式学习算法是指</p> <p>在训练时,每个数据记录都有一个标签,表示这个数据记录属于哪个类别。训练过程中,会不断比较预测值与标签值的差异,根据差异不</p>	<p>基于“合作—参与”计算认知模型的半监督学习算法研究与应用 邓超-《哈尔滨工业大学博士论文》-2009-06-01 (是否引证:否)</p> <p>1.图象识别、自然语言理解、生物信息学、计算机辅助诊断等。半监督机器学习方法是受应用驱动而产生的新方法。传统的机器学习方法包括监督学习和无监督学习。其中监督学习方法,如 ID3 决策树、贝叶斯分类算法、BP 神经网络、遗传算法等分类算法需要大量的带标记数据作训练集,对</p> <p>在线社会网络中基于属性的重叠社区发现算法研究与应用 王宇欢-《东北大学硕士论文》-2014-06-01 (是否引证:否)</p> <p>1.了比较详尽的阐述。-17-东北大学硕去学位论文 第</p>

		<p>3章动态属性群算法第3章动态属性群算法<b>机器学习算法</b>可<sup>^</sup>式分为有<b>监督学习</b>和<b>无监督学习</b>算法两大类。所谓有<b>监督学习</b>，是指从已经标注好类别的数据样本中学习，而无监督学习，是指根据数据本身的内在特点进行学习，样本事先并没有清晰的分类。</p> <p>机器学习常用算法总结 - 竭尽全力的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )</p> <p>1.，这样可以让人们在建模和算法选择的时候考虑能根据输入数据来选择最合适的算法来获得最好的结果。在机器学习领域，有几种主要的<b>学习方式</b>：<b>监督学习</b>，<b>非监督学习</b>，<b>半监督学习</b>和强化学习。1、监督式学习：在监督式学习下，输入数据被称为“训练数据”，每组训练数据有一个明确的标识或结果，如对防垃</p>
4	<p><b>此处有 35 字相似</b></p> <p>数据不要求记录都需要打上标签，而是根据数据本身的分布特性，能够自称为一类。非监督不需要目标标签，而是学出数据自身类特性，<b>最后对类特性进行映射作为最终的预测结果。典型算法例如K-Means算法和Apriori算法[25]。</b></p> <p>2.1.3 手势识别方法对比</p> <p>手势识别算法没有绝对的优劣之分，对于不同的场景、不同类</p>	<p>K-Means算法改进及其在森林健康评价中的应用 谭浩 - 《中南林业科技大学硕士论文》 - 2015-05-01 ( 是否引证：否 )</p> <p>1.称之为动态聚类或者逐步聚类方法,其基本思想是开始先粗略的分类,然后按照某种最优的原则修改不合理的分类,直到所有分类比较合理为止,此时形成<b>最终</b>的分类结果U”。同时,K-Means算法也是基于距离聚类算法的典型代表,将距离的大小作为两个数据对象进行相似性评价的重要指标,即距离越近的数据对象,其相似性越</p>
5	<p><b>此处有 53 字相似</b></p> <p>是数据挖掘中一类重要的算法研究领域。聚类算法，主要是模拟自然数据的一种现象，即“物以类聚，人以群分”。聚类实质上是对数据<b>空间进行划分，是以子空间的内部数据点之间的相似性尽可能大，子空间之间的数据相似性尽可能的小为目标将数据空间</b></p> <p>划分为若干个子空间。聚类典型的特点是无需标记数据，基于数据分布特点学习，自动将数据归类。聚类算法整体需要经过如下四个步骤</p>	<p>基于免疫遗传算法的模糊神经网络研究 赵红 - 《哈尔滨工程大学硕士论文》 - 2007-12-01 ( 是否引证：否 )</p> <p>1. 研究3.1模糊聚类的含义及应用 聚类就是根据某种相似性准则将样本空间分成多个子<b>空间，使每个子空间内部样本点尽可能相似，不同子空间内样本点之间差异尽可能大，将样本空间中的样本数据集划分为可以用来表示系统行为的一些子集，其实质是寻找隐藏在数据中不同的数据模型，是一个无监督学</b></p>
6	<p><b>此处有 46 字相似</b></p> <p>学习中十大经典算法之一，也是聚类算法中的典型，更是工业上应用较多的算法之一。K-Means算法目的是将一个数据集划分为<b>若干个簇类，让属于同一簇类的数据对象之间能够高度相似，不同类的数据对象之间能够有较大的差异。</b></p> <p>该算法的基本思想是首先取定K个中心以及K个中心的初始位置，再计算每个数据点离各个中心点的距离，根据距离的远近将其划分到不</p>	<p>12212780吴余 - 《学术论文联合比对库》 - 2014-04-23 ( 是否引证：否 )</p> <p>1.有用信息和知识的一种有效手段。聚类分析就是基于“物以类聚”的思想，根据某种定义的准则，将数据集进行分类，形成<b>若干个组，在同一组中的数据对象彼此之间相似，而不同组间的数据对象存在较大的区别，最后分组得到的结果成为一个聚类[2]。</b>它是一种无监督学习过程，我们在对数据进行分析之前是没有先验知识的，</p>
7	<p><b>此处有 31 字相似</b></p> <p>能终端身份识别方法。</p> <p>2.4 本章小结</p>	<p>安卓平台下基于传感器的手势识别技术应用研究 李正山 - 《北京邮电大学硕士论文》 - 2014-01-08 ( 是否引证：否 )</p> <p>1.骤是整个系统的核心，识别算法的好坏直接关系到整个手势识别系统的精度和效率。在第二章基础理论中</p>

	本章介绍了本文所用的一些关键技术，其中第一小节包括动态手势识别的简介，几种 <b>常见的动态手势识别方法，包括模板匹配、人工神经网络和机器学习。</b> 接下来对比了几种手势识别方法的优劣。第二小节详细介绍了两种相关性分析的方法，有互信息法和皮尔森相关系数，为之后的特征选择	，我们简单介绍了 <b>常见的手势识别算法，包括模板匹配、隐马尔可夫模型、人工神经网络和机器学习</b> ，并从计算开销、识别精度等方面对这几类算法做了简单地比较分析。相关研究表明，模板匹配类型的手势识别算法31
--	---	--

指 标
疑似剽窃文字表述

1.	该模型以并行分布的处理能力、高容错性、智能化和自学习等能力为特征，将信息的加工和存储结合在一起，以其独特的知识表示方式和智能化的自适应学习能力，引起各学科领域的关注。它实际上是一个有大量简单元件相互连接而成的复杂网络，具有高度的非线性，能够进行复杂的逻辑操作和非线性关系实现的系统。
2.	空间进行划分，是以子空间的内部数据点之间的相似性尽可能大，子空间之间的数据相似性尽可能的小为目标将数据空间
3.	若干个簇类，让属于同一簇类的数据对象之间能够高度相似，不同类的数据对象之间能够有较大的差异。

3. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第3部分	总字数：10828
相似文献列表	文字复制比：0%(0) 疑似剽窃观点：(0)

4. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第4部分	总字数：4046
相似文献列表	文字复制比：0%(0) 疑似剽窃观点：(0)

5. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第5部分	总字数：2749
相似文献列表	文字复制比：0%(0) 疑似剽窃观点：(0)

6. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第6部分	总字数：837
相似文献列表	文字复制比：3.5%(29) 疑似剽窃观点：(0)

1	城市轨道交通对沿线住宅价格的影响研究 李林芸(导师：任波) - 《重庆大学硕士论文》 - 2011-10-01	3.5% ( 29 ) 是否引证：否
---	--	-----------------------

原文内容	相似内容来源
此处有 29 字相似 以此公式作为判决真伪用户的依据。同时还为新的预测样本增加了回收机制，对于判决的结果与真实情况不符的样本进行加权回收，加大 <b>模型学习遗漏的用户行为样本的学习的力度，使模型能够得到不断</b> 的完善。 3、基于动态手势的身份识别平台的系统实现。主要实现了以下几个模块：1 ) 客户端 ( Android系统 ) ，主要承担	城市轨道交通对沿线住宅价格的影响研究 李林芸 - 《重庆大学硕士论文》 - 2011-10-01 ( 是否引证：否 ) 1.价格水平与预测值间的比较，对模型估计效果进行检验。最后，将新建楼盘加入模?52 型并更新 <b>模型，随着样本数量的增加，模型也能够得到不断完善。</b> 另一方面，特征价格模型对同一城市的不同轨道交通项目，或者其他具有类似经济、环境特征的轨道交通项目沿线楼盘

7. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第7部分	总字数：9196
相似文献列表	文字复制比：7.1%(651) 疑似剽窃观点：(0)

1	基于云计算的分布式存储技术 唐京伟; - 《中国传媒科技》 - 2013-08-08	2.7% ( 249 ) 是否引证：否
2	云计算在教育信息化中的应用研究 张鹏; - 《信息通信》 - 2012-12-15	2.5% ( 232 ) 是否引证：否
3	Hadoop YARN资源分配与调度的研究 李媛祯(导师：杨群) - 《南京航空航天大学硕士论文》 - 2015-01-01	2.4% ( 221 ) 是否引证：否

4	浅析广电行业如何应用云计算 林宗达; - 《中国有线电视》 - 2012-06-15	2.3% ( 216 ) 是否引证 : 否
5	关于云计算及其建设的探讨 张建国; - 《山西电子技术》 - 2012-10-15	2.0% ( 188 ) 是否引证 : 否
6	基于Google的云计算实例分析 蔡键;王树梅; - 《电脑知识与技术》 - 2009-09-05	2.0% ( 188 ) 是否引证 : 否
7	云计算在图书馆建设与信息服务中潜在价值探析 李征; - 《大学图书馆学报》 - 2011-01-21	2.0% ( 186 ) 是否引证 : 否
8	基于云计算的数字图书馆信息服务平台 陈宫;牛秦洲; - 《情报科学》 - 2012-05-05	2.0% ( 185 ) 是否引证 : 否
9	实达云计算海量数据挖掘效率大大提升_hulala9876 - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2014	2.0% ( 182 ) 是否引证 : 否
10	【智慧建筑4MC之云计算Cloud&nbsp;Computing】_智慧运维-smartFM - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2015	1.9% ( 172 ) 是否引证 : 否
11	云计算在人才管理中的展望 朱建新; - 《上海企业》 - 2011-12-10	1.9% ( 172 ) 是否引证 : 否
12	浅谈云计算技术在服务业中的应用 赵红; - 《信息系统工程》 - 2012-10-20	1.8% ( 170 ) 是否引证 : 否
13	创新与安全,云计算的两只跷跷板 薛芊; - 《信息安全与通信保密》 - 2009-06-10	1.8% ( 167 ) 是否引证 : 否
14	浅析云计算 李婧; - 《科技信息》 - 2011-05-15	1.5% ( 142 ) 是否引证 : 否
15	云计算在高校教学中的应用 沈沛; - 《企业家天地(下半月刊)》 - 2014-06-23	1.5% ( 142 ) 是否引证 : 否
16	云计算在图书馆的应用初探 沈奎林;杜瑾; - 《图书情报工作》 - 2010-12-31	1.5% ( 135 ) 是否引证 : 否
17	云计算在数字图书馆应用 赵力; - 《科技信息》 - 2011-04-15	1.5% ( 135 ) 是否引证 : 否
18	云计算在中小企业的电子商务中的应用研究 姚天祥;徐运红;刘双霞; - 《电脑知识与技术》 - 2011-05-15	1.5% ( 135 ) 是否引证 : 否
19	云计算时代的数字图书馆发展探析 韦妙; - 《中国教育技术装备》 - 2011-10-15	1.5% ( 135 ) 是否引证 : 否
20	云计算与江西昌北高校图书馆联盟 李雪萍;吴青林; - 《江西图书馆学刊》 - 2012-11-30	1.5% ( 135 ) 是否引证 : 否
21	浅析云计算在故障诊断中的应用 王春键;卢文忠;刘丙杰; - 《计算机与网络》 - 2010-05-26	1.5% ( 135 ) 是否引证 : 否
22	公安云计算平台的建设探讨 吕益民; - 《中国人民公安大学学报(自然科学版)》 - 2013-02-15	1.5% ( 135 ) 是否引证 : 否
23	基于云计算技术的高校招生管理系统研究 石允剑;袁家斌; - 《中国教育信息化》 - 2013-02-10	1.5% ( 135 ) 是否引证 : 否
24	对国内IT行业在“低碳”面前尴尬处境的思考 汤慧; - 《警官文苑》 - 2010-09-15	1.5% ( 135 ) 是否引证 : 否
25	大数据与云计算 李永宏; - 《统计与管理》 - 2013-12-20	1.5% ( 135 ) 是否引证 : 否
26	云计算环境下的信息安全问题研究 刘宏;邱平; - 《中国信息界》 - 2014-01-20	1.4% ( 128 ) 是否引证 : 否
27	一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15	0.5% ( 50 ) 是否引证 : 否
28	基于Hadoop一种移动云计算本地化调度算法的研究 刘瑞祥;汤艳; - 《计算机应用与软件》 - 2015-07-15	0.4% ( 35 ) 是否引证 : 否
29	基于模式匹配和协议分析的入侵检测系统的军事应用研究 曾宪金(导师:桂卫华) - 《中南大学硕士论文》 - 2006-03-01	0.4% ( 33 ) 是否引证 : 否
30	基于多维评价模型及改进蚁群优化算法的云计算资源调度策略 张乐乾 - 《学术论文联合比对库》 - 2015-04-01	0.4% ( 33 ) 是否引证 : 否
31	无线传感器网络覆盖优化与目标定位技术研究	0.3% ( 31 )



	原文内容	相似内容来源
1	<p>此处有 33 字相似</p> <p>景和意义,明确工作定位与内容。然后说明本课题的创新点。最后将阐述本论文的组织结构。</p> <p>7.1 课题背景与意义</p> <p>近年来,</p> <p>随着互联网技术与应用的飞速兴起与发展,网络服务已经成为普通居民生活中不可或缺的一部分,不仅参与网络活动的用户数量在持续增长,单个用户的网络活动内容也在持续增多。随着用户网络活动的增多,由</p>	<p>基于模式匹配和协议分析的入侵检测系统的军事应用研究 曾宪金 - 《中南大学硕士论文》 - 2006-03-01 (是否引证：否)</p> <p>1.学位论文;学校可根据国家或湖南省有关部门规定送交学位论文。一娅导师/畔日期:一‘日★随着网络技术的快速发展,网络在成为人们日常生活一部分的同时,也逐步融入军事指挥系统这一神经中枢。信息战是现代战争的一个重要环节,它直接关系到整个战局的成败。而信息战中</p>
2	<p>此处有 85 字相似</p> <p>靠服务的分布式集群的技术也逐渐成熟,“云计算”的概念也就在此背景下传播开来。现在云计算已经广泛应用于电子商务、金融、天文</p> <p>计算等各个领域。</p> <p>云计算 ( Cloud Computing ) 是分布式计算 ( Distributed Computing ) 、并行计算 ( Parallel Computing ) 、</p> <p>网络存储 ( Network Storage Technologies ) 、虚拟化 ( Virtualization ) 、负载均衡 (</p>	<p>浅析广电行业如何应用云计算 林宗达; - 《中国有线电视》 - 2012-06-15 (是否引证：否)</p> <p>1.计算与广电行业牵手时,可能会出现什么样的情况?本文结合云计算的特点分析如何在广电行业中发展云计算的应用。1云计算1.1云计算的定义云计算(Cloud Computing):是分布式处理(Distributed Computing)、并行处理(Parallel Computing)和网格计算(Grid Computing)的发展,或者说是这些计算机科学概念的商业实现,是指基于互联网的超级计算模</p> <p>云计算在教育信息化中的应用研究 张鹏; - 《信息通信》 - 2012-12-15 (是否引证：否)</p> <p>1.教育云所需的基础资源及算法,以及云计算在教育信息化中的具体应用等多方面的研究,为教育云产业未来的推进和发展提供参考。1云计算的概念云计算(Cloud Computing)是分布式处理(Distributed Computing)、并行处理(Parallel Computing)和网格计算(Grid Computing)的发展,或者说是这些计算机科学概</p> <p>基于云计算的数字图书馆信息服务平台 陈宫;牛秦洲; - 《情报科学》 - 2012-05-05 (是否引证：否)</p> <p>1.据需要获取计算力、存储空间和各种软件服务,让用户脱离技术与部署上的复杂性而获得应用,从而较好地解决了以上的问题。云计算是分布式处理(Distributed Computing)、并行处理(Parallel Computing)和网格计算(Grid Computing)的发展,从本质上讲,云计算是指用户终端通过远程连接获取存储、计算、数据库</p> <p>基于云计算的分布式存储技术 唐京伟; - 《中国传媒科技》 - 2013-08-08 (是否引证：否)</p> <p>1.策、资金、项目等方面的大力支持。云计算的概念与特点云计算是网格计算(Grid Computing)、分布式计算(Distributed Computing)、并行计算(Parallel Computing)、效用计算(Utility Computing)、网络存储(Network Storage Technologies)</p> <p>关于云计算及其建设的探讨 张建国; - 《山西电子技术》 - 2012-10-15 (是否引证：否)</p> <p>1.存储服务器、宽带资源等等。云计算将所有的计算资源集中起来,并由软件实现自动管理,无需人为参与。从技术角度看,云计算是并行计算(Parallel Computing)、</p>

		<p>分布式计算(Distributed Computing)和网格计算(GridComputing)的发展,或者说是这些计算机科学概念的商业实现。云计算是虚拟化(Virtualization)</p> <p>基于Google的云计算实例分析 蔡键;王树梅; - 《电脑知识与技术》 - 2009-09-05 (是否引证:否)</p> <p>1.技术和产品的普及。本文介绍一些关于云计算的一些基本概念及Google提出的云计算模型及实现机制。1云计算综述云计算是并行计算(Parallel Computing)、分布式计算(Distributed Computing)和网格计算(Grid Computing)的发展,或者说是这些计算机科学概念的商业实现。云计算是虚拟化(Virtualization)、效用计</p> <p>云计算在图书馆建设与信息服务中潜在价值探析 李征; - 《大学图书馆学报》 - 2011-01-21 (是否引证:否)</p> <p>1.布在一个国家的各个地区,甚至世界其他国家和地区。云计算中的“云”字是相对互联网而言,用以比喻互联网的复杂结构。而具体讲云计算(Cloud Computing)是分布式处理(Distributed Computing)、并行处理(Parallel Computing)和网格计算(Grid Computing)的发展,或者说是这些计算机科学概</p>
3	<p>此处有 95 字相似</p> <p>并行计算 ( Parallel Computing ) 、网络存储 ( Network Storage Technologies ) 、虚拟化 ( Virtualization ) 、负载均衡 ( Load Balance ) 等传统计算机和网络技术发展融合的产物。云计算是通过使计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中,云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运</p>	<p>基于云计算的分布式存储技术 唐京伟; - 《中国传媒科技》 - 2013-08-08 (是否引证:否)</p> <p>1.ity Computing)、网络存储(Network Storage Technologies)、虚拟化(Virtualization)、负载均衡(LoadBalance)等传统计算机和网络技术发展融合的产物。广义云计算指服务的交付和使用模式,指通过网络以按需要、易扩展的方式获得服务。这种服务可以是IT软件和互联网相关服务,也可以是其他服</p> <p>实达云计算海量数据挖掘效率大大提升_hulala9876 - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - (是否引证:否)</p> <p>1.实达云计算的能力不断发展,正在让更多的企业的办公变得更加安全与便利。云计算是通过使计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中,企业数据中心的运行将与互联网更相似。这使得企业能台词会<a href="http://www.cfch.info">http://www.cfch.info</a>够将资源切换到需要的应用上</p> <p>【智慧建筑4MC之云计算Cloud&amp;Computing】 智慧运维-smartFM - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - (是否引证:否)</p> <p>1.puting) 是基于互联网的相关服务的增加、使用和交付模式,通常涉及通过互联网来提供动态易扩展且经常是虚拟化的资源。云计算是通过使计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中,企业数据中心的运行将与互联网更相似。这使得企业能够将资源切换到需要的应用上,根据需求访问计算机和存储系统。云计算特点有</p> <p>云计算在人才管理中的展望 朱建新; - 《上海企业》 - 2011-12-10 (是否引证:否)</p> <p>1.首要问题。云计算是分布式处理、并行处理和网格计算的发展,或者说是这些计算机科学概念的商业实现。云计算的基本原理是,将数据计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中,企业数据</p>

		<p>中心的运行将与互联网相似，这使得企业能够将资源切换到需要的应用上，根据需求访问计算机和存储系统。这就好比是从古老</p> <p>浅谈云计算技术在服务业中的应用 赵红;-《信息系统工程》-2012-10-20(是否引证:否)</p> <p>1.络,服务器,存储,应用软件,服务),这些资源能够被快速提供,只需投入很少的管理工作,或服务供应商进行很少的交互。通过使<b>计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中</b>,企业数据中心的运行将与互联网更相似。这使得企业能够将资源切换到需要的应用上,根据需求访问计算机和存储系统。2.云计算的特</p> <p>云计算在教育信息化中的应用研究 张鹏;-《信息通信》-2012-12-15(是否引证:否)</p> <p>1.g)和网格计算(Grid Com-puting)的发展,或者说是这些计算机科学概念的商业实现。云计算的基本原理是,通过使<b>计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中</b>,企业数据中心的运行将更与互联网相似。这使得企业能够将资源切换到需要的应用上,根据需求访问计算机和存储系统。云计算提供了最</p> <p>创新与安全,云计算的两只跷跷板 薛芊;-《信息安全与通信保密》-2009-06-10(是否引证:否)</p> <p>1.到底何谓云计算呢?通俗地讲,云计算就是庞大的服务器集群,根据用户需要动态调动资源,提供各种IT服务。云计算的基本原理是,通过使<b>计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中</b>,企业数据中心的运行将更与互联网相似。这使得企业能够将资源切换到需要的应用上,根据需求访问计算机和存储系统。“云”就是计算</p>
4	<p>此处有 147 字相似</p> <p>e)等传统计算机和网络技术发展融合的产物。云计算是通过使计算分布在大量的分布式计算机上,而非本地计算机或远程服务器中,云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。</p> <p>在云计算的领域中,</p> <p>国外的公司一直都是先行者和开拓者。谷歌、微软和亚马逊等大型互联网公司在早期就开始探索并布局云计算方面的应用。云计算服务的</p>	<p>实达云计算海量数据挖掘效率大大提升_hulala9876 -《网络(<a href="http://blog.sina.com">http://blog.sina.com</a>)》-(是否引证:否)</p> <p>1.nfo够将资源切换到需要的应用上,根据需求访问计算机和存储系统。云技术可以让用户在不同设备上体验同样的优质办公方式,云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。云计算的日益强大,为企业办公节省了硬件的投入,由于“云”的特殊容错措施可以采用极其廉价的节点来构成云,“云”的自动化集中式管理使</p> <p>浅析云计算 李婧;-《科技信息》-2011-05-15(是否引证:否)</p> <p>1.十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。再有,云计算特点在于它的虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。另外</p>

		<p>,云计算特点还有高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可</p>
		<p>云计算在高校教学中的应用 沈沛; - 《企业家天地(下半月刊)》 - 2014-06-23 (是否引证:否)</p>
		<p>1.十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。再有,云计算特点在于它的虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。另外,云计算特点还有高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可</p>
		<p>【智慧建筑4MC之云计算Cloud&amp;nbsp;Computing】 智慧运维-smartFM - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - (是否引证:否)</p>
		<p>1.“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化:云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性:“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机</p>
		<p>云计算在图书馆的应用初探 沈奎林;杜瑾; - 《图书情报工作》 - 2010-12-31 (是否引证:否)</p>
		<p>1.oo等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。②虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。③高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠。④</p>
		<p>云计算在图书馆建设与信息服务中潜在价值探析 李征; - 《大学图书馆学报》 - 2011-01-21 (是否引证:否)</p>
		<p>1.)最小化、共享效率高、个人隐私保护得到加强、用户得到个人信息控制权等[8]。所有这些,在云计算理念中完全可以解决,因云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一</p>



	<p>台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。3.8用户将享受更多全新的服务方式体验以Google为代表的搜索引擎提供给用户的“一站式”搜索体验在学术领域由于知识产权</p>
	<p>浅析广电行业如何应用云计算 林宗达; - 《中国有线电视》 - 2012-06-15 ( 是否引证 : 否 )</p> <p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化:云计算支持用户在任意位置使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性:“云”使用了数据多副本容错、负载均衡、计算节点同构可互换等措施来保障服务的高可靠性。(4)通用性:云计算</p>
	<p>云计算在数字图书馆应用 赵力; - 《科技信息》 - 2011-04-15 ( 是否引证 : 否 )</p> <p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。1.2虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。1.3高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
	<p>云计算在中小企业的电子商务中的应用研究 姚天祥;徐运红;刘双霞; - 《电脑知识与技术》 - 2011-05-15 ( 是否引证 : 否 )</p> <p>1.o等的“云”均拥有几十万台服务器,企业“云”一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。2)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。3)高可靠性和安全性。云计算提供了最可靠、最安全的数据存储中心,用户不用再担心数据丢失、病毒入侵等麻烦。在“云”的另一端</p>
	<p>云计算在人才管理中的展望 朱建新; - 《上海企业》 - 2011-12-10 ( 是否引证 : 否 )</p> <p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)灵活性。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是</p>

		<p>固定的有形的实体。应用在“云”中某处运行，但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机，就可以通过网络服务来实现我们需要的一切，甚至包括超级计算这样的任务。</p> <p>(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性，使用云计算比使用本地计算机可靠</p>
		<p>云计算时代的数字图书馆发展探析 韦妙; - 《中国教育技术装备》 - 2011-10-15 ( 是否引证：否 )</p>
		<p>1.一定的相似性,因此,“云”是一个相当贴切的比喻。通常云计算应该具备以下特征。1)基于虚拟化技术快速部署资源或获得服务。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。2)数据易于共享。不同操作系统之间的数据共享较为复杂,但对于云计算,重要的是数据本身,而不是操作系统,用户可以将</p> <p>Wind</p>
		<p>基于云计算的数字图书馆信息服务平台 陈宫;牛秦洲; - 《情报科学》 - 2012-05-05 ( 是否引证：否 )</p>
		<p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>浅谈云计算技术在服务业中的应用 赵红; - 《信息系统工程》 - 2012-10-20 ( 是否引证：否 )</p>
		<p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>关于云计算及其建设的探讨 张建国; - 《山西电子技术》 - 2012-10-15 ( 是否引证：否 )</p>
		<p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化。云计算支持用户在任意位置、使用各</p>

		<p>种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>云计算在教育信息化中的应用研究 张鹏;-《信息通信》-2012-12-15 (是否引证:否)</p>
		<p>1.轻松实现不同设备间的数据与应用共享。最后,云计算为我们使用网络提供了几乎无限多的可能。2云计算的技术优势(1)虚拟化:云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(2)高可靠性:“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>云计算与江西昌北高校图书馆联盟 李雪萍;吴青林;-《江西图书馆学刊》-2012-11-30 (是否引证:否)</p>
		<p>1.“云”也都拥有几十万台服务器。企业私有云一般拥有数百至上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一部手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>浅析云计算在故障诊断中的应用 王春键;卢文忠;刘丙杰;-《计算机与网络》-2010-05-26 (是否引证:否)</p>
		<p>1.别是一些超级计算机的大量投资,可以节约本企业资金来用于发展。(2)虚拟化。云计算对用户端的设备要求低,使用起来也方便。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。满足不同用户的需要,并不强制要求用户使用配置很高的终端,或是一定位于某个位置,同时也不必要“亲临现场”。(s)高可靠性。</p>
		<p>公安云计算平台的建设探讨 吕益民;-《中国人民公安大学学报(自然科学版)》-2013-02-15 (是否引证:否)</p>
		<p>1.等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化。云计算支持用户在任意位置、使用各</p>

		<p>种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>基于云计算技术的高校招生管理系统研究 石允剑;袁家斌; - 《中国教育信息化》- 2013-02-10 (是否引证:否)</p> <p>1.最终使用户终端简化成一个单纯的输入输出设备,并能按需享受“云”的强大计算处理能力。云计算的主要特点有:(1)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(2)高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠</p>
		<p>基于云计算的分布式存储技术 唐京伟; - 《中国传媒科技》- 2013-08-08 (是否引证:否)</p> <p>1.的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。(2)虚拟化:云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。(3)高可靠性:“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可</p>
		<p>对国内IT行业在“低碳”面前尴尬处境的思考 汤慧; - 《警官文苑》- 2010-09-15 (是否引证:否)</p> <p>1.享受“云”的低成本优势,经常只要花费几百美元、几天时间就能完成以前需要数万美元、数月时间才能完成的任务。(2)虚拟化。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。2.2PUE(数据中心的整体功耗/IT设备的功耗)求,同时也可以减轻企业的IT运维成本。从技术的角度而言,就是提高数据中</p>
		<p>大数据与云计算 李永宏; - 《统计与管理》- 2013-12-20 (是否引证:否)</p> <p>1.o等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。2、虚拟化。云计算支持用户在任意位置、使用</p>



		<p>各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。3、高可靠性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠。</p> <p>云计算环境下的信息安全问题研究 刘宏;邱平; - 《中国信息界》 - 2014-01-20 ( 是否引证 : 否 )</p> <p>1.。除以上对比的优点外,云计算还隐藏着以下几个安全性隐患:一、不确定性。用户需求的计算在何处运行具有不确定性。云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的、有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置,只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。二、高危险性。云计算服务除了提供计算服务外,还必须提供存储服务功能。如果云计算服务当前控制在私人机构(企业)手中,这对政</p> <p>创新与安全,云计算的两只跷跷板 薛芊; - 《信息安全与通信保密》 - 2009-06-10 ( 是否引证 : 否 )</p> <p>1.的“云”均拥有几十万台服务器。正是因为Google拥有超大规模的服务器,才造就了它搜索引擎全球第一的地位。二是虚拟化。云计算支持用户在任意位置使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体;应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置;只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。三是高可靠性和高可扩展性。“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地</p> <p>基于Google的云计算实例分析 蔡键;王树梅; - 《电脑知识与技术》 - 2009-09-05 ( 是否引证 : 否 )</p> <p>1.le云计算已经拥有100多万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。虚拟化:云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。高可靠性:“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性,使用云计算比使用本地计算机可靠。通用</p>
5	<p>此处有 29 字相似</p> <p>也通过搭建自己的小规模集群来处理实验数据等工作,在缩短数据处理时间和提高工作效率方面有着良好的效果。</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证 : 否 )</p> <p>1.doop资源管理、数据存储、作业调度、性能优化和安全性等方面进行了大量的研究[10-14]。随着应用的扩展</p>

	<p>随着Hadoop集群应用和规模的不断扩展，其在资源管理和作业调度上的问题也</p> <p>逐渐暴露出来，成为影响集群性能提升的主要瓶颈之一，对本问题的改进能使集群在不扩展规模和提升单机性能的基础上有比较明显的性</p>	<p>,Hadoop<b>集群的规模和结构也在不断扩大,其资源管理和集群性能问题也</b>日益受到重视。在Hadoop MRv1以前的版本中,资源调度和作业控制功能完全由Hadoop集群的主节点负责;Hadoo</p>
6	<p>此处有 61 字相似</p> <p>，提高了Hadoop集群的工作性能。</p> <p>目前，Hadoop的作业调度器作为核心组件之一，只提供了三种常用的资源分配算法：</p> <p><b>FIFO算法，Capacity调度算法，Fair调度算法。随着集群应用范围的不断扩大，系统内置的几种资源分配算法并不能满足</b></p> <p>广大用户和作业的差异化分配需求。FIFO调度算法虽然有简单易行，资源调度开销较小的优点，但是对于调度的公平性上有所不足，</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证：否)</p> <p>1.,第二层为任务级资源分配,其中第二层中任务的资源分配由用户应用程序决定。目前,Hadoop只提供了三种常用资源分配算法:<b>FIFO算法、Capacity调度算法、Fair调度算法,但随着应用范围不断扩大,这几种资源分配算法并不能满足</b>对集群性能和运行效率要求的不断提高。在Hadoop资源调度研究领域,大量学者致力于作业级的资源调度算法[15-18],即</p>
7	<p>此处有 36 字相似</p> <p>所谓作业调度就是将系统资源分配给相应的作业满足其运行需求的过程，也可以被称为资源调度。在Hadoop集群中通常会有大量的</p> <p><b>作业同时提交运行，每个作业又被分为若干个Map任务和Reduce子任务，</b></p> <p>并且作业类型和资源需求也不尽相同，在此种情况下，设计一个好的资源管理和作业调度器，对于提高集群的资源利用率，降低作业的水平</p>	<p>基于Hadoop一种移动云计算本地化调度算法的研究 刘瑞祥;汤艳; - 《计算机应用与软件》 - 2015-07-15 (是否引证：否)</p> <p>1.2 Hadoop作业请求调度算法Hadoop平台下的作业请求都作为MapReduce过程进行,称为一个MapReduce<b>作业,每个作业可分为若干个Map任务和Reduce任务。请</b>求调度算法,就是MapReduce编程模型中的作业调度算法,是管理节点Job Tracker对Map和Reduce任务</p>
8	<p>此处有 31 字相似</p> <p>传布谷鸟优化算法，在布谷鸟算法的基础上，引入遗传算法。保留布谷鸟算法的强全局搜索能力的同时，结合了遗传算法局部收敛性好的</p> <p><b>特点，加快了算法在后期的收敛速度。标准测试函数表明，本文提出的</b></p> <p>混合遗传布谷鸟算法相比布谷鸟算法，加快了后期的收敛速度，求解精度得到提高，算法的整体性能得到提升。</p> <p>二、资源调度问题是</p>	<p>无线传感器网络覆盖优化与目标定位技术研究 黄月 - 《东北大学博士论文》 - 2013-07-01 (是否引证：否)</p> <p>1.没有特殊要求，使算法应用更具有广泛性。由于PSO算法的随机搜索本质，使得它更不容易落入局部最优。同时，其基于概率<b>进化的特征又保证了算法的快速性。因此，本文提出的</b>基于粒子群优化的定位算法对求解目标定位问-65-东北大学博去学位论文 第4章基于粒子群优化算法的声巧目标</p>
9	<p>此处有 50 字相似</p> <p>明，本文提出的混合遗传布谷鸟算法相比布谷鸟算法，加快了后期的收敛速度，求解精度得到提高，算法的整体性能得到提升。</p> <p>二、</p> <p><b>资源调度问题是一个NP-Hard问题，采用传统的调度算法并不能很好的对系统资源进行分配。由于智能算法</b></p> <p>是一种有效地求解最优化问题的手段，本论文通过对Hadoop YARN的资源管理和分配机制进行深入分析，将混合遗传布谷鸟算</p>	<p>一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15 (是否引证：否)</p> <p>1.和Fair调度器.但是,由于应用的多样性,这些调度器并不能很好地满足用户合理分配资源与减低作业执行时间的需求[3,4].<b>资源调度问题是一个NP难问题,采用传统的调度算法并不能很好地分配系统资源,由于智能算法是</b>求解最优化问题的一种有效手段,为得到资源分配问题的最优解,本文采用智能优化算法设计和实现HadoopYarn资源调度器</p> <p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空</p>

		<p>航天大学硕士论文》- 2015-01-01 (是否引证:否)</p> <p>1.air调度器。但是,由于应用的多样性,这些调度器并不能很好地满足用户合理分配资源与减低作业执行时间的需求[45,46]。<b>资源调度问题是一个NP难问题,采用传统的调度算法并不能很好地分配系统资源,由于智能算法是求解最优化问题的一种有效手段,为得到资源分配问题的最优解,本章提出了一种新的Hadoop YARN资源调度方法:基于蚁</b></p> <p><b>基于多维评价模型及改进蚁群优化算法的云计算资源调度策略 张乐乾 -《学术论文联合比对库》- 2015-04-01 (是否引证:否)</b></p> <p>1.计算中的资源调度策略需要有处理动态资源的能力,对计算资源具有很强的灵活性和适应性,并及时的对用户请求进行响应。大数据中的<b>资源调度问题是一个NP完全问题,传统的资源调度算法无法很好的完成。</b>因此,要实现低成本、高效率、高易用性的云计算平台,依然面临着许多挑战和问题。§1.1.1研究意义在云计算中应用了</p>
10	<p><b>此处有 47 字相似</b></p> <p>务,提高其调度优先级,避免资源量需求较大的任务长时间得不到调度以及即将完成的任务陷入长时间等待的困局。在论文的最后进行了<b>实验验证。实验结果表明,采用本文的算法能有效地提高系统资源利用率,缩短集群的作业执行时间。</b></p> <p>7.4 论文结构</p> <p>本文主要阐述了基于布谷鸟算法的改进型混合遗传布谷鸟算法的研究与实现,并将该算法应用于Hadoop资源</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 -《南京航空航天大学硕士论文》- 2015-01-01 (是否引证:否)</p> <p>1.算法,同时根据蚁群算法全局最优解的波动趋势动态调整信息素挥发系数,采用这些方法以提高算法的收敛速度和提高算法解的精确性。<b>实验结果表明,上述方法能有效地调度Hadoop集群资源,缩短集群的作业执行时间。对比Hadoop YARN现有的调度器,无论是处理数据密集型作业还是计算密集型作业,该算法的作业集执行时间至少缩短10%</b></p>
11	<p><b>此处有 37 字相似</b></p> <p>平台发展和架构及其核心技术的介绍。首先介绍了Hadoop平台及其发展背景,并重点介绍了其核心技术和重要概念。在本章的最后<b>阐述了当前Hadoop平台内置的三种默认的调度算法,并对现有资源调度算法的不足进行了分析。</b></p> <p>第三章研究对群智能算法的改进。NP-Hard问题是不能在多项式时间内可以解决的问题,随着问题规模的增</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 -《南京航空航天大学硕士论文》- 2015-01-01 (是否引证:否)</p> <p>1.研究意义,最后介绍了本文的主要研究内容和章节安排。第二章介绍Hadoop相关知识。首先详细介绍了Hadoop的框架结构,<b>阐述了当前Hadoop平台中三种默认的调度算法,并分析了现有资源调度算法的实现方式及不足。</b>第三章研究如何合理分配资源。本章首先介绍蚁群算法与粒子群算法的实现原理,并指出算法在资源调度上优势与不足</p>

## 指 标

### 疑似剽窃观点

1. **实验结果表明,采用本文的算法能有效地提高系统资源利用率,缩短集群的作业执行时间。**

### 疑似剽窃文字表述

1. 计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”,而不是固定的有形的实体。应用在“云”中某处运行,但实际上用户无需了解、也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机,就可以通过网络服务来实现我们需要的一切,甚至包括超级计算这样的任务。  
在云计算的领域中,
2. FIFO算法, Capacity调度算法, Fair调度算法。随着集群应用范围的不断扩大,系统内置的几种资源分配算法并不能满足

3. 资源调度问题是一个NP-Hard问题，采用传统的调度算法并不能很好的对系统资源进行分配。由于智能算法

## 8. 80022497643220302\_基于动态手势的智能终端身份识别技术研究\_第8部分 总字数：9421

相似文献列表 文字复制比：22.2%(2090) 疑似剽窃观点：(0)

1	03700_1120379028_赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28	8.4% ( 796 ) 是否引证：否
2	Hadoop环境下的动态资源管理研究 赵颖(导师：吴刚) - 《上海交通大学硕士论文》 - 2015-01-14	6.9% ( 650 ) 是否引证：否
3	BH20109263109_李孟庭 李孟庭 - 《学术论文联合比对库》 - 2013-01-05	5.7% ( 541 ) 是否引证：否
4	李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》 - 2013-01-15	5.5% ( 518 ) 是否引证：否
5	HADOOP调度算法及其改进策略研究 李盼(导师：余重秀) - 《北京邮电大学硕士论文》 - 2013-01-14	4.9% ( 457 ) 是否引证：否
6	异构环境下MapReduce离线调度算法的研究 陈珩(导师：过敏意) - 《上海交通大学硕士论文》 - 2014-01-01	4.3% ( 403 ) 是否引证：否
7	BH20099257125_卡哇伊 卡哇伊 - 《学术论文联合比对库》 - 2012-12-28	4.0% ( 377 ) 是否引证：否
8	马鹏将 马鹏将 - 《学术论文联合比对库》 - 2015-01-04	1.3% ( 123 ) 是否引证：否
9	基于Hadoop的文本分类研究 刘丛山(导师：杨煜普) - 《上海交通大学硕士论文》 - 2012-02-01	0.7% ( 69 ) 是否引证：否
10	通信网云计算平台资源调度策略与算法研究 汪洋(导师：陈海) - 《南昌大学硕士论文》 - 2013-05-28	0.7% ( 69 ) 是否引证：否
11	基于HDFS的电子文件集中存储和检索系统 张宸(导师：谢立;茅兵) - 《南京大学硕士论文》 - 2012-05-01	0.7% ( 67 ) 是否引证：否
12	20121236535-王德龙-Hadoop平台下作业调度算法的研究与改进 王德龙 - 《学术论文联合比对库》 - 2015-04-07	0.6% ( 60 ) 是否引证：否
13	主资源公平调度算法在YARN中应用研究 陈昕;刘朝晖; - 《电脑知识与技术》 - 2016-04-12 1	0.6% ( 56 ) 是否引证：否
14	基于Hadoop云平台的数据挖掘技术在天气数据的应用研究 杨利娟(导师：张永军) - 《北京邮电大学硕士论文》 - 2015-01-04	0.4% ( 40 ) 是否引证：否
15	基于hadoop的地震数据分布式存储策略的研究 冯翔(导师：文必龙) - 《东北石油大学硕士论文》 - 2014-06-08	0.4% ( 38 ) 是否引证：否
16	基于Hadoop的海量数据管理系统 多雪松;张晶;高强; - 《微计算机信息》 - 2010-05-05	0.4% ( 34 ) 是否引证：否
17	基于HADOOP的数据挖掘研究 杨宸铸(导师：向宏) - 《重庆大学硕士论文》 - 2010-11-01	0.4% ( 34 ) 是否引证：否
18	云计算背景下基于FPGA的文件管理系统与Web缓存的紧耦合研究与分析 孔雪(导师：祝永新) - 《上海交通大学硕士论文》 - 2011-12-28	0.3% ( 32 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 49 字相似</p> <p>一种分治的思想，将数据处理分散到各个计算节点。该计算模型封装了简单的数据处理接口，屏蔽了处理分布式计算所需要注意的细节，</p> <p>用户只需要实现两个简单的函数Map和Reduce，即可高效的实现分布式计算。用户无需关心框架是如何处理数据在各个节点上的存储和在节点间的复制或移动，如何对单个任务或计算节点的异常进行处理，极大地节省了开发人员的精力。</p>	<p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士论文》 - 2014-01-01 ( 是否引证：否 )</p> <p>1.的 MapReduce 计算框架就是 Google 为了解决这一问题而提出的。该计算模型封装出了简单的接口，用户只需实现简单的两个函数：map 和 reduce，就可以实现高效的分布式计算，而无需关心系统如何分布式存放输入数据，如何进行并行计算，如何进行异常处理。MapReduce 计算模型的处理对象是以 &amp;lt;key</p>
2	<p>此处有 73 字相似</p> <p>不会因为格式问题陷入僵局。MapReduce框架只需用</p>	<p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士论文》 - 2014-01-01 ( 是否引证：否 )</p> <p>1.y, value&gt; 形式存储。MapReduce 框架的用户只需</p>



	<p>户实现两个自定义函数map和reduce，分别对数据进行两个规约， 即可以并行的方式处理输入数据集。通常，map函数负责处理输入的&lt;key,value&gt;键值对，生成一系列以&lt;key,value&gt;形式存储的中间结果。 再经过shuffling过程，将相同key值的键值对分配到一个计算节点，在此过程中，MapReduce框架不对程序运行中</p>	<p>实现两个用户自定义函数 map 和 reduce 即可以并行的方式处理所有的输入数据。其中 map 函数会处理输入的 &lt;key, value&gt; 对，并生成一系列以 &lt;key, value&gt; 形式存储的中间结果。MapReduce 会根据中间结果的 key 值，将拥有相同 key 值的 &lt;key, value&gt; 合并交给 r</p>
3	<p>此处有 47 字相似</p> <p>节点，在此过程中，MapReduce框架不对程序运行中间结果做更改。最后由reduce程序对shuffling的结果进行处理并产生输出文件。在以上过程中，MapReduce框架会自动将输入文件划分为M份小的输入文件存入HDFS文件系统，同时产生M个map任务待调度执行。在资源充足的集群中，map任务可以在M个计算节点上并行运行，生成</p>	<p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士论文》- 2014-01-01 (是否引证：否)</p> <p>1.给 reduce 函数处理。reduce 函数也由用户定义，其接受拥有相同 key 值的中间结果，用户可以根据需要处理输入数据产生输出文件。MapReduce 框架会自动将输入文件划分为 M 份小的输入文件，M 个 map 任务——</p> <p>11 —异构环境下 MAPREDUCE 离线调度算法的研究 上海交通大学硕士学位论文</p>
4	<p>此处有 48 字相似</p> <p>作的详细流程描述：(1) 首先用户需要实现数据处理所需要的map函数和reduce函数，之后连同数据集一起提交到集群。(2 ) 当用户提交任务并启动之后，MapReduce框架会将输入数据集按照一定的大小划分为M个小文件，用户可以按照需求自行配置划分文件的大小。(3) 在MapReduce集群系统中，节点机器会被划分为两类不同职责的机器，一类</p>	<p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士论文》- 2014-01-01 (是否引证：否)</p> <p>1.educe 函数，处理一个输入文件时，MapReduce 框架按如下过程执行(图中的数字对应于下文中的步骤)：(1) 当用户启动一个 MapReduce 程序时，MapReduce 框架首先将输入文件划分成 M 个小文件，文件的大小可以由用户根据需要配置。而后，系统将在机器中的多个机器中运行该 MapReduce程序。(2) 在</p>
5	<p>此处有 123 字相似</p> <p>并启动之后，MapReduce框架会将输入数据集按照一定的大小划分为M个小文件，用户可以按照需求自行配置划分文件的大小。 (3) 在MapReduce集群系统中，节点机器会被划分为两类不同职责的机器，一类是控制节点，在一般情况下由一台机器来执行控制节点的职能；其余的所有机器被划为另一类，即工作节点。控制节点的主要任务是将map和Reduce任务分配给工作节点。当其发现有资源闲置并且有待调度的map或者reduce任务存在时，控制节点会给待调度任务分配运算资源以运行任务。这样，同一个输入</p>	<p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士论文》- 2014-01-01 (是否引证：否)</p> <p>1.文件，文件的大小可以由用户根据需要配置。而后，系统将在机器中的多个机器中运行该 MapReduce程序。(2) 在 MapReduce 系统中，上文中运行的机器可以被划分为两类，一类是控制节点，通常只有一台机器充当控制节点。剩下的所有机器是另一类，称之为工作节点。控制节点主要负责分配 map 或 reduce 任务给工作节点。当发现有机器空闲且有 map 或 reduce 任务待分配时，控制节点会选择一个任务分配给工作节点。(3) 对于每个被分配</p>
6	<p>此处有 111 字相似</p> <p>推送给用户。通常来讲，输出结果会是R份文件，这R份文件可以简单合并，也可以作为下一个MapReduce任务的输入直接写入文件系统。 8.3 Hadoop的分布式文件系统HDFS Hadoop的数据存储系统是HDFS (Hadoop Distribute</p>	<p>李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》- 2013-01-15 (是否引证：否)</p> <p>1.点研究这两项关键技术的工作原理，分析现有的Hadoop 架构缺陷，提出改进策略。3.1 Hadoop 分布式文件系统Hadoop 分布式文件系统 (Hadoop Distributed File System, HDFS) 是 Hadoop云计算平台的一项关键技术，用于提供高吞吐率、高传输速率、高容错性和高扩展性的数据存储服务，它是 Google 的 GFS 技术</p>

	<p>File System ) , 它也是Hadoop云计算平台的一项核心技术。HDFS能提供高吞吐量的数据存储访问服务, 作为大数据处理集上的分布式存储系统来说非常有优势。HDFS的高吞吐率是通过“一次写入多次读取”</p>	<p>的开源实现。3.1.1 H</p> <p>基于Hadoop的文本分类研究 刘丛山 - 《上海交通大学硕士学位论文》- 2012-02-01 ( 是否引证 : 否 )</p> <p>1.据作了基础准备。 2.2.2 Hadoop 生生生态态系系统 Hadoop 的基本组成部分是 <b>Hadoop 分布式文件系统 ( Hadoop Distributed File System, HDFS )</b> 和并行编程范式 MapReduce。这两部分实际上是 Google 的 GFS 和MapReduc</p> <p>通信网云计算平台资源调度策略与算法研究 汪洋 - 《南昌大学硕士学位论文》- 2013-05-28 ( 是否引证 : 否 )</p> <p>1.11第3章 Hd aoop概要 第3章Hadoop概要3.1 HDFS文件系统HDFS (基于 <b>Hadoop 的分布式文件系统 , Hadoop Distirbuted File System</b>)是一套已经成功应用于Hadoop框架的文件系统。与GFS (谷歌文件系统 ) 类似, 也是为了实现云框架而设计 ,</p> <p>异构环境下MapReduce离线调度算法的研究 陈珩 - 《上海交通大学硕士学位论文》- 2014-01-01 ( 是否引证 : 否 )</p> <p>1.p 公共模块 -HadoopCommon, 主要提供一些供其他 Hadoop 模块调用的公共类库及组件; ( 2 ) <b>Hadoop 分布式文件系统 -Hadoop Distributed File System ( HDFS )</b> , 主要负责存储 MapReduce 模块将要处理的输入数据和其运行中产生的中间数据和结果 , HDFS 为 Had</p>
7	<p>此处有 67 字相似</p> <p>8.3.1 HDFS的工作机制</p> <p>HDFS被设计成一种Master/Slaver结构 ( 即主从结构 ) , 一个典型的系统包括</p> <p>一个名字节点 ( NameNode ) 和N个数据节点 ( DataNode ) , 如图2.4所示, NameNode中存储有包含完整的文件系统命名空间和文件块映射表的元数据, 是文件系统的控制和监控中心; 文件数据则按照一定的规则存在诸多DataNode上, 同时DataNo</p>	<p>基于HDFS的电子文件集中存储和检索系统 张宸 - 《南京大学硕士学位论文》- 2012-05-01 ( 是否引证 : 否 )</p> <p>1.群中的读取负载。NameNode 和 DataNodeHDFS集群有两类节点, 并以管理者-工作者模式运行, 即一个 <b>NameNode (管理者)</b> 和多个<b>DataNode (工作者)</b>。NameNode负责管理文件系统的命名空间。它维护着文件系统中文件和目录的树型结构, 并在本地磁盘上以两个文件形式永久保存这些信息: 命名空间镜像文件和编辑日志文</p>
8	<p>此处有 35 字相似</p> <p>了因为节点故障而导致数据丢失的风险, 为 MapReduce程序的运行提供了良好的基础。</p> <p>Hadoop的文件目录管理结构是</p> <p><b>传统的层次型组织结构, 文件系统命名空间与现在常见的大多数文件系统相像,</b></p> <p>符合用户的使用习惯, 用户可以较为方便的移动、创建和删除文件。不过Hadoop不支持对用户访问文件权限的控制, 也不支持多用</p>	<p>云计算背景下基于FPGA的文件管理系统与Web缓存的紧耦合研究与分析 孔雪 - 《上海交通大学硕士学位论文》- 2011-12-28 ( 是否引证 : 否 )</p> <p>1.就不需要通过 Namenode ( 也就是说文件数据的读写是直接 在 Datanode 上 )。HDFS 支持<b>传统的层次型文件组织, 与大多数其他文件系统</b>类似: 用户可以创建目录, 并在其间创建、删除、移动和重命名文件。Namenode 负责维护文件系统的 na</p>
9	<p>此处有 36 字相似</p> <p>名空间的修改都以日志的方式记录在本地磁盘。</p> <p>Editlog事务日志即是用于记录HDFS系统对文件操作记录的, Fslmage</p> <p><b>文件则存有整个文件系统的命名空间, 以及数据块与文</b></p>	<p>基于Hadoop的海量数据管理系统 多雪松;张晶;高强; - 《微计算机信息》- 2010-05-05 ( 是否引证 : 否 )</p> <p>1.3HDFS架构2)文件系统元数据任何对文件系统元数据产生修改的操作,管理节点都会使用编辑日志(Editlog)记录下来。 <b>文件系统的名字空间,包括数据块到文件的映射</b></p>

	<p>件的映射、文件的属性等。</p> <p>当客户端发起任何此类操作，NameNode会通过记录的映射关系确定操作的数据块或文件的位置，进而根据用户的需求，进行相应</p>	<p>、文件的属性等,都存储在映像文件(FsImage)中。内存中保存着整个文件系统的名字空间和文件数据块映射(Blockmap)的映像。当管</p> <p>基于HADOOP的数据挖掘研究 杨宸铸 -《重庆大学硕士论文》- 2010-11-01 (是否引证：否)</p> <p>1. EditLog 的事务日志记录下来。Namenode在本地操作系统的文件系统中存储这个 Editlog。整个文件系统的名字空间，包括数据块到文件的映射、文件的属性等，都存储在一个称为 FsImage 的文件中，这个文件也是放在 Namenode 所在的本地文件系统上。</p>
10	<p>此处有 41 字相似</p> <p>de上的存储信息。该文件系统会定期或在启动时将事务日志中记录的文件操作与元数据进行合并，生成新的FsImage保存到本地</p> <p>文件系统并对旧的日志进行清除，这个过程称为一个检查点 ( checkpoint )。</p> <p>8.3.2 HDFS的可靠性分析</p> <p>Hadoop分布式文件系统 ( HDFS ) 可以在大量廉价的通用硬件所组成的集群上可靠运行</p>	<p>基于hadoop的地震数据分布式存储策略的研究 冯翔 -《东北石油大学硕士论文》- 2014-06-08 (是否引证：否)</p> <p>1.了一定的帮助。二级名称节点主要用来保存名称节点中元数据的备份，会周期性的从名称节点上进行镜像文件的下载和日志文件的合并，这个过程被称为是一次检查点 ( checkpoint )，得到新的镜像文件后会自动的上传到系统的名称节点中，替换掉原来名称节点中的镜像文件，使 Namenode 上的镜像文</p>
11	<p>此处有 50 字相似</p> <p>领域故障的发生也被认为是常态而不是异常，在HDFS设计与实现时就对故障的发生原因和可能导致的结果做了充分的考虑和准备，在</p> <p>系统中设计和实现了较为完善的错误处理机制来保证系统能提供可靠的存储服务。</p> <p>( 1 ) 数据块冗余机制</p> <p>副本的存放是HDFS高可靠性和高读取性能的关键。在HDFS中，文件不是以整体存放于数据节点，而是被分割成大小相等的数据块</p>	<p>BH20109263109 李孟庭 李孟庭 -《学术论文联合比对库》- 2013-01-05 (是否引证：否)</p> <p>1.以运行在大量普通硬件构成的集群上，在这样的条件下，硬件故障的发生率很高。HDFS在设计实现时充分考虑到了各种故障因素，为系统增加了比较完善的故障处理机制，确保了系统的可靠性。1) 数据块冗余机制 HDFS中文件被切割开来，以数据块的形式存储在不同的节点上。当存储着文件数据块的节点发生故障时，文件将无法访问。为了避免</p> <p>HADOOP调度算法及其改进策略研究 李盼 -《北京邮电大学硕士论文》- 2013-01-14 (是否引证：否)</p> <p>1.普通硬件构成的集群上，在这样的条件下，硬件故障的发生率很高。HDFS在设计实现时充分考虑到了各种故障因素，为系统增加了比较完善的故障处理机制，确保了系统的可靠性。?数据块冗余机制.... HDFS中文件被切割开来，以数据块的形式存储在不同的节点上。当存储着文件数据块的节点发生故障时，文件将无法访问。为</p>
12	<p>此处有 29 字相似</p> <p>自动进行复制备份，分散其到其他运行良好的数据节点中。</p> <p>大型的HDFS集群往往布置在多个机架上，甚至有异地的数据中心，在</p> <p>同一机架上的数据节点的通讯效率远远高于位于不同机架的节点。</p> <p>然而由于不同机架同时发生故障的概率更为微小，为了避免在单个机架出现断电等故障导致服务无法继续运行，将数据块的副本备份到不</p>	<p>BH20099257125 卡哇伊 卡哇伊 -《学术论文联合比对库》- 2012-12-28 (是否引证：否)</p> <p>1.小的副本数目时，系统将对此数据块启动复制过程，以增加其副本数。一般情况下，HDFS运行在由多个机架组成的集群上，而同一个机架上的机器之间的通信效率要远高于不同机架间的通信效率。为了避免机器或机架故障时数据资源无法访问，在存放数据块副本时，要采取一定的策略。最简单的想法是将不同的副本放置到不同的机</p> <p>BH20109263109 李孟庭 李孟庭 -《学术论文联合比对库》- 2013-01-05 (是否引证：否)</p>



		<p>1.小的副本数目时，系统 will 对此数据块启动复制过程，以增加其副本数。一般情况下，HDFS运行在由多个机架组成的集群上，而<b>同一个机架上的机器之间的通信效率要远高于不同机架间的通信效率</b>。为了避免机器或机架故障时数据资源无法访问，在存放数据块副本时，要采取一定的策略。最简单的想法是将不同的副本放置到不同的机</p> <p>HADOOP调度算法及其改进策略研究 李盼 - 《北京邮电大学硕士论文》 - 2013-01-14 (是否引证：否)</p> <p>1.本数目时，系统 will 对此数据块启动复制过程，以增加其副本数。一般情况下，HDFS运行在由多个机架组成的集群上，而<b>同一个机架上的机器之间的通信效率要远高于不同机架间的通信效率</b>。为了避免机器或机架故障时数据资源无法访问，在存放数据块副本时，要采取一定的策略。最简单的想法是将不同的副本放</p> <p>李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》 - 2013-01-15 (是否引证：否)</p> <p>1.数目时，系统 will 对此数据块启动复制过程，以增加其副本数。一般情况下，HDFS 运行在由多个机架组成的集群上，而<b>同一个机架上的机器之间的通信效率要远高于不同机架间的通信效率</b>。为了避免机器或机架故障时数据资源无法访问，在存放数据块副本时，要采取一定的策略。最简单的想法是将不同的副本放</p>
13	<p>此处有 45 字相似</p> <p>机制保证了系统存在故障节点时的可靠性，同时系统发现故障节点之后能够快速恢复数据块的冗余程度。而系统对故障节点的发现依赖于</p> <p><b>心跳通信机制。在HDFS中，NameNode与DataNode通过心跳机制来进行信息交流。</b></p> <p>每隔固定的时间，DataNode将会发送一个心跳数据包，汇报本节点当前的状态。NameNode也会定期查看处理各Data</p>	<p>BH20099257125_卡哇伊 卡哇伊 - 《学术论文联合比对库》 - 2012-12-28 (是否引证：否)</p> <p>1.，HDFS尽量选择离本地较近的副本。这样的策略既增加了数据的可靠性，同时减少了写文件带来的机架间的数据传输。2) <b>心跳通信机制在HDFS中NameNode与DataNode通过心跳的机制进行交流。</b> DataNode端执行一个简单的循环，每隔一定的时间向NameNode发送信息，汇报自己的工作状态。如果NameNode</p> <p>BH20109263109_李孟庭 李孟庭 - 《学术论文联合比对库》 - 2013-01-05 (是否引证：否)</p> <p>1.时，HDFS尽量选择离本地较近的副本。这样的策略既增加了数据的可靠性，同时减少了写文件带来的机架间的数据传输。2) <b>心跳通信机制在HDFS中NameNode与DataNode通过心跳的机制进行交流。</b> DataNode端执行一个简单的循环，每隔一定的时间向NameNode发送信息，汇报自己的工作状态。如果NameNode</p> <p>HADOOP调度算法及其改进策略研究 李盼 - 《北京邮电大学硕士论文》 - 2013-01-14 (是否引证：否)</p> <p>1.HDFS尽量选择离本地较近的副本。这样的策略既增加了数据的可靠性，同时减少了写文件带来的机架间的数据传输。<b>心跳通信机制在HDFS中NameNode与DataNode通过心跳的机制进行交流。</b> DataNode端执行一个简单的循环，每隔一定的时间向NameNode发送信息，汇报自己的工作状态。如果Na</p>



		<p>李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》- 2013-01-15 (是否引证:否)</p> <p>1.DFS 尽量选择离本地较近的副本。这样的策略既增加了数据的可靠性,同时减少了写文件带来的机架间的数据传输。<b>心跳通信机制在 HDFS 中 NameNode 与 DataNode 通过心跳的机制进行交流。</b>DataNode端执行一个简单的循环,每隔一定的时间向 NameNode 发送信息,汇报自己的工作状态。如果</p>
14	<p>此处有 47 字相似</p> <p>。而系统对故障节点的发现依赖于心跳通信机制。在 HDFS中,NameNode与DataNode通过心跳机制来进行信息交流。</p> <p><b>每隔固定的时间,DataNode将会发送一个心跳数据包,汇报本节点当前的状态。NameNode</b></p> <p>也会定期查看处理各DataNode的心跳包,如果NameNode发现心跳信号缺失,并在规定时间内没有再与之建立连接,则认</p>	<p>BH20099257125_卡哇伊 卡哇伊 - 《学术论文联合比对库》- 2012-12-28 (是否引证:否)</p> <p>1.信机制在HDFS中NameNode与DataNode通过心跳的机制进行交流。DataNode端执行一个简单的循环,<b>每隔一定的时间向NameNode发送信息,汇报自己的工作状态。如果NameNode</b>在规定的时间内没有接收到某个DataNode传递过来的心跳信息,则判定DataNode所在的节点为死亡节点。该节点上所有的</p> <p>BH20109263109_李孟庭 李孟庭 - 《学术论文联合比对库》- 2013-01-05 (是否引证:否)</p> <p>1.信机制在HDFS中NameNode与DataNode通过心跳的机制进行交流。DataNode端执行一个简单的循环,<b>每隔一定的时间向NameNode发送信息,汇报自己的工作状态。如果NameNode</b>在规定的时间内没有接收到某个DataNode传递过来的心跳信息,则判定DataNode所在的节点为死亡节点。该节点上所有的</p> <p>HADOOP调度算法及其改进策略研究 李盼 - 《北京邮电大学硕士论文》- 2013-01-14 (是否引证:否)</p> <p>1.在HDFS中NameNode与DataNode通过心跳的机制进行交流。DataNode端执行一个简单的循环,<b>每隔一定的时间向NameNode发送信息,汇报自己的工作状态。如果NameNode</b>在规定的时间内没有接收到某个DataNode传递过来的心10北京邮电人学硕士学位论文跳信息,则判定D</p> <p>李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》- 2013-01-15 (是否引证:否)</p> <p>1.DFS 中 NameNode 与 DataNode 通过心跳的机制进行交流。DataNode端执行一个简单的循环,<b>每隔一定的时间向 NameNode 发送信息,汇报自己的工作状态。如果 NameNode</b> 在规定的时间内没有接收到某个DataNode 传递过来的心跳信息,则判定 DataNode 所在的节点为死亡节点。</p>
15	<p>此处有 37 字相似</p> <p>统从中取出数据块并对其进行校验,将校验结果与预存的校验和进行比对,比对无误则说明数据块无损坏,如果比对失败,则系统将会向</p> <p><b>其他节点发起副本传输的请求。</b></p> <p><b>(4) 元数据冗余机制</b></p> <p><b>在HDFS中,数据</b></p> <p>可靠性的保证十分依赖于元数据存储文件FsImage和事务日志Editlog这两个核心文件。如果这两个文件发生损坏,则整个</p>	<p>BH20109263109_李孟庭 李孟庭 - 《学术论文联合比对库》- 2013-01-05 (是否引证:否)</p> <p>1.出文件数据块时,它首先计算该数据块的校验和并与预先保存的校验和相比较,如果匹配失败,客户端则向其他的工作节点索取该数据块<b>另外的副本。</b>4) <b>元数据冗余机制HDFS中元数据</b>的同步与可靠性是由镜像文件和事务日志共同保障的。然而,当镜像文件和事务日志发生损坏时,将严重影响HDFS的工作。针对这样的</p>

16	<p>此处有 40 字相似</p> <p>kTracker是MRv1中的另一个重要的组件，其在每台计算节点上均有一个实例，它的主要工作就是监控所在机器的资源现状及任务的运行情况，并定时向JobTracker进行汇报。由此可见，JobTracker不仅要管理系统内所有的计算资源，还要兼顾作业调度的实现，在MapReduce任务非常多的时候，会对JobTracker</p>	<p>基于Hadoop云平台的数据挖掘技术在天气数据的应用研究 杨利娟 - 《北京邮电大学硕士论文》 - 2015-01-04 (是否引证：否)</p> <p>1.一方面，需要执行的各种命令通过与JobTracker进行通信而获得；另一方面，及时地通过心跳机制将数据节点当前作业任务的执行情况向JobTracker汇报。它与JobTracker和Task之间采用了RPC协议进行通信。一个集群中可以有多个TaskTracker。(4)分布式文件系</p>
17	<p>此处有 50 字相似</p> <p>duce管理框架YARN ( Yet Another Resource Negotiator ) 诞生了。YARN的一个重大改变就是将MRv1中的JobTracker的两个主要功能，即资源管理和作业控制（包括作业监控、容错等），拆分为两个单独的守护进程进行运行管理，这两个进程分别是一个全局的资源管理器（Resource Manager，RM）和每</p>	<p>03700_1120379028_赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证：否)</p> <p>1.源模型以及向后兼容性。其中提高集群资源利用率是YARN的首要需求。YARN相比于MRv1中最大的变化就是将MRv1中JobTracker的两个主要功能，即资源管理和作业控制（包括作业监控、容错等），分拆成两独立的进程[14]，这样YARN只需要负责实现资源管理的部分，而作业控制则由各个应用程序自行管理。</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士论文》 - 2015-01-14 (是否引证：否)</p> <p>1.框架多样性,灵活的资源模型以及向后兼容性。其中提高集群资源利用率是YARN的首要需求。YARN相比于MRv1中最大的变化就是将MRv1中JobTracker的两个主要功能,即资源管理和作业控制(包括作业监控、容错等),分拆成两独立的进程[14],这样YARN只需要负责实现资源管理的部分,而作业控制则由各个应用程序自行管理。</p> <p>2.2.1 Y</p>
18	<p>此处有 158 字相似</p> <p>分，大大降低了MapReduce任务增多时对主节点的压力，从而使得集群能够容纳更多的计算节点和承载更多的作业运行，这也是YARN的基本设计思想。</p> <p>YARN的基本架构如图2.5所示，它由一个资源管理器RM和多个节点管理器（Node Manager）组成，节点管理器在每个计算节点上都会运行，因此YARN在总体架构上仍然与第一代相同，是一个Master/Slaver结构，资源管理器RM为Master角色，节点管理器NM为Slaver，在Master与Slaver之间通过心跳机制来传递信息，资源管理器通过了解汇总各节点资源的使用和空闲情况，结合待调度任务</p>	<p>03700_1120379028_赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证：否)</p> <p>1.[14]，这样YARN只需要负责实现资源管理的部分，而作业控制则由各个应用程序自行管理。2.2.1 YARN的架构 YARN的基本架构如图2-1所示，它由一个资源管理器（ResourceManager）和多个节点管理器（NodeManager）组成，节点管理器在每个节点上都会运行。在第一代MapReduce中，由JobTracker负责所有资源的管理、分配和任务调度，YARN在设计时采用了两级</p> <p>2.一个应用特有的ApplicationMaster(AM)负责，负责对自己的子任务进行资源分配和调度。YARN总体上仍然是一个Master/Slave结构，在整个资源管理架构中，资源管理器为Master，节点管理器为Slave，它们通过心跳来维持状态和发送消息，资源管理器负责对每个节点管理器上的资源进行统一调度和 — 8 — 上海</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士论文》 - 2015-01-14 (是否引证：否)</p> <p>1.,分拆成两独立的进程[14],这样YARN只需要负责实现资源管理的部分,而作业控制则由各个应用程序自行管理。2.2.1 YARN的架构YARN的基本架构如图2-1所示,它由一个资源管理器(ResourceManager)和多个节点管理</p>

		<p>器(NodeManager)组成,节点管理器在每个节点上都会运行。在第一代MapReduce中,由JobTracker负责所有资源的管理、分配和任务调度,YARN在设计时采用了两级调度策</p> <p>2.,第二层调度则交给每一个应用特有的ApplicationMaster(AM)负责,负责对自己的子任务进行资源分配和调度。<b>YARN总体上仍然是一个Master/Slave结构,在整个资源管理架构中,资源管理器为Master,节点管理器为Slave</b>,它们通过心跳来维持状态和发送消息,资源管理器负责对每个节点管理器上的资源进行统一调度和——8——图2-1 YARN基本架构[</p>
19	<p>此处有 32 字相似</p> <p>er/Slaver结构,资源管理器RM为Master角色,节点管理器NM为Slaver,在Master与Slaver之间<b>通过心跳机制来传递信息,资源管理器通过了解汇总各节点资源的使用和</b>空闲情况,结合待调度任务的需求来分配资源和进行任务调度。对比在MRv1架构设计中,由JobTracker负责所有节点资</p>	<p>03700_1120379028_赵颖_赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证:否)</p> <p>1. 止,并更新调度器内的集群资源记录,然后向AM返回已经为应用程序分配好的容器记录。2. 节点管理器<b>通过心跳向资源管理器汇报节点的资源使用情况和容器的运行状态</b>,如果资源管理器发现节点上有符合条件的空闲资源,则进行资源分配,并将分配好的资源保存下来,</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士学位论文》 - 2015-01-14 (是否引证:否)</p> <p>1.要释放的容器列表,将容器停止,并更新调度器内的集群资源记录,然后向AM返回已经为应用程序分配好的容器记录。2.节点管理器<b>通过心跳向资源管理器汇报节点的资源使用情况和容器的运行状态</b>,如果资源管理器发现节点上有符合条件的空闲资源,则进行资源分配,并将分配好的资源保存下来,等待AM下一次心</p>
20	<p>此处有 51 字相似</p> <p>源管理器通过了解汇总各节点资源的使用和空闲情况,结合待调度任务的需求来分配资源和进行任务调度。对比在MRv1架构设计中,<b>由JobTracker负责所有节点资源的管理、分配和任务调度,YARN框架在设计中采用了两级调度策略</b>:第一层调度由资源管理器RM管理执行,管理所有系统的资源并进行调度分配;第二层调度交由每个应用对应的应用管理器AM来执行</p>	<p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士学位论文》 - 2015-01-14 (是否引证:否)</p> <p>1.ager)和多个节点管理器(NodeManager)组成,节点管理器在每个节点上都会运行。在第一代MapReduce中,<b>由JobTracker负责所有资源的管理、分配和任务调度,YARN在设计时采用了两级调度策略</b>,也即把这些繁重的任务分为两部分,第一层调度由资源管理器负责,统管资源分配和管理,第二层调度则交给每一个应用特有的Appl</p> <p>03700_1120379028_赵颖_赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证:否)</p> <p>1.个节点管理器(NodeManager)组成,节点管理器在每个节点上都会运行。在第一代MapReduce中,<b>由JobTracker负责所有资源的管理、分配和任务调度,YARN在设计时采用了两级调度策略</b>,也即把这些繁重的任务分为两部分,第一层调度由资源管理器负责,统管资源分配和管理,第二层调度则交给每一个应用特</p>
21	<p>此处有 225 字相似</p> <p>调度由资源管理器RM管理执行,管理所有系统的资源并进行调度分配;第二层调度交由每个应用对应的应用管理器AM来执行调度,将<b>申请到的资源分配给自己的子任务,同时AM也需要通</b></p>	<p>03700_1120379028_赵颖_赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证:否)</p> <p>1.己内部程序资源的ApplicationMaster(AM),由AM负责向YARN的资源管理器申请资源,并把<b>申请到的资源分配给自己的子任务,同时它需要通知已分配到的资源所在节点的节点管理器来启动相应的任务</b>。我们将详细</p>

	<p>知对应资源所在节点的管理器NM来启动对应的任务进行运行。接下来将详细讲述此三个YARN的重要组成部分。</p> <p>8.4.1 资源管理器 (Resouce Manager, RM)</p> <p>在YARN资源管理框架中，资源管理器RM拥有系统级别的资源管理和分配的全部权限，它由两大重要部分构成：调度器(Scheduler)和应用程序管理器(ApplicationsManager)。调度器是RM实现资源调度的核心组件，</p> <p>它会根据集群的资源容量、作业队列、任务优先级等约束条件，并依据相应的策略完成将集群资源分配给各个应用程序的过程。在RM中</p>	<p>介绍 YARN 这几个基本组成部分。2.2.1.1 资源管理器 (ResourceManager) 在 YARN 中，所有系统层面的资源管理和分配都是由资源管理器完成的。它主要由两个组件构成：调度器 (Scheduler) 和应用程序管理器 (Applicat</p> <p>2.ResourceManager) 在 YARN 中，所有系统层面的资源管理和分配都是由资源管理器完成的。它主要由两个组件构成：调度器 (Scheduler) 和应用程序管理器 (Applications-Manager)。调度器是资源管理器的核心组件之一，是真正负责对全局资源进行调度和分配的地方。它会根据集群资源容量、队列设置、优先级等不同情况，采取一定的策略将</p>
		<p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士论文》 - 2015-01-14 ( 是否引证：否 )</p> <p>1.于跟踪和管理自己内部程序资源的</p> <p>ApplicationMaster(AM),由AM负责向YARN的资源管理器申请资源,并把申请到的资源分配给自己的子任务,同时它需要通知已分配到的资源所在节点的节点管理器来启动相应的任务。我们将详细介绍YARN这几个基本组成部分。2.2.1.1资源管理器(ResourceManager)在YARN中,所有系统层面的资源管理和分配都是由资源管理器完成的。它主要由两个组件构成:调度器(Schedule</p> <p>2.1.1资源管理器(ResourceManager)在YARN中,所有系统层面的资源管理和分配都是由资源管理器完成的。它主要由两个组件构成:调度器(Scheduler)和应用程序管理器(Applications-Manager)。调度器是资源管理器的核心组件之一,是真正负责对全局资源进行调度和分配的地方。它会根据集群资源容量、队列设置、优先级等不同情况,采取一定的策略将资源分配给各</p>
		<p>马鹏将 马鹏将 - 《学术论文联合比对库》 - 2015-01-04 ( 是否引证：否 )</p> <p>1.onMaster和Container等几个组件构成。1)</p> <p>ResourceManager ( RM ) RM是一个全局的资源管理器，负责整个系统的资源管理和分配。它主要由两个组件构成：调度器 ( Scheduler ) 和应用程序管理器 ( ApplicationsManager , ASM ) 。a) 调度器调度器根据容量、队列等限制条件 ( 如每个队列分配一定的资源，最多执行一定数量的作业等 )，将系</p>
		<p>主资源公平调度算法在YARN中应用研究 陈昕;刘朝晖; - 《电脑知识与技术》 - 2016-04-12 1 ( 是否引证：否 )</p> <p>1.ource Manager(RM)Resource Manager作为Hadoop全局的资源管理器,主要是负责YARN的资源管理和分配。它由两个组件构成:调度器(Scheduler)和应用程序管理器(Applications Manager,ASM)。1)调度器调度器和MRv1的一样,可以根据多种限制条件(如容量、队列等),将资源分配给系统中各</p>
22	<p>此处有 63 字相似</p> <p>器(Scheduler)和应用程序管理器 (ApplicationsManager)。调度器是RM实现资源调度的</p>	<p>03700_1120379028 赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 ( 是否引证：否 )</p> <p>1.ns-Manager)。调度器是资源管理器的核心组件之一</p>



	<p>核心组件， 它会根据集群的资源容量、作业队列、任务优先级等约束条件，并依据相应的策略完成将集群资源分配给各个应用程序的过程。在RM中，调度器只完成纯粹的资源调度功能，并不负责监控和跟踪应用程序的后续运行状态。此外也不会对由于应用程序bug或硬件故障等原因导致</p>	<p>，是真正负责对全局资源进行调度和分配的地方。它会根据集群资源容量、队列设置、优先级等不同情况，采取一定的策略将资源分配给各个应用程序。但这里的调度器只是一个纯粹的调度器，并不参与任何与应用程序相关的具体工作，比如对应用程序的监控或者对应用程序的状态进行追</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士学位论文》 - 2015-01-14 (是否引证：否)</p> <p>1.Applications-Manager)。调度器是资源管理器的核心组件之一,是真正负责对全局资源进行调度和分配的地方。它会根据集群资源容量、队列设置、优先级等不同情况,采取一定的策略将资源分配给各个应用程序。但这里的调度器只是一个纯粹的调度器,并不参与任何与应用程序相关的具体工作,比如对应用程序的监控或者对应用程序的状态进行追踪,同时它也</p>
23	<p>此处有 38 字相似</p> <p>资源分配给各个应用程序的过程。在RM中，调度器只完成纯粹的资源调度功能，并不负责监控和跟踪应用程序的后续运行状态。此外也不会对由于应用程序bug或硬件故障等原因导致的任务失败做出响应。调度器仅依赖于应用程序提出的资源申请和系统设定的资源分配算法进行工作，它的资源调度单位是容器（container）。容器是一个对系统</p>	<p>03700_1120379028 赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证：否)</p> <p>1.粹的调度器，并不参与任何与应用程序相关的具体工作，比如对应用程序的监控或者对应用程序的状态进行追踪，同时它也不会对由于软件或者硬件原因造成应用程序运行中的失败做出任何反应。调度器工作的依据仅仅是自己的资源分配算法和各个应用程序申请的资源量。YARN 资源分配的单位是容器（container），它</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士学位论文》 - 2015-01-14 (是否引证：否)</p> <p>1.器只是一个纯粹的调度器,并不参与任何与应用程序相关的具体工作,比如对应用程序的监控或者对应用程序的状态进行追踪,同时它也不会对由于软件或者硬件原因造成应用程序运行中的失败做出任何反应。调度器工作的依据仅仅是自己的资源分配算法和各个应用程序申请的资源量。YARN资源分配的单位是容器(container),它</p> <p>—9—图2-2</p>
24	<p>此处有 32 字相似</p> <p>素。使用该容器的进程对节点资源的使用量不能超过该容器的上限，同时容器也提供了一个相对隔离的环境来运行任务，从而实现了任务隔离的目的。在本论文所采用的YARN版本中只提供了对CPU和内存的容器封装支持。在YARN中内置了三个可以直接使用的资源调度器：先进先出（FIFO）调度器、公平（Fair）调度器和计算</p>	<p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士学位论文》 - 2015-01-14 (是否引证：否)</p> <p>1.源进行抽象封装,容器内进程对资源的使用量不能超过容器的资源量,因此这样可以限定每个运行在容器中进程的资源使用量,达到资源隔离的目的。本课题所采用的YARN版本中仅支持CPU和内存。目前YARN本身也提供了多种直接可用的调度器,典型的调度器如先进先出调度器、公平调度器和容量调度器,默认为容量调度器。此</p>
25	<p>此处有 50 字相似</p> <p>CPU和内存的容器封装支持。在YARN中内置了三个可以直接使用的资源调度器：先进先出（FIFO）调度器、公平（Fair）调度器和计算能力（Capacity）调度器，默认使用的调度器为Capacity Scheduler。此外，调度器的调度策略被设计成一个可插拔的组件，用户可以根据自身的需求来设计符合需要的调度策略</p>	<p>BH20109263109 李孟庭 李孟庭 - 《学术论文联合比对库》 - 2013-01-05 (是否引证：否)</p> <p>1.kerJobQueueWaiting JobsRunning JobsFinished Jobs图4-2 FIFO调度模型2) 计算能力调度算法 (Capacity Scheduler) 计算能力调度算法是Yahoo公司研发的调度算法[26]。算法的基本调度模型如图4-3所示。JobQue</p>

	来替换现有的调度器。 8	
26	<p>此处有 38 字相似</p> <p>器、公平 ( Fair ) 调度器和计算能力 ( Capacity ) 调度器，默认使用的调度器为Capacity Scheduler。</p> <p>此外，调度器的调度策略被设计成一个可插拔的组件，用户可以根据自身的需求来设计符合需要的调度策略来替换现有的调度器。</p> <p>8.4.2 应用管理器(Application Master, AM) 应用</p>	<p>马鹏将 马鹏将 - 《学术论文联合比对库》 - 2015-01-04 ( 是否引证：否 )</p> <p>1. , Container是一个动态资源分配单位，它将内存、CPU、磁盘、网络等资源封装在一起，从而限定每个任务使用的资源量。此外，该调度器是一个可插拔的组件，用户可根据自己的需要设计新的调度器，YARN提供了多种可直接使用的调度器，比如FairScheduler和CapacityScheduler等。</p> <p>BH20099257125 卡哇伊 卡哇伊 - 《学术论文联合比对库》 - 2012-12-28 ( 是否引证：否 )</p> <p>1.er加载作业的程序，并启动一个独立的Child JVM来执行任务。2 ) 调度器编写步骤为了满足不同客户的差异化调度需求，调度器被设计成一个可插拔的组件，用户可以根据特有的需求进行定制化，只需要在配置文件mapred-site.xml中的mapred.jobtracker.taskSchedul</p> <p>BH20109263109 李孟庭 李孟庭 - 《学术论文联合比对库》 - 2013-01-05 ( 是否引证：否 )</p> <p>1.。？在目标作业中进行任务选择。4.5.2 DPBCScheduler算法实现为了满足不同客户的差异化调度需求，调度器被设计成一个可插拔的组件，用户可以根据特有的需求进行定制化，只需要在配置文件mapred-site.xml中的mapred.jobtracker.taskSchedul</p> <p>HADOOP调度算法及其改进策略研究 李盼 - 《北京邮电大学硕士论文》 - 2013-01-14 ( 是否引证：否 )</p> <p>1.f&amp;gt;&amp;gt;在目标作业巾进行任务选择。4.4.2 DPBCScheduler 算法实现为了满足不同客户的差异化调度需求，调度器被设计成一个可插拔的组件，用户可以根据特有的需求进行定制化，只需要在配置文件mapred-site.xml中的mapred.jobtracker.taskSche</p> <p>李盼_10Z0700 Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》 - 2013-01-15 ( 是否引证：否 )</p> <p>1.北京邮电大学硕士学位论文394.5.2 DPBCScheduler 算法实现为了满足不同客户的差异化调度需求，调度器被设计成一个可插拔的组件，用户可以根据特有的需求进行定制化，只需要在配置文件 mapred-site.xml 中的mapred.jobtracker.taskSc</p>
27	<p>此处有 92 字相似</p> <p>设计符合需要的调度策略来替换现有的调度器。</p> <p>8.4.2 应用管理器(Application Master, AM)</p> <p>应用程序管理器 ( ApplicationsManager)负责接受作业的提交，并且负责于应用程序申请分配得到的第一个容器上启动应用程序的专属的 ApplicationMaster ( AM ) ，并提供任务失败时重新启动AM的服务。AM的主要有以下功能：与RM管理器提交包含一定资源需求的容器申请，其中容器所包含的资</p>	<p>03700_1120379028 赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 ( 是否引证：否 )</p> <p>1.度器原理大致相同，但其资源分配排序的依据是任务已使用资源与已分配资源的差值，并且支持异构的队列。应用程序管理器 ( ApplicationsManager) 则是负责接受应用程序的提交，并且给应用程序分配第一个容器来运行应用程序特有的 ApplicationMaster。此外，它还负责应用程序的 AM 失败后的重启。2.2.1.2 ApplicationMaster(AM)</p> <p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士论文》 - 2015-01-14 ( 是否引证：否 )</p>

		<p>1.r):容器调度器与公平调度器原理大致相同,但其资源分配排序的依据是任务已使用资源与已分配资源的差值,并且支持异构的队列。应用程序管理器 (ApplicationsManager)则是负责接受应用程序的提交,并且给应用程序分配第一个容器来运行应用程序特有的 ApplicationMaster。此外,它还负责应用程序的AM失败后的重启。2.2.1.2 Applicatio</p>
28	<p>此处有 59 字相似</p> <p>AM容器之后,AM负责将申请到的容器进行再分配来运行子任务;在任务运行在所申请的容器后,与容器所在节点保持心跳连接,同时</p> <p>监控和跟踪任务的运行状态,如果出现任务运行失败的情况,AM将重新提交包含任务运行所需资源的申请,以便重新运行任务。</p> <p>8.4.3 节点管理器 (Node Manager, NM)</p> <p>节点管理器在所有可用的计算节点上均有一个实例运行。它的主</p>	<p>Hadoop环境下的动态资源管理研究与实现 赵颖 - 《上海交通大学硕士论文》 - 2015-01-14 (是否引证:否)</p> <p>1.?将从资源管理器处申请到的资源进行二次分配给内部的子任务。?与被分配容器所在节点的节点管理器通信,从而启动/停止任务。?监控所有任务的运行状态,当有任务失败时,AM将重新申请任务需要的资源然后重新运行任务。当前YARN自带了MapReduce应用程序的AM实现——MRAppMaster,其它一些框架对应的AM也由对应的开发人</p> <p>03700_1120379028 赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证:否)</p> <p>1.请到的资源进行二次分配给内部的子任务。?与被分配容器所在节点的节点管理器通信,从而启动/停止任务。? 监控所有任务的运行状态,当有任务失败时,AM将重新申请任务需要的资源然后重新运行任务。当前YARN 自带了 MapReduce 应用程序的 AM 实现——MRAppMaste</p>
29	<p>此处有 64 字相似</p> <p>跟踪任务的运行状态,如果出现任务运行失败的情况,AM将重新提交包含任务运行所需资源的申请,以便重新运行任务。</p> <p>8.4.</p> <p>3 节点管理器 (Node Manager, NM)</p> <p>节点管理器在所有可用的计算节点上均有一个实例运行。它的主要功能是监控容器</p> <p>内进程的资源使用状况,并通过心跳包向RM进行汇报,具体包含各个Container的运行状态、运行的 Application</p>	<p>03700_1120379028 赵颖 赵颖 - 《学术论文联合比对库》 - 2014-12-28 (是否引证:否)</p> <p>1.r, 其它一些框架对应的 AM 也由对应的开发人员开发完成,如 spark, storm 等。2.2.1.3 节点管理器 (NodeManager) 节点管理器会在每一个可用于计算的节点的运行,主要功能是负责容器,即需要监控容器内进程的资源使用情况(如内存、CPU、网络、硬盘)。节点管理器通过心跳信息与资源管理器保持联络,同时</p>
30	<p>此处有 260 字相似</p> <p>失败信息或调度算法的计算结果告知NM需要启动和关闭container列表、Application列表等信息。</p> <p>8.5</p> <p>Hadoop现有的作业调度算法</p> <p>作业调度器是的关键组件之一,它的职能是当检测到中存在空闲的工作槽时,根据特定的调度算法从作业队列中选取作业,再进一步选取一个或多个任务作为回复,派发给去执行。调度器性能的好坏将直接影响整个系统的吞吐量和效率。本小节将重点研究的作业调度算法及其改进策略。</p> <p>当前,应用比较广泛的Hadoop作业调度算法包括:FIFO调度算法、Yahoo公司研发的容量调度 (Capacity Scheduler) 算法和Facebook公司研发的公</p>	<p>BH20109263109 李孟庭 李孟庭 - 《学术论文联合比对库》 - 2013-01-05 (是否引证:否)</p> <p>1.要介绍Hadoop的关键技术,深入介绍了Hadoop分布式文件系统和MapReduc框架的架构和工作原理。第四章 Hadoop作业调度算法改进作业调度器是Hadoop的关键组件之一,它的职能是当检测到TaskTracker中存在空闲的工作槽时,根据特定的调度算法从作业队列中选取作业,再进一步选取一个或多个任务作为回复,派发给TaskTracker去执行。调度器性能的好坏将直接影响整个Hadoop系统的吞吐量和效率。本章将重点研究Hadoop的作业调度算法及其改进策略。4.1 调度算法分析Hadoop作业调度的主要目的是将合适作业的合适任务分配到合适的Slave上去。任务分配是一个</p>

<p>平份额调度 ( Fair Scheduler ) 算法。</p> <p>8.5.1 FIFO调度算法</p> <p>在Hadoop发展的初期，MapReduce框架所处理的作业主要是单用户提交的大规模批处理</p>	<p>2.作业调度器成为一个可配置的独立组件，开发者可以根据实际调度需求定制自己的作业调度器。4.1.1 三种调度算法当前，应用比较广泛的Hadoop作业调度算法包括：FIFO调度算法、Yahoo公司研发的容量调度 ( Capacity Scheduler ) 算法和Facebook公司研发的公平份额调度 ( Fair Scheduler ) 算法。1 ) 先入先出调度算法 ( FIFO Scheduler ) Hadoop最早应用时，处理的业务通常是单用户提交的大型批处理</p> <p>HADOOP调度算法及其改进策略研究 李盼 - 《北京邮电大学硕士论文》 - 2013-01-14 ( 是否引证：否 )</p> <p>1.会使用默认的Patritioner (应用hash函数 )。16北j ; !邮电大学硕士学位论文 第三章 Hadoop作业调度算法作业调度器是Hadoop的关键组件之一，它的职能是当检测到TaskTracker中存在空闲的工作槽时，根据特定的调度算法从作业队列中选取作业，再进一步选取一个或多个任务作为回复，派发给TaskTracker去执行。调度器性能的好坏将直接影响整个Hadoop系统的吞吐量和服务效率。本章将重点研究Hadoop的作业调度算法及其改进策略。3.1 Hadoop作业调度流程f y ,v,  submitiobMp T 品忠ce jobClien</p> <p>2.发现Job执行成功，便从runJob方法返回，把运行的结果发送给用户。3.2 Hadoop现有调度算法当前，应用比较广泛的Hadoop作业调度算法包括：FIFO调度算法、Yahoo公司研发的容量调度 ( Capacity Scheduler)算法和Facebook公司研发的公平份额调度 ( Fair Scheduler)算法。3.2.1先入先出调度算法 Hadoop最早应用时，处理的业务通常是单用户提交的大型批处理作业，因此作</p> <p>李盼_10Z0700_Hadoop关键技术及其改进策略研究 李盼 - 《学术论文联合比对库》 - 2013-01-15 ( 是否引证：否 )</p> <p>1.会使用默认的 Partitioner ( 应用 hash函数 )。北京邮电大学硕士学位论文25第四章 Hadoop 作业调度算法改进作业调度器是 Hadoop 的关键组件之一，它的职能是当检测到 TaskTracker中存在空闲的工作槽时，根据特定的调度算法从作业队列中选取作业，再进一步选取一个或多个任务作为回复，派发给 TaskTracker 去执行。调度器性能的好坏将直接影响整个 Hadoop 系统的吞吐量和服务效率。本章将重点研究 Hadoop 的作业调度算法及其改进策略。4.1 调度算法分析Hadoop 作业调度的主要目的是将合适作业的合适任务分配到合适的Slave上去。任务</p> <p>2.度器成为一个可配置的独立组件，开发者可以根据实际调度需求定制自己的作业调度器。4.1.1 三种调度算法当前，应用比较广泛的 Hadoop 作业调度算法包括：FIFO 调度算法、Yahoo公司研发的容量调度 ( Capacity Scheduler ) 算法和 Facebook 公司研发的公平份额调度 ( Fair Scheduler ) 算法。1 ) 先入先出调度算法 ( FIFO Scheduler ) Hadoop 最早应用时，处理的业务通常是单用户提交的大型批处</p>
--	---



	<p>BH20099257125 卡哇伊 卡哇伊 - 《学术论文联合比对库》 - 2012-12-28 ( 是否引证：否 )</p> <p>1.Hadoop的关键技术，深入介绍了Hadoop分布式文件系统和MapReduce框架的架构和工作原理。第五章 Hadoop作业调度算法改进作业调度器是Hadoop的关键组件之一，它的主要职能是在收到TaskTracker发来的心跳信息时，根据特定的调度算法从作业队列中选取作业，再进一步选取一个或多个任务作为回</p> <p>2.adoop的关键组件之一，它的主要职能是在收到TaskTracker发来的心跳信息时，根据特定的调度算法从作业队列中选取作业，再进一步选取一个或多个任务作为回复，派发给TaskTracker去执行。调度器性能的好坏将直接影响整个Hadoop系统的吞吐量和服务效率。本章将重点研究Hadoop的作业调度算法，通过分析调度算法的不足，提出自己的改进策略。系统作业的执行流程Hadoop系统中Master端 ( JobTra</p> <p>3.作业调度作为一个可插拔的组件独立出来，使用户能够根据各自的需要提供不同的实现。4.2.1 三种调度算法当前，应用比较广泛的Hadoop作业调度算法包括：FIFO调度算法、Yahoo公司研发的容量调度 ( Capacity Scheduler ) 算法和Facebook公司研发的公平份额调度 ( Fair Scheduler ) 算法。1) 先入先出调度算法 ( FIFO Scheduler ) Hadoop最早应用时，面对的基本是单用户提交的大型批处理作业</p> <p>20121236535-王德龙-Hadoop平台下作业调度算法的研究与改进 王德龙 - 《学术论文联合比对库》 - 2015-04-07 ( 是否引证：否 )</p> <p>1.Hadoop目前重要的内置作业调度算法有：先入先出调度算法 ( FIFO Scheduler )、由雅虎公司研发的容量调度 ( Capacity Scheduler ) 算法和由Facebook公司研发的公平份额调度 ( Fair Scheduler ) 算法。这些内置的作业调度算法配置简单，得到广泛应用。2.2.1 FIFO调度算法在Hadoop发展初期，单用户提交的作</p>
--	--

指 标

疑似剽窃文字表述

1. 用户只需要实现两个简单的函数Map和Reduce，即可高效的实现分布式计算。用户无需关心框架是如何
2. 即可以并行的方式处理输入数据集。通常，map函数负责处理输入的<key,value>键值对，生成一系列以<key,value>形式存储的中间结果。
3. 处理并产生输出文件。在以上过程中，MapReduce框架会自动将输入文件划分为M份小的输入文件
4. 在MapReduce集群系统中，节点机器会被划分为两类不同职责的机器，一类是控制节点，在一般情况下由一台机器来执行控制节点的职能；其余的所有机器被划为另一类，即工作节点。控制节点的主要任务是将map和Reduce任务分配给工作节点。当其发现
5. 心跳通信机制。在HDFS中，NameNode与DataNode通过心跳机制来进行信息交流。每隔固定的时间，DataNode将会发送一个心跳数据包，汇报本节点当前的状态。NameNode
6. 任务的运行情况，并定时向JobTracker进行汇报。由此可见，JobTrack
7. 由JobTracker 负责所有节点资源的管理、分配和任务调度，YARN框架在设计中采用了两级调度策略
8. 申请到的资源分配给自己的子任务，同时AM也需要通知对应资源所在节点的管理器NM来启动对应的任务进行运行。接下来将详细讲述此三个YARN的重要组成部分。
9. 调度器是RM实现资源调度的核心组件，它会根据集群的资源容量、作业队列、任务优先级等约束条件，并依据相应的策略完成将集群资源分配给各个应用程序的过程。在RM中，调度

10. 监控和跟踪任务的运行状态，如果出现任务运行失败的情况，AM将重新提交包含任务运行所需资源的申请，以便重新运行任务。
11. Hadoop现有的作业调度算法
- 作业调度器是的关键组件之一，它的职能是当检测到中存在空闲的工作槽时，根据特定的调度算法从作业队列中选取作业，再进一步选取一个或多个任务作为回复，派发给去执行。调度器性能的好坏将直接影响整个系统的吞吐量和服务效率。本小节将重点研究的作业调度算法及其改进策略。

## 9. 80022497643220302\_基于动态手势的智能终端身份识别技术研究是实现\_第9部分 总字数：2829

相似文献列表 文字复制比：14.3%(405) 疑似剽窃观点：(0)

1	Hadoop YARN资源分配与调度的研究 李媛祯(导师：杨群) - 《南京航空航天大学硕士论文》 - 2015-01-01	13.1% ( 371 ) 是否引证：否
2	5-颜廷帅 颜廷帅 - 《学术论文联合比对库》 - 2014-12-22	1.2% ( 33 ) 是否引证：否
3	S312060010+颜廷帅 颜廷帅 - 《学术论文联合比对库》 - 2014-12-31	1.2% ( 33 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 40 字相似</p> <p>在MRv2中，FIFO调度器支持任务的优先级调度，通过设置5个不同的优先级</p> <p>队列，根据队列的优先级的高低，从高到低依次选择不同队列，在同级队列中，按照作业的提交顺序进行作业选择调度和资源分配。FIFO调度算法的设计思想简单易实现，且用户无需对系统进行额外的配置，调度器运行算法</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证：否 )</p> <p>1.的队列设置了5个优先等级,分别为:Very Low、Low、Normal、High及Very High。每个等级对应一个队列,按照队列的优先级从高到低选取队列,在同级队列中,按照提交作业的时间先后顺序提取并执行。FIFO调度算法[40]的设计思想简单、易于实现,整个系统无需额外配置且调度开销较小。与此同时,</p>
2	<p>此处有 68 字相似</p> <p>的优先级的高低，从高到低依次选择不同队列，在同级队列中，按照作业的提交顺序进行作业选择调度和资源分配。FIFO调度算法的</p> <p>设计思想简单易实现，且用户无需对系统进行额外的配置，调度器运行算法的代价也较小。相对而言，FIFO调度算法也存在比较明显的不足之处：首先</p> <p>该算法对各个作业的资源需求考虑不足，容易造成系统不同计算资源利用率的失衡，例如集群中计算密集型作业过多使得内存资源闲置，</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证：否 )</p> <p>1.个队列,按照队列的优先级从高到低选取队列,在同级队列中,按照提交作业的时间先后顺序提取并执行。FIFO调度算法[40]的设计思想简单、易于实现,整个系统无需额外配置且调度开销较小。与此同时,FIFO调度算法的缺点也显而易见,第一,其未考虑不同作业的需求差异,对所有作业都一视同仁,这对小作业的执行非常不利,如图2.5所示;第二,算法的优先级不支持抢占</p>
3	<p>此处有 63 字相似</p> <p>同时也可以设置资源使用上限防止队列无限制申请资源。每个队列并不是严格按照配置分配固定的资源，而是根据需求动态分配，当其他</p> <p>队列有剩余资源未能加以利用时，调度器会将资源暂时回收，分配给其他队列来加速集群的运行。当原有队列有新的任务提交需要资源运行时，</p> <p>调度器将会对其他队列发起资源回收请求，在被占用资源上的容器中的任务运行结束后，资源将会被返回原有队列，如果该队列在等待一</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证：否 )</p> <p>1.2)支持资源抢占:与Capacity Scheduler算法相同,Fair Scheduler算法也支持资源抢占,当某个队列中有剩余资源时,调度器会将这些资源共享给其他队列,而当该队列中有新的应用程序提交时,调度器便为它回收资源。但如果该队列在等待一段时间后尚未有归还的资源,则进行资源抢占。(</p> <p>5-颜廷帅 颜廷帅 - 《学术论文联合比对库》 - 2014-12-22 ( 是否引证：否 )</p> <p>1.序。并且，管理员可以指定整个集群系统的管理员和队列的管理员。如果某个队列中出现了空闲的资源，这些空间的资源可以暂付共享给其他资源紧缺的队列使用。当该队列有新的应用程序提交时，被共享的资源在完成任务后会被归还到它所属的队列中。这种方式不但使</p>

		<p>得集群中的资源得到了共享，同时提高了整个集群中资源利用率</p> <p>S312060010+颜廷帅 颜廷帅 - 《学术论文联合比对库》 - 2014-12-31 (是否引证：否)</p> <p>1.序。并且，管理员可以指定整个集群系统的管理员和队列的管理员。如果某个队列中出现了空闲的资源，这些空间的资源可以暂付共享给其他资源紧缺的队列使用。当该队列有新的应用程序提交时，被共享的资源在完成任务后会被归还到其所属的队列中。这种方式不但使得集群中的资源得到了共享，同时提高了整个集群中资源利用率</p>
4	<p>此处有 36 字相似</p> <p>，默认地，Capacity调度器将会强制停止其它队列上的一些任务来返回之前征用的资源。</p> <p>Capacity调度器实际上是</p> <p>三级资源分配模型，调度流程如图2.8所示，当一个节点上有空闲资源的时候，</p> <p>系统会依次对Job Queue（或用户队列）、Application（或作业）和Container（或请求）进行三次分配</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.Capacity Scheduler会强制停止占用资源的任务并将资源归还给队列。Capacity Scheduler采用三级资源分配策略,如图2.6所示,当一个节点上有空闲资源时,会依次选择队列、应用程序(如作业)和Container(请求)使用该资源,下面介绍三级资源分配策略。(1)选择队列YAR</p>
5	<p>此处有 45 字相似</p> <p>当的调整；其次，在系统选择一个队列之后，Capacity Scheduler按照队列内的调度算法对作业进行调度排序。通常</p> <p>队列内的调度算法为FIFO算法或者DRF ( Dominant Resource Fairne</p> <p>ss ) 算法。DRF算法是一种考虑多种资源（主要是CPU和内存）需求调度的公平资源分配算法。由于不是研究重点，在此不做详细</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.有剩余时,可暂时将剩余资源共享给其他队列。</p> <p>Capacity Scheduler支持多个队列,每个队列可分配一定的资源量,队列中资源的调度策略可以为FIFO或DRF(Dominant Resource Fairness),多用户下,为防止同一个用户提交的作业独占队列的所有资源,Capacity Scheduler对每个用户可提交的作业可分配</p>
6	<p>此处有 73 字相似</p> <p>city Scheduler也增加了对于作业和文件的权限管理，有效地降低了单个用户误操作对其他用户造成影响。</p> <p>8.5.</p> <p>3 Fair调度算法</p> <p>Fair调度算法是由Facebook研发和推广的多用户调度器，其实现方式相比Capacity Scheduler较为相似，也采用了以队列划分资源的方式。Fair Scheduler设计的初衷是让系统中的各个作业能够公平地分配到系统资源。Fair</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.定队列管理员和集群系统管理员。(5)动态更新配置文件:管理员可根据需要动态修改各种配置参数,以实现在线集群管理。2.3.3 Fair调度算法2.3.3.1 Fair调度算法思想Fair Scheduler是Facebook开发的多用户调度器[42],同Capacity Scheduler相似,都以队列为单位划分资源,且每个队列可设定一定比例的资源最低保证和使用上限。Facebook设计FairScheduler</p>
7	<p>此处有 80 字相似</p> <p>他用户在其抢占资源上的运行程序，并返回资源给等待用户。Fair Scheduler也有工作良好的负载均衡机制，其作用是将</p> <p>系统中作业所含的任务尽可能平均地分配到各个计算节点上，用户也可以依据自己的需求进行负载均衡的定制。</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.,则进行资源抢占。(3)任务负载均衡:Fair Scheduler提供了一个基于任务数目的负载均衡机制,该进制尽可能的将系统中的任务均匀的分配到集群的各个节点上,同时也支持用户根据自身需要设计新的负载均衡机制。2.3.4 Hadoop常用调度算法的不足上述分析中可知,Hadoop常</p>



8.6 Hadoop常用调度算法的不足 由上述分析可知， YARN内置的三种调度算法均能完成基础的资源管理和作业调度功能，但是在分配的过程中均存在某些缺陷。 FIFO算法是应用最早	用的三种调度算法能够顺利完成资源分配功能,但在分配过程中仍存在资源分配不合理的缺陷。从作业执行上来看,Ha
---	---

指 标
疑似剽窃文字表述
1. 队列，根据队列的优先级的高低，从高到低依次选择不同队列，在同级队列中，按照作业的 2. 设计思想简单易实现，且用户无需对系统进行额外的配置，调度器运行算法的代价也较小。相对而言，FIFO调度算法也存在比较明显的不足之处：首先 3. 队列有剩余资源未能加以利用时，调度器会将资源暂时回收，分配给其他队列来加速集群的运行。当原有队列有新的任务提交需要资源运行时， 4. 3 Fair调度算法 Fair调度算法是由Facebook研发和推广的多用户调度器，其实现方式相比Capacity Scheduler较为相似， 5. 系统中作业所含的任务尽可能平均地分配到各个计算节点上，用户也可以依据自己的需求进行负载均衡的定制。 8.6 Hadoop常用调度算法的不足 由上述分析可知，

10. 80022497643220302_基于动态手势的智能终端身份识别技术研究_第10部分 总字数：4607		
相似文献列表 文字复制比：31.3%(1443) 疑似剽窃观点：(0)		
1	<u>Hadoop平台下基于遗传算法的作业调度的研究与改进</u> 燕明磊(导师：薛涛) - 《西安工程大学硕士论文》 - 2015-05-24	18.9% ( 869 ) 是否引证：否
2	<u>燕明磊_2012076_Hadoop平台下基于遗传算法的作业调度的研究与改进</u> 燕明磊 - 《学术论文联合比对库》 - 2015-03-11	18.9% ( 869 ) 是否引证：否
3	<u>燕明磊_2012076_Hadoop平台下基于遗传算法的作业调度的研究与改进</u> 燕明磊 - 《学术论文联合比对库》 - 2015-03-11	18.9% ( 869 ) 是否引证：否
4	<u>一种自适应步长布谷鸟搜索算法</u> 郑洪清;周永权; - 《计算机工程与应用》 - 2012-05-21 1	7.9% ( 365 ) 是否引证：否
5	<u>一种小规模多种群布谷鸟算法</u> 郑巧燕;莫愿斌;刘付永;马彦追; - 《计算机应用与软件》 - 2014-10-15	2.5% ( 113 ) 是否引证：否
6	<u>基于CS-SVM的网络热点话题变化趋势预测</u> 邱仕坦; - 《福州大学学报(自然科学版)》 - 2014-05-28 1	2.1% ( 98 ) 是否引证：否
7	<u>改进布谷鸟搜索算法求解批量流水线调度问题</u> 郑洪清; - 《计算机系统应用》 - 2014-10-15	2.0% ( 94 ) 是否引证：否
8	<u>随机交叉粒子群优化算法</u> 王联国;洪毅; - 《计算机工程与应用》 - 2009-06-01	1.9% ( 88 ) 是否引证：否
9	<u>基于觅食算子的粒子群优化算法</u> 王联国;洪毅;赵付青; - 《计算机应用与软件》 - 2009-11-15	1.7% ( 79 ) 是否引证：否
10	<u>一种多智能体混合蛙跳算法</u> 王联国;代永强; - 《计算机工程》 - 2013-07-15	1.5% ( 71 ) 是否引证：否
11	<u>求解无约束优化问题的改进布谷鸟搜索算法</u> 苏英华;刘云连;伍铁斌; - 《计算机工程》 - 2014-05-15	1.5% ( 71 ) 是否引证：否
12	<u>基于高斯扰动的布谷鸟搜索算法</u> 王凡;贺兴时;王燕; - 《西安工程大学学报》 - 2011-08-25	1.1% ( 52 ) 是否引证：否
13	<u>基于布谷鸟搜索算法的主题爬虫策略设计</u> 钱竞远;杨辉华;刘振丙; - 《仪器仪表用户》 - 2017-06-08	1.1% ( 51 ) 是否引证：否
14	<u>基于混沌优化与人工鱼群算法的混合算法研究</u> 石鸿雁;邢东亚; - 《微计算机信息》 - 2012-09-15	0.8% ( 35 ) 是否引证：否
15	<u>求解工程结构优化问题的改进布谷鸟搜索算法</u> 陈乐;龙文; - 《计算机应用研究》 - 2013-11-05 1	0.8% ( 35 ) 是否引证：否
原文内容		相似内容来源



1	<p>此处有 54 字相似</p> <p>程和莱维飞行而产生的。布谷鸟算法具有以下优点：操作简单，参数少，无需在处理优化问题时重新匹配参数。布谷鸟寻觅巢穴的路径是类似随机搜索的方式。为了模拟和简化布谷鸟寻巢的方式，规定以下3个理想的状态：(1)每只布谷鸟每次只产一个卵，即一个解。鸟巢孵化位置遵循随机选择的方式。(2)位置最好的鸟巢即最优的解将被保留到下一代。(3)可供寄生的巢数量n是固</p>	<p>基于高斯扰动的布谷鸟搜索算法 王凡;贺兴时;王燕; - 《西安工程大学学报》 - 2011-08-25 ( 是否引证：否 )</p> <p>1.上,提出了基于高斯扰动的布谷鸟搜索算法(GCS).1基本CS算法在自然界中,布谷鸟寻找适合自己产卵的鸟窝位置是随机的或是类似随机的方式,为了模拟布谷鸟寻窝的方式,首先,需要设定以下3个理想的状态[4]:(1)布谷鸟一次只产一个卵,并随机选择鸟窝来孵化它;(2)在随机选择的一组鸟窝中,最好的鸟窝将会被保留到下一代;(3)可利用的鸟窝数量n是固定的,一</p> <p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》 - 2012-05-21 1 ( 是否引证：否 )</p> <p>1.长布谷鸟搜索算法具有较快的收敛速度和较高的寻优精度。1基本CS算法在自然界中,布谷鸟寻找适合自己产卵的鸟窝位置是随机的或类似随机的方式,为了模拟布谷鸟寻窝的行为,首先,设定以下3个理想的状态[4]:(1)布谷鸟一次只产一个卵,并随机选择鸟窝位置来孵化它。(2)在随机选择的一组鸟窝中,最好的鸟窝将会被保留到下一代。(3)可利用的鸟窝数量n是固定的</p> <p>一种小规模多种群布谷鸟算法 郑巧燕;莫愿斌;刘付永;马彦追; - 《计算机应用与软件》 - 2014-10-15 ( 是否引证：否 )</p> <p>1.解精度和收敛成功率等均优于改进前的算法。1基本CS算法在自然界中,布谷鸟寻找适合自己产卵的鸟窝位置是随机的或者是类似随机的方式,为了模拟布谷鸟寻窝的方式,首先,需要设定以下3个理想的状态[1]:(1)布谷鸟一次只产一个卵,并随机选择鸟窝来孵化它;(2)在随机选择的一组鸟窝中,最好的鸟窝将会被保留到下一代;(3)可利用的鸟窝数量是固定的,一个</p> <p>基于CS-SVM的网络热点话题变化趋势预测 邱仕坦; - 《福州大学学报(自然科学版)》 - 2014-05-28 1 ( 是否引证：否 )</p> <p>1.最优SVM参数,建立预测精度高的网络热点话题变化趋势预测模型.<a href="http://xbzrb.fzu.edu.cn">http://xbzrb.fzu.edu.cn</a>1.2布谷鸟搜索算法为了模拟布谷鸟寻巢的行为,需设定以下3个理想的状态:1)布谷鸟一次只产一个卵,并随机选择鸟巢位置来孵化它;2)在随机选择的一组鸟巢中,最好位置的鸟巢将会保留到下一代;3)可利用的鸟巢数量N是固定的,</p> <p>改进布谷鸟搜索算法求解批量流水线调度问题 郑洪清; - 《计算机系统应用》 - 2014-10-15 ( 是否引证：否 )</p> <p>1.)()(maxmaxcc????,????2基本的布谷鸟搜索算法在自然界中,布谷鸟寻找适合自己产卵的鸟巢位置是随机的或类似随机的方式,为了模拟布谷鸟寻窝的方式,首先,需要设定以下3个理想的状态[6]:1)布谷鸟每一次只能产一个卵,并随机选择鸟巢位置来孵化它;2)在随机选择的一组鸟巢中,最好的鸟巢将会被保留到下一代;3)可利用的鸟</p>
2	<p>此处有 53 字相似</p> <p>谷鸟寻巢的方式，规定以下3个理想的状态：(1)每只布谷鸟每次只产一个卵，即一个解。鸟巢孵化位置遵循随机选择的方式。(2)位置最好的鸟巢即最优的解将被保留到下一代。(3)可供寄生的巢数量n是固定的，布谷鸟蛋被鸟巢主人发现</p>	<p>基于CS-SVM的网络热点话题变化趋势预测 邱仕坦; - 《福州大学学报(自然科学版)》 - 2014-05-28 1 ( 是否引证：否 )</p> <p>1.的行为,需设定以下3个理想的状态:1)布谷鸟一次只产一个卵,并随机选择鸟巢位置来孵化它;2)在随机选择的一组鸟巢中,最好位置的鸟巢将会保留到下一代;3)可利用的鸟巢数量N是固定的,一个鸟巢主人能发现一个外来</p>

	<p>的概率</p> <p>也是固定的，为<math>P_a</math> (<math>P_a \in [0, 1]</math>)。在以上三种规则的基础上，布谷鸟寻窝路径和位置更新公式如下：</p> $X_i(t+1)$	<p>鸟蛋的概率为<math>P_a</math>[11]。在CS算法中，一个鸟巢的卵表示一个候选解，一个布谷鸟的卵表示一个新的解。在上述三个理想</p> <p>改进布谷鸟搜索算法求解批量流水线调度问题 郑洪清；-《计算机系统应用》- 2014-10-15 (是否引证：否)</p> <p>1.需要设定以下3个理想的状态[6]:1)布谷鸟每一次只能产一个卵,并随机选择鸟巢位置来孵化它;2)在随机选择的一组鸟巢中,最好的鸟巢将会被保留到下一代;3)可利用的鸟巢数量<math>n</math>是固定的,一个鸟巢的主人能发现一个外来鸟蛋的概率<math>P_a</math>。在这3个理想状态的基础上,布谷鸟寻窝的路径和位置更新公式如下:<math>n_i L_{xxtiti}</math>,</p> <p>基于布谷鸟搜索算法的主题爬虫策略设计 钱竞远;杨辉华;刘振丙;-《仪器仪表用户》- 2017-06-08 (是否引证：否)</p> <p>1.鸟搜索算法的实现定义了3个理想化的规则:1)每只布谷鸟每次只产1枚蛋,并且将它存放在一个随机选择的鸟巢中。2)有最好的蛋(解)的鸟巢将会被保存到下一代。3)可用的鸟巢的数目是固定的,且巢的主人发现外来蛋的概率为<math>P_a \in [0, 1]</math>,在这3个理想状态的基础上,布谷鸟寻窝的路径和位置更新公式如下:其中<math>X_i(t)</math>表示第<math>i</math>个个鸟</p>
3	<p>此处有 37 字相似</p> <p>式如下：</p> $X_i(t+1) = X_i(t) + \alpha \oplus L(\lambda) \quad (i = 1, 2, \dots, n) \quad (3-1)$ <p>式中 <math>X_i(t)</math></p> <p>表示第<math>i</math>个鸟窝在第<math>t</math>代的鸟窝位置；<math>\alpha</math>表示步长控制量；<math>\oplus</math>为点乘；<math>L(\lambda)</math></p> <p>服从Levy分布，并且<math>L \sim u = t^{-\lambda} \quad (1 \leq \lambda \leq 3)</math>。在鸟巢更新位置后，将随机数<math>r \in [0, 1]</math></p>	<p>求解工程结构优化问题的改进布谷鸟搜索算法 陈乐;龙文;-《计算机应用研究》- 2013-11-05 (是否引证：否)</p> <p>1.想状态的基础上,布谷鸟寻窝的路径和位置更新公式如下:<math>x_{t+1i} = x_{ti} + \alpha \oplus L(\lambda) \quad i=1,2,\dots,N</math>(2)其中:<math>x_{ti}</math>表示第<math>i</math>个鸟窝在第<math>t</math>代的鸟窝位置;<math>\alpha</math>为步长控制量;<math>\oplus</math>为点积;<math>L(\lambda)</math>为随机步长路径,其服从Lévy分布:<math>Lévy \sim u = t^{-\lambda} \quad 1 \leq \lambda \leq 3</math>(3)通过位置更新后,用随机数<math>r \in [0, 1]</math>与鸟窝的</p> <p>求解无约束优化问题的改进布谷鸟搜索算法 苏英华;刘云连;伍铁斌;-《计算机工程》- 2014-05-15 (是否引证：否)</p> <p>1.础上,布谷鸟寻窝的路径和位置更新公式如下:1)<math>i=1,2,\dots,n</math>其中:<math>x_{t+1i} = x_{ti} + \alpha \oplus L(\lambda) \quad i=1,2,\dots,n</math>其中:<math>x_{ti}</math>表示第<math>i</math>个鸟窝在第<math>t</math>代的鸟窝位置;<math>\alpha</math>为步长控制量;<math>\oplus</math>为点积;<math>L(\lambda)</math>为随机步长路径,其服从Lévy分布:<math>Lévy \sim u = t^{-\lambda} \quad 1 \leq \lambda \leq 3</math>(2)通过位置更新后,用随机数<math>r \in [0, 1]</math>与</p>
4	<p>此处有 872 字相似</p> <p><math>y_i(t+1)</math>进行保留，此时仍记为<math>X_i(t+1)</math>。</p> <p>9.2 遗传算法</p> <p>9.2.1 遗传算法产生的背景</p> <p>遗传算法</p> <p>(Genetic Algorithms GA)源自上世纪六七十年代，由 John Holland教授与其同事、学生们提出，依据达尔文的进化论的主要思想—物竞天择，适者生存，目的是通过模拟生物进化过程构造人工系统模型。后来随着相关理论著作的出版，以及在机器学习和性能优化等领域的应用，该模型理论正式定名为遗传算法。</p> <p>9.2.2 遗传算法的原理</p> <p>9.2.3 遗传算法的特点</p> <p>遗传算法是一种优化算法，因而需要满足最优性原理。最优性原理是针对现实生活中的这一类问题：这个问题可被分为若干个子问题，对于任意的子问题<math>t</math>都只依赖于</p>	<p>燕明磊_2012076_Hadoop平台下基于遗传算法的作业调度的研究与改进 燕明磊 -《学术论文联合比对库》- 2015-03-11 (是否引证：否)</p> <p>1.C遗传算法在思想上是一致的。4基于遗传算法的Hadoop作业调度的研究4.1遗传算法产生背景遗传算法[63](Genetic Algorithms GA)源自上世纪六七十年代，由John Holland教授与其同事、学生们提出，依据达尔文的进化论的主要思想“物竞天择，适者生存”，目的是通过模拟生物进化过程构造人工系统模型。后来随着相关理论著作的出版，以及在机器学习和性能优化等领域的应用，该模型理论正式定名为遗传算法。4.2遗传算法的特点遗传算法是一种优化算法，因而需要满足“最优性原理”。“最优性原理”是针对现实生活中的这一类问题：这个问题可被分为若干个子问题，对于任意的子问题<math>t</math>都只依赖于当前阶段的状态，而与<math>t</math>之前的所有其他子问题的状态均无关，这是一种多阶段决策的问题。这类问题中一种最通用但是低效的方法是枚举法，该方法</p>

<p>当前阶段的状态，而与t之前的所有其它子问题的状态均无关，这是一种多阶段决策的问题。这类问题中一种最通用但是低效的方法是枚举法，该方法的基本思想是穷举解集范围内的所有解，然后依据求解目标找出最优解；这种算法的一种改进算法是搜索算法，它的基本思想是把搜索范围缩小到解集的某一个子集或几个子集的范围内，通过缩小搜索域的方式可以大大的减少计算量；此外，这类问题还有另外一种效率较高但不具有通用性的解决办法，启发式算法：它针对每个待求解的问题必须明确它的特定的启发式规则。遗传算法兼顾了以上两类算法的优点，具有通用性，高效性，并且可以应对问题规模的动态增长，此外还具有以上两类算法不具备的一些特点：</p> <p>(1) 遗传算法以种群作为出发点，从本质上来说是并行的，这是遗传算法最本质的特点。根据进化理论，种群是物种进化的基本单位，种群间的遗传操作是并行的；同时在种群内部，个体间的遗传操作也是并行的，因此具有种群与个体两个层面上的并行。</p> <p>(2) 遗传算法具有智能性，物种在进化中遵守物竞天择，适者生存的进化法则。适应环境的个体有更大的几率生存下去，并且把自己的基因传给后代；而适应环境能力差的个体生存的几率较小或者被淘汰，因此进化的过程就是一个智能选择优良个体淘汰劣质个体的过程。</p> <p>(3) 遗传算法具有灵活性，遗传算法可以通过目标函数和适应度函数来控制种群进化的方向，因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。</p>	<p>的基本思想是穷举解集范围内的所有解，然后依据求解目标找出最优解；这种算法的一种改进算法是搜索算法，它的基本思想是把搜索范围缩小到解集的某一个子集或几个子集的范围内，通过缩小搜索域的方式可以大大的减少计算量；此外，这类问题还有另外一种效率较高但不具有通用性的解决办法，启发式算法：它针对每个带求解的问题必须明确它的特定的启发式规则。遗传算法兼顾了以上两类算法的优点，具有通用性，高效性，并且可以应对问题规模的动态增长，此外还具有以上两类算法不具备的一些特点：(1) 遗传算法以种群作为出发点，从本质上来说是并行的，这是遗传算法最本质的特点。根据进化理论，种群是物种进化的基本单位，种群间的遗传操作是并行的；同时在种群内部，个体间的遗传操作也是并行的，因此具有种群与个体两个层面上的并行。(2) 遗传算法具有“智能性”，物种在进化中遵守“物竞天择，适者生存”的进化法则。适应环境的个体有更大的几率生存下去，并且把自己的基因给后代；而适应环境能力差的个体生存的几率较小或者被淘汰，因此进化的过程就是一个智能选择优良个体淘汰劣质个体的过程。(3) 遗传算法具有灵活性，遗传算法可以通过目标函数和适应度函数来控制种群进化的方向，因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。调度领域是遗传算法的经典应用领域之一。在Hadoop作业调度中，用MapReduce处理的数据集必须具备这样的特点：待处</p>
<p>9.3 混合遗传布谷鸟算法</p> <p>9.3.1 混合遗传布谷鸟算法模型</p> <p>在基本的布谷鸟算法中，步长是由levy flight</p>	<p>燕明磊_2012076_Hadoop平台下基于遗传算法的作业调度的研究与改进 燕明磊 - 《学术论文联合比对库》 - 2015-03-11 (是否引证：否)</p> <p>1.C遗传算法在思想上是一致的。4基于遗传算法的Hadoop作业调度的研究4.1遗传算法产生背景遗传算法[63](Genetic Algorithms GA)源自上世纪六七十年代，由John Holland教授与其同事、学生们提出，依据达尔文的进化论的主要思想“物竞天择，适者生存”，目的是通过模拟生物进化过程构造人工系统模型。后来随着相关理论著作的出版，以及在机器学习和性能优化等领域的应用，该模型理论正式定名为遗传算法。4.2遗传算法的特点遗传算法是一种优化算法，因而需要满足“最优性原理”。“最优性原理”是针对现实生活中的这一类问题：这个问题可被分为若干个子问题，对于任意的子问题t都只依赖于当前阶段的状态，而与t之前的所有其他子问题的状态均无关，这是一种多阶段决策的问题。这类问题中一种最通用但是低效的方法是枚举法，该方法的基本思想是穷举解集范围内的所有解，然后依据求解目标找出最优解；这种算法的一种改进算法是搜索算法，它的基本思想是把搜索范围缩小到解集的某一个子集或几个子集的范围内，通过缩小搜索域的方式可以大大的减少计算量；此外，这类问题还有另外一种效率较高但不具有通用性的解决办法，启发式算法：它针对每个带求解的问题必须明确它的特定的启发式规则。遗传算法兼顾了以上两类算法的优点，具有通用性，高效性，并且可以应对问题规模的动态增长，此外还具有以上两类算法不具备的一些特点：(1) 遗传算法以种群作为出发点，从本质上来说是并行的，这是遗传算法最本质</p>

的特点。根据进化理论，种群是物种进化的基本单位，种群间的遗传操作是并行的；同时在种群内部，个体间的遗传操作也是并行的，因此具有种群与个体两个层面上的并行。(2) 遗传算法具有“智能性”，物种在进化中遵守“物竞天择，适者生存”的进化法则。适应环境的个体有更大的几率生存下去，并且把自己的基因给后代；而适应环境能力差的个体生存的几率较小或者被淘汰，因此进化的过程就是一个智能选择优良个体淘汰劣质个体的过程。(3) 遗传算法具有灵活性，遗传算法可以通过目标函数和适应度函数来控制种群进化的方向，因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。调度领域是遗传算法的经典应用领域之一。在Hadoop作业调度中，用MapReduce处理的数据集必须具备这样的特点：待处

Hadoop平台下基于遗传算法的作业调度的研究与改进 燕明磊 - 《西安工程大学硕士论文》 - 2015-05-24 (是否引证：否)

1.合,这与CHC遗传算法在思想上是一致的。4基于遗传算法的Hadoop作业调度的研究4.1遗传算法产生背景遗传算法[63](Genetic Algorithms GA)源自上世纪六七十年代,由John Holland教授与其同事、学生们提出,依据达尔文的进化论的主要思想—物竞天择,适者生存Ⅱ,目的是通过模拟生物进化过程构造人工系统模型。后来随着相关理论著作的出版,以及在机器学习和性能优化等领域的应用,该模型理论正式定名为遗传算法。4.2遗传算法的特点遗传算法是一种优化算法,因而需要满足—最优性原理Ⅱ。—最优性原理Ⅱ是针对现实生活中的这一类问题:这个问题可被分为若干个子问题,对于任意的子问题t都只依赖于当前阶段的状态,而与t之前的所有其它子问题的状态均无关,这是一种多阶段决策的问题。这类问题中一种最通用但是低效的方法是枚举法,该方法的基本思想是穷举解集范围内的所有解,然后依据求解目标找出最优解;这种算法的一种改进算法是搜索算法,它的基本思想是把搜索范围缩小到解集的某一个子集或几个子集的范围,通过缩小搜索域的方式可以大大的减少计算量;此外,这类问题还有另外一种效率较高但不具有通用性的解决办法,启发式算法:它针对每个待求解的问题必须明确它的特定的启发式规则。遗传算法兼顾了以上两类算法的优点,具有通用性,高效性,并且可以应对问题规模的动态增长,此外还具有以上两类算法不具备的一些特点:(1)遗传算法以种群作为出发点,从本质上来说是并行的,这是遗传算法最本质的特点。根据进化理论,种群是物种进化的基本单位,种群间的遗传操作是并行的;同时在种群内部,个体间的遗传操作也是并行的,因此具有种群与个体两个层面上的并行。(2)遗传算法具有—智能性Ⅱ,物种在进化中遵守—物竞天择,适者生存Ⅱ的进化法则。适应环境的个体有更大的几率生存下去,并且把自己的基因传给后代;而适应环境能力差的个体生存的几率较小或者被淘汰,因此进化的过程就是一个智能选择优良个体淘汰劣质个体的过程。(3)遗传算法具有灵活性,遗传算法可以通过目标函数和适应度函数来控制种群进化的方向,因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。调度领域是遗传算法的经典应用领域之



		一。在Hadoop作业调度中,用MapReduce处理的数据集必须具备这样的特点:待处
5	<p>此处有 43 字相似</p> <p>向, 因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。</p> <p>9.3 混合遗传布谷鸟算法</p> <p>9.3.1 混合遗传布谷鸟算法模型</p> <p>在基本的布谷鸟算法中, 步长是由levy flight产生的随机长度。</p> <p>因此在搜索过程中, 布谷鸟算法能比较容易跳出局部最优值, 进而增大了搜索到全局最优的概率。然而在局部搜索的过程中由于levy</p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》- 2012-05-21 1 (是否引证: 否)</p> <p>1.t+1)i进行随机改变,反之不变。最后保留测试值较好的一组鸟窝位置<math>y(t+1)_i</math>,此时仍记为<math>x(t+1)_i</math>。2自适应步长布谷鸟搜索算法在基本CS算法中,采用Levy flight产生随机步长,但这种步长时大时小,在搜索过程中,步长越大,越容易搜索全局最优,但同时降低了搜索精度,甚至有时会出现震荡现象;步长越小,</p>
6	<p>此处有 34 字相似</p> <p>适应值较差的值继续应用levy flight随机游走, 保持算法的全局搜索性。</p> <p>9.3.2 混合遗传布谷鸟算法的实现</p> <p>根据第二章的分析, GCS算法的具体过程与步骤如下: 步骤1 (初始化)</p> <p>) : 首先对解空间进行适当的编码, 以便适应遗传算法的交叉遗传的需要。随机生成n个鸟窝, 搜索空间的维数为m, 最大迭代次为it</p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》- 2012-05-21 1 (是否引证: 否)</p> <p>1.应性。因此,只有根据搜索的步长实现自适应动态调整,算法的搜索速度和寻优精度都有较大的提高。</p> <p>3ASCS算法的实施过程与步骤根据第2章分析,ASCS算法的实施过程与步骤如下:步骤1(初始化)随机生成n个鸟窝,搜索空间维数为m,最大迭代次数iter max,最大步长stepmax,最小步长stepmin,并找</p>
7	<p>此处有 69 字相似</p> <p>需要。随机生成n个鸟窝, 搜索空间的维数为m, 最大迭代次为itermax, 适应值较好的阈值为fv (<math>fv \in [0, 1]</math>), 找出当前最优鸟窝位置<math>Xb(0)</math> (<math>b \in \{1, 2, \dots, n\}</math>)和最优解fmax。</p> <p>步骤2 (循环体) : 保留上一代最优鸟窝的位置<math>Xb(t-1)</math>, t为正整数, 表示迭代的代数。对适应值比较好的前p个 (<math>p = n \times fv</math>)解, 应用遗传算法, 使其尽快收敛到局部最优值。取</p>	<p>一种小规模多种群布谷鸟算法 郑巧燕;莫愿斌;刘付永;马彦追; - 《计算机应用与软件》- 2014-10-15 (是否引证: 否)</p> <p>1.步骤1设置鸟窝的个数为n,搜索空间维数为d,初始化鸟窝的位置<math>p_0 = [x(0)_1, x(0)_2, \dots, x(0)_n]^T</math>,并找出最优鸟窝的位置<math>x(0)_b, b \in \{1, 2, \dots, n\}</math>和最优解fmin。步骤2循环体1)位置更新保留上代最优鸟窝的位置<math>x_{t-1b}</math>,t为整数,并利用式(1)对其他的鸟窝位置进行更新,得到一组新的鸟窝的位置,对这组鸟窝的位置进行测试,与上一代产生的一组鸟</p>
8	<p>此处有 81 字相似</p> <p>巢<math>Xb(t)</math>, 并判断是否满足输出条件 (迭代次数达到限制或是其他限制条件)。</p> <p>9.4 算法性能测试与分析</p> <p>9.4.1 实验测试设计</p> <p>本文通过求四个标准测试函数的最小值为例进行仿真实验, 对该算法的性能进行评估, 测试平台为matlab7.1和windows7, 机器的cpu主频为2.10GHz, 内存为4.00 GB。测试函数如下:</p> <p><math>f1(x) = xi^2</math></p> <p><math>f2(x) = 10d + i=1d[xi</math></p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》- 2012-05-21 1 (是否引证: 否)</p> <p>1.优值fmin和对应的全局最优位置<math>x(t)_b</math>。反之,则返回步骤2循环体继续进行迭代更新,直到达到精度为止。4仿真实验4.1实验设计本文求八个标准测试函数的最小值为例,进行仿真实验评估该算法的优化性能,测试平台为Matlab7.1和Win-dows7,机器主频1.90GHz,内存2.00GB。 <math>f1(x)=xi^2</math><math>f2(x)=10d+i=1d[xi^2-10\cos(2\pi xi)]f3(x)</math></p> <p>随机交叉粒子群优化算法 王联国;洪毅; - 《计算机工程与应用》- 2009-06-01 (是否引证: 否)</p> <p>1.条件(通常为达到预定最大进化次数或足够好的适应值),若条件不满足,则转步骤2。否则输出最优解,算法终止</p>

		<p>。4仿真实验4.1实验设计本文以求3个基准测试函数的最小值为例,进行仿真实验来评价随机交叉粒子群优化算法的性能。测试平台为C++,机器主频为P4(1.7G)。</p> <p>(1)<math>f(1X)=D-1i=1\sum(100(x_i^2+1-x_i)^2+(1-x_i)^2)</math>,<math>-100\leq</math></p> <p>基于觅食算子的粒子群优化算法 王联国;洪毅;赵付青; - 《计算机应用与软件》 - 2009-11-15 (是否引证:否)</p> <p>1.通常为达到预定最大进化次数或足够好的适应值),若条件不满足,则转Step 2;否则输出最优解,算法终止。3仿真实验3.1实验设计本文以求6个基准测试函数的最小值为例,进行仿真实验,测试软件平台为Visual C++和Windows XP,机器主频为P4(1.7G),内存为512M。<math>f_1(x)=\sum</math></p> <p>一种多智能体混合蛙跳算法 王联国;代永强; - 《计算机工程》 - 2013-07-15 (是否引证:否)</p> <p>1.条件(通常为达到预定最大进化次数或足够好的适应值),若条件满足,输出最优解,算法终止;否则转Step3。3仿真实验3.1实验设计本文以求6个基准测试函数的最小值为例,进行仿真实验,评价多智能体混合蛙跳算法的优化性能[16]。<math>211()niif x x,12 2 22 11()(100()(1))ni</math></p>
9	<p>此处有 80 字相似</p> <p>windows7, 机器的cpu主频为2.10GHz, 内存为4.00 GB。测试函数如下:</p> <p><math>f_1(x) = x_i^2</math></p> <p><math>f</math></p> <p><math>2(x) = 10d + i=1d[x_i^2-10\cos(2\pi x_i)]</math></p> <p><math>f_3(x) = i=1n-1[100x_i+1-x_i^2+(1-x_i)^2]</math></p> <p><math>f_4(x)</math></p> <p><math>= i=1d x_i^2-10\cos 2\pi x_i+10</math></p> <p>实验中算法参数设置为:鸟窝规模为20, 阈值fV为0.2, 用于遗传算法按照</p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》 - 2012-05-21 1 (是否引证:否)</p> <p>1.化性能,测试平台为Matlab7.1和Windows7,机器主频1.90GHz,内存2.00GB。<math>f_1(x)=x_i^2f_2(x)=10d+i=1d[x_i^2-10\cos(2\pi x_i)]f_3(x)=i=1n-1[100(x_i+1-x_i^2)+(1-x_i)^2]f_4(x)=i=1n[x_i^2-10\cos(2\pi x_i)+10]f_5(x)=0.5+\sin 2x_{12}+x_2-0.5[1+0.001(x_1</math></p>
10	<p>此处有 51 字相似</p> <p>数设置为:鸟窝规模为20, 阈值fV为0.2, 用于遗传算法按照函数的不同设置不同的精度进行编码, 精度按照表3-1所示。使用</p> <p>基本CS算法和GCS算法对上述四个测试函数分别进行寻优测试, 最终测试结果采用独立运行20次后的平均值。</p> <p>当算法的全局最优值收敛到目标精度值时停止算法。</p> <p>表3-1 用于遗传算法的精度</p> <p>函数维数搜索范围理论最优值遗传算法精度</p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权; - 《计算机工程与应用》 - 2012-05-21 1 (是否引证:否)</p> <p>1.法参数设置为:鸟窝规模为20,pa=0.25,步长按公式(2)、公式(3)动态调整,各函数优化的参数和目标精度见表1。用基本CS算法和ASCS算法分别对上述八个测试函数的最小值寻优,最终测试结果采用独立运行20次后的平均值。性能评价采用如下方法:(1)固定进化迭代次数,评估算法收敛速度和精度;固定收敛精度目标值,评估算法达到该精度所需的迭代次</p> <p>基于觅食算子的粒子群优化算法 王联国;洪毅;赵付青; - 《计算机应用与软件》 - 2009-11-15 (是否引证:否)</p> <p>1.ep按式(7)动态调整,s=3;各函数优化的参数和目标精度见表1。用基本PSO算法和基于觅食算子的粒子群优化(PPSO)算法分别对上述六个测试函数的极小值寻优,最终测试结果采用独立运行50次后的平均值。表1用于测试改进算法的基准函数函数维数搜索范围理论最优值目标精度f130[-100,100]0 10-5f230[-60</p> <p>随机交叉粒子群优化算法 王联国;洪毅; - 《计算机工程与应用》 - 2009-06-01 (是否引证:否)</p> <p>1.=2,pc=0.35,随机交叉概率为0.25。用基本粒子群优化</p>

		<p>算法(PSO)和随机交叉粒子群优化算法(SCPPSO)分别对上述3个测试函数的极小值寻优,最终测试结果采用独立运行50次后的平均值。性能评价采用如下方法:(1)固定进化迭代次数,评估算法收敛速度和精度;(2)固定收敛精度目标值,评估算法达到该精度所需要</p> <p>一种多智能体混合蛙跳算法 王联国;代永强;-《计算机工程》-2013-07-15 (是否引证:否)</p> <p>1.()[sin(50())1]ni i iif x x x x用SFLA、文献[8]中的ISFLA和MSFLA分别对上述6个测试函数的最小值进行优化,最终测试结果采用独立运行30次后的平均值。用于测试改进算法的基准函数参数如表1所示。表1用于测试改进算法的基准函数参数函数维数搜索范围理论最优值目标精度f130[</p> <p>基于混沌优化与人工鱼群算法的混合算法研究 石鸿雁;邢东亚;-《微计算机信息》-2012-09-15 (是否引证:否)</p> <p>1.的Visual=2.5,Step=3;函数的Visual=25,Step=3。用基本人工鱼群算法和本文提出算法分别求上述四个测试函数的极小值,最终测试结果采用独立运行20次后的平均值。4.2实验结果及分析固定进化迭代数为50,算法独立运行20次,实验结果如图1、图2和表1所示。图1函数的平均值的进化曲线</p>
11	<p>此处有 32 字相似</p> <p>S在收敛速度上较快,在寻优精度上表现较好。综上所述,改进后的GCS对比CS在寻优精度和收敛速度方面都较为优秀。</p> <p>布谷鸟</p> <p>算法尽管表现优秀,但也存在后期收敛速度较慢、精度有待提高等缺陷。</p> <p>本文提出了一种新的混合遗传布谷鸟算法(GCS)。经过测试表明,改进后的混合遗传布谷鸟算法在后期收敛速度和寻优精度上有所提</p>	<p>一种自适应步长布谷鸟搜索算法 郑洪清;周永权;-《计算机工程与应用》-2012-05-21 1 (是否引证:否)</p> <p>1.少、易于实现,并成功应用于工程优化等实际问题中[5-7],逐渐发展成为群智能算法领域的一个新亮。但CS算法同其他一些群智能算法一样,也存在后期搜速度慢、精度不高等缺点。针对这些不足,本文提出一自适应步长布谷鸟搜索算法(Self-Adaptive Step Cuckooearch algo</p>
12	<p>此处有 37 字相似</p> <p>后对该算法的优势与缺点进行了阐述。针对其后期收敛较慢,局部计算精度不高的缺点,提出结合遗传算法进行修正。最后将布谷鸟算法</p> <p>进行改进,并且通过实验对改进的布谷鸟算法与原有的布谷鸟算法进行了性能比较,</p> <p>结果显示改进的布谷鸟算法的性能要优于原本的布谷鸟算法。</p>	<p>求解无约束优化问题的改进布谷鸟搜索算法 苏英华;刘云连;伍铁斌;-《计算机工程》-2014-05-15 (是否引证:否)</p> <p>1.部寻优,令t=t+1,返回Step3。4数值实验为评估本文提出的改进布谷鸟搜索(MCS)算法的性能,选取4个标准测试函数进行仿真实验,并与标准布谷鸟搜索(CS)算法进行比较,各测试函数的主要特征如表1所示。表1 4个标准测试函数函数名函数维数搜索范围理论最优值Sphere f1(x)20[-1</p>

指 标
疑似剽窃文字表述
1. 类似随机搜索的方式。为了模拟和简化布谷鸟寻巢的方式,规定以下3个理想的状态:(1)每只布谷鸟每次只产一个卵,后来随着相关理论著作的出版,以及在机器学习和性能优化等领域的应用,该模型理论正式定名为遗传算法。
9.2.2 遗传算法的原理
9.2.3 遗传算法的特点
遗传算法是一种优化算法,因而需要满足最优性原理。最优性原理是针对现实生活中的这一类问题:这个问题可被分为若干个子问题,对于任意的子问题t都只依赖于当前阶段的状态,而与t之前的所有其它子问题的状态均无关,这是一种



多阶段决策的问题。这类问题中一种最通用但是低效的方法是枚举法，该方法的基本思想是穷举解集范围内的所有解，然后依据求解目标找出最优解；这种算法的一种改进算法是搜索算法，它的基本思想是把搜索范围缩小到解集的某一个子集或几个子集的范围内，通过缩小搜索域的方式可以大大的减少计算量；此外，这类问题还有另外一种效率较高但不具有通用性的解决办法，启发式算法：它针对每个待求解的问题必须明确它的特定的启发式规则。遗传算法兼顾了以上两类算法的优点，具有通用性，高效性，并且可以应对问题规模的动态增长，此外还具有以上两类算法不具备的一些特点：

- (1) 遗传算法以种群作为出发点，从本质上来说是并行的，这是遗传算法最本质的特点。根据进化理论，种群是物种进化的基本单位，种群间的遗传操作是并行的；同时在种群内部，个体间的遗传操作也是并行的，因此具有种群与个体两个层面上的并行。
- (2) 遗传算法具有智能性，物种在进化中遵守物竞天择，适者生存的进化法则。适应环境的个体有更大的几率生存下去，并且把自己的基因传给后代；而适应环境能力差的个体生存的几率较小或者被淘汰，因此进化的过程就是一个智能选择优良个体淘汰劣质个体的过程。
- (3) 遗传算法具有灵活性，遗传算法可以通过目标函数和适应度函数来控制种群进化的方向，因此在实际问题中可通过修改控制这两个函数来应对不同的应用场景。

### 3. 布谷鸟算法模型

在基本的布谷鸟算法中，步长是由levy flight产生的随机长度。

### 4. 实验测试设计

本文通过求四个标准测试函数的最小值为例进行仿真实验，对该算法的性能进行评估，测试平台为matlab7.1和windows7，机器的cpu主频为2.

### 5. 基本CS算法和GCS算法对上述四个测试函数分别进行寻优测试，最终测试结果采用独立运行20次后的平均值。

## 11. 80022497643220302\_基于动态手势的智能终端身份识别技术研究与实践\_第11部分 总字数：7366

相似文献列表 文字复制比：75.3%(5548) 疑似剽窃观点：(0)

1	Hadoop YARN资源分配与调度的研究 李媛祯(导师：杨群) - 《南京航空航天大学硕士论文》 - 2015-01-01	75.0% ( 5527 ) 是否引证：否
2	一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15	8.3% ( 613 ) 是否引证：否
3	Hadoop Yarn 框架原理及运作机制 - Focus,Just For Today - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	2.7% ( 198 ) 是否引证：否
4	马鹏将 马鹏将 - 《学术论文联合比对库》 - 2015-01-04	2.0% ( 149 ) 是否引证：否
5	李芳林_并行推荐算法的研究与实践 李芳林 - 《学术论文联合比对库》 - 2015-01-02	2.0% ( 149 ) 是否引证：否
6	大数据计算框架及其资源调度机制研究 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2016	2.0% ( 147 ) 是否引证：否
7	Hadoop YARN大数据计算框架及其资源调度机制研究 董春涛;李文婷;沈晴霓;吴中海; - 《信息通信技术》 - 2015-02-15	1.3% ( 98 ) 是否引证：否
8	SA12225077_徐磊_1 徐磊 - 《学术论文联合比对库》 - 2014-08-19	1.2% ( 87 ) 是否引证：否
9	SA12225077徐磊 徐磊 - 《学术论文联合比对库》 - 2015-01-04	1.2% ( 87 ) 是否引证：否
10	573+刘盼红 - 《学术论文联合比对库》 - 2015-03-24	0.9% ( 65 ) 是否引证：否
11	SA12225077_徐磊_2 徐磊 - 《学术论文联合比对库》 - 2014-09-18	0.7% ( 50 ) 是否引证：否
12	152130200058.徐辉.学术硕士 - 《学术论文联合比对库》 - 2015-07-27	0.7% ( 49 ) 是否引证：否
13	基于Hadoop的密度聚类算法并行化分析与研究 徐辉 - 《学术论文联合比对库》 - 2015-05-19	0.7% ( 49 ) 是否引证：否
14	基于Hadoop的密度聚类算法并行化分析与研究 徐辉 - 《学术论文联合比对库》 - 2015-06-08	0.7% ( 49 ) 是否引证：否
15	基于Hadoop的密度聚类算法并行化分析与研究 徐辉 - 《学术论文联合比对库》 - 2015-06-09	0.7% ( 49 ) 是否引证：否
16	基于Hadoop的密度聚类算法并行化分析与研究 徐辉 - 《学术论文联合比对库》 - 2015-06-09	0.7% ( 49 ) 是否引证：否



17	yarn的工作流程 - 军军的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	0.6% ( 41 ) 是否引证：否
18	Spark入门实战系列--4.Spark运行架构 - yirenboy的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	0.5% ( 38 ) 是否引证：否
19	并行图挖掘算法的研究与实现 王虹旭(导师：吴斌) - 《北京邮电大学硕士论文》 - 2015-03-10	0.4% ( 31 ) 是否引证：否
20	王虹旭-2012110718-并行图挖掘算法的研究与实现-0112 王虹旭 - 《学术论文联合比对库》 - 2015-01-12	0.4% ( 31 ) 是否引证：否
21	面向多用户环境的MapReduce集群调度算法研究 陈重韬; - 《高技术通讯》 - 2017-04-15	0.4% ( 31 ) 是否引证：否

	原文内容	相似内容来源
1	<p>此处有 2294 字相似</p> <p>十章基于GCS的Hadoop作业调度算法的设计与实现</p> <p>10.1 Hadoop资源调度介绍</p> <p>10.1.1 Hadoop资源调度模型分析</p> <p>Hadoop2.x支持内存和CPU两种不同资源类型的管理和分配，资源调度器采用动态资源分配机制，通过处理由应用程序提交的Request请求来分配这两种资源。</p> <p>当NM通过ResourceTracker协议向RM注册、汇报节点状态时，它会提交本节点中可分配的CPU和内存资源总量。在系统中默认值是每个NM具有8GB内存，8个虚拟CPU核心。NM周期性地主动向RM发出请求并汇报资源信息，而应用程序通过ResourceRequest向RM请求资源，一个请求包含了以下字段：</p> <p>( 1 ) Priority：表示资源优先级（0优先级最大，数字越大优先级越小）。</p> <p>( 2 ) ResourceName：表示请求资源的位置（*：表示任意位置资源；/rack-name：在这个机架上任意一个节点申请资源；/host-name：在某个节点申请资源）。</p> <p>( 3 ) Capability：资源的规格，包括上面提到的两种资源的量（例如：2GB内存，一个虚拟CPU核）。</p> <p>( 4 ) Num_Container：请求满足上面资源规格描述的Container数量。</p> <p>( 5 ) Relax_locality：是否强制要求本地性。因为AM会根据数据副本的数量提交多个资源请求，在响应ResourceRequest时，会按照本地作业（NODE_LOCAL），本机架作业（RACK_LOCAL）和任意位置（OFF_SWITCH）的顺序进行响应。这样有一个被响应就会使得对应的较大范围的请求被撤回（例如本地作业请求被响应则本机架作业请求和任意位置请求被撤回）。</p> <p>调度器会根据一系列条件进行调度，首先是选择队列，由于队列具有以下两种属性，调度器需要考虑这两种属性来确定队列：</p> <p>( 1 ) 最小容量：调度器需要努力满足这个最小容量的值，在一个多队列的环境下，调度器会选择队列实际容量和最小容量差值最大的那些队列，把资源分配给这些队列，这样做是为了保证这些队列内的作业能够保持执行的效率。当一条队列空转时，其资源会分配到其他队列供其他队列的应用使用，直到此队列再一次有应用提交并申请资源。</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证：否 )</p> <p>1.收敛速度,本章算法正是利用粒子群算法快速收敛的优势,提高了蚁群算法的收敛速度,弥补了蚁群算法收敛速度慢的缺陷。3.2.3资源调度的请求与模型Hadoop 2.x支持内存和CPU两种不同资源类型的管理和分配,资源调度器采用动态资源分配机制,通过处理由应用程序提交的Request请求来分配这两种资源。当NM通过ResourceTracker协议向RM注册、汇报节点状态时,它会提交本节点中可分配的CPU和内存资源总量。在系统中默认值是每个NM具有8GB内存,8个虚拟CPU核心。NM周期性地主动向RM发出请求并汇报资源信息,而应用程序通过ResourceRequest向RM请求资源,一个请求包含了以下字段:(1)Priority:表示资源优先级(0优先级最大,数字越大优先级越小)。(2)ResourceName:表示请求资源的位置(*:表示任意位置资源;/rack-name:在这个机架上任意一个节点申请资源;/host-name:在某个节点申请资源)。(3)Capability:资源的规格,包括上面提到的两种资源的量(例如:2GB内存,一个虚拟CPU核)。(4)Num_Container:请求满足上面资源规格描述的Container数量。(5)Relax_locality:是否强制要求本地性。因为AM会根据数据副本的数量提交多个资源请求,在响应ResourceRequest时,会按照本地作业(NODE_LOCAL),本机架作业(RACK_LOCAL)和任意位置(OFF_SWITCH)的顺序进行响应。这样有一个被响应就会使得对应的较大范围的请求被撤回(例如本地作业请求被响应则本机架作业请求和任意位置请求被撤回)。调度器会根据一系列条件进行调度,首先是选择队列,由于队列具有以下两种属性,调度器需要考虑这两种属性来确定队列:(1)最小容量:调度器需要努力满足这个最小容量的值,在一个多队列的环境下,调度器会选择队列实际容量和最小容量差值最大的那些队列,把资源分配给这些队列,这样做是为了保证这些队列内的作业能够保持执行的效率。当一条队列空转时,其资源会分配到其他队列供其他队列的应用使用,直到此队列再一次有应用提交并申请资源。(2)最大容量:在资源情况比较紧张时,队列也会设置一个最大容量。最大容量是队列的一个硬性要求值。在任何时刻,队列的资源占有量都不能超过这个值。一般而言,为了提高队列效率,这个值不会被设置,这样,一个比较繁忙队列能够在其他队列都不工作时完全使用集群资源(占用100%的资源),在其他队列有应用时再逐步释放。在队列中,存放了该队列的用户提交的应用,调度器会</p>

(2) 最大容量：在资源情况比较紧张时，队列也会设置一个最大容量。最大容量是队列的一个硬性要求值。在任何时刻，队列的资源占有量都不能超过这个值。一般而言，为了提高队列效率，这个值不会被设置，这样，一个比较繁忙队列能够在其他队列都不工作时完全使用集群资源（占用100%的资源），在其他队列有应用时再逐步释放。在队列中，存放了该队列的用户提交的应用，调度器会根据资源情况和应用的请求响应资源请求。新的HadoopYARN实现的是两层调度机制：在第一层中，上述的ResourceRequest会被应用程序提交到RM中，而调度器的唯一工作就是结合由应用提交的ResourceRequest和各节点的资源情况，确定是否对ResourceRequest进行响应，如果响应这些请求则会在节点上启动一个到多个Container；如果没有响应则在下一次节点状态更新的时候再判断是否需要响应这些请求。在第二层中，应用程序的AM从RM中定期取回分配到的Container，接着在应用内部将这些Container分配给自己要完成的任务。在HadoopYARN中，分配资源要经过以下步骤：

- (1) NM用心跳信息向RM汇报节点情况。
- (2) RM中的ResourceTracker服务会返回一个应答，包含要启动和释放Container的信息。
- (3) 服务向调度器转发NODE\_UPDATE事件。
- (4) 调度器把释放的资源重新标记为可用，再按照一定策略将节点上的资源分配给应用程序。
- (5) AM向RM发送心跳信息。
- (6) RM把分配的Container应答给AM。
- (7) AM把Container分配给自己内部的任务。

设计调度器的关键就是完成第四步的策略，选择合适的调度策略进行资源分配，可以使得资源被更为合理的使用，从而提高应用的执行效率。

#### 10.1.2 资源调度存在的问题

资源调度是Hadoop中一个重要的研究领域。不同于Hadoop0.x中的JobTracker组件，HadoopYARN中的资源调度器是一个“纯调度器”，不再从事任何与具体应用程序相关的工作，如不负责监控或跟踪应用程序的执行状态，不负责重新启动因应用程序执行失败或硬件故障而产生的失败任务等，而是仅仅负责根据各个应用程序的资源需求对整个集群的资源进行统一的管理和分配。资源分配的单位用“资源容器”（Container）表示，Container是一个资源分配单位，它将内存与CPU资源封装在一起以限定每个任务使用的资源量，所以如何对资源进行合理的分配，使得哪个资源分配给哪个Container执行，关系到整个集群中的作业执行效率。当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序，首先是启动AM，其次是由AM创建应用程序，并为它申请资源，监控整个过程直到运行完成。在AM资源申请过程中，调度器采用轮询的方式为其分配资源，而并未考虑到是否有其他更优的资源可供AM使用。图4.1描述了当前Hadoop调度器的资源分配策略。当AM A向RM中的Scheduler组件申请Container时，假设当前轮询的位置的节点Node2且Node2中可分配资源满足AM A所需资源量

根据资源情况和应用的请求响应资源请求。新的Hadoop YARN实现的是两层调度机制：在第一层中，上述的ResourceRequest会被应用程序提交到RM中，而调度器的唯一工作就是结合由应用提交的ResourceRequest和各节点的资源情况，确定是否对ResourceRequest进行响应，如果响应这些请求则会在节点上启动一个到多个Container；如果没有响应则在下一次节点状态更新的时候再判断是否需要响应这些请求。在第二层中，应用程序的AM从RM中定期取回分配到的Container，接着在应用内部将这些Container分配给自己要完成的任务。在Hadoop YARN中，分配资源要经过以下步骤：(1) NM用心跳信息向RM汇报节点情况。(2) RM中的ResourceTracker服务会返回一个应答，包含要启动和释放Container的信息。(3) 服务向调度器转发NODE\_UPDATE事件。(4) 调度器把释放的资源重新标记为可用，再按照一定策略将节点上的资源分配给应用程序。(5) AM向RM发送心跳信息。(6) RM把分配的Container应答给AM。(7) AM把Container分配给自己内部的任务。设计调度器的关键就是完成第四步的策略，选择合适的调度策略进行资源分配，可以使得资源被更为合理的使用，从而提高应用的执行效率。

#### 3.2.4 资源调度存在的问题

资源调度是Hadoop中一个重要的研究领域。不同于Hadoop 0.x中的JobTracker组件，Hadoop YARN中的资源调度器是一个“纯调度器”，不再从事任何与具体应用程序相关的工作，如不负责监控或跟踪应用程序的执行状态，不负责重新启动因应用程序执行失败或硬件故障而产生的失败任务等，而是仅仅负责根据各个应用程序的资源需求对整个集群的资源进行统一的管理和分配。资源分配的单位用“资源容器”（Container）表示，Container是一个资源分配单位，它将内存与CPU资源封装在一起以限定每个任务使用的资源量，所以如何对资源进行合理的分配，使得哪个资源分配给哪个Container执行，关系到整个集群中的作业执行效率。当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序，首先是启动AM，其次是由AM创建应用程序，并为它申请资源，监控整个过程直到运行完成。在AM资源申请过程中，调度器采用轮询的方式为其分配资源，而并未考虑到是否有其他更优的资源可供AM使用。图3.1描述了当前Hadoop调度器的资源分配策略。当AM A向RM中的Scheduler组件申请Container时，假设当前轮询的位置的节点Node2且Node2中可分配资源满足AM A所需资源量，Scheduler将Node2中的资源分配给AM A，然而从图3.1可以看出，相对于Node2，Node 3的资源负载更小CPU处理速度更快，故将Node 3中资源分配给AM A更合理。ResourceInformationSchedulerNode 1Load:0.8Data:AM A/BCPU:1.8

Hadoop Yarn 框架原理及运作机制 - Focus,Just For Today  
- 博客频道 - CSDN.NET - 《网络 ( <http://blog.csdn.net> ) 》 -  
( 是否引证：否 )

1.件（如每个队列分配一定的资源，最多执行一定数量的作业等），将系统中的资源分配给各个正在运行的应用程序。需要注意的是，该调度器是一个“纯调度器”，它不再从事任何与具体应用程序相关的工作，比如不

<p>，Scheduler将Node2中的资源分配给AM A，然而从图4.1可以看出，相对于Node2，Node3的资源负载更小CPU处理速度更快，故将Node3中资源分配给AM A更合理。</p> <p>对于Hadoop集群资源调度问题，可以把Container看作是一个关于资源分配的映射目标，可行解空间就是资源分配策略的</p>	<p>负责监控或者跟踪应用的执行状态等，也不负责重新启动因应用执行失败或者硬件故障而产生的失败任务，这些均交由应用程序相关的ApplicationMaster完成。调度器仅根据各个应用程序的资源需求进行资源分配，而资源分</p> <p>2.的ApplicationMaster完成。调度器仅根据各个应用程序的资源需求进行资源分配，而资源分配单位用一个抽象概念“资源容器”(Resource Container，简称Container)表示，Container是一个动态资源分配单位，它将内存、CPU、磁盘、网络等资源封装在一起，从而限定每个任务使用的资源量。此外，该调度器是一个可插拔的组件，用户可根据自己的需要设计新的调度器，YARN提供了多种直接可用的调度器，比如Fair</p> <p>3.本书完成时，YARN仅支持CPU和内存两种资源，且使用了轻量级资源隔离机制Cgroups进行资源隔离。</p> <p>1.3 YARN工作流程 当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序：第一个阶段是启动ApplicationMaster；第二个阶段是由ApplicationMaster创建应用程序，为它申请资源，并监控</p>
	<p>李芳林 并行推荐算法的研究与实现 李芳林 - 《学术论文联合比对库》 - 2015-01-02 (是否引证：否)</p> <p>1.条件(如每个队列分配一定的资源，最多执行一定数量的作业等)，将系统中的资源分配给各个正在运行的应用程序。需要注意的是，该调度器是一个“纯调度器”，它不再从事任何与具体应用程序相关的工作，比如负责监控或者跟踪应用的执行状态等，它也不负责重新启动因应用的执行失败或者硬件故障而产生的失败任务，这些均交由应用程序相关的ApplicationMaster去完成。调度器仅根据各个应用程序的资源需求进行资源分配，而资源</p> <p>2.ApplicationMaster去完成。调度器仅根据各个应用程序的资源需求进行资源分配，而资源分配单位用一个抽象概念“资源容器”(Resource Container，简称Container)表示，Container是一个动态资源分配单位，它将内存、CPU、磁盘、网络等资源封装在一起，从而限定每个任务使用的资源量。此外，该调度器是一个可插拔的组件，用户可根据自己的需要设计新的调度器，YARN提供了多种可直接使用的调度器，比如Fair</p>
	<p>马鹏将 马鹏将 - 《学术论文联合比对库》 - 2015-01-04 (是否引证：否)</p> <p>1.条件(如每个队列分配一定的资源，最多执行一定数量的作业等)，将系统中的资源分配给各个正在运行的应用程序。需要注意的是，该调度器是一个“纯调度器”，它不再从事任何与具体应用程序相关的工作，比如负责监控或者跟踪应用的执行状态等，它也不负责重新启动因应用的执行失败或者硬件故障而产生的失败任务，这些均交由应用程序相关的ApplicationMaster去完成。调度器仅根据各个应用程序的资源需求进行资源分配</p>



	<p>，而资源</p> <p>2.ApplicationMaster去完成。调度器仅根据各个应用程序的资源需求进行资源分配，而资源分配单位用一个抽象概念“资源容器” ( Resource Container，简称 Container ) 表示，Container是一个动态资源分配单位，它将内存、CPU、磁盘、网络等资源封装在一起，从而限定每个任务使用的资源量。此外，该调度器是一个可插拔的组件，用户可根据自己的需要设计新的调度器，YARN提供了多种可直接使用的调度器，比如Fair</p>
	<p>大数据计算框架及其资源调度机制研究 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2016-8-3 0:47:59 ( 是否引证：否 )</p>
	<p>1.container以心跳应答的形式返回给ApplicationMaster。ApplicationMaster收到新分配的Container列表后,将这些Container分配给内部任务。3.2层级队列管理机制用户和资源管理机制是任何资源调度器的基础。在YARN中,用户以层级队列的形式组织。该队列组织方式</p> <p>2.点上的资源,当AM向RM申请资源时,RM为AM返回的资源以Container表示ResourceManager的资源调度器是一个“纯调度器”,它不再从事任何与具体应用程序相关的工作,比如不负责监控或者跟踪应用的执行状态,也不负责重新启动运行失败的任务,这些均交由ApplicationMaster完成。1.2 YARN的工作流程当用户向YARN提交应用程序时,YARN分</p> <p>3.踪应用的执行状态,也不负责重新启动运行失败的任务,这些均交由ApplicationMaster完成。1.2 YARN的工作流程当用户向YARN提交应用程序时,YARN分两个阶段运行该应用程序,第一个阶段启动ApplicationMaster,第二个阶段由ApplicationMaster为应用程序申请资源,并监控整个运行过程</p>
	<p>Hadoop YARN大数据计算框架及其资源调度机制研究 董春涛;李文婷;沈晴霓;吴中海; - 《信息技术》 - 2015-02-15 ( 是否引证：否 )</p>
	<p>1.tainer以心跳应答的形式返回给Application Master。Application Master收到新分配的Container列表后,将这些Container分配给内部任务。3.2层级队列管理机制用户和资源管理机制是任何资源调度器的基础。在YARN中,用户以层级队列的形式组织。该队列组织方式具</p> <p>2.资源时,RM为AM返回的资源以Container表示ResourceManager的资源调度器是一个“纯调度器”,它不再从事任何与具体应用程序相关的工作,比如不负责监控或者跟踪应用的执行状态,也不负责重新启动运行失败的任务,这些均交由Application Master完成。1.2 YARN的工作流程当用户向YARN提交应用程序时,Y</p>
	<p>SA12225077_徐磊_1 徐磊 - 《学术论文联合比对库》 - 2014-08-19 ( 是否引证：否 )</p>



		<p>1.r, ASM)。调度器是根据容量、队列等限制条件,将系统中的资源分配给各个正在运行的应用程序。需要注意的是,该调度器是个“单纯的”调度器,不再从事任何与具体应用程序相关的工作,比如不负责监控或者跟踪应用的执行状态。调度器是个可插拔的组件,用户可根据自己的需要设计新的调度器,YARN 提供了多种直接可用的调度器,比如 Fa</p> <p>2.YARN 仅支持 CPU 和内存两种资源,并且使用了轻量级资源隔离机制 Cgroups进行资源隔离[14]。当用户向 YARN 提交个应用后,YARN 将分两个阶段运行该应用程序:第一个阶段是启动 ApplicationMaster;第二个阶段是用 ApplicationMaster 创建应</p>
		<p>SA12225077徐磊 徐磊 -《学术论文联合比对库》- 2015-01-04 (是否引证:否)</p> <p>1.ter, ASM)。调度器是根据容量、队列等限制条件,将系统中的资源分配给各个正在运行的应用程序。需要注意的是,该调度器是个“单纯的”调度器,不再从事任何与具体应用程序相关的工作,比如不负责监控或者跟踪应用的执行状态。摘要Incremental computingframework under heterogeneouspl</p> <p>2.RN 仅支持 CPU 和内存两种资源,并且使用了轻量级资源隔离机制 Cgroups进行资源隔离[14]。当用户向 YARN 提交个应用后,YARN 将分两个阶段运行该应用程序:第一个阶段是启动 ApplicationMaster;第二个阶段是用 ApplicationMaster 创建应</p>
		<p>SA12225077_徐磊 2 徐磊 -《学术论文联合比对库》- 2014-09-18 (是否引证:否)</p> <p>1.度器是根据容量、队列等限制条件,将系统中的资源分配给各个正在运行的应用程序。需要注意的是,该调度器是一个“单纯的”调度器,不再从事任何与具体应用程序相关的工作,比如不负责监控或者跟踪应用的执行状态。调度器是一个可插拔的组件,用户可根据自己的需要设计新的调度器,YARN 提供了多种直接可用的调度器,比如 F</p>
		<p>基于Hadoop的密度聚类算法并行化分析与研究 徐辉 -《学术论文联合比对库》- 2015-05-19 (是否引证:否)</p> <p>1.YARN的基本架构Fig. 2-5 The basic architecture of YARN(2) YARN工作流程当用户向YARN中提交一个应用程序后,YARN将分两个阶段运行该应用程序:第一个阶段是启动ApplicationMaster;第二个阶段是由 ApplicationMaster创建应用程序,为它申请资源,并监控它</p>
		<p>基于Hadoop的密度聚类算法并行化分析与研究 徐辉 -《学术论文联合比对库》- 2015-06-08 (是否引证:否)</p> <p>1.YARN的基本架构Fig. 2-5 The basic architecture of YARN(2) YARN工作流程当用户向YARN中提交一个应用程序后,YARN将分两个阶段运行该应用程序:第一个阶段是启动ApplicationMaster;第二个阶段是由 ApplicationMaster创建应用程序,为它申请资源,并监</p>

		控它
		基于Hadoop的密度聚类算法并行化分析与研究 徐 辉 - 《学术论文联合比对库》 - 2015-06-09 ( 是否引证：否 )
		1.YARN的基本架构Fig. 2-5 The basic architecture of YARN(2) YARN工作流程当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序：第一个阶段是启动ApplicationMaster；第二个阶段是由ApplicationMaster创建应用程序，为它申请资源，并监控它
		基于Hadoop的密度聚类算法并行化分析与研究 徐 辉 - 《学术论文联合比对库》 - 2015-06-09 ( 是否引证：否 )
		1.YARN的基本架构Fig. 2-5 The basic architecture of YARN(2) YARN工作流程当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序：第一个阶段是启动ApplicationMaster；第二个阶段是由ApplicationMaster创建应用程序，为它申请资源，并监控它
		152130200058.徐辉.学术硕士 - 《学术论文联合比对库》 - 2015-07-27 ( 是否引证：否 )
		1.YARN的基本架构Fig. 2-5 The basic architecture of YARN(2) YARN工作流程当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序：第一个阶段是启动ApplicationMaster；第二个阶段是由ApplicationMaster创建应用程序，为它申请资源，并监控它
		Spark入门实战系列--4.Spark运行架构 - yirenboy的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )
		1.ourceManager申请注销并关闭自己。 2.2.3 YARN-Cluster 在YARN-Cluster模式中，当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序：第一个阶段是把Spark的Driver作为一个ApplicationMaster在YARN集群中先启动；第二个阶段是由App
		yarn的工作流程 - 军军的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )
		1.用不同，一类直接运行数据处理程序，一类用于部署服务（服务之上再运行数据处理程序），但运行在YARN上的流程是相同的。当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序： 1 ) 启动ApplicationMaster；2 ) 由ApplicationMaster创建应用程序，为它申请资源，并监控它的整个运
		王虹旭-2012110718-并行图挖掘算法的研究与实现-0112 王虹旭 - 《学术论文联合比对库》 - 2015-01-12 ( 是否引证：否 )
		1.制条件（如每个队列对于有限的资源，执行作业的数量有限等），调度器将给系统中各个正在运行的应用程序分配资源。我们注意到，该调度器是一个“纯调度器”，它不再从事任何与具体应用程序相关的工作，比如对应用的执行状态进行监控和跟踪，对因应用的执行失败或者硬件故障而产生的失败任务的重新启动，这些均交

		<p>由应用程序相关的A</p> <p>并行图挖掘算法的研究与实现 王虹旭 - 《北京邮电大学硕士论文》 - 2015-03-10 (是否引证:否)</p> <p>1.个队列对于有限的资源,执行作业的数量有限等),调度器将给系统中各个正在运行的应用程序分配资源。我们注意到,该调度器是“一个纯调度器”,它不再从事任何与具体应用程序相关的工作,比如对应用的执行状态进行监控和跟踪,对因应用的执行失败或者硬件故障而产生的失败任务的重新启动,这些均交由应用</p>
2	<p>此处有 142 字相似</p> <p>1可以看出,相对于Node2,Node3的资源负载更小CPU处理速度更快,故将Node3中资源分配给AM A更合理。</p> <p>对于Hadoop集群资源调度问题,可以把Container看作是一个关于资源分配的映射目标,可行解空间就是资源分配策略的集合。在可行解空间中,如何为每一个Container提高最优的资源,这是一个无法确定的问题,同时由于无法确定最优解,使得资源的分配问题是一个NP-Hard问题。</p> <p>10.1.3 合理调度分配资源的重要性</p> <p>当前,Hadoop 技术越来越成熟,资源调度问题也越来越重要。在 Hadoop</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证:否)</p> <p>1.e 4Load:0.8Data:AM A/BCPU:1.8AM AAM AAM A图</p> <p>3.1 Hadoop调度器资源分配图对于Hadoop集群资源调度问题,可以把Container看作是一个关于资源分配的映射目标,可行解空间就是资源分配策略的集合。在可行解空间中,如何为每一个Container提高最优的资源,这是一个无法确定的问题,同时由于无法确定最优解,使得资源的分配问题是一个NP-hard问题。3.3 Hadoop资源调度算法原理3.3.1合理分配与调度资源的重要性当前,Hadoop技术越来越成熟,资源调度问题也</p>
3	<p>此处有 539 字相似</p> <p>优的资源,这是一个无法确定的问题,同时由于无法确定最优解,使得资源的分配问题是一个NP-Hard问题。</p> <p>10.1.3</p> <p>合理调度分配资源的重要性</p> <p>当前,Hadoop 技术越来越成熟,资源调度问题也越来越重要。在 Hadoop YARN 中,集群资源以大小不等的Container 为单位进行分配,随着集群规模越来越大,集群中各个节点不可能是同构的,同一作业在不同计算能力的节点资源上执行时,其执行效率有很大的差异,所以选择一个合适的节点为 Container 分配资源非常重要。Hadoop 集群中,影响作业执行时间的因素有很多,但其中一个重要的因素为运行作业中所属任务的资源的性能。如果一个作业中的任务被分配到一个性能相对较差的节点上,如负载高、CPU 处理速度慢或当前节点上没有运行任务所需的数据副本,此时执行该任务需要花费较长的时间,这在一定程度上大大的延长了作业的执行时间,甚至造成了集群资源的浪费。针对上述问题,本章首先分析 Hadoop 资源调度的请求机制与模型,详细剖析了资源调度中存在的问题,在此基础上,引入蚁群算法和粒子群算法,实现 Hadoop 资源的合理化分配。最终,根据本章算法编写相应的 Hadoop 资源调度器,并进行实验验证,实验表明本章调度器能够有效的提高集群资源分配的合理化,达到缩短作业执行时间的目的。</p> <p>10.2 基于GCS的资源调度算法设计</p> <p>10.</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证:否)</p> <p>1.问题,同时由于无法确定最优解,使得资源的分配问题是一个NP-hard问题。3.3 Hadoop资源调度算法原理3.3.1合理分配与调度资源的重要性当前,Hadoop技术越来越成熟,资源调度问题也越来越重要。在Hadoop YARN中,集群资源以大小不等的Container为单位进行分配,随着集群规模越来越大,集群中各个节点不可能是同构的,同一作业在不同计算能力的节点资源上执行时,其执行效率有很大的差异,所以选择一个合适的节点为Container分配资源非常重要。Hadoop集群中,影响作业执行时间的因素有很多,但其中一个重要的因素为运行作业中所属任务的资源的性能。如果一个作业中的任务被分配到一个性能相对较差的节点上,如负载高、CPU处理速度慢或当前节点上没有运行任务所需的数据副本,此时执行该任务需要花费较长的时间,这在一定程度上大大的延长了作业的执行时间,甚至造成了集群资源的浪费。针对上述问题,本章首先分析Hadoop资源调度的请求机制与模型,详细剖析了资源调度中存在的问题,在此基础上,引入蚁群算法和粒子群算法,实现Hadoop资源的合理化分配。最终,根据本章算法编写相应的Hadoop资源调度器,并进行实验验证,实验表明本章调度器能够有效的提高集群资源分配的合理化,达到缩短作业执行时间的目的。3.3.2基于ACO与PSO的资源调度算法调度模型本节将蚁群算法应用于资源调度问题,算法各参数与资源调度各参数的介绍如下。集群环境模型记为<math>G=\{A,R,J,C\}</math>,</p>

	<p>2.1 Hadoop资源调度模型分析</p> <p>本节将混合遗传布谷鸟算法应用于Hadoop资源调度问题，Hadoop集群模型化描</p>	<p>面向多用户环境的MapReduce集群调度算法研究 陈重韬; -《高技术通讯》- 2017-04-15 (是否引证:否)</p> <p>1.确定各资源池的权重分配剩余资源,最终使集群资源分配更加合理。实验结果表明,运用面向多用户环境的MapReduce集群作业调度算法,能够有效地缩短集群作业的整体执行时间,提高集群利用率。(1)(2)男,1986年生,博士生;研究方向:海量数据的存储与处理,联系人,E-mail:chenzh</p>
4	<p>此处有 466 字相似</p> <p>如下:</p> <p>集群环境模型记为<math>G=\{E,R,J,C\}</math>,其中,<math>E</math>为布谷鸟所产的卵的位置的集合,即可行解的集合;<math>R</math>为Hadoop节点资源集合;<math>J</math>为待调度作业集;<math>C</math>为容器(container)集合,资源集合<math>R=\{R_1,R_2,\dots,R_n\}</math>由<math>n</math>个节点及其资源构成,其中,第<math>i</math>个节点上的资源<math>R_i=\{cpuSpeed_i,M_i,resMi,load_i\}</math>。<math>cpuSpeed_i</math>为节点<math>i</math>上的CPU运行速率,节点<math>i</math>的总资源量用<math>M_i</math>表示,<math>resMi</math>为节点<math>i</math>的空闲资源量,<math>resMi \leq M_i</math>,<math>0 \leq i &lt; n</math>,<math>load_i</math>表示节点<math>i</math>的负载。作业集<math>J=\{J_1,J_2,\dots,J_s\}</math>表示当前集群上运行的作业量为<math>s</math>,第<math>j</math>个作业<math>J_j=\{resTotalJ_j,resMJ_j,prog_j\}</math>,<math>0 \leq j &lt; s</math>,其中,<math>resTotalJ_j</math>为<math>J_j</math>所需资源的总量;<math>resMJ_j</math>为分配给<math>J_j</math>的实际资源量;<math>prog_j</math>表示<math>J_j</math>的运行进度。容器集<math>C=\{C_1,C_2,\dots,C_t\}</math>表示集群中所有申请了资源的容器集合,第<math>m</math>个容器<math>C_m=\{Jm_j,resMCm\}</math>,<math>0 \leq j &lt; s</math>,<math>0 \leq m &lt; t</math>,<math>Jm_j</math>为<math>C_m</math>所属的作业,<math>resMCm</math>为容器<math>C_m</math>所申请的资源量。对容器进行资源分配时,对任意<math>R_i \in R</math>,<math>C_m \in C</math>,如果资源<math>R_i</math>可以满足容器<math>C_m</math>所申请的资源量,则允许<math>R_i</math>为容器<math>C_m</math>分配执行任务所需要的资源。满足容器<math>C_m</math>的资源<math>R_i</math>可能不止一个。容器选择资源节点可以表示成为一个</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 -《南京航空航天大学硕士论文》- 2015-01-01 (是否引证:否)</p> <p>1.于资源调度问题,算法各参数与资源调度各参数的介绍如下。集群环境模型记为<math>G=\{A,R,J,C\}</math>,其中,<math>A</math>为蚂蚁的集合;<math>R</math>为节点资源集合;<math>J</math>为作业集,<math>C</math>为容器集。资源集合<math>R=\{R_1,R_2,\dots,R_n\}</math>由<math>n</math>个节点及其资源构成,其中,第<math>i</math>个节点上的资源<math>R_i=\{cpuSpeed_i,M_i,resMi,load_i\}</math>。<math>cpuSpeed_i</math>为节点<math>i</math>上的CPU运行速率,节点<math>i</math>的总资源量用<math>M_i</math>表示,<math>resMi</math>为节点<math>i</math>的空闲资源量,<math>resMi \leq M_i</math>,<math>0 \leq i &lt; n</math>,<math>load_i</math>表示节点<math>i</math>的负载。作业集<math>J=\{J_1,J_2,\dots,J_s\}</math>表示当前集群上运行的作业量为<math>s</math>,第<math>j</math>个作业<math>J_j=\{resTotalJ_j,resMJ_j,prog_j\}</math>,<math>0 \leq j &lt; s</math>,其中,<math>resTotalJ_j</math>为<math>J_j</math>所需资源的总量;<math>resMJ_j</math>为分配给<math>J_j</math>的实际资源量;<math>prog_j</math>表示<math>J_j</math>的运行进度。容器集<math>C=\{C_1,C_2,\dots,C_t\}</math>表示集群中所有申请了资源的容器集合,第<math>m</math>个容器<math>C_m=\{Jm_j,resMCm\}</math>,<math>0 \leq j &lt; s</math>,<math>0 \leq m &lt; t</math>,<math>Jm_j</math>为<math>C_m</math>所属的作业,<math>resMCm</math>为容器<math>C_m</math>所申请的资源量。定义3.1(资源分配)?<math>R_i?R,C_m?C</math>,如果资源<math>R_i</math>可以满足容器<math>C_m</math>的调度要求,则记为?<math>(R_i,C_m)&gt;0</math>,表示允许<math>R_i</math>为容器<math>C_m</math>分配资源以执行任务。满足容器<math>C_m</math>的资源<math>R_i</math>可能有多,为实现</p> <p>一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; -《电子学报》- 2016-05-15 (是否引证:否)</p> <p>1.的自适应Ha-doop资源调度算法3.1集群环境描述集群环境模型表示为<math>G=\{A,R,J,C\}</math>,其中,<math>A</math>为蚂蚁的集合;<math>R</math>为节点资源集合;<math>J</math>为作业集,<math>C</math>为容器集.具体特征描述如下:资源集合<math>R=\{R_1,R_2,\dots,R_n\}</math>由<math>n</math>个节点及其资源构成,其中,第<math>i</math>个节点上的资源<math>R_i=\{cpu Speed_i,M_i,res Mi,load_i\}</math>。<math>cpu Speed_i</math>为节点<math>i</math>上的CPU运行速率,节点<math>i</math>的总资源量用<math>M_i</math>表示,<math>res Mi</math>为节点<math>i</math>的空闲资源量,<math>res Mi \leq M_i</math>,<math>0 \leq i &lt; n</math>,<math>load_i</math>表示节点<math>i</math>的负载.作业集<math>J=\{J_1,J_2,\dots,J_s\}</math>表示当前集群上运行的作业量为<math>s</math>,第<math>j</math>个作业<math>J_j=\{res Total J_j,res Total MJ_j,prog_j\}</math>,<math>0 \leq j &lt; s</math>,其中,<math>res Total J_j</math>为<math>J_j</math>所需资源的总量;<math>res MJ_j</math>为分配给<math>J_j</math>的实际资源量;<math>prog_j</math>表示<math>J_j</math>的运行进度.容器集<math>C=\{C_1,C_2,\dots,C_t\}</math>由<math>t</math>个容器构成,表示集群中所有申请了资源的容器集合,第<math>m</math>个容器<math>C_m=\{Jm,j,res MCm\}</math>,<math>0 \leq j &lt; s</math>,<math>0 \leq m &lt; t</math>,<math>Jm,j</math>为<math>C_m</math>所属的作业,<math>res MCm</math>为容器<math>C_m</math>所申请的资源量.3.2 SRSAPH算法模型 本文将蚁群算法应用于资源调度问题,算法各参数与资源调度各参数的对应关系如下.定义1(资源分配</p>
5	<p>此处有 35 字相似</p> <p>器<math>C_m</math>所申请的资源量。</p> <p>对容器进行资源分配时,对任意<math>R_i \in R</math>,<math>C_m \in C</math>,如果资</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 -《南京航空航天大学硕士论文》- 2015-01-01 (是否引证:否)</p> <p>1.量。定义3.1(资源分配)?<math>R_i?R,C_m?C</math>,如果资源<math>R_i</math>可以</p>



	<p>源<math>R_i</math>可以满足容器<math>C_m</math>所申请的资源量，则允许<math>R_i</math>为容器<math>C_m</math>分配执行任务所需要的资源。满足容器<math>C_m</math>的资源<math>R_i</math>可能不止一个。</p> <p>容器选择资源节点可以表示成为一个<math>t</math>维度的向量 (<math>X_d = X_1X_2X_3...X_t</math>) 其中 <math>0 \leq X_i \leq m</math> (<math>d = 1, 2, 3, \dots, t</math>)</p>	<p>满足容器<math>C_m</math>的调度要求,则记为<math>\Theta(R_i, C_m) &gt; 0</math>,表示允许<math>R_i</math>为容器<math>C_m</math>分配资源以执行任务。满足容器<math>C_m</math>的资源<math>R_i</math>可能有一个或多个,为实现集群最大程度的性能优化,<math>C_m</math>以概率<math>p_{im}</math>被调度到资源<math>R_i</math>上。定义3.2(信息素矩阵)信息素矩阵由一个<math>n \times t</math>维矩阵构成</p> <p>一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15 (是否引证:否)</p> <p>1.系如下.定义1(资源分配)<math>R_i \in R, C_m \in C</math>,如果资源<math>R_i</math>可以满足容器<math>C_m</math>的调度要求,则记为<math>\Theta(R_i, C_m) &gt; 0</math>,表示允许<math>R_i</math>为容器<math>C_m</math>分配资源以执行任务.满足容器<math>C_m</math>的资源<math>R_i</math>可能有一个或多个,为实现集群最大程度的性能优化,<math>C_m</math>以概率<math>p_{i,m}</math>被调度到资源<math>R_i</math>上.定义2(信息素矩阵)信息素矩阵由一个<math>n \times t</math>维矩阵构成</p>
6	<p>此处有 49 字相似</p> <p>个容器没有选择节点，第三个容器选择第二个节点，第四个容器选择了第四个节点。</p> <p>为了保证节点资源得到充分、合理的分配，使得集群可以正常运行。本论文规定：当算法满足以下两个约束条件时，单个布谷鸟的一次迭代过程视为结束。</p> <p>条件一：对所有的资源节点，不能分配任何额外容器。即当前所有资源节点已经饱和分配。条件二：所有的容器都已经分配给指定的资源</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证:否)</p> <p>1.<math>m \leq 1(3.9)0, t_i \in m \text{ im } R \times resMC</math>  <math>resM \text{ ??????}(3.10)</math>定义6(终止条件)为保证集群正常运行,文中规定:当算法满足约束条件(3.11)或(3.12)时,单个蚂蚁的一次迭代过程结束。当为<math>R_i</math>选择容器分配资源时,为避免<math>R_i</math>因无法选择到满足调度条件的容器而陷入死循环,算法增加了控制参数<math>attemptCon</math></p> <p>一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15 (是否引证:否)</p> <p>1.<math>m \in AT, x_i, m + y_i, m \leq 1(2)R_i \in R, \sum t_m = 0x_i, m.resMC_m \leq Mi(3)</math>定义6(终止条件)为保证集群正常运行,文中规定:当算法满足约束条件(4)或(5)时,单个蚂蚁的一次迭代过程结束.当为<math>R_i</math>选择容器分配资源时,为避免<math>R_i</math>因无法选择到满足调度条件的容器而陷入死循环,算法增加了控制参数<math>attemptCo</math></p>
7	<p>此处有 332 字相似</p> <p>Hadoop资源调度，以及针对具体的细节进行算法的调整。在GCS算法的应用中，一次布谷鸟搜索和一次遗传算法的求解表示一次资源分配方案的生成。下面详细介绍本章算法的具体流程：</p> <p>(1) 初始化种群：集群Hadoop环境下，无论是Map或Reduce，节点的CPU速率、内存量与负载等因素对任务的执行效果具有至关重要的影响。一般而言，任务不分配到CPU执行速度低、内存量小、负载较重的节点上执行，而应该在高性能节点（如CPU速率高、负载小、对应作业失败率小的节点）上启动，以此减少任务执行时间。</p> <p>Hadoop YARN资源调度框架中，AM负责作业启动与监控，RM负责集群中所有资源的管理和分配，并监控、管理各节点上AM的状态。AM与RM之间的通信通过心跳传输机制实现。本章设计并实现了一个资源调度器，通过该资源调度器从NM获取节点CPU速率、作业失败记录、内存容量及负载情况等信息。</p> <p>在算法初始化时，随机生成<math>k</math>组解，每组解的形式如4.2.1小节中描述的<math>t</math>维向量，表示各个容器选择资源节</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证:否)</p> <p>1.精确度。本节提出的一种基于蚁群算法和粒子群算法的自适应资源调度算法中,蚂蚁表示调度者,负责分配资源给申请资源的容器以获取资源分配方案。下面详细介绍本章算法的具体流程。(1)初始化信息素集群  Hadoop环境下,无论是Map或Reduce,节点的CPU速率、内存量与负载等因素对任务的执行效果具有至关重要的影响。一般而言,任务不分配到CPU执行速度低、内存量小、负载较重的节点上执行,而应该在高性能节点(如CPU速率高、负载小,对应作业失败率小的节点)上启动,以此减少任务执行时间。Hadoop YARN资源调度框架中,AM负责作业启动与监控,RM负责集群中所有资源的管理和分配,并监控、管理各节点上AM的状态。AM与RM之间的通信通过心跳传输机制实现。根据上述特点,本章设计并实现了一个资源调度器,通过该资源调度器从NM获取节点CPU速率、作业失败记录、内存容量及负载情况等信息,在获取全部相关信息后计算各容器在节点上的执行能力,并将结果以信息素值的形式存储于信息素矩阵,完成信息素矩阵<math>PH_{n \times t} = (p</math></p> <p>一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15 (是否引证:否)</p>

	<p>点的情况。抛弃不符合终</p>	<p>1.m=0(6)3.3 SRSAPH算法思想本文提出的SRSAPH算法中,蚂蚁表示调度者,负责分配资源给申请资源的容器以获取资源分配方案.下面详细介绍本文算法的具体流程.(1)初始化信息素根据Hadoop资源调度框架,本文设计并实现了一个资源调度器,通过该资源调度器从NM(Node Manager)</p> <p>2.资源的容器以获取资源分配方案.下面详细介绍本文算法的具体流程.(1)初始化信息素根据Hadoop资源调度框架,本文设计并实现了一个资源调度器,通过该资源调度器从NM(Node Manager)获取节点CPU速率、作业失败记录、内存容量及负载情况等信息,在获取全部相关信息后计算各容器在节点上的执行能力,并将结果以信息素值的形式存储于信息素矩阵,完成信息素矩阵<math>PH_{n \times t} = (p</math></p>
8	<p>此处有 1322 字相似</p> <p>4.6 ) 所示。</p> <p>( 4.6 )</p> <p>10.3 基于GCS的资源调度算法实现</p> <p>10.3.1 主要数据结构介绍</p> <p>Hadoop</p> <p>调度器负责资源分配, 本节以Capacity调度器的源代码为例介绍主要的实现类:</p> <p>( 1 ) CapacityScheduler类主要作为Capacity调度器的资源管理。当RM收到来自NM发送的心跳信息后, 将向CapacityScheduler发送一个Scheduler EventType的NODE_UPDATE事件。调度器接收到NODE_UPDATE事件后, 首先释放已成功或失败的Container。如果此时调度器有资源空闲, 调度器就会尝试在这个节点上按照调度分配策略响应应用程序提交的Resource Request。</p> <p>( 2 ) SchedulerNode类用于节点对资源的管理。在Hadoop YARN中, 调度器通过汇集各个应用程序的资源请求进行资源分配。集群初始化时, 程序从状态机rmNode中获得相关信息, 然后不断维护节点上的SchedulerNode以更新集群资源情况, 并确认是否能在此节点上分配资源。当一个Container在节点上启动时, SchedulerNode类会调整与可分配资源相关的内部属性信息值, 并记录这个Container的情况。</p> <p>( 3 ) Scheduler Application类主要用于记录应用程序对集群资源的占用情况和程序所在队列情况。调度器通过维护SchedulerApplication以监控集群资源的分配, 当某个应用程序过度占用资源时, 调度器会通过心跳信息通知应用程序释放Container, 如果应用程序仍不释放资源, 调度器将请求RM杀死此Container。同时, SchedulerApplication中包含了此应用程序提交的ResourceRequest, 在每次allocate接口被调用时, AM发来的程序请求会被更新到SchedulerApplication中, 以供调度算法在调度时进行计算。</p> <p>10.3.2 关键接口的实现</p> <p>资源调度器是RM的一个核心部件, 通过RM的组件和服务与其他组件进行交互。Hadoop保留了一系列的接口</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 ( 是否引证 : 否 )</p> <p>1.adoop资源调度上的有效性,需要修改Hadoop源代码,并选取实验数据进行测试。3.4.1主要数据结构介绍</p> <p>Hadoop调度器负责资源分配,本节以Capacity调度器的源代码为例介绍主要的实现类:(1)CapacityScheduler类主要作为Capacity调度器的资源管理。当RM收到来自NM发送的心跳信息后,将向CapacityScheduler发送一个SchedulerEventType的NODE_UPDATE事件。调度器接收到NODE_UPDATE事件后,首先释放已成功或失败的Container。如果此时调度器有资源空闲,调度器就会尝试在这个节点上按照调度分配策略响应应用程序提交的ResourceRequest。(2)SchedulerNode类用于节点对资源的管理。在Hadoop YARN中,调度器通过汇集各个应用程序的资源请求进行资源分配。集群初始化时,程序从状态机rmNode中获得相关信息,然后不断维护节点上的SchedulerNode以更新集群资源情况,并确认是否能在此节点上分配资源。当一个Container在节点上启动时,SchedulerNode类会调整与可分配资源相关的内部属性信息值,并记录这个Container的情况。</p> <p>(3)SchedulerApplication类主要用于记录应用程序对集群资源的占用情况和程序所在队列情况。调度器通过维护SchedulerApplication以监控集群资源的分配,当某个应用程序过度占用资源时,调度器会通过心跳信息通知应用程序释放Container,如果应用程序仍不释放资源,调度器将请求RM杀死此Container。同时,SchedulerApplication中包含了此应用程序提交的ResourceRequest,在每次allocate接口被调用时,AM发来的程序请求会被更新到SchedulerApplication中,以供调度算法在调度时进行计算。3.4.2关键接口的实现资源调度器是RM的一个核心部件,通过RM的组件和服务与其他组件进行交互。Hadoop保留了一系列的接口供RM其他模块调用,本章资源调度器借鉴Capacity调度器的实现方式,实现了与资源调度相关的接口,具体接口如下:(1)AntSchedulerNode:节点信息接口,继承自SchedulerNode,保留了节点当前硬件状态和资源状态,为信息素矩阵的初始化提供相关的属性信息。在资源调度过程中,AntSchedulerNode会根据分配或释放</p>

	<p>供RM其他模块调用，本章资源调度器借鉴Capacity调度器的实现方式，实现了与资源调度相关的接口，具体接口如下：</p> <p>(1) GcsSchedulerNode：节点信息接口，继承自SchedulerNode，保留了节点当前硬件状态和资源状态，为信息素矩阵的初始化提供相关的属性信息。在资源调度过程中，GcsSchedulerNode会根据分配或释放Container情况动态更新节点资源情况，使调度器能够正确显示集群资源情况。</p> <p>(2) GcsSchedulerApp：应用程序接口；包含了该应用程序提交的资源请求信息和当前已获取的资源信息。在调度器进行资源分配时，获取应用程序中的最新资源请求信息以更新数据结构，为下一步资源分配提高数据参考，从而避免因资源请求超出节点容量而造成的资源分配失败。GcsSchedulerNode与GcsSchedulerApp均以映射表Map的形式存储在调度器中，以便在资源分配时供调度器查阅。</p> <p>(3) GcsScheduler：本章资源调度器的关键结构，类似于CapacityScheduler类。GcsSchedulerHadoop调度器负责资源分配，本节以Capacity调度器的源代码为例介绍主要的实现类：实现了调度器与RM的其他组件交互的接口，主要包</p>	<p>Container情况动态更新节点资源情况,使调度器能够正确显示集群资源情况。(2)AntSchedulerApp:应用程序接口;包含了该应用程序提交的资源请求信息和当前已获取的资源信息。在调度器进行资源分配时,获取应用程序中的最新资源请求信息以更新数据结构,为下一步资源分配提高数据参考,从而避免因资源请求超出节点容量而造成的资源分配失败。AntSchedulerNode与AntSchedulerApp均以映射表Map的形式存储在调度器中,以便在资源分配时供调度器查阅。</p> <p>(3)AntScheduler:本章资源调度器的关键结构,类似于CapacityScheduler类。AntScheduler实现了调度器与RM的其他组件交互的接口,主要包括handle函数和allocate函数。handle函数是调度器与Event di</p> <p>573+刘盼红 - 《学术论文联合比对库》 - 2015-03-24 (是否引证：否)</p> <p>1.程序所属用户提交的应用程序数目未超过管理员设定的上限。当ResourceManager收到来自NodeManager发送的心跳信息后，将向Capacity Scheduler发送一个SchedulerEventType.NODE_UPDATE事件，Capacity Scheduler收到该事件后，会依次进行以下操作。(1)处理心跳信息。NodeManager发送</p>
9	<p>此处有 34 字相似</p> <p>ler：本章资源调度器的关键结构，类似于CapacityScheduler类。GcsSchedulerHadoop调度器负责资源分配，本节以Capacity调度器的源代码为例介绍主要的实现</p> <p>类：实现了调度器与RM的其他组件交互的接口，主要包括handle函数和allocate函数。handle函数是调度器与E</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.op资源调度上的有效性,需要修改Hadoop源代码,并选取实验数据进行测试。3.4.1主要数据结构介绍Hadoop调度器负责资源分配,本节以Capacity调度器的源代码为例介绍主要的实现类:(1)CapacityScheduler类主要作为Capacity调度器的资源管理。当RM收到来自NM发送的心跳信息后,</p>
10	<p>此处有 281 字相似</p> <p>r类。GcsSchedulerHadoop调度器负责资源分配，本节以Capacity调度器的源代码为例介绍主要的实现类：实现了调度器与RM的其他组件交互的接口，主要包括handle函数和allocate函数。handle函数是调度器与Eventdispatcher的接口，通过这个接口，Eventdispatcher会把调度器事件(SchedulerEvent)以参数的形式转发给调度器，调度器会根据事件的不同做出相应的动作。allocate函数主要用于ApplicationMasterService的调用，通过该函数，调度器汇集从AM发来的各种请求，然后根据这些信息进行相应操作，如释放Container，更新ResourceRequest，把节点从黑名单中加入或者移除等。</p> <p>10.4 本章小结</p> <p>本章针对Hadoop集群的资源调度问题，将第三章提出的混合遗传布谷鸟算法应用于Hadoop资源调度</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.AntScheduler:本章资源调度器的关键结构,类似于CapacityScheduler类。AntScheduler实现了调度器与RM的其他组件交互的接口,主要包括handle函数和allocate函数。handle函数是调度器与Event dispatcher的接口,通过这个接口,Event dispatcher会把调度器事件(SchedulerEvent)以参数的形式转发给调度器,调度器会根据事件的不同做出相应的动作。allocate函数主要用于ApplicationMasterService的调用,通过该函数,调度器汇集从AM发来的各种请求,然后根据这些信息进行相应操作,如释放Container,更新ResourceRequest,把节点从黑名单中加入或者移除等。</p> <p>3.4.3 Hadoop编译为重新编译Hadoop YARN代码,需要安装很多不同的第三方库,如maven安装编译过程中</p>



11	<p>此处有 54 字相似</p> <p>资源调度。算法在应用的过程中，充分考虑了节点的各个因素对布谷鸟选择新巢的影响，在修正莱维飞行和遗传算法产生的新解时考虑了</p> <p>节点的CPU频率、内存、负载及容器所属作业在节点上失败率等因素，做到了对集群性能较为全面的评估。以此为前提，</p> <p>资源分配能够更加充分地利用系统资源。在最后简略阐述了算法实现所需要注意的事项和实现的接口。</p>	Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士论文》 - 2015-01-01 (是否引证：否)
		1.粒子群算法的自适应Hadoop资源调度算法。本章算法主要特点表现在:(1)算法较全面的考虑了各个因素对信息素的影响,包括节点内存、负载、CPU速率及容器所属作业在节点上失败率,对整个集群性能做出较为系统的评估,以此指导蚂蚁资源探索过程;(2)引入粒子群算法更新规则中的三要素:粒子惯性、自身经验、社会经验,吸收了其精度高、收敛快的特点
		一种Hadoop Yarn的资源调度方法研究 李媛祯;杨群;赖尚琦;李博涵; - 《电子学报》 - 2016-05-15 (是否引证：否)

指 标	
疑似剽窃观点	
<div>1. 当AMA向RM中的Scheduler组件申请Container时，假设当前轮询的位置的节点Node2且Node2中可分配资源满足AM A所需资源量，Scheduler将Node2中的资源分配给AM A，然而从图4.1可以看出，相对于Node2，Node3的资源负载更小CPU处理速度更快，故将Node3中资源分配给AM A更合理。</div> <div>2. 最终，根据本章算法编写相应的 Hadoop 资源调度器，并进行实验验证，实验表明本章调度器能够有效的提高集群资源分配的合理化，达到缩短作业执行时间的目的。</div>	
疑似剽窃文字表述	
<div>1. 资源调度模型分析</div> <div>Hadoop2.x支持内存和CPU两种不同资源类型的管理和分配，资源调度器采用动态资源分配机制，通过处理由应用程序提交的Request请求来分配这两种资源。</div> <div>当NM通过ResourceTracker协议向RM注册、汇报节点状态时，它会提交本节点中可分配的CPU和内存资源总量。在系统中默认值是每个NM具有8GB内存，8个虚拟CPU核心。</div> <div>2. 当一条队列空转时，其资源会分配到其他队列供其他队列的应用使用，直到此队列再一次有应用提交并申请资源。</div> <div>3. 在队列中，存放了该队列的用户提交的应用，调度器会根据资源情况和应用的请求响应资源请求。新的HadoopYARN实现的是两层调度机制：在第一层中，上述的ResourceRequest会被应用程序提交到RM中，而调度器的唯一工作就是结合由应用提交的ResourceRequest和各节点的资源情况，确定是否对ResourceRequest进行响应，如果响应这些请求则会在节点上启动一个到多个Container；如果没有响应则在下一次节点状态更新的时候再判断是否需要响应这些请求。在第二层中，应用程序的AM从RM中定期取回分配到的Container，接着在应用内部将这些Container分配给自己要完成的任务。在HadoopYARN中，分配资源要经过以下步骤：</div> <div>(1)NM用心跳信息向RM汇报节点情况。</div> <div>(2)RM中的ResourceTracker服务会返回一个应答，包含要启动和释放Container的信息。</div> <div>(3)服务向调度器转发NODE_UPDATE事件。</div> <div>(4)调度器把释放的资源重新标记为可用，再按照一定策略将节点上的资源分配给应用程序。</div> <div>(5)AM向RM发送心跳信息。</div> <div>(6)RM把分配的Container应答给AM。</div> <div>(7)AM把Container分配给自己内部的任务。</div> <div>设计调度器的关键就是完成第四步的策略，选择合适的调度策略进行资源分配，可以使得资源被更为合理的使用，从而提高应用的执行效率。</div> <div>10.1.2 资源调度存在的问题</div> <div>资源调度是Hadoop中一个重要的研究领域。不同于Hadoop0.x中的JobTracker组件，HadoopYARN中的资源调度器是一个“纯调度器”，不再从事任何与具体应用程序相关的工作，如不负责监控或跟踪应用程序的执行状态，不负责重新启</div>	



- 动因应用程序执行失败或硬件故障而产生的失败任务等，而是仅仅负责根据各个应用程序的资源需求对整个集群的资源进行统一的管理和分配。
- 当用户向YARN中提交一个应用程序后，YARN将分两个阶段运行该应用程序，首先是启动AM，其次是由AM创建应用程序，并为它申请资源，监控整个过程直到运行完成。在AM资源申请过程中，调度器采用轮询的方式为其分配资源，而并未考虑到是否有其他更优的资源可供AM使用。图4.1描述了当前Hadoop调度器的资源分配策略。
  - 对于Hadoop集群资源调度问题，可以把Container看作是一个关于资源分配的映射目标，可行解空间就是资源分配策略的集合。在可行解空间中，如何为每一个Container提高最优的资源，这是一个无法确定的问题，同时由于无法确定最优解，使得资源的分配问题是一个NP-Hard问题。
  - 合理调度分配资源的重要性  
当前，Hadoop 技术越来越成熟，资源调度问题也越来越重要。在 Hadoop YARN 中，集群资源以大小不等的 Container 为单位进行分配，随着集群规模越来越大，集群中各个节点不可能是同构的，同一作业在不同计算能力的节点资源上执行时，其执行效率有很大的差异，所以选择一个合适的节点为 Container 分配资源非常重要。Hadoop 集群中，影响作业执行时间的因素有很多，但其中一个重要的因素为运行作业中所属任务的资源的性能。如果一个作业中的任务被分配到一个性能相对较差的节点上，如负载高、CPU 处理速度慢或当前节点上没有运行任务所需的数据副本，此时执行该任务需要花费较长的时间，这在一定程度上大大的延长了作业的执行时间，甚至造成了集群资源的浪费。针对上述问题，本章首先分析 Hadoop 资源调度的请求机制与模型，详细剖析了资源调度中存在的问题，在此基础上，引入蚁群算法和粒子群算法，实现 Hadoop 资源的合理化分配。
  - cpuSpeed<sub>i</sub>为节点i上的CPU运行速率，节点i的总资源量用M<sub>i</sub>表示，resM<sub>i</sub>为节点i的空闲资源量，resM<sub>i</sub>≤M<sub>i</sub>，0≤i<n，load<sub>i</sub>表示节点i的负载。作业集J={J<sub>1</sub>,J<sub>2</sub>,...,J<sub>s</sub>}表示当前集群上运行的作业量为s，第j个作业J<sub>j</sub>={resTotal<sub>j</sub>,resM<sub>j</sub>,prog<sub>j</sub>},0≤j<s，其中，resTotal<sub>j</sub>为J<sub>j</sub>所需资源的总量；resM<sub>j</sub>为分配给J<sub>j</sub>的实际资源量；prog<sub>j</sub>表示J<sub>j</sub>的运行进度。容器集C={C<sub>1</sub>,C<sub>2</sub>,...,C<sub>t</sub>}表示集群中所有申请了资源的容器集合，第m个容器C<sub>m</sub>={J<sub>mj</sub>,resMC<sub>m</sub>},0≤j<s,0≤m<t，J<sub>mj</sub>为C<sub>m</sub>所属的作业，resMC<sub>m</sub>为容器C<sub>m</sub>所申请的资源量。对容器进行资源分配时，对任意R<sub>i</sub>∈R，C<sub>m</sub>∈C，如果资源R<sub>i</sub>可以满足容器C<sub>m</sub>所申请的资源量，
  - 集群可以正常运行。本论文规定：当算法满足以下两个约束条件时，单个布谷鸟的一次迭代过程视为结束。
  - Hadoop YARN资源调度框架中，AM负责作业启动与监控，RM负责集群中所有资源的管理和分配，并监控、管理各节点上AM的状态。AM与RM之间的通信通过心跳传输机制实现。本章设计并实现了一个资源调度器，通过该资源调度器从NM获取节点CPU速率、作业失败记录、内存容量及负载情况等信息。
  - 在
  - 当RM收到来自NM发送的心跳信息后，将向CapacityScheduler发送一个Scheduler EventType的NODE\_UPDATE事件。调度器接收到NODE\_UPDATE事件后，首先释放已成功或失败的Container。如果此时调度器有资源空闲，调度器就会尝试在这个节点上按照调度分配策略响应应用程序提交的Resource Request。
  - 在Hadoop YARN中，调度器通过汇集各个应用程序的资源请求进行资源分配。集群初始化时，程序从状态机rmNode中获得相关信息，然后不断维护节点上的SchedulerNode以更新集群资源情况，并确认是否能在此节点上分配资源。当一个Container在节点上启动时，SchedulerNode类会调整与可分配资源相关的内部属性信息值，并记录这个Container的情况。
  - 调度器通过维护SchedulerApplication以监控集群资源的分配，当某个应用程序过度占用资源时，调度器会通过心跳信息通知应用程序释放Container，如果应用程序仍不释放资源，调度器将请求RM杀死此Container。同时，SchedulerApplication中包含了此应用程序提交的ResoureRequest，在每次allocate接口被调用时，AM发来的程序请求会被更新到SchedulerApplication中，以供调度算法在调度时进行计算。
- ### 10.3.2 关键接口的实现
- 资源调度器是RM的一个核心部件，通过RM的组件和服务与其他组件进行交互。
- 在资源调度过程中，GcsSchedulerNode会根据分配或释放Container情况动态更新节点资源情况，使调度器能够正确显示集群资源情况。
  - 在调度器进行资源分配时，获取应用程序中的最新资源请求信息以更新数据结构，为下一步资源分配提高数据参考，从而避免因资源请求超出节点容量而造成的资源分配失败。GcsSchedulerNode与GcsSchedulerApp均以映射表Map的形式存储在调度器中，以便在资源分配时供调度器查阅。
  - 实现了调度器与RM的其他组件交互的接口，主要包括handle函数和allocate函数。
  - allocate函数主要用于ApplicationMasterService的调用，通过该函数，调度器汇集从AM发来的各种请求，然后根据这些信息进行相应操作，如释放Container，更新ResourceRequest，把节点从黑名单中加入或者移除等。
  - 节点的CPU频率、内存、负载及容器所属作业在节点上失败率等因素，做到了对集群性能较为全面的评估。以此为前提

原文内容		相似内容来源
1	<p>此处有 145 字相似</p> <p>unt。WordCount是字数统计测试用例，是一个典型的数据密集型作业集。实验的测试数据采用了一组2015年7月的网络</p> <p>新闻汇总，数据来源为数据堂网站公开分享。本实验将数据进行分词处理后上传到Hadoop文件系统HDFS。在试验测试中分别同时提交8个和16个WordCount作业，每个作业设置为处理一天的数据信息。每一天的信息数据由至少300个txt文档组成，每个文档对应一个Map任务进行处理。因此8个作业构成的作业集中的Map数量要超过2400，此时集群已不可能同时全部进行调度执行，集群负载处于超额状态。分别将作业集提交运</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1..36 969 1108Capacity 1121.05 1009 1443数据密集型实验测试数据为2013年11月中国新闻汇总,下载自数据堂网站。实验将数据用分词器分词并处理后上传到Hadoop文件系统HDFS,分别提交4个与8个WordCount作业。每个作业处理一天的数据信息,每一天的信息数据由至少500个txt文档组成,每个文档对应一个Map任务,故每个作业在执行时需创建并处理至少500个Map。4个WordCount组成的作业集中Map数量大致为2000,此时集群的负载处于</p>
2	<p>此处有 64 字相似</p> <p>可能同时全部进行调度执行，集群负载处于超额状态。分别将作业集提交运行20次的最终测试结果如下表5-2所示：</p> <p>表5-2</p> <p>不同规模WordCount作业集在集群上的运行时间</p> <p>作业集(WordCount) 算法平均时间(s) 最优时间(s) 最差时间(s)</p> <p>8个 GCS 624 553 689</p> <p>Capacity 679 603 768</p> <p>16个 GCS 1219</p>	<p>Hadoop YARN资源分配与调度的研究 李媛祯 - 《南京航空航天大学硕士学位论文》 - 2015-01-01 (是否引证：否)</p> <p>1.法对不同类型问题的解决效果,除计算密集型作业外,实验还对数据密集型作业集(文本统计WordCount)进行测试。表3.2不同规模文本统计WordCount作业的各算法求解结果作业集(WordCount)算法平均时间(s)最优时间(s)最差时间(s)4个SRSAPH 584 542 625Capacity 647 601 6998个SRSAPH 1042.36</p>

## 指 标

## 疑似剽窃文字表述

1. 新闻汇总，数据来源为数据堂网站公开分享。本实验将数据进行分词处理后上传到Hadoop文件系统HDFS。在试验测试中分别同时提交8个和16个WordCount作业，每个作业设置为处理一天的数据信息。每一天的信息数据由至少300个txt文档组成，每个文档对应一个Map任务进行处理。因此8个作业

## 13. 80022497643220302\_基于动态手势的智能终端身份识别技术研究\_第13部分 总字数：1229

相似文献列表 文字复制比：2.5%(31) 疑似剽窃观点：(0)

1	G11113007-王学锋-基于布谷鸟搜索算法的智能组卷系统-软件工程-崔更申 王学锋 - 《学术论文联合比对库》 - 2015-04-13	2.5% ( 31 ) 是否引证：否
---	---	-----------------------

原文内容		相似内容来源
1	<p>此处有 31 字相似</p> <p>能力。然而布谷鸟算法仍然存在些许不足。本论文详细介绍了布谷鸟算法和遗传算法的实现原理和特点，分析了两种群智能算法的优势与</p> <p>不足，提出了一种基于布谷鸟算法的改进算法，即混合遗传布谷鸟算法</p> <p>。针对布谷鸟算法在迭代后期收敛较慢、计算精度不足的缺点，在算法中混合遗传算法，对布谷鸟算法的局部搜索性进行改进。通过标准</p>	<p>G11113007-王学锋-基于布谷鸟搜索算法的智能组卷系统-软件工程-崔更申 王学锋 - 《学术论文联合比对库》 - 2015-04-13 (是否引证：否)</p> <p>1.行为，会对组卷效率产生严重的影响，因此也会导致我们组成的试卷成功率很低，不能很好地控制我们的试卷。鉴于现有算法的很多不足之处，我们提出一种新的智能组卷算法，这种算法的思想使得我们组卷更加顺利，成功率也相对较高。§3.2组卷模型组卷问题的实质是从整个试题库中选择一组试题满足我们</p>

--	--	--

14. 80022497643220302\_基于动态手势的智能终端身份识别技术研究\_第14部分 总字数：8600

相似文献列表 文字复制比：43.6%(3749) 疑似剽窃观点：(0)

1	基于操作码序列的静态恶意代码检测方法的研究 卢占军(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2012-12-01	35.6% ( 3063 ) 是否引证：否
2	10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12	34.9% ( 2998 ) 是否引证：否
3	BH2009925453 - 《学术论文联合比对库》 - 2012-12-11	34.9% ( 2998 ) 是否引证：否
4	BH2009921397 - 《学术论文联合比对库》 - 2012-12-04	34.7% ( 2983 ) 是否引证：否
5	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27	26.7% ( 2295 ) 是否引证：否
6	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14	26.7% ( 2295 ) 是否引证：否
7	基于操作码行为深度学习的恶意代码检测方法 陈晨(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2013-12-01	26.5% ( 2276 ) 是否引证：否
8	基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛(导师：王勇军) - 《国防科学技术大学硕士论文》 - 2012-09-01	2.7% ( 230 ) 是否引证：否
9	姓 名 莫冬春 学 号 201317170102 班 级 计网1301 专 业 计算机网络技术... - 《互联网文档资源 ( <a href="http://www.worlduc.c">http://www.worlduc.c</a> ) 》 - 2016	1.5% ( 126 ) 是否引证：否
10	恶意代码行为挖掘关键技术研究 解培岱(导师：卢锡城) - 《国防科学技术大学博士论文》 - 2013-10-01	1.1% ( 96 ) 是否引证：否
11	基于滑动窗口的数据流预测聚集查询处理的研究 肖裕权(导师：周肆清) - 《中南大学硕士论文》 - 2011-06-30	0.9% ( 80 ) 是否引证：否
12	基于语义的Deep Web数据源分类研究 刘伟平(导师：王宇平) - 《西安电子科技大学硕士论文》 - 2012-01-01	0.6% ( 53 ) 是否引证：否
13	TEA+MEA混合胺液脱除天然气中CO <sub>2</sub> 吸收性能 唐建峰;青霞;张新军;徐明海;黄彬;李晶;史泽林;杨帆; - 《天然气化工(C1化学与化工)》 - 2015-10-14 1	0.4% ( 35 ) 是否引证：否
14	太阳能—地源热泵系统运行特性实验研究 高原(导师：李素芬) - 《大连理工大学硕士论文》 - 2014-05-01	0.4% ( 34 ) 是否引证：否
15	足部三维重构的关键技术研究 田苗苗(导师：杨秋翔;刘磊) - 《中北大学硕士论文》 - 2013-05-25	0.4% ( 31 ) 是否引证：否
16	基于程序语义的静态恶意代码检测系统的研究与实现 袁雪冰(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2009-12-01	0.4% ( 31 ) 是否引证：否
17	可控僵尸网络模拟平台的研究与实现 丁澄天(导师：刘贵松) - 《电子科技大学硕士论文》 - 2012-03-01	0.3% ( 29 ) 是否引证：否

	原文内容	相似内容来源
1	<p>此处有 36 字相似</p> <p>种网络安全事件频发，给个人、企业甚至政府机关带来了巨大经济损失。在众多的攻击事件中，恶意代码是其中的主要的攻击手段。</p> <p>随着信息共享技术的发展及使用，恶意代码的发展速度和更新频率变得越来越快，</p> <p>各种未知恶意代码层出不穷，给分析人员带来巨大的挑战。一般来说，恶意代码分析技术分为静态分析和动态分析，静态分析技术从语法</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》 - 2012-09-01 ( 是否引证：否 )</p> <p>1.序列行为依赖图。该方法大大提高了恶意代码分析的效率，并为恶意代码的检测提供了有力支持。1.3 研究现状随着软件技术的发展，恶意代码也变得越来越复杂，对恶意代码的检测也变得更加困难。恶意代码特征码能唯一标识某一恶意代码，被广泛应用于基于特征码扫描的反病毒产品</p>
2	<p>此处有 47 字相似</p> <p>中，恶意代码是其中的主要的攻击手段。</p> <p>随着信息共享技术的发展及使用，恶意代码的发展速度</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》 - 2012-09-01 ( 是否引证：否 )</p> <p>1.恶意代码行为越来越复杂，这增加了恶意代码行为分</p>



	<p>和更新频率变得越来越快，各种未知</p> <p>恶意代码层出不穷，给分析人员带来巨大的挑战。一般来说，恶意代码分析技术分为静态分析和动态分析，静态分析技术从语法、语义的层面去理解程序的行为，以期获取程序在运行过程中的信息，而不需要运行程序。动态分析技术指在可控</p>	<p>析的难度。本章首先给出了恶意代码的定义和分类，在此基础上介绍了主要的恶意代码分析方法。一般来说，恶意代码分析包括静态分析和动态分析。这两种分析方法各有优缺点，在实际分析过程中可以将两者结合起来。最后，介绍了恶意代码采用的反分析技术。2.1</p>
3	<p>此处有 40 字相似</p> <p>的快速发展，已经有许多研究将这些技术用于恶意代码的检测，并且取得了不错的效果。如何应对恶意代码的爆发式增长，尤其是对未知</p> <p>恶意代码及其变种得到较好的检测效果成为恶意代码检测技术的研究重点。</p> <p>本文对基于</p> <p>机器学习算法的恶意代码检测方法做出了改进，并根据该方法设计并实现了一个恶意代码检测系统。与 традиция方法不同的是，本文提供多于</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.在虚拟环境下的检测也使得动态检测技术无能为力。因此，如何应对恶意代码爆炸式增长，尤其是应对利用混淆技术产生的恶意代码的变种问题成为恶意代码检测技术研究的重点。★本文研究了基于操作码序列的静态恶意代码检测技术，与以往的静态检测技术不同的是本文提取了基于程序控制流程图的操作码序列作</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.躲避在虚拟环境下的检测也使得一些动态检测技术无能为力。因此，如何应对恶意代码爆炸式增长，尤其是应对一些利用混淆技术产生的恶意代码的变种成为恶意代码检测技术研究的重点。本文研究了基于操作码序列的静态恶意代码检测技术，与以往的静态检测技术不同的是本文提取了基于程序的控制流程图的操作码序列作为恶意代码的特</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.意行为来躲避在虚拟环境下的检测也使得动态检测技术无能为力。因此，如何应对恶意代码爆炸式增长，尤其是应对利用混淆技术产生的恶意代码的变种成为恶意代码检测技术研究的重点。本文研究了基于操作码序列的静态恶意代码检测技术，与以往的静态检测技术不同的是本文提取了基于程序控制流程图的操作码序列作为恶意代码的特征</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.意行为来躲避在虚拟环境下的检测也使得动态检测技术无能为力。因此，如何应对恶意代码爆炸式增长，尤其是应对利用混淆技术产生的恶意代码的变种成为恶意代码检测技术研究的重点。本文研究了基于操作码序列的静态恶意代码检测技术，与以往的静态检测技术不同的是本文提取了基于程序控制流程图的操作码序列作为恶意代码的特征</p>
4	<p>此处有 50 字相似</p> <p>果最为明显的中间码序列，并根据此中间码序列生成概率矩阵，然后以此概率矩阵作为代表恶意代码的特征。首先，对恶意代码样本进行</p> <p>查壳和脱壳处理。其次，对恶意代码样本进行反汇编处理，得到样本的汇编文本，并从中提取出汇编操作码序列；</p> <p>然后根据定义的多种抽象化方式分别对汇编操作码序列</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。(2) 本文完成了恶意代码的查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在 vs2010 平台下结合 IDA SDK 开发了基于控制流程图和基于文本的</p>



<p>进行抽象处理，分别得到各抽象方式对应中间码序列；接着，使用n-gram算</p>	<p>操作码序列提取的两个插件。（3）本</p>
	<p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。</p> <p>(2) 本文完成了恶意程序的<b>查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的操作码序列提取的两个插件</b>。</p>
	<p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。</p> <p>(2) 本文完成了恶意程序的<b>查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的操作码序列提取的两个插件</b>。</p>
	<p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。</p> <p>(2) 本文完成了恶意程序的<b>查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的操作码序列提取的两个插件</b>。</p>
	<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1.论基础和系统实现，并将深度学习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的<b>查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本</b>，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性</p>
	<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.论基础和系统实现，并将深度学习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的<b>查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本</b>，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性</p>
	<p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证：否)</p> <p>1.础和系统实现，并将深度学习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的<b>查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本</b>，提取操作码数据</p>

		集 4000。(2) 将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性
5	<p>此处有 34 字相似</p> <p>，并使用Eclat算法对中间码特征序列的频繁项集进行分析，分析出对分类效果最明显的中间码特征序列，并依据该中间码特征序列</p> <p>生成概率矩阵作为样本特征；最后，使用机器学习算法实现恶意代码的检测。</p> <p>在实验中，通过准确率、精确率和召回率三个指标对实验结果进行评估。通过实验结果的对比分析，本文使用的方法相比传统方法有一定</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》- 2012-09-01 (是否引证：否)</p> <p>1.关键 API 函数的参数。得到参数以后，对其进行污点判断，将污染情况输出到文件中。通过分析这些污点信息，可以生成恶意代码行为依赖图。最后，设计并实现了恶意代码行为依赖图挖掘系统，该系统能够有效的挖掘出恶意代码 API 序列的依赖关系，并形成恶意代码行为依赖图。和手工方法相</p>
6	<p>此处有 34 字相似</p> <p>据该中间码特征序列生成概率矩阵作为样本特征；最后，使用机器学习算法实现恶意代码的检测。在实验中，通过准确率、精确率和召回</p> <p>率三个指标对实验结果进行评估。通过实验结果的对比分析，本文使用的方法</p> <p>相比传统方法有一定的优越性。</p> <p>飞速发展的互联网技术推动人类社会的不断进步，但凡事皆有利弊，互联网技术的不断革新也促进了</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.分类方法实现恶意代码的检测。在实验中，通过选择特征的数量及其他方法来对比分析实验，实验结果通过正确率、误报率、漏报率三个指标进行评价。通过实验结果的对比和分析，本文提出的方法收到了很好的实验效果。操作码;;恶意代码检测;;控制流程图★With the wide spread of the</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.持向量机等方法实现恶意代码的分类。在实验中，通过选择特征的数量及其他方法来对比分析实验，实验结果通过正确率、误报率、漏报率三个指标进行评价。通过实验结果的对比和分析，本文提出的方法收到了很好的实验效果。关键词：操作码；恶意代码检测；控制流程图Abstract With the</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.持向量机等方法实现恶意代码的分类。在实验中，通过选择特征的数量及其他方法来对比分析实验，实验结果通过正确率、误报率、漏报率三个指标进行评价。通过实验结果的对比和分析，本文提出的方法收到了很好的实验效果。关键词：操作码；恶意代码检测；控制流程图Abstract With the</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1.最后使用深度置信网络进行分类检测。在实验中，通过选择特征的数量及其他方法来对比分析实验，实验结果通过正确率、误报率、漏报率三个指标进行评价。通过实验结果的对比和分析，本文提出的方法收到了很好的实验效果。基于深度置信网络的恶意代码检测模型包含三个模块：数据预处理、opcode特征提取、深度置信网</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.最后使用深度置信网络进行分类检测。在实验中，通过选择特征的数量及其他方法来对比分析实验，实验结</p>

		果通过正确率、误判率、漏报率三个指标进行评价。通过实验结果的对比和分析，本文提出的方法收到了很好的实验效果。基于深度置信网络的恶意代码检测模型包含三个模块：数据预处理、opcode特征提取、深度置信网
7	<p>此处有 33 字相似</p> <p>获得，开发变得越来越方便，而传播方式也由原来的被动传播进化为主动传播，使得用户极易被感染，据统计 80% 以上的用户曾遭受过</p> <p>恶意代码的侵袭。第三，获利方便，恶意代码出现的初衷是为了技术炫耀，</p> <p>如今在利益的驱动下，更多的恶意代码作者有了经济利益诉求，促进了恶意代码的发展，这也是勒索软件出现的原因。现如今,提高互联</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》- 2012-09-01 (是否引证：否)</p> <p>1. 了理论上的支持。然后，介绍了恶意代码的定义，并对几种常见的恶意代码进行了介绍。通过这些内容，可以大体了解到不同类型恶意代码的不同特点。最后，介绍了恶意代码的分析技术和反分析技术，使我们对恶意代码分析技术的现状以及优缺点有了一定理解。第 13 页国防科学技术大学研究生</p>
8	<p>此处有 31 字相似</p> <p>，危害他人以达到经济或者政治诉求。据统计，中国木马产业链一年的收入已逾上百亿元,黑色产业链正在逐渐成型。研究出一种可以对</p> <p>恶意代码进行有效检测的方法已经迫在眉睫。</p> <p>综上所述,恶意代码</p> <p>的危害无处不在,不仅给个人、企业带来了巨大的损失,甚至可能给国家安全带来不可预期的危害。对于个人,恶意代码的入侵会导致个</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》- 2012-09-01 (是否引证：否)</p> <p>1. 能，掌握恶意代码对受害系统所进行的破坏活动，为受害主机的系统恢复提供重要信息。另一方面，恶意代码分析工作又是恶意代码检测工作的技术基础。现有的恶意代码检测技术主要以检测特征码为主，掌握恶意代码的行为信息，可以为提取更加有效地特征码提供帮助。然而，由于恶意代码</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1. 分的数目，可执行部分的数量以及 PE 头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种新</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1. 和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带了很多挑战。本文提出了一种新的恶意代码检测</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1. 和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带了很多挑战。本文提出了一种新的恶意代码检测</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》-</p>

		<p>2013-11-14 (是否引证:否)</p> <p>1.但无法检测出恶意代码的家族,并且难于实现。</p> <p>2.2.7机器学习和数据挖掘的检测方法机器学习、数据挖掘的方法随着<b>恶意代码检测技术的进步,已经开始在恶意代码检测过程中使用</b>。主要应用分类、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证:否)</p> <p>1.但无法检测出恶意代码的家族,并且难于实现。</p> <p>2.2.7机器学习和数据挖掘的检测方法机器学习、数据挖掘的方法随着<b>恶意代码检测技术的进步,已经开始在恶意代码检测过程中使用</b>。主要应用分类、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证:否)</p> <p>1.测出恶意代码的家族,并且难于实现。2.2.7 机器学习和数据挖掘的检测方法机器学习、数据挖掘的方法随着<b>恶意代码检测技术的进步,已经开始在恶意代码检测过程中使用</b>。主要应用分类、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意</p>
9	<p>此处有 56 字相似</p> <p>此拉开了恶意代码检测与反检测的战争序幕,众多的学者开始投入大量的精力和时间到恶意代码的对抗中。矛与盾从来都是相互促进的,</p> <p><b>恶意代码的发展同样促进了检测技术的进步,目前已经有各种各样的检测技术被应用在了商业环境中。</b></p> <p><b>Sung[3]等人提出</b></p> <p>了一种针对恶意代码变种的静态检测技术,该方法的主要是基于系统调用来检测恶意代码的,首先将恶意代码通过逆向工程技术得到其汇</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证:否)</p> <p>1.981 年第一个病毒 Apple II 出现以来[5],国内外许多计算机安全的学者便投身于与恶意代码的对抗过程中。<b>恶意代码编写技术的发展也推动了检测技术的发展,直到现在已经有很多恶意代码检测技术被广泛应用。</b><b>Sung[6]等人提出了</b>基于系统调用的静态恶意代码检测方法,主要针对恶意代码的变种。该方法是将恶意代码反汇编并根据反汇编后的文本信息提取</p> <p>BH2009921397 -《学术论文联合比对库》- 2012-12-04 (是否引证:否)</p> <p>1.状自1981年第一个病毒Apple II出现以来[5],国内外许多计算机安全的学者便投身于与恶意代码的对抗过程中。<b>恶意代码编写技术的发展也推动了检测技术的发展,直到现在已经有很多恶意代码检测技术被广泛应用。</b><b>Sung[6]等人提出了</b>基于系统调用的静态恶意代码检测方法,主要针对恶意代码的变种。该方法是将恶意代码反汇编并根据反汇编后的文本信息提取系统调</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 (是否引证:否)</p> <p>1.状自1981年第一个病毒Apple II出现以来[5],国内外许多计算机安全的学者便投身于与恶意代码的对抗过程中。<b>恶意代码编写技术的发展也推动了检测技术的发展,直到现在已经有很多恶意代码检测技术被广泛应用。</b><b>Sung[6]等人提出了</b>基于系统调用的静态恶意代码检测方法,主要针对恶意代码的变种。该方法是将恶意代码反汇编并根据反汇编后的文本信息提取系统调</p>



		<p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.状自1981年第一个病毒Apple II出现以来[5]，国内外许多计算机安全的学者便投身于与恶意代码的对抗过程中。<b>恶意代码编写技术的发展也推动了检测技术的发展，直到现在已经有很多恶意代码检测技术被广泛应用。Sung[6]等人提出了基于系统调用的静态恶意代码检测方法，主要针对恶意代码的变种。该方法是将恶意代码反汇编并根据反汇编后的文本信息提取系统调</b></p>
10	<p><b>此处有 31 字相似</b></p> <p>一种针对恶意代码变种的静态检测技术，该方法的主要是基于系统调用来检测恶意代码的，首先将恶意代码通过逆向工程技术得到其汇编<b>文本文件，然后通过分析汇编文本信息，得到系统调用序列，最后根据</b>序列相似度判断该程序是否是恶意的。张波云[4]等人也提出了相似的检测方法，不同的是该方法在提取系统调用信息时程序处于运行</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 -《国防科学技术大学硕士论文》- 2012-09-01 ( 是否引证：否 )</p> <p>1.定义模块中进行相应的处理。6. 处理模块能够提取API 函数的参数，进行污点记录与检查，并将污点信息输出到文件中。<b>7. 通过对污点文件进行分析，可以得到API 序列之间的依赖关系。根据依赖关系手动生成恶意代码行为依赖图。在上述实验条件下，为了验证本方法的有效性，我们对一个哈希值为1</b></p>
11	<p><b>此处有 45 字相似</b></p> <p>序是否是恶意的。张波云[4]等人也提出了相似的检测方法，不同的是该方法在提取系统调用信息时程序处于运行状态，一般是通过在<b>虚拟环境中执行恶意代码程序，提取系统的调用序列，然后使用n-gram算法提取调用序列特征，</b>在检测恶意代码时程序处于运行状态的分析方法被称为动态分析方法。对于恶意代码的分析方法，恶意代码从来都不缺少应对手段，混淆</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )</p> <p>1.文本信息提取系统调用序列，并通过系统调用序列的相似度来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用 n-gram 算法来提取特征，</b>然后进行分类。张波云[7]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用 n-gram 算法提取特征，</p> <p>BH2009921397 -《学术论文联合比对库》- 2012-12-04 ( 是否引证：否 )</p> <p>1.据反汇编后的文本信息提取系统调用序列，并通过系统调用序列的相似度来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用n-gram算法来提取特征，</b>然后进行分类。张波云[7]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用n-gram算法提取特征，使用粗糙集理</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.据反汇编后的文本信息提取系统调用序列，并通过系统调用序列的相似度来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用n-gram算法来提取特征，</b>然后进行分类。张波云[7]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用n-gram算法提取特征，使用粗糙集理</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.据反汇编后的文本信息提取系统调用序列，并通过系</p>

		<p>统调用序列的相似度来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用n-gram算法来提取特征</b>，然后进行分类。张波云[7]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用n-gram算法提取特征，使用粗糙集理</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 ( 是否引证：否 )</p> <p>1. 恶意样本进行反汇编分析，从中提取出API序列，通过分析API序列来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用n-gram算法来提取特征</b>，然后进行分类。张波云[4]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用n-gram算法提取特征，使用粗糙集理</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 ( 是否引证：否 )</p> <p>1. 恶意样本进行反汇编分析，从中提取出API序列，通过分析API序列来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用n-gram算法来提取特征</b>，然后进行分类。张波云[4]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用n-gram算法提取特征，使用粗糙集理</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 ( 是否引证：否 )</p> <p>1. 反汇编分析，从中提取出 API 序列，通过分析API 序列来判断。基于系统调用序列方法也可以用在动态检测过程中，在<b>虚拟环境中执行恶意程序时可以提取执行时的系统调用序列，并使用 n-gram 算法来提取特征</b>，然后进行分类。张波云[4]等人在虚拟环境中动态获得可执行文件的系统调用序列，并使用 n-gram 算法提取特征，使</p>
12	<p>此处有 78 字相似</p> <p>序处于运行状态的分析方法被称为动态分析方法。对于恶意代码的分析方法，恶意代码从来都不缺少应对手段，混淆技术应运而生。同样</p> <p>为了解决混淆技术带来的困惑，有学者提出了基于语义分析的检测方法，该方法试图分析指令运行时的语义，一般的分析方法有符号执行[5]、模型检验[6]、逻辑推理证明</p> <p>等。Cousot.P 和 Cousot.R[7]等人提出了程序分析构造和逼近不动点语义理论，使得程序的语义分析有了理论依</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否 )</p> <p>1. 行文件的系统调用序列，并使用 n-gram 算法提取特征，使用粗糙集理论对特征降维并使用支持向量机实施分类。为了解决混淆技术带来的困惑，一些学者研究基于程序的语义分析方法。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和 Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证：否 )</p> <p>1. 维并使用支持向量机实施分类。为了解决混淆技术带来的困惑，一些学者研究基于程序的语义来分析。语义分析是通过形式化抽象<b>指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明</b>等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构</p>

		造和逼近不动点语义理论，这为程序
		BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)
		1.维并使用支持向量机实施分类。为了解决混淆技术带来的困惑，一些学者研究基于程序的语义来分析。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序
		10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)
		1.维并使用支持向量机实施分类。为了解决混淆技术带来的困惑，一些学者研究基于程序的语义来分析。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)
		1.征降维并使用支持向量机实施分类。针对恶意代码混淆技术，一些学者研究基于程序的语义分析方法。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)
		1.征降维并使用支持向量机实施分类。针对恶意代码混淆技术，一些学者研究基于程序的语义分析方法。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的
		基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)
		1.用支持向量机实施分类。针对恶意代码混淆技术，一些学者研究基于程序的语义分析方法。语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和 Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这
13	<p>此处有 79 字相似</p> <p>基于语义分析的检测方法，该方法试图分析指令运行时的语义，一般的分析方法有符号执行[5]、模型检验[6]、逻辑推理证明等。</p> <p>Cousot.P 和 Cousot.R[7]等人提出了程序分析构造和逼近不动点语义理论，使得程序的语义分析有了理论依据。M.Christodorescu[8]</p> <p>使用行为自动机表述恶意代码，并引入了一种抽象模式库来表示自动机的符号，最终恶意代码将会被表达为库</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和 Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后</p>

<p>中符号表示的自动机，最后</p>	<p>他提</p> <tr> <td data-bbox="829 112 1548 179"> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> </td><td data-bbox="829 179 1548 526"> <p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p> </td></tr> <tr> <td data-bbox="829 526 1548 593"> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> </td><td data-bbox="829 593 1548 940"> <p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p> </td></tr> <tr> <td data-bbox="829 940 1548 1008"> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> </td><td data-bbox="829 1008 1548 1355"> <p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p> </td></tr> <tr> <td data-bbox="829 1355 1548 1422"> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p> </td><td data-bbox="829 1422 1548 1769"> <p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p> </td></tr> <tr> <td data-bbox="829 1769 1548 1836"> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p> </td><td data-bbox="829 1836 1548 2152"> <p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p> </td></tr>	<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>	<p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>	<p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>	<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>	<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>
<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>										
<p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>										
<p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[8]、模型检验[9]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[10]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>										
<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>										
<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p>	<p>1. 语义分析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种</p>										



		<p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)</p> <p>1.析是通过形式化抽象指令运行时的语义，通过符号执行[5]、模型检验[6]、逻辑推理证明等方法来分析程序的语义信息。Cousot.P 和 Cousot.R[7]提出了程序分析构造和逼近不动点语义理论，这为程序的语义分析提供了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提</p>
14	<p>此处有 91 字相似</p> <p>odorescu[8]使用行为自动机表述恶意代码，并引入了一种抽象模式库来表示自动机的符号，最终恶意代码将会被表达为库中</p> <p>符号表示的自动机，最后使用模型检验实现预测。之后他又提出了一种基于语义的恶意代码检测方法，恶意代码的行为通过迹语义来描述，采用抽象解释方法检测恶意代码的行为[9]。D.Preda[10]同样使用了抽象解释的思想对恶意代码进行检测，再度证明了该方法对于混淆技术具有一定的对抗能力。Singh[11]通过分</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用迹语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[14]通过分析反汇</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用迹语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用迹语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1.提供了理论基础。M.Christodorescu[11]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用迹语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文</p>

		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 ( 是否引证：否 )
		1.了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[9]。 D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。 Singh[11]通过分析反汇编文本
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 ( 是否引证：否 )
		1.了理论基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[9]。 D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。 Singh[11]通过分析反汇编文本
		基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 ( 是否引证：否 )
15	此处有 95 字相似	1.基础。M.Christodorescu[8]引入抽象模式库作为恶意行为自动机的符号，将恶意行为表示为带未解释符号的自动机，最后使用模型检验来实现检测。随后他提出一种基于语义的检测方法，用语义来描述恶意代码的行为，采用抽象解释方法检测恶意行为[9]。D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[11]通过分析反汇
	9]。D.Preda[10]同样使用了抽象解释的思想对恶意代码进行检测，再度证明了该方法对于混淆技术具有一定的对抗能力。 Singh[11]通过分析存在于汇编文本中的数据流信息，利用线性时态逻辑语义模型检测程序是否存在恶意行为。Kinder[12]结合程序流程图和函数之间的调用关系，使用计算逻辑树描述恶意代码。 李佳静等人[13]通过分析函数调用及其调用序列之间的依赖关系，使用有穷自动机描述程序的行为信息。王晓洁和王海峰[14]通	基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否 )
		1.为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[14]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测恶意行为。Kinder[15]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[16]等人提出了一种基于语义的行为分析方法，对函数调用及函数
		BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证：否 )
		1.意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测来检测恶意行为。Kinder[15]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[16]等人提出了一种基于语义的行为分析方法，对函数调用及函数调用序列之间
		BH2009925453 - 《学术论文联合比对库》 - 2012-12-

16	11 ( 是否引证 : 否 )	1.意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测来检测恶意行为。Kinder[15]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[16]等人提出了一种基于语义的行为分析方法，对函数调用及函数调用序列之间
	10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 ( 是否引证 : 否 )	1.意行为[12]。D.Preda[13]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码的检测的正确性和完备性。Singh[14]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测来检测恶意行为。Kinder[15]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[16]等人提出了一种基于语义的行为分析方法，对函数调用及函数调用序列之间
	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 ( 是否引证 : 否 )	1.检测恶意行为[9]。D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[11]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测恶意行为。Kinder[12]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[13]等人提出了一种基于语义的行为分析方法，对函数调用及函数调用序列之间
	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 ( 是否引证 : 否 )	1.检测恶意行为[9]。D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[11]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测恶意行为。Kinder[12]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[13]等人提出了一种基于语义的行为分析方法，对函数调用及函数调用序列之间
	基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 ( 是否引证 : 否 )	1.意行为[9]。D.Preda[10]也借鉴了抽象解释的思想，证明了关于混淆技术产生的恶意代码检测的正确性和完备性。Singh[11]通过分析反汇编文本的数据流信息，利用线性时态逻辑语义模型检测恶意行为。Kinder[12]分析了程序的控制流程图和函数之间的调用关系，用计算逻辑树描述恶意行为并公式化，最终使用模型检测方法检测。李佳静[13]等人提出了一种基于语义的行为分析方法，对函数调用及函数
	基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证 : 否	

16	此处有 47 字相似	基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证 : 否
----	------------	---

	<p>。李佳静等人[13]通过分析函数调用及其调用序列之间的依赖关系，使用有穷自动机描述程序的行为信息。王晓洁和王海峰[14]</p> <p>通过语义描述恶意代码的行为，实验证明该方法对使用了混淆技术的恶意代码也具有相当明显的检测效果。</p> <p>孔德光等人[15]提出了一种签名提取算法，对于恶意代码检测的准确率有了较大的提高。G.Tahan等人[16]提出了一种自</p>	<p>)</p> <p>1.态空间以提高分析精度，最终使用模型检测器实现检测。王晓洁和王海峰[17]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[18]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测-3-哈尔滨工业大学工学硕士学位论文</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.局状态空间以提高分析精度，最终使用模型检测器检测是否是恶意的。王晓洁和王海峰[17]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[18]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。G.Tahan[19]等人提出了</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.局状态空间以提高分析精度，最终使用模型检测器检测是否是恶意的。王晓洁和王海峰[17]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[18]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。G.Tahan[19]等人提出了</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1.局状态空间以提高分析精度，最终使用模型检测器检测是否是恶意的。王晓洁和王海峰[17]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[18]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。G.Tahan[19]等人提出了</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p> <p>1.程序的全局状态空间以提高分析精度，最终使用模型检测器实现检测。王晓洁和王海峰[14]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[15]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。G.Tahan[16]等人提出了</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p> <p>1.程序的全局状态空间以提高分析精度，最终使用模型检测器实现检测。王晓洁和王海峰[14]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[15]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。</p>
--	--	--



		<p>G.Tahan[16]等人提出了</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 ( 是否引证：否 )</p> <p>1.全局状态空间以提高分析精度，最终使用模型检测器实现检测。王晓洁和王海峰[14]提出一种基于语义模型匹配的检测算法，通过语义描述恶意代码的行为，这样对经过代码混淆技术处理的恶意代码的检测有很好的效果。孔德光[15]等人提出一种结合语义的多态蠕虫的签名提取算法，提高了检测的鲁棒性和准确性。</p> <p>G.Tahan[16]等人</p>
17	<p>此处有 1788 字相似</p> <p>人[16]提出了一种自动签名提取算法，该算法主要被应用在高速恶意代码的过滤装置中。Y.Tang等人[17]提出了一种基于正则表达式的签名算法，这种方法能否更加准确的产生基于漏洞的签名。Y.Chen[18]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于 n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。</p> <p>Robert[19]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[20]。Schultz[21]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器 Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来 Kolter[22]使用 n-gram 算法提取字节序列作为特征，改进的决策树算法取得了很好的分类效果。在参考文献[23]中，作者提出了使用 n-gram 算法提取特征后使用信息增益的方法来选择一些分类效果好的特征，并使用 K 近邻，基于 TFIDF 的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和 Maloof[22]研究了恶意代码的家族的分类，基于恶意代码的功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[24]中作者提出了一个层次特征选择的方法，即使用 n-gram 算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Categorical Proportional Difference )。CPD 用来度量一个特征的区分能力，最终分类效果相对比较好。Dolev[26]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[27]。将操作码提取</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否 )</p> <p>1.针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[20]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[21]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于 n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[22]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[23]。Schultz[24]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器 Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来 Kolter[25]使用 n-gram 算法提取字节序列作为特征，改进的决策树算法取得了很好的分类效果。在参考文献[26]中，作者提出了使用 n-gram 算法提取特征后使用信息增益的方法来选择一些分类效果好的特征，并使用 K 近邻，基于 TFIDF 的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和 Maloof[25]研究了恶意代码的家族的分类，基于恶意代码的功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[27]中作者提出了一个层次特征选择的方法，即使用 n-gram 算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Categorical Proportion</p> <p>2.反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Categorical ProportionalDifference )。CPD 用来度</p>

<p>出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[19]，实验使用三种分类算法取得了很好的实验效果。在文献[28]中，作者提出了使用变长的指令序列作为特征，并使用 Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[29]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[30]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail B[31]提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[32]利用程序控制流程的拓扑图的同构来实现检测。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[33]提出了一个新的检测未知恶意代码族的方法。该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[34]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及 PE 头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[35]。</p> <p>综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。</p> <p>本文改进了一种基于机器学习算法的恶意代码检测方法，并基于该方法实现了一个恶意代码检测系统。方法主要是使用汇编操作码的抽象化技术将汇编码表示为中间码，并结合Eclat算法对</p>	<p>量一个特征的区分能力，最终分类效果相对比较好。Dolev[29]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[30]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[22]，实验使用三种分类算法取得了很好的实验效果。在文献[31]中，作者提出了使用变长的指令序列作为特征，并使用 Bagging-4-哈尔滨工业大学工学硕士学位论文算法得到了很好的实验效果。也有人使用了十六进制码作为特征[32]。在恶意</p> <p>3.]中，作者提出了使用变长的指令序列作为特征，并使用 Bagging-4-哈尔滨工业大学工学硕士学位论文算法得到了很好的实验效果。也有人使用了十六进制码作为特征[32]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[1]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail[33]提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[34]利用程序控制流程的拓扑图的同构来实现检测。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[35]提出了一个新的检测未知恶意代码族的方法。该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[36]等人提出了从 PE 文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及 PE 头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。</p> <p>1.3 本文的主要内容与结构</p> <p>1.3.1 本文的主要内容</p> <p>本文提出了一种新的静态恶意代码检测方法，主要是结合程序的控制流程</p> <div>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 ( 是否引证：否 )</div> <p>1.法主要针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[20]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[21]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要</p>
--	---

不断的更新特征库。现在大部分研究用基于n-gram序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[22]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[23]。Schultz[24]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作

2.后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作者提出了使用n-gram提取特征后使用信息增益的方法来选择一些分类效果好的特征，并使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和Maloof[25]研究了恶意代码的家族的分类，基于恶意代码分功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[27]中作者提出了一个层次特征选择的方法，即使用n-gram算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意

3.征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Difference )。CPD用来度量一个特征对区分一个类和别的类的的能力，最终分类效果相对比较好。Dolev[29]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[30]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[22]，实验使用三种分类算法取得了很好的准确率。在文献[31]中，作者提出了使用变长的指令序列作为特征，并使用Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[32]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[1]中，作者使用程序的控制流程图并



用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail[33]提出了使用程序控制流程图和函数调用拓扑用于将未知的恶意代码归类。使用函数调用的拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[34]利用程序控制流程的拓扑图的同构来实现检测。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[35]提出了一个新的检测未知恶意代码族的方法。该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[36]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码检测工具[37]。基于恶意代码的PE文件信息和结构特征的检测方法还有很多，其中PE文件的跟多信息都存放在PE头文件。综上所述，现有的恶意代码检测

4.代码的PE文件信息和结构特征的检测方法还有很多，其中PE文件的跟多信息都存放在PE头文件。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带了很多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习和数据挖掘的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。 1.3 本文的主要内容与结构1.3.1 本文的主要内容本文主要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制

BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)

1.法主要针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[20]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[21]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于n-gram序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[22]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[23]。Schultz[24]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作



2.后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作者提出了使用n-gram提取特征后使用信息增益的方法来选择一些分类效果好的特征，并使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和Maloof[25]研究了恶意代码的家族的分类，基于恶意代码分功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[27]中作者提出了一个层次特征选择的方法，即使用n-gram算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意

3.征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Difference )。CPD用来度量一个特征对区分一个类和别的类的的能力，最终分类效果相对比较好。Dolev[29]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[30]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[22]，实验使用三种分类算法取得了很好的准确率。在文献[31]中，作者提出了使用变长的指令序列作为特征，并使用Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[32]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[1]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail[33]提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[34]利用程序控制流程的拓扑图的同构来实现检测。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[35]提出了一个新的检测未知恶意代码族的方法。该方法是基于操作码序列出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[36]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及PE头文

件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带了很多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习和数据挖掘的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。

1.3 本文的主要内容与结构

1.3.1 本文的主要内容本文主要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制

10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 ( 是否引证：否 )

1.法主要针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[20]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[21]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于n-gram序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[22]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[23]。Schultz[24]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作

2.后来Kolter[25]使用n-gram算法代替不覆盖序列提高了实验的准确率，并证明在所有的分类算法中改进的决策树算法的分类效果最好。在参考文献[26]中，作者提出了使用n-gram提取特征后使用信息增益的方法来选择一些分类效果好的特征，并使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和Maloof[25]研究了恶意代码的家族的分类，基于恶意代码分功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[27]中作者提出了一个层次特征选择的方法，即使用n-gram算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意

	<p>3.征后选择那些出现频率高于某个阈值的特征，这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Difference )。CPD用来度量一个特征对区分一个类和别的类的的能力，最终分类效果相对比较好。Dolev[29]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[30]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[22]，实验使用三种分类算法取得了很好的准确率。在文献[31]中，作者提出了使用变长的指令序列作为特征，并使用Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[32]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[1]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail[33]提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[34]利用程序控制流程的拓扑图的同构来实现检测。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[35]提出了一个新的检测未知恶意代码族的方法。该方法是基于操作码序列的出现频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[36]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[37]。综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带了很多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习和数据挖掘的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。</p> <p>1.3 本文的主要内容与结构</p> <p>1.3.1 本文的主要内容本文主要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 ( 是否引证：否 )</p> <p>1.法主要针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[17]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[18]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检</p>
--	--

测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[19]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[20]。Schultz[21]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来Kolter[22]使用n-gram算法提取字节序列作为特征，改进的决策树算法取得了很好的分类效果。在参考文献[23]中，作者提出了采用n-gram算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、

2.在参考文献[23]中，作者提出了采用n-gram算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和Maloof[22]研究了恶意代码的家族的分类，基于恶意代码的功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[24]中作者提出了一个层次特征选择的方法，即使用n-gram算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Difference )。CPD用来度量一个特征的区分能力，最终分类效果相对比较好。Dolev[26]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[27]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[19]，实验使用三种分类算法取得了很好的实验效果。在文献[28]中，作者提出了使用变长的指令序列作为特征，并使用Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[29]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[30]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类[31]。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避



	<p>检测。Halvar[32]利用程序控制流程的拓扑图的同构来实现检测[32]。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。</p> <p>Igor[33]提出了一个新的检测未知恶意代码族的方法[33]。该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[34]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[35]。</p> <p>1.3 本文的主要研究内容由恶意代码检测国内外研究现状可知，现有的恶意代码检测技术有很多，但每一种方法都有自身的优</p> <p>3.们开发了一个快速统计恶意代码的检测工具[35]。1.3 本文的主要研究内容由恶意代码检测国内外研究现状可知，现有的恶意代码检测技术有很多，但每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种基于深度置信网络的恶意代码检测方法，结合了特征码与深度学习的方法，提出了基于深度置信网络的恶意代码检测方法。一方面深度置信网络为半监督学习，能利用大量的无类标的恶意样本集与正常样本，另一方面深度置</p>
	<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p> <p>1.法主要针对恶意的可执行文件，被应用到高速恶意代码过滤装置中。Y.Tang[17]等人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能产生更加准确的基于漏洞的签名。Y.Chen[18]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[19]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[20]。Schultz[21]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器Pipper、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来Kolter[22]使用n-gram算法提取字节序列作为特征，改进的决策树算法取得了很好的分类效果。在参考文献[23]中，作者提出了采用n-gram算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用K近邻，基于TFIDF的分类器、朴素贝叶斯、支持向量机、</p> <p>2.。在参考文献[23]中，作者提出了采用n-gram算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用K近邻，基于TFIDF的分类器、朴素贝</p>

叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和Maloof[22]研究了恶意代码的家族的分类，基于恶意代码的功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[24]中作者提出了一个层次特征选择的方法，即使用n-gram算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Difference )。CPD用来度量一个特征的区分能力，最终分类效果相对比较好。Dolev[26]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[27]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[19]，实验使用三种分类算法取得了很好的实验效果。在文献[28]中，作者提出了使用变长的指令序列作为特征，并使用Bagging算法得到了很好的实验效果。也有人使用了十六进制码作为特征[29]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[30]中，作者使用程序的控制流程图并用三种不同版本的黑客防御工具设计了一个分类算法，并取得了很好的实验效果。Ismail提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类[31]。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[32]利用程序控制流程的拓扑图的同构来实现检测[32]。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[33]提出了一个新的检测未知恶意代码族的方法[33]。该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[34]等人提出了从PE文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及PE头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[35]。1.3 本文的主要研究内容由恶意代码检测国内外研究现状可知，现有的恶意代码检测技术有很多，但每一种方法都有自身的优

3.们开发了一个快速统计恶意代码的检测工具[35]。1.3 本文的主要研究内容由恶意代码检测国内外研究现状可知，现有的恶意代码检测技术有很多，但每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种基于深度置信网络的恶意代码检测方法，结合了特征码与深度学习的方法，提出了基于深度置信网络的恶意代码检测方法。一方

面深度置信网络为半监督学习，能利用大量的无类标的恶意样本集与正常样本，另一方面深度置

基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)

1.人提出了一种利用多序列对比技术的简化的正则表达式签名算法，这种方法能2哈尔滨工业大学硕士学位论文产生更加准确的基于漏洞的签名。Y.Chen[18]等人提出了在网络层没有任何主机分析的蠕虫执行的脆弱性驱动的签名，实验效果非常好。现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于 n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。Robert[19]等人提出了用操作码序列作为特征，然后使用文本分类的方法实现检测，并解决了数据不平衡问题[20]。Schultz[21]等人第一次提出了应用数据挖掘模型来检测恶意代码，他们提取三种特征并使用不同的分类方法：程序的头文件信息，字符串信息，字节的序列，应用基于签名、基于规则的学习器 Pippet、朴素贝叶斯等方法进行分类。研究表明使用机器学习方法能提高准确率。后来 Kolter[22]使用 n-gram 算法提取字节序列作为特征，改进的决策树算法取得了很好的分类效果。在参考文献[23]中，作者提出了采用 n-gram 算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用 K 近邻，基于文档频率的分类器、朴素贝叶斯

2.参考文献[23]中，作者提出了采用 n-gram 算法构造操作码序列，然后使用信息增益来选择分类效果好的特征。并且使用 K 近邻，基于文档频率的分类器、朴素贝叶斯、支持向量机、决策树等分类方法，并取得了很好的实验效果。Kolterh 和 Maloof[22]研究了恶意代码的家族的分类，基于恶意代码的功能行为，使用多分类方法将恶意代码分为蠕虫、木马、后门、病毒等，这更加细化了分类的结果，有助于对每一种恶意代码的研究，发现它们的共性，这也为以后的语义分析等方法奠定了基础。文献[24]中作者提出了一个层次特征选择的方法，即使用 n-gram 算法提取特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Categorical Proportional Difference )。CPD用来度量一个特征的区分能力，最终分类效果相对比较好。Dolev[26]提倡使用操作码来作为恶意代码的中间表示。操作码是机器语言的一个操作的一部分，它包含着指令的行为和程序的控制。近年来，操作码特征已经被用来检测蠕虫的变种和一些间谍软件[27]。将操作码提取出来作为标签，然后产生签名来判别恶意代码的变种。后来有些学者提取操作码并将其转化成操作码序列来检测未知的恶意代码[19]，实验使用三种分类算法取得了很好的实验效果。在文献[28]中，作者提出了使用变长

		<p>的指令序列作为特征，并使用Bagging 算法得到了很好的实验效果。也有人使用了十六进制码作为特征[29]。在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。在文献[30]中，作者使用程序的控制流程图并用三种不同版本的黑客防御3哈尔滨工业大学硕士学位论文工具设计了一个分类算法，并取得了很好的实验效果。Ismail 提出了将程序控</p> <p>3.在文献[30]中，作者使用程序的控制流程图并用三种不同版本的黑客防御3哈尔滨工业大学硕士学位论文工具设计了一个分类算法，并取得了很好的实验效果。Ismail 提出了将程序控制流程图和函数调用拓扑用于将未知的恶意代码归类[31]。使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。Halvar[32]利用程序控制流程的拓扑图的同构来实现检测[32]。因为同一种族的恶意代码的拓扑图基本相似，这种方法也是适合检测恶意代码的变种。Igor[33]提出了一个新的检测未知恶意代码族的方法[33]。该方法是基于操作码序列的出现频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是非常有效的。Perdisci[34]等人提出了从 PE 文件提取一些特征，如标准和非标准部分的数目，可执行部分的数量以及 PE 头文件的熵信息，并使用不同的机器学习模型实现分类。后来他们开发了一个快速统计恶意代码的检测工具[35]。深度学习在计算机视觉和语音识别方面都取得了良好的效果。在计算机视觉方面，Krizhevsky[36]等人提出了采用</p> <p>4.单位，也在进行深度学习在语音识别上的研究。1.3 本文的主要研究内容由恶意代码检测国内外研究现状可知，现有的恶意代码检测技术有很多，但每一种方法都有自身的优缺点。这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种基于深度置信网络的恶意代码检测方法，结合了特征码与深度学习的方法，提出了基于深度置信网络的恶意代码检测方法。一方面深度置信网络为半监督学习，能利用大量的无类标的恶意样本集与正常样本，另</p>
18	<p>此处有 64 字相似</p> <p>实验结果，给出结论。实验主要步骤有：首先，为了逃避病毒检测系统的检测，一般的恶意代码作者都会对恶意程序进行加壳处理，所以</p> <p>本文的第一步就是对恶意代码进行查壳和脱壳处理；其次对恶意代码样本进行反汇编处理，得到样本的汇编文本，并从中提取出汇编操作码序列；</p> <p>然后根据已配置的抽象方式对汇编操作码序列进行抽象化处理，得到各抽象方式对应的中间码序列；接着使用 n-gram 算法获得中间</p>	<p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.较好，而支持向量机的效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。(2) 本文完成了恶意程序的查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的操作码序列提取的两个插件。(</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.而支持向量机的效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和</p>



		漏报率。(2) 本文完成了恶意代码的查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在 vs2010 平台下结合 IDA SDK 开发了基于控制流程图和基于文本的操作码序列提取
		11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 (是否引证：否)
		1.习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性，同时另一
		11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证：否)
		1.习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性，同时另一
		基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证：否)
		1.法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，提取操作码数据集 4000。(2) 将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性，同时另一
		BH2009925453 -《学术论文联合比对库》- 2012-12-11 (是否引证：否)
		1.较好，而支持向量机的效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。(2) 本文完成了恶意程序的查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的
19	此处有 31 字相似 据本文提出的方法设计并实现了一个恶意代码检测系统。本文的工作有以下几点： 第一：收集实验样本，并对样本进行预处理操作，然后提取基于文本的汇编操作码序列。 第二：本文改进了一种基于	10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 (是否引证：否)
		1.较好，而支持向量机的效果相对差一些。从整个实验结果分析，本文提出的方法有较高的准确率，较低的误报率和漏报率。(2) 本文完成了恶意程序的查壳和脱壳工作，对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，并在vs2010平台下结合IDA SDK开发了根据控制流程图提取操作码序列以及基于文本的
		基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)
		1.哈尔滨工业大学工学硕士学位论文第一：收集实验样本并对其做了查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析，然后提取基于文本的操作码序列。第二：本文提出了一种新的基于操作码序列的静态恶意代码

	机器学习算法的恶意代码检测方法。通过汇编操作码抽象化技术、Eclat算法频繁项集分析和n-gram算法提取特征序列获得概	检测方法。结合程序的控制流程图提取操作码序列。这样既考虑到了程序的文本特征，也分析了
		BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)
		1.实施分类。本文的主要工作有以下几点：第一：收集实验样本并对其做了查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析，然后提取基于文本的操作码序列。第二：本文提出了一种新的基于操作码序列的静态恶意代码检测方法。结合程序的控制流程图提取操作码序列。这样既考虑到了程序的文本特征，也分析了程序的行为特
		BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)
20	<p>此处有 54 字相似</p> <p>算法提取特征序列获得概率矩阵，并以此作为代表恶意代码的特征，最后使用机器学习算法构建分类模型。</p> <p>第三：对本文提出的方法进行实验仿真，并对实验结果进行对比分析，得出结论。</p> <p>第四：结合实验仿真的结果分析，针对本文提出的改进方法，采用模块化编程技术实现了一个恶意代码检测系统。</p> <p>本文总共分为四章：</p> <p>第一章为绪论，主要介绍课题背景及意义。并且详细阐</p>	10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)
		1.实施分类。本文的主要工作有以下几点：第一：收集实验样本并对其做了查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析，然后提取基于文本的操作码序列。第二：本文提出了一种新的基于操作码序列的静态恶意代码检测方法。结合程序的控制流程图提取操作码序列。这样既考虑到了程序的文本特征，也分析了程序的行为特
		基于语义的Deep Web数据源分类研究 刘伟平 - 《西安电子科技大学硕士论文》- 2012-01-01 (是否引证：否)
		1.别设计了改进方法。第五章：实验结果与分析。首先介绍了本文的实验数据集、实验评价标准以及实验平台；然后对数据集进行了三组实验，并对实验结果进行对比分析，验证本文所设计方法的可行性和高效性。第六章：总结与展望。本章对本文所作的研究工作进行总结，并指明未来的工作方向。8 基于
		基于滑动窗口的数据流预测聚集查询处理的研究 肖裕权 - 《中南大学硕士论文》- 2011-06-30 (是否引证：否)
		1.据流管理系统BorealiS实现了预测子模块，采集著名流量库中的网络流量数据进行了测试，按照不同参数设计多组对照实验，并对实验结果进行对比分析，可以看出本文提出的数据流预测聚集查询方法在处理速度和预测精度两方面均有较好的表现。同时也发现该模型还有许多可改进的方面。如多数
		TEA+MEA混合胺液脱除天然气中CO <sub>2</sub> 吸收性能 唐建峰;青霞;张新军;徐明海;黄彬;李晶;史泽林;杨帆; - 《天然气化工(C1化学与化工)》- 2015-10-14 1 (是否引证：否)
		1.TEA+1mol/L MEA、3mol/L TEA、2mol/L TEA、1mol/L MEA四种胺液进行相同条件下的吸收实验,并对实验结果进行对比分析。本节吸收实验所选用的实验参

		数依据本课题前期研究基础确定:原料气为纯CO2气体,胺液量为100m L,吸收压力为0.21MPa,操作温度为50
21	<p>此处有 75 字相似</p> <p>结合实验仿真的结果分析,针对本文提出的改进方法,采用模块化编程技术实现了一个恶意代码检测系统。本文总共分为四章:</p> <p>第一章为绪论,主要介绍课题背景及意义。并且详细阐述了国内外对于恶意代码检测技术的研究现状和存在的问题,最后介绍了本文的研究内容和章节安排。</p> <p>第二章是</p> <p>相关理论与关键技术。首先介绍了恶意代码的定义和分类,然后对现有恶意代码分析技术、检测技术和反检测技术做了相关介绍。</p> <p>第</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证:否)</p> <p>1.三种机器学习分类算法实现了恶意代码检测系统。</p> <p>1.3.2 本文的结构本文共分为四章,全文的结构安排如下:第一章是绪论,主要介绍了恶意代码的危害和发展及课题的研究意义,并详细阐述了国内外的研究现状。最后介绍本文研究内容及其结构安排。第二章是恶意代码检测技术的研究,主要介绍恶意代码定义及分类和现有的一些恶意代码检测技术的原理及优缺点。第三章是基于操</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证:否)</p> <p>1.四章,全文的结构安排如下:第一章是绪论,主要介绍了恶意代码的危害和发展,简单介绍了现阶段恶意代码检测的技术及研究的意义并详细阐述了国内外的研究现状。最后介绍本文研究内容及其结构安排。第二章是恶意代码检测技术的研究,主要介绍恶意代码定义及分类和现有的一些恶意代码的检测技术的原理及优缺点。第三章是基于操作码</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证:否)</p> <p>1.四章,全文的结构安排如下:第一章是绪论,主要介绍了恶意代码的危害和发展,简单介绍了现阶段恶意代码检测的技术及研究的意义并详细阐述了国内外的研究现状。最后介绍本文研究内容及其结构安排。第二章是恶意代码检测技术的研究,主要介绍恶意代码定义及分类和现有的一些恶意代码的检测技术的原理及优缺点。第三章是基于操作码</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证:否)</p> <p>1.最优。最后使用深度置信网络模型进行测试集分类,得出实验结果并分析。1.4 本文的组织结构第 1 章 绪论——首先介绍课题的研究背景和意义,然后阐述恶意代码检测国内外的研究现状和综述,最后概述本文的主要工作。第 2 章 恶意代码检测方法研究概述——介绍了恶意代码检测需要对恶意样本进行的预处理,阐述了现有的</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证:否)</p> <p>1.全文的结构安排如下:第一章是绪论,主要介绍了恶意代码的危害和发展,简单介绍了现阶段恶意代码检测的技术及研究的意义。详细阐述了国内外的研究现状。最后介绍本文研究内容及其结构安排。第二章是恶意代码检测技术的研究,主要介绍恶意代码定义及分类和现有的一些恶意代码的检测技术的原理及优缺点。第三章是基于操作码</p> <p>基于滑动窗口的数据流预测聚集查询处理的研究 肖裕权 -</p>

		<p>《中南大学硕士论文》- 2011-06-30 (是否引证: 否)</p> <p>1. 第一章为绪论, 首先介绍了数据流模型, 包括数据流的特点和数据流的典型应用, 接着介绍了课题在国内外的研究现状, 然后给出了本文的研究内容。第二章详细介绍了数据流的相关技术, 如数据流约简技术、数据流预测技术等。第三章设计了基于隐马尔可夫模型</p>
		<p>足部三维重构的关键技术研究 田苗苗 - 《中北大学硕士论文》- 2013-05-25 (是否引证: 否)</p> <p>1. 的足部测量系统, 提出了对足部测量的要求, 并阐述了足部三维重构技术的发展趋势, 其次介绍了本文研究的足部三维重构的关键技术的理论依据和实验设备, 最后介绍了本文的研究内容和章节安排。2. 足部测量系统的硬件设计2.1 引言众所周知, 具有视觉的生物是用一双眼睛观察周围的物体, 从而确定深度或远近</p>
		<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-14 (是否引证: 否)</p> <p>1. 利用信息增益和文档频率两种特征选择方法分别选取样本opcode特征序列, 构建训练集和测试集。(2) 归纳总结了以往恶意代码检测的各种方法和技术, 以及前人的工作和研究的发展, 构建了基于深度置信网络的恶意代码检查模型, 采用训练集中样本操作码特征序列数据进行训练, 并通过深度信念网络的反馈调节使权值</p>
		<p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-27 (是否引证: 否)</p> <p>1. 利用信息增益和文档频率两种特征选择方法分别选取样本opcode特征序列, 构建训练集和测试集。(2) 归纳总结了以往恶意代码检测的各种方法和技术, 以及前人的工作和研究的发展, 构建了基于深度置信网络的恶意代码检查模型, 采用训练集中样本操作码特征序列数据进行训练, 并通过深度信念网络的反馈调节使权值</p>
22	<p>此处有 35 字相似</p> <p>内容和章节安排。</p> <p>第二章是相关理论与关键技术。首先介绍了恶意代码的定义和分类, 然后对现有恶意代码分析技术、检测技术和反</p> <p>检测技术做了相关介绍。</p> <p>第三章是基于机器学习算法的恶意代码检测方法。</p> <p>首先对汇编操作码的特点进行了分析, 并介绍了有关学者对汇编码抽象化的一些研究; 然后概要的介绍了本文提出方法的大体流程, 接着</p>	<p>恶意代码行为挖掘关键技术研究 解培岱 - 《国防科学技术大学博士论文》- 2013-10-01 (是否引证: 否)</p> <p>1. 网络安全具有极其重要的意义, 对未来恶意代码研究和APT检测具有指导意义。论文的主要工作包括以下方面: (1) 对抗型壳代码的检测技术研究提出了基于缺失ELM算法的加壳恶意代码检测方法A-ELM, 用于对抗恶意代码的运行时壳行为。已有的基于机器学习的加壳恶意代码检测方法在目标样本二进制代码结构与属性的特征向量空间上作</p>
23	<p>此处有 34 字相似</p> <p>出方法的大体流程, 接着对方法的各部分进行了详细说明, 包括数据预处理、概率矩阵的生成过程和恶意代码的分类; 最后根据本文方法</p> <p>进行了实验仿真, 并和传统方法的实验结果进行了对比分析, 得出结论。</p> <p>第四章是系统设计与实现。首先介绍整个系统的架构设计, 包括系统的功能分析及组成模块, 并对各模块的功</p>	<p>太阳能—地源热泵系统运行特性实验研究 高源 - 《大连理工大学硕士论文》- 2014-05-01 (是否引证: 否)</p> <p>1. ture recovery rate of 4 depths4. 5本章小结本章对太阳能—地源热泵联合供暖实验系统进行了实验研究, 并对实验结果进行了对比和分析。进行了四组工况的实验研究: 太阳能与地源热泵系统串联运行和三组不同流量比例的并联运行实验。通过对比, 分析了联合</p>



	能进行了简单介绍；接着对	
24	<p>此处有 32 字相似</p> <p>和恶意代码的分类；最后根据本文方法进行了实验仿真，并和传统方法的实验结果进行了对比分析，得出结论。</p> <p>第四章是系统设计与实现。首先介绍整个系统的架构设计，包括系统的功能分析及组成模块，并对各模块的功能进行了简单介绍；接着对各模块的实现方式进行了详细介绍；最后对本章进行总结。</p> <p>第五章为总结与展望。概括总</p>	<p>可控僵尸网络模拟平台的研究与实现 丁澄天 - 《电子科技大学硕士论文》 - 2012-03-01 (是否引证：否)</p> <p>1.信节点的返回机制具有更好的适用价值。4.5 本章小结本章介绍了基于超节点 P2P 可控僵尸网络模拟平台的设计与实现。首先介绍了整个系统的架构设计，该系统由两部分组成：控制端和僵尸网络终端。然后分别介绍了控制端和僵尸网络终端的设计与实现，并重点从功能结构、模块结构、重要数</p>
25	<p>此处有 30 字相似</p> <p>与展望。概括总结了本文的主要研究成果和不足，对未来的可研究方向进行了展望。</p> <p>本章首先介绍了恶意代码相关概念，并对现有的恶意代码检测技术以及反检测技术进行了详细介绍；然后对恶意代码的反检测技术做了相关介绍；最后对恶意代码的分析技术做了总结。</p> <p>恶意代码也成为恶意软件，是对各种敌对和入侵软件的概括性术</p>	<p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》 - 2012-09-01 (是否引证：否)</p> <p>1.定义，并对几种常见的恶意代码进行了介绍。通过这些内容，可以大体了解到不同类型恶意代码的不同特点。最后，介绍了恶意代码的分析技术和反分析技术，使我们对恶意代码分析技术的现状以及优缺点有了一定理解。第 13 页国防科学技术大学研究生院工程硕士学位论文第三章</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Catego</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Dif</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应用数据挖掘方法实现恶意代码的检测，他通过反汇编技术提取了恶意代码的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Dif</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1.这种方法对于检测变种的恶意代码家族很有效，因为一般一个恶意家族的代码都大致相似。Raja[28]等人应</p>

		<p>用数据挖掘方法实现<b>恶意代码的检测</b>，他通过反汇编技术提取了<b>恶意代码</b>的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Dif</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 ( 是否引证：否 )</p> <p>1.特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现<b>恶意代码的检测</b>，他通过反汇编技术提取了<b>恶意代码</b>的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Dif</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 ( 是否引证：否 )</p> <p>1.特征后选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现<b>恶意代码的检测</b>，他通过反汇编技术提取了<b>恶意代码</b>的操作码序列，使用了一种新的在文本分类领域的特征选择方法CPD ( Categorical Proportional Dif</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 ( 是否引证：否 )</p> <p>1.选择那些出现频率高于某个阈值的特征，这种方法对于检测恶意代码的变种很有效。Raja[25]等人应用数据挖掘方法实现<b>恶意代码的检测</b>，他通过反汇编技术提取了<b>恶意代码</b>的操作码序列，使用了一种新的在文本分类领域的特征选择方法 CPD ( Categorical Proportional</p>
26	<p>此处有 32 字相似</p> <p>检测技术做了相关介绍；最后对恶意代码的分析技术做了总结。</p> <p>恶意代码也成为恶意软件，是对各种敌对和入侵软件的概括性术语。</p> <p><b>包括各种形式的计算机病毒、蠕虫、特洛伊木马、勒索软件、间谍软件、</b></p> <p>广告软件以及其他的恶意软件。形式上多种多样，可以是可执行文件、脚本、插件等等。其违背使用者的意愿去执行一些操作，损害用户</p>	<p>恶意代码行为挖掘关键技术研究 解培岱 -《国防科学技术大学博士论文》- 2013-10-01 ( 是否引证：否 )</p> <p>1.代码、执行行为、恶意代码分析、行为分析、恶意行为和行为挖掘的概念,理清了概念的相互关系。第二,介绍了恶意代码的常见类别,<b>包括病毒、蠕虫、木马、僵尸程序、间谍软件</b>和Rootkit等,比较了各类别的典型特征和主要目标。第三,从整体代码结构、模块代码结构、局部代码结构和代码量四个方面总</p>
27	<p>此处有 96 字相似</p> <p>多样，可以是可执行文件、脚本、插件等等。其违背使用者的意愿去执行一些操作，损害用户的利益以达到入侵者不可告人的目的。</p> <p><b>根据不同的依据，恶意代码有很多种不同的分类方法，没有一种标准的分法，但是常见的种类有：计算机病毒、蠕虫、特洛伊木马、间谍软件、勒索软件等等。下面对几种恶意代码做简要介绍：</b></p> <p><b>(1)计算机病毒。</b></p> <p>病毒是早期产生的最主要的恶意代码之一，病毒是能够自我繁殖并寄生在其他程序中的代码，这个被寄存的程序被称为宿主程序，但是病</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )</p> <p>1.统特定功能的代码[40]。恶意代码大部分是以二进制的可执行文件的形式存在。2.1.2 恶意代码的分类恶意代码<b>根据不同的依据有很多种分类方法，没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等</b>，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：( 1 ) 病毒</p> <p>2.一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、</p>

	<p>Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期恶意代码的主要形式就是病毒，病毒是指能够自我复制并将其嵌入被感染宿主程序，宿主文件一旦感染病毒就会感染其他文件</p>
	<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.统特定功能的代码[40]。恶意代码大部分是以二进制的可执行文件的形式存在。2.1.2 恶意代码的分类恶意代码根据不同的依据，有很多种分类方法，也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期</p> <p>2.也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期恶意代码的主要形式就是病毒，病毒是指能够自我复制并将其嵌入被感染宿主程序，宿主文件一旦感染病毒就会感染其他文件。病毒</p>
	<p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.统特定功能的代码[40]。恶意代码大部分是以二进制的可执行文件的形式存在。2.1.2 恶意代码的分类恶意代码根据不同的依据，有很多种分类方法，也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期</p> <p>2.也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期恶意代码的主要形式就是病毒，病毒是指能够自我复制并将其嵌入被感染宿主程序，宿主文件一旦感染病毒就会感染其他文件。病毒</p>
	<p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1.统特定功能的代码[40]。恶意代码大部分是以二进制的可执行文件的形式存在。2.1.2 恶意代码的分类恶意代码根据不同的依据，有很多种分类方法，也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期</p> <p>2.也没有一个统一的说法，但是常见的种类有：病毒、后门、蠕虫、特洛伊木马等，此外还有广告软件、恶意网页脚本、Rootkit、间谍软件等[41]。下面对几种恶意代码做简单介绍：（1）病毒。早期恶意代码的主要形</p>

		<p>式就是病毒，病毒是指能够自我复制并将其嵌入被感染宿主程序，宿主文件一旦感染病毒就会感染其他文件。病毒</p> <p>恶意代码行为挖掘关键技术研究 解培岱 - 《国防科学技术大学博士论文》 - 2013-10-01 ( 是否引证：否 )</p> <p>1.定义了恶意代码、执行行为、恶意代码分析、行为分析、恶意行为和行为挖掘的概念,理清了概念的相互关系。第二,介绍了恶意代码的<b>常见类别,包括病毒、蠕虫、木马、僵尸程序、间谍软件和Rootkit等</b>,比较了各类别的典型特征和主要目标。第三,从整体代码结构、模块代码结构、局部代码结构和代码量四个方面总</p>
28	<p>此处有 58 字相似</p> <p>激活宿主程序并满足一定条件下，病毒就能干扰电脑正常工作，扰乱或破坏已有存储的信息，甚至引起整个系统不能正常工作。一般而言</p> <p><b>计算机病毒通常由三个单元和一个标志构成：引导模块、感染模块、破坏表现模块和感染指标。1、引导模块是指将计算机病毒感染</b></p> <p>的宿主程序设法引导安装到</p> <p>计算机操作系统中，为以后的感染、破坏两个后期模块提供前期的有效准备，一般而言不同的计算机病毒</p>	<p>姓 名 莫冬春 学 号 201317170102 班 级 计网1301 专 业 计 算机网络技术... - 《互联网文档资源 ( <a href="http://www.worlduc.c">http://www.worlduc.c</a> ) 》 - 2016/7/23 8:22:51 ( 是否引证：否 )</p> <p>1.行状态或窃取系统控制权。病毒的主动传染和破坏作用，都是动态病毒的“杰作”。 - 3 -1.2.1 计算机病毒的组成模块 <b>计算机病毒由三个单元和一个标志构成：引导模块、感染模块、破坏模块和感染标志。[4] 引导模块：计算机病毒感染前</b>，需要先通过识别感染标志判断计算机系统是否被感染，若判断没有感染则将病毒的主体设法引导安装在计算机系统，为其感染模块和破坏</p>
29	<p>此处有 69 字相似</p> <p>断部分。该模块提供一个感染的标志，用来判断计算机是否被感染。另一个是执行感染功能部分。这一部分主要的功能就是监控宿主</p> <p><b>满足条件的时机，并及时的将计算机病毒存入到系统特定的位置。3、破坏表现模块与感染模块一样包括两个部分，一是具有触发破坏表现功能的判断部分。</b></p> <p>二是具有破坏表现功能的实施部分。计算机病毒一般具有寄生性、传染性、隐藏性、破坏性、潜伏性等特征。</p> <p>( 2 ) 特洛伊木马。木</p>	<p>姓 名 莫冬春 学 号 201317170102 班 级 计网1301 专 业 计 算机网络技术... - 《互联网文档资源 ( <a href="http://www.worlduc.c">http://www.worlduc.c</a> ) 》 - 2016/7/23 8:22:51 ( 是否引证：否 )</p> <p>1.有一部分是感染功能的实施部分，对未感染的计算机系统，设法将病毒侵入内存后获得运行控制并进行监视，当发现被传染的目标且判断<b>满足传染条件时，及时将计算机病毒存入系统的特定位置。破坏表现模块：一是具有触发破坏表现功能的判断部分</b>，主要判断病毒是否满足触发条件且适合破坏表现。二是具有破坏表现功能的实施部分，当病毒满足触发条件且适合破坏表现时，病毒便开</p>
30	<p>此处有 484 字相似</p> <p>表现功能的判断部分。二是具有破坏表现功能的实施部分。计算机病毒一般具有寄生性、传染性、隐藏性、破坏性、潜伏性等特征。</p> <p><b>( 2 ) 特洛伊木马。木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP 型木马以及破坏型木马等。</b></p> <p><b>( 3 ) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为</b></p>	<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证：否 )</p> <p>1.2病毒及Unix病毒等；按代码形式分为源码病毒、中间码病毒及目标码病毒等；按照宿主可以分为引导型病毒和文件型病毒。<b>( 2 ) 特洛伊木马。木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP型木马以及破坏型木马等。( 3 ) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为无害型、消耗型和破坏</b></p>



<p>无害型、消耗型和破坏型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的 CPU 和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。</p> <p>(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。</p>	<p>型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的CPU和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。</p> <p>(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程</p>
	<p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.2病毒及Unix病毒等；按代码形式分为源码病毒、中间码病毒及目标码病毒等；按照宿主可以分为引导型病毒和文件型病毒。(2) 特洛伊木马。木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP型木马以及破坏型木马等。(3) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为无害型、消耗型和破坏型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的CPU和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。</p> <p>(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程</p>
	<p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.2病毒及Unix病毒等；按代码形式分为源码病毒、中间码病毒及目标码病毒等；按照宿主可以分为引导型病毒和文件型病毒。(2) 特洛伊木马。木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP型木马以及破坏型木马等。(3) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。它通过网络、共享文件、电子邮件、移动存储设备以及有</p>

		<p>漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为无害型、消耗型和破坏型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的CPU和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。</p> <p>(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.x 病毒等；按代码形式分为源码病毒、中间码病毒及目标码病毒等；按照宿主可以分为引导型病毒和文件型病毒。(2) 特洛伊木马。木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP 型木马以及破坏型木马等。(3) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。-7-哈尔滨工业大学工学硕士学位论文它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复</p> <p>2.3) 蠕虫。蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。-7-哈尔滨工业大学工学硕士学位论文它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为无害型、消耗型和破坏型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的 CPU 和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远</p> <p>基于动态污点分析的恶意代码行为依赖图挖掘技术的研究与实现 尤作赛 - 《国防科学技术大学硕士论文》- 2012-09-01 (是否引证：否)</p> <p>1.是针对服务端来说的，因为要使木马的运行必须借助于服务端的支持。木马程序一般由客户端程序和服务端程序两部分组成。客户端是控制端，安装在攻击者的机器上，负责给服务端程序发送指令并配置服务端。而服务端被安装在受害者的机器上，负责接收客户端指令</p>
--	--	---

		，非法获取操作权限。攻击者正是利
--	--	------------------

指 标

疑似剽窃观点

1. 综上所述，现有的恶意代码检测技术有很多，每一种方法都有自身的优缺点。

疑似剽窃文字表述

1. 恶意代码层出不穷，给分析人员带来巨大的挑战。一般来说，恶意代码分析技术分为静态分析和动态分析，
2. 恶意代码及其变种得到较好的检测效果成为恶意代码检测技术的研究重点。
- 本文对基于
3. 查壳和脱壳处理。其次，对恶意代码样本进行反汇编处理，得到样本的汇编文本，并从中提取出汇编操作码序列；
4. 恶意代码的发展同样促进了检测技术的进步，目前已经有各种各样的检测技术被应用在了商业环境中。
5. 现有的基于签名的恶意代码检测技术通过特殊的字符串特征来判断，其准确率非常高，但是其缺点是不能检测新出现的恶意代码，并且需要不断的更新特征库。现在大部分研究用基于 n-gram 序列的字节序列代替二进制特征码序列，这会提高分类的准确率。
6. 在恶意代码检测技术中使用操作码序列作为特征的研究相对还是比较少的，但是研究结果发现操作码序列是一种比较好的特征表示方法。
7. 使用函数调用拓扑的缺点是，攻击者能使用相似的函数调用或者改变函数调用的序列来逃避检测。
8. 该方法是基于操作码序列的出现的频率，并挖掘了每一个操作码序列的相关性。通过大量实验对比分析，该方法是行之有效的。
9. 这为后面的研究提供了基础的同时也带来了许多挑战。本文提出了一种新的恶意代码检测方法，结合了特征码、行为及机器学习的方法，提出了基于操作码序列的静态恶意代码检测方法，能更好的检测恶意代码。
- 本文改进了一种基于机器学习算法的恶意代码检测方法，
10. 本文的第一步就是对恶意代码进行查壳和脱壳处理；其次对恶意代码样本进行反汇编处理，得到样本的汇编文本，并从中提取出汇编操作码序列；
11. 进行实验仿真，并对实验结果进行对比分析，得出结论。
- 第四：结合实验仿真的结果分析，针对本文提出的改进方法，
12. 第一章为绪论，主要介绍课题背景及意义。并且详细阐述了国内外对于恶意代码检测技术的研究现状和存在的问题，最后介绍了本文的研究内容和章节安排。
- 第二章是
13. 根据不同的依据，恶意代码有很多种不同的分类方法，没有一种标准的分法，但是常见的种类有：计算机病毒、蠕虫、特洛伊木马、间谍软件、勒索软件等等。下面对几种恶意代码做简要介绍：
- (1)计算机病毒。
14. 计算机病毒通常由三个单元和一个标志构成：引导模块、感染模块、破坏表现模块和感染指标。1、引导模块是指将计算机病毒感染
15. 满足条件的时机，并及时的将计算机病毒存入到系统特定的位置。3、破坏表现模块与感染模块一样包括两个部分，一是具有触发破坏表现功能的判断部分。
16. 木马分为客户端和服务端，客户端安装在攻击者的主机上是控制端，服务端安装在受害者的机器上。木马可以使攻击者远程控制受害者的主机，造成受害者信息丢失等问题。木马有很好的隐蔽性，通过模仿正常的系统文件命名、与其他程序绑定、进程注入及拦截系统调用的方法伪装自己。木马也有很好的自启动性和自恢复性。常见木马有远程访问型木马、键盘记录型木马、密码发送型木马、FTP 型木马以及破坏型木马等。
17. 蠕虫是一种可以独立运行、自我复制及自动传播的恶意程序。它通过网络、共享文件、电子邮件、移动存储设备以及有漏洞的主机等自我复制和传播。蠕虫的传播速度非常快，根据它的危害性可以简单分为无害型、消耗型和破坏型。无害型蠕虫感染主机后会产生很多垃圾文件减少系统的可用空间；消耗型蠕虫感染主机后，发送大量扫描数据包，消耗主机的 CPU 和内存资源，与此同时增加了网络的负载，降低网络的性能；破坏型蠕虫感染主机后会删除和破坏程序和文件，有时会泄露一些重要信息。
18. 它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。

15. 80022497643220302\_基于动态手势的智能终端身份识别技术研究与实践\_第15部分 总字数：8875

相似文献列表 文字复制比：57.8%(5132) 疑似剽窃观点：(0)

1	基于操作码序列的静态恶意代码检测方法的研究 卢占军(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2012-12-01	26.0% ( 2307 ) 是否引证：否
2	10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12	25.2% ( 2233 ) 是否引证：否
3	BH2009925453	25.2% ( 2233 )

	- 《学术论文联合比对库》- 2012-12-11	是否引证：否
4	BH2009921397 - 《学术论文联合比对库》- 2012-12-04	24.8% ( 2204 ) 是否引证：否
5	恶意代码检测中若干关键技术研究 陈良(导师：李斌;陈斌) - 《扬州大学硕士论文》- 2012-04-01	12.3% ( 1091 ) 是否引证：否
6	094616006_冯本慧 冯本慧 - 《学术论文联合比对库》- 2013-09-23	11.4% ( 1008 ) 是否引证：否
7	基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧(导师：王加阳) - 《中南大学硕士论文》- 2013-09-01	10.7% ( 946 ) 是否引证：否
8	基于操作码行为深度学习的恶意代码检测方法 陈晨(导师：丁宇新) - 《哈尔滨工业大学硕士论文》- 2013-12-01	9.4% ( 832 ) 是否引证：否
9	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-27	8.6% ( 766 ) 是否引证：否
10	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-14	8.6% ( 766 ) 是否引证：否
11	结合语义的统计机器学习方法在代码安全中应用研究 孔德光(导师：奚宏生) - 《中国科学技术大学博士论文》- 2010-05-01	8.6% ( 764 ) 是否引证：否
12	恶意代码行为挖掘关键技术研究 解培岱(导师：卢锡城) - 《国防科学技术大学博士论文》- 2013-10-01	2.0% ( 181 ) 是否引证：否
13	浅谈病毒与反病毒 梅净缘; - 《电脑与信息技术》- 2012-04-15	1.4% ( 124 ) 是否引证：否
14	液压支架电液控制系统的研究与实现 郭科伟(导师：李志敏) - 《重庆大学硕士论文》- 2012-04-01	0.4% ( 36 ) 是否引证：否
15	基于沙箱技术的恶意代码行为自动化检测方法 李志勇(导师：李伟明) - 《华中科技大学硕士论文》- 2015-05-01	0.3% ( 31 ) 是否引证：否
16	基于污点跟踪的固件漏洞定位研究 戴忠华;费永康;赵波;王婷; - 《山东大学学报(理学版)》- 2016-04-18 1	0.3% ( 30 ) 是否引证：否

原文内容		相似内容来源
1	此处有 603 字相似	基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )
	<p>后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。</p> <p>( 5 ) Rootkit。它是指帮助攻击者获取主机管理权限后，实现维持拥有管理权限的程序[36]。通常攻击者通过后门获取管理权限，并使用 Rootkit 维持管理权限使的恶意代码能隐藏在目标系统中。Rootkit 分为用户模式和内核模式。用户模式通过通道插入恶意代码、覆盖文件、API 钩子和 DLL 注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。</p> <p>( 6 ) 间谍软件。它是在未授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。</p> <p>( 7 ) 广告软件。它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。</p> <p>( 8 ) 恶意网页脚本。它是指在网页中嵌入一些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本</p>	<p>1。( 4 ) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。</p> <p>( 5 ) Rootkit。它是指帮助攻击者获取主机管理权限后，实现维持拥有管理权限的程序[42]。通常攻击者通过后门获取管理权限，并使用 Rootkit 维持管理权限使的恶意代码能隐藏在目标系统中。Rootkit 分为用户模式和内核模式。用户模式通过通道插入恶意代码、覆盖文件、API 钩子和 DLL 注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。</p> <p>( 6 ) 间谍软件。它是在未授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。</p> <p>( 7 ) 广告软件。它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。</p> <p>( 8 ) 恶意网页脚本。它是指在网页中嵌入一</p>



<p>的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。</p> <p>目前用于商业的恶意代码检测软件中，一般采用的都是基于“特征码”的检测技术，基本思想是，当新的恶意代码被发现后，对其进行采</p>	<p>些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。下面对几种恶意代码做简单对比：表 2-1 几种恶意代码的比较名称 特征 存在形式 传播途径 传播方式 典型代</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.要信息。(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。(5) Rootkit。它是指帮助攻击者获取主机管理权限后，实现维持拥有管理权限的程序[42]。通常攻击者通过后门获取管理权限，并使用Rootkit维持管理权限的恶意代码能隐藏在目标系统中。Rootkit分为用户模式的Rootkit和内核模式的Rootkit。用户模式通过通道插入恶意代码、覆盖文件、API钩子和DLL注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。(6) 间谍软件。它是在未授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。(7) 广告软件。它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。(8) 恶意网页脚本。它是指在网页中嵌入一些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。下面对几种恶意代码做简单对比：表2-1几种恶意代码的比较表2-1 (续表) 2.2 恶意代码的检测技术</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.要信息。(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。(5) Rootkit。它是指帮助攻击者获取主机管理权限后，实现维持拥有管理权限的程序[42]。通常攻击者通过后门获取管理权限，并使用Rootkit维持管理权限的恶意代码能隐藏在目标系统中。Rootkit分为用户模式和内核模式。用户模式通过通道插入恶意代码、覆盖文件、API钩子和DLL注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。(6) 间谍</p>
--	---

		<p>软件。它是在未授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。(7) 广告软件。它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。(8) 恶意网页脚本。它是指在网页中嵌入一些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。下面对几种恶意代码做简单对比：表2-1几种恶意代码的比较表2-1 (续表) 2.2恶意代码的检测技术</p>
	10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 (是否引证：否)	<p>1.要信息。(4) 后门。它是一种运行在目标系统中，能够绕过安全控制机制获得对系统的访问权，为攻击者提供通道的恶意代码。后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。(5) Rootkit。它是指帮助攻击者获取主机管理权限后，实现维持拥有管理权限的程序[42]。通常攻击者通过后门获取管理权限，并使用Rootkit维持管理权限使的恶意代码能隐藏在目标系统中。Rootkit分为用户模式和内核模式。用户模式通过通道插入恶意代码、覆盖文件、API钩子和DLL注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。(6) 间谍软件。它是在未授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。(7) 广告软件。它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。(8) 恶意网页脚本。它是指在网页中嵌入一些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。下面对几种恶意代码做简单对比：表2-1几种恶意代码的比较表2-1 (续表) 2.2恶意代码的检测技术</p>
	基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 -《中南大学硕士论文》- 2013-09-01 (是否引证：否)	<p>1.验样本集。部分恶意代码利用PEid查壳的结果如图4-8所示，从图4-8可以看出，利用PEid工具不仅可以查出恶意代码加壳的信息，也可以得编写恶意代码程序的编译器的相关息—_■^ - ——C；\select\DR. Delphi. Gen.</p>

2	<p>此处有 94 字相似</p> <p>后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。</p> <p>目前用于商业的恶意代码检测</p> <p>软件中，一般采用的都是基于“特征码”的检测技术，基本思想是，当新的恶意代码被发现后，对其进行采集取样，分析代码构成，提取有用的特征码，然后将新的特征码加入已有数据库中，用户更新病毒库之后，就会使用新的特征库去匹配恶意代码，如果匹配成功则进行相应的处理。但是特征码检测技术的缺点是只能对已知恶意代码进行有效的将</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.也在成倍增加，在信息安全领域如何有效的检测恶意代码成为研究的重点。目前的恶意代码的研究中，用于商业的病毒检测软件一般采用的都是基于“特征码”的检测技术，它的基本思想是当新的病毒被发现后，对其进行采集取样，分析代码的构成，提取有用的特征码，然后将新提取的特征加入的病毒库中，用户在对病毒库进行更新后在下一次扫描计算机的时候就会拿病毒库中的特征去匹配病毒，如果匹配成功则提示发现病毒并进行相应的处理。但是</p>
3	<p>此处有 57 字相似</p> <p>后，对其进行采集取样，分析代码构成，提取有用的特征码，然后将新的特征码加入已有数据库中，用户更新病毒库之后，就会使用新的</p> <p>特征库去匹配恶意代码，如果匹配成功则进行相应的处理。但是特征码检测技术的缺点是只能对已知恶意代码进行有效的将测，对</p> <p>与未知或者稍加变动的恶意代码无能为力。因此，在恶意代码检测领域提出了启发式检测算法来预防和检测新的恶意代码。根据对恶意代</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.取有用的特征码，然后将新提取的特征加入的病毒库中，用户在对病毒库进行更新后在下一次扫描计算机的时候就会拿病毒库中的特征去匹配病毒，如果匹配成功则提示发现病毒并进行相应的处理。但是特征码检测技术的缺点就是只能检测已知的恶意代码，对新出现的恶意代码则无能为力。因此，在恶意代码检测领域提出了启发式检测算法来预防和检测新的恶意代码。根据对恶意代码分</p>
4	<p>此处有 122 字相似</p> <p>相应的处理。但是特征码检测技术的缺点是只能对已知恶意代码进行有效的将测，对与未知或者稍加变动的恶意代码无能为力。因此，在</p> <p>恶意代码检测领域提出了启发式检测算法来预防和检测新的恶意代码。根据对恶意代码分析原理的不同对现有恶意代码检测方法进行分类，主要分为基于特征码的检测技术、基于行为的检测技术、基于启发式的检测技术、基于语义的检测方法和基于机器学习算法的检测技术等。</p> <p>下面将详细介绍几种检测技术。</p> <p>(1) 基于特征码的检测技术</p> <p>基于特征码的检测方法是使用最古老和最广泛的方法。被Syma</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.现病毒并进行相应的处理。但是特征码检测技术的缺点就是只能检测已知的恶意代码，对新出现的恶意代码则无能为力。因此，在恶意代码检测领域提出了启发式检测算法来预防和检测新的恶意代码。根据对恶意代码分析原理的A6 扬州大学硕士学位论文不同对现有的恶意代码检测方法进行分类主要分为基于特征码的检测方法、基于函数特征的检测方法、启发式检测方法、基于行为的检测方法等。1.2.1基于特征码的检测法基于特征码的检测方法[7]是使用最广泛和最古老的方法，这种方法通过密罐系统提取恶意代码的样木，并分析采集他</p> <p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 - 《中南大学硕士论文》 - 2013-09-01 (是否引证：否)</p> <p>1.为基于网络的检测技术和基于主机的检测检测技术两类，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码检测技术主要有基于特征码的检测、基于完整性的检测、基于启发式和基于语义的检测等方法，但这些检测方法存在过于依赖特征码、过于依赖分析人员的经验、无法检测未知恶意代码等问题，面对</p> <p>094616006_冯本慧 冯本慧 - 《学术论文联合比对库》 - 2013-09-23 (是否引证：否)</p> <p>1.主要分为基于网络的检测技术和基于主机的检测检测技术两类，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码检测技术主要有基于特征码的检测、基于完整性的检测、基于启发式和基于语义的检测等方法，但这些检测方法存在过于依赖特征码、过于依</p>

5		<p>赖分析人员的经验、无法检测未知恶意代码等问题，面对当前数</p> <p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》 - 2010-05-01 (是否引证：否)</p> <p>1.壮，比仅仅的静态污点分析方法更易于定量和相似性比较。N4:提出了一种结合控制流和系统调用特征的迷惑<b>恶意代码检测算法</b>，用于检测迷惑后的恶意代码。控制流和系统调用抓取了代码的语义特征，而同时结合的统计特征，相互补充，尽可能准确地进行迷惑恶意代码的类别检测</p>
	<p>此处有 46 字相似</p> <p>检测技术、基于行为的检测技术、基于启发式的检测技术、基于语义的检测方法和基于机器学习算法的检测技术等。下面将详细介绍几种检测技术。</p> <p>(1) 基于特征码的检测技术</p> <p>基于特征码的检测方法是使用最古老和最广泛的方法。被Symantec等多有著名病毒检测厂商所使用，是目前已知的所有恶意代码检测方法中最简单、开销最小的方法，广泛用于文件类</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.为基于特征码的检测方法、基于函数特征的检测方法、启发式检测方法、基于行为的检测方法等。1.2.1基于特征码的<b>检测法基于特征码的检测方法[7]是使用最广泛和最古老的方法</b>，这种方法通过密罐系统提取恶意代码的样本，并分析采集他们的独有的特征指令序列，当反病毒软件扫描文件时，将当前的</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) <b>特征码检测方法。基于特征码的检测技术</b>，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。<b>2.2.1 特征码检测技术基于特征码的检测技术</b>，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) <b>特征码检测方法。基于特征码的检测技术</b>，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) <b>特征码</b></p>



		检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过
6	<p>此处有 67 字相似</p> <p>技术等。下面将详细介绍几种检测技术。</p> <p>(1) 基于特征码的检测技术</p> <p>基于特征码的检测方法是使用最古老和最广泛的方法。被</p> <p>Symantec等多有著名病毒检测厂商所使用，是目前已知的所有恶意代码检测方法中最简单、开销最小的方法，广泛用于文件类型的病毒检测中。</p> <p>检测软件的核心就是恶意代码特征库的完全性，当需要扫描某个程序是否有恶意企图时，启动特征扫描提取特征，然后再与特征库进行匹</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.有特征代码法、校验和法、行为监测法、软件模拟法等。特征代码法 特征代码法被symantec[8]等所有著名病毒检测厂商使用，是目前已知的检测已知恶意代码的最简单、开销最小的方法，广泛用于文件类型的病毒检测中。防毒厂商是建立尽可能全的已知病毒的病毒码特征库，当需要扫描某程序是否恶意企图时，启动特征扫描与匹配程序，通过</p>
7	<p>此处有 32 字相似</p> <p>使用，是目前已知的所有恶意代码检测方法中最简单、开销最小的方法，广泛用于文件类型的病毒检测中。检测软件的核心就是恶意代码</p> <p>特征库的完全性，当需要扫描某个程序是否有恶意企图时，启动特征扫描</p> <p>提取特征，然后再与特征库进行匹配，如果匹成功，则判断该程序是恶意的。此技术的关键在于如何选取最能代表恶意程序的特征值。采</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.已知恶意代码的最简单、开销最小的方法，广泛用于文件类型的病毒检测中。防毒厂商是建立尽可能全的已知病毒的病毒码特征库，当需要扫描某程序是否恶意企图时，启动特征扫描与匹配程序，通过字符串或者正则表达式匹配，如果匹配上，则判定该程序已被感染。此技术的关键在于如何选取最能表</p>
8	<p>此处有 45 字相似</p> <p>。检测软件的核心就是恶意代码特征库的完全性，当需要扫描某个程序是否有恶意企图时，启动特征扫描提取特征，然后再与特征库进行</p> <p>匹配，如果匹成功，则判断该程序是恶意的。此技术的关键在于如何选取最能代表恶意程序的特征值。</p> <p>采用该方法，检验结果准确，鲜有误报情况，但该方法对于未知或者变形恶意代码无能为力。还有，这种方法使得特征库不断增加，这需</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.全的已知病毒的病毒码特征库，当需要扫描某程序是否恶意企图时，启动特征扫描与匹配程序，通过字符串或者正则表达式匹配，如果匹配上，则判定该程序已被感染。此技术的关键在于如何选取最能表征恶意代码特征的特征(字符串)值，鲜有厂商愿意公开此技术细节。字符串相关的匹配和优化技术能够减少运行开销，提高效率。如何建立</p>
9	<p>此处有 217 字相似</p> <p>在于如何选取最能代表恶意程序的特征值。采用该方法，检验结果准确，鲜有误报情况，但该方法对于未知或者变形恶意代码无能为力。</p> <p>还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。</p> <p>(2) 基于行为的检测技术</p> <p>基于行为的检测方法是利用恶意代码的特有行为来检测恶意代码的方法。恶意代码的行为有相对的稳定性和易于检测的特点，比如特定的系统调用，恶意代码要完成自身逻辑功能，即完成对系统的入侵和破坏，就必须获取系统非法权限，调用系统的资源[37]，这样通过分析</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》- 2012-04-01 (是否引证：否)</p> <p>1.商业的杀毒技术中并进行了推广，根据用户反馈来看，资源占用严重仍然是该技术有待提高的主要方面之一。</p> <p>1.2.4基于行为的检测法基于行为的检测方法是利用恶意代码的特有行为特征来检测病毒的方法。恶意代码程序的行为有相对稳定性和易于检测性的特点，比如系统调用，恶意代码要完成对系统的入侵和破坏行为就必须调用系统的资源"1】，这样通过分析恶意代码的行为就可以方便的分析8扬州大学硕士学位论文 检测出恶意代码。计算机被入侵大都是因为对系统资源和系统权限的滥用。由于对主机的权限</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -</p>

<p>恶意代码的行为就可以方便的分析</p> <p>检测出恶意代码。当程序运行时，监控其行为，如果发现异常行为，则立即报警。一般用于检测恶意代码的行为特征如下：</p> <p>1)</p>	<p>《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )</p> <p>1.有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。从图 2-1 中可以看出，当恶意代码入侵时，首先扫面其特征与特征库匹配，如果检测该样本为恶意样本则将其清除，否则分</p> <p>2.过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。( 1 )特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过扫描</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。从图2-1中可以看出，当恶意代码入侵时，首先扫面其特征与特征库匹配，如果检测该样本为恶意样本，并将其清除，则分离恶意代码</p> <p>2.码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。( 1 )特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过扫描样本的相关特</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。从图2-1中可以看出，当恶意代码入侵时，首先扫面其特征与特征库匹配，如果检测该样本为恶意样本，并将其清除，则分离恶意代码</p> <p>2.码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。( 1 )特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通过扫描</p>
--	--

	<p>样本的相关特</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。从图2-1中可以看出，当恶意代码入侵时，首先扫面其特征与特征库匹配，如果检测该样本为恶意样本，并将其清除，则分离恶意代码</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p> <p>1.术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。2.2.3启发式法检验启发式检验是结合简化行为判断引擎和简化虚拟机的一种检测技术，主要目的在于检测已知恶意代码的变</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p> <p>1.术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。2.2.3启发式法检验启发式检验是结合简化行为判断引擎和简化虚拟机的一种检测技术，主要目的在于检测已知恶意代码的变</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)</p> <p>1.有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。2.2.3 启发式法检验启发式检验[4]是结合简化行为判断引擎和简化虚拟机的一种检测技术，主要目的在于检测已</p> <p>094616006_冯本慧 冯本慧 - 《学术论文联合比对库》 - 2013-09-23 (是否引证：否)</p> <p>1.性的检测方法。各种不同检测技术有各自的优点，也有各自的局限，因此往往将各种检测技术进行结合，取长补短，提高恶意代码检测的精度。(4)基于语法的检测技术基于语法的检测技术是通过静态分析或动态分析(如动态污点分析抽象恶意代码的静态或动态行为语义，从恶意代码的逻辑关系出发利用形式化的方法</p>
--	--

		<p>浅谈病毒与反病毒 梅净缘; -《电脑与信息技术》- 2012-04-15 (是否引证: 否)</p> <p>1.,而且此法会影响文件的运行速度。2.3行为检测行为检测,行为检测最近在国内外都有大量的宣传,但其实行为检测并不是什么新的技术。行为检测是利用病毒的特有行为特征检测病毒的方法,也称为人工智能陷阱。通过对病毒多年的观察、研究,研究者发现病毒有一些行为是病毒的共同行为,而且及其特殊。在正常程序中,这</p> <p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 -《中国科学技术大学博士论文》- 2010-05-01 (是否引证: 否)</p> <p>1.法对隐蔽性病毒也无效。隐蔽性病毒进驻内存后,会自动剥去染毒程序中的病毒代码,使校验和法失效。基于行为的检测技术 利用恶意代码的一些特定行为,来对病毒进行监测,而这些行为["],在正常程序中出现比较罕见。此方法,一般需要动态执行程序或者至少虚拟执行</p>
10	<p>此处有 44 字相似</p> <p>的入侵和破坏,就必须获取系统非法权限,调用系统的资源[37],这样通过分析恶意代码的行为就可以方便的分析检测出恶意代码。</p> <p>当程序运行时,监控其行为,如果发现了异常行为,则立即报警。一般用于检测恶意代码的行为特征如下:</p> <p>1) 对特定文件执行写操作:有些恶意代码时依附而生,所以在其执行时,就要将自身代码附加在感染文件中,可以监控是</p>	<p>浅谈病毒与反病毒 梅净缘; -《电脑与信息技术》- 2012-04-15 (是否引证: 否)</p> <p>1.究,研究者发现病毒有一些行为是病毒的共同行为,而且及其特殊。在正常程序中,这些行为十分罕见。当程序运行时,我们可以监视其行为,一旦发现了特征行为,立即报警或阻塞可疑程序。用于检测病毒的行为特征主要有以下几点:盗用截流系统中断、修改内存总量和内存控制块、对可执行文件做写入操作、写引导扇区或执行格式化磁盘等可疑动作</p>
11	<p>此处有 76 字相似</p> <p>常行为,则立即报警。一般用于检测恶意代码的行为特征如下:</p> <p>1) 对特定文件执行写操作:有些恶意代码时依附而生,所以在其</p> <p>执行时,就要将自身代码附加在感染文件中,可以监控是否有异常写操作。</p> <p>2) 监控系统调用序列:某些系统调用序列可以体现某种程度的程序语义。系统调用是用户</p> <p>态和内核态的唯一接口,恶意代码想要获取高级权限实施破坏行为,就必然要经过系统调用接口。</p> <p>3) 修改内存总量:恶意代码为</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 -《中国科学技术大学博士论文》- 2010-05-01 (是否引证: 否)</p> <p>1.统及其它应用 程序不能占据其空间。3.对特定文件执行写操作:由于病毒是依附而生,那么在病毒执行时,就要将自身代码附加在被感染文件之中,可以监视是否有写操作。4.监测系统调用序列,某些系统调用序列是否体现某种程度的恶意语义。系统调用是用户应用程序与操作系统的唯一接口。例如shallcode中经常使用的 监视函数GetproeAddress·</p>
12	<p>此处有 77 字相似</p> <p>用序列可以体现某种程度的程序语义。系统调用是用户态和内核态的唯一接口,恶意代码想要获取高级权限实施破坏行为,就必然要经过</p> <p>系统调用接口。</p> <p>3) 修改内存总量:恶意代码为了完成特定的恶意意图,经常会常驻在内存中,并且不能被覆盖,那么将会减少系统内存的总量,使得该段内存不受系统内核控制。</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 -《中国科学技术大学博士论文》- 2010-05-01 (是否引证: 否)</p> <p>1.引导会首先加载病毒代码,完成驻留后,会将自身代码隐藏起来,继续执行系统正常功能。2.修改系统内存总量:病毒为了完成感染与破坏等特定功能,会常驻在内存中,并且不能被覆盖掉。病毒程序将减少系统内存总量,使系统及其它应用 程序不能占据其空间。3.对特定文件执行写操作:由于病毒是依附而生,那么在病毒执</p>



	<p>(3) 基于启发式的检测技术</p> <p>启发式检测方法是对恶意代码特征提前设定一个阈值，在对文件进行扫描后，当提取的</p>	
13	<p>此处有 137 字相似</p> <p>且不能被覆盖，那么将会减少系统内存的总量，使得该段内存不受系统内核控制。</p> <p>(3) 基于启发式的检测技术</p> <p>启发式检测方法</p> <p>是对恶意代码特征提前设定一个阈值，在对文件进行扫描后，当提取的特征和恶意代码特征的相似度达到一定的值，这认定该文件是恶意代码。例如一些恶意代码都会固定的对一些内核函数进行调用，通常这些调用的顺序是有一定的规律，因此利用对内核函数的名称和调用次数进行分析，可以构建一个恶意代码</p> <p>对内核函数的特征。启发式方法属于主动防御技术，对未知的恶意代码检测具有明显的效果，因此，这种方法在现今的商业开发被重点</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士学位论文》 - 2012-04-01 (是否引证：否)</p> <p>1. 件使用了哪些有威胁的函数，并大致可以估计其功能和属于哪种类型。1.2.3 启发式检测法启发式检测方法的主要思想是对恶意代码特征提前设定一个阈值，在对文件进行扫描的时候，当文件和恶意代码特征的相似程度达到一定的值，就将它当作是恶意代码。比如对于一些恶意代码，都会固定的对一些内核函数进行调用。通常情况下这些调用的函数在代码中出现的顺序也是有一定的规律，因此利用其对内核函数的名称或者次数的调用进行陈良：恶意代码检测中若干关键问题研究 7 分析，可以构建一个恶意代码对内核函数调用的集合，进而比较待查的程序调用的内核函数与</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士学位论文》 - 2012-12-01 (是否引证：否)</p> <p>1. 险系数，通过统计设定一个风险阈值，然后根据程序的行为累加各种行为的危险值，当超过预先设定的风险阈值时则判定为恶意代码。例如，一些恶意代码执行时，会调用一些特定的内核函数，而往往这些函数的顺序也有规律，可以根据这些特征来指定一些规则检测同类恶意程序。这种方法也需要人工的参与分析，如果是自动提取也难免会受到混淆</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1. 一些行为的危险系数，通过统计设定一个风险阈值，然后根据程序的行为累加各种行为的危险值，当超过预先设定的风险阈值时则判定为恶意代码。例如，一些恶意代码执行时，会调用一些特定的内核函数，而往往这些函数的顺序也有规律，可以根据这些特征来指定一些规则检测同类恶意程序。这种方法也需要人工的参与分析，如果是自动提取也难免会受到混淆技术的干扰。</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1. 一些行为的危险系数，通过统计设定一个风险阈值，然后根据程序的行为累加各种行为的危险值，当超过预先设定的风险阈值时则判定为恶意代码。例如，一些恶意代码执行时，会调用一些特定的内核函数，而往往这些函数的顺序也有规律，可以根据这些特征来指定一些规则检测同类恶意程序。这种方法也需要人工的参与分析，如果是自动提取也难免会受到混淆技术的干扰。</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p> <p>1. 一些行为的危险系数，通过统计设定一个风险阈值，然后根据程序的行为累加各种行为的危险值，当超过预先设定的风险阈值时则判定为恶意代码。例如，一些恶意代码执行时，会调用一些特定的内核函数，而往往</p>

		<p>这些函数的顺序也有规律，可以根据这些特征来指定一些规则检测同类恶意程序。这种方法也需要人工的参与分析，如果是自动提取也难免会受到混淆技术的干扰。</p>
14	<p>此处有 128 字相似</p> <p>用，通常这些调用的顺序是有一定的规律，因此利用对内核函数的名称和调用次数进行分析，可以构建一个恶意代码对内核函数的特征。</p> <p>启发式方法属于主动防御技术，对未知的恶意代码检测具有明显的效果，因此，这种方法在现如今的商业开发被重点应用。启发式检测可分为静态启发和动态启发。静态启发方法其实是对传统的特征识别方法的一种扩展，通过分析程序对系统API的调用序列作为特征，有领域专家根据自身经验，研究总结出某些恶意代码的行为特征，当对行为进行监控时，此类特征一旦被发现，就立即报警并做相应的处理。这种方法能</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.码对内核函数调用的集合，进而比较待查的程序调用的内核函数与数据库中已才在的恶意代码的内核函数调用集合的相似度程度；启发式方法也是属于主动防御技术，它是对未知的恶意代码进行检测的重要手段，因此，这种方法在现如今的商业开发中被重点应用。启发式检测方法又可称为行为监测法，它是通过对程序的行为进行监测判别的方法，主要分为静态启发和动态启发两种方法。静态启发方法其实</p> <p>2.现如今的商业开发中被重点应用。启发式检测方法又可称为行为监测法，它是通过对程序的行为进行监测判别的方法，主要分为静态启发和动态启发两种方法。静态启发方法其实就是对传统的特征识别方法的一种扩展，这种方法是通过分析程序对系统API的调用序列作为特征，由领域专家系统根据长期对恶意代码的行为的观察、研究并总结出某些恶意代码的共有的行为特征，当对行为监测时，此类特征一旦被发现，就</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.战，需要不断的创新技术来应对，而恶意代码的增长主要是由混淆技术产生的变种导致，所以检测恶意代码的变种成为人们研究的焦点。恶意代码检测技术主要分为静态和动态两种检测方法。静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。现在使</p> <p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)</p> <p>1.的挑战，需要不断的创新技术来应对，而恶意代码的增长主要是由混淆技术产生的变种导致的，所以检测恶意代码的变种成为人们研究的焦点。恶意代码检测技术主要分为静态和动态两种检测方法。静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。现在使用最广泛的静</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.的挑战，需要不断的创新技术来应对，而恶意代码的增长主要是由混淆技术产生的变种导致的，所以检测恶意代码的变种成为人们研究的焦点。恶意代码检测技术主要分为静态和动态两种检测方法。静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。现在使用最广泛的静</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)</p>

		<p>1.的挑战，需要不断的创新技术来应对，而恶意代码的增长主要是由混淆技术产生的变种导致的，所以检测恶意代码的变种成为人们研究的<b>焦点</b>。<b>恶意代码检测技术主要分为静态和动态两种检测方法</b>。静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。现在使用最广泛的静</p>
15	<p>此处有 108 字相似</p> <p>静态启发方法其实是对传统的特征识别方法的一种扩展，通过分析程序对系统API的调用序列作为特征，有领域专家根据自身经验，</p> <p><b>研究总结出某些恶意代码的行为特征，当对行为进行监控时，此类特征一旦被发现，就立即报警并做相应的处理。这种方法能够有效的检测出已知的恶意代码，并发现部分未知的恶意代码，但在发现恶意代码的时候，系统往往已经被感染。另外，</b></p> <p>行为检测是对系统进行实时的监控，因此可能会持续占用大量内存、CPU等系统资源。在商业领域中，该方法主要用于辅助性检测。</p>	<p>恶意代码检测中若干关键技术研究 陈良 -《扬州大学硕士学位论文》- 2012-04-01 (是否引证：否)</p> <p>1.种扩展，这种方法是通过分析程序对系统API的调用序列作为特征，由领域专家系统根据长期对恶意代码的行为的观察、<b>研究并总结出某些恶意代码的共有的行为特征，当对行为监测时，此类特征一旦被发现，就立即报警并且终止进程。这种方法能够有效的检测出已知的恶意代码，并发现部分未知的恶意代码，但在发现恶意代码的时候，系统常常已经被感染，并且其漏报率与误警率依赖于所制定行为特征的准确性和广泛度。另外，</b>行为监测是实时的对系统进行监测，因此监测时易</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士学位论文》- 2012-12-01 (是否引证：否)</p> <p>1.是自动生成，但是有一些需要该领域的-9-哈尔滨工业大学工学硕士学位论文专家人工提取比较好的特征；其次，<b>这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；</b>再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这</p> <p>BH2009921397 -《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.要方式。但是这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，<b>这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；</b>再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.要方式。但是这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，<b>这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；</b>再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.要方式。但是这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，<b>这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；</b>再次，该</p>

		<p>技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1.要方式。但是这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.要方式。但是这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增加，这需要用</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证：否)</p> <p>1.这种技术也有很多缺点：首先，特征提取时有的是自动生成，但是有一些需要该领域的专家人工提取比较好的特征；其次，这种技术只能检测已有的恶意代码，对未知的恶意代码种类束手无策；再次，该技术提取的特征的泛化能力不足，很容易受到混淆技术的干扰；还有，这种方法使得特征库不断增</p>
16	<p>此处有 135 字相似</p> <p>为检测是对系统进行实时的监控，因此可能会持续占用大量内存、CPU等系统资源。在商业领域中，该方法主要用于辅助性检测。</p> <p>动态启发式技术主要的工作原理是在计算机系统中划分出一各独立的虚拟环境，当发现可疑程序时，并不立即停止，而是让其继续运行。“沙盒”技术就是动态启发式技术的一种，沙盒会对可疑程序的行为进行记录，直到恶意代码完全暴露后，它在执行回滚操作，使计算机恢复到执行可疑程序之前的状态。</p> <p>近年来病毒检测厂商已经将沙盒技术应用与商业的查杀工具中并进行了推广。</p> <p>(4) 基于语义的检测技术</p> <p>基于语义的检测技术是</p>	<p>恶意代码检测中若干关键技术研究 陈良 -《扬州大学硕士论文》- 2012-04-01 (是否引证：否)</p> <p>1.工具，用户一般和其它的反病毒软件联合使用，在一定程度上能够保护计算机避免严重的侵害，但是它的稳定性需要改进。动态启发式技术的主要工作原理是在计算机系统中划分出一个虚拟的环境，当发现程序有可疑的行为时，并不对其进行终止，而是让可疑行为继续运行。“沙盒”技术[9]就是动态启发式技术的一种，沙盒会对可疑行为的各个动作进行记录，直到病毒完全的暴露后，它会执自带的回滚操作以使计算机恢复到可疑行为出现前的状态。沙盒技术因为使用了虚拟机的技术，所以要考虑资源的占用问题，对其行为监控的过程也要严格的控制，因此沙盒技术一直</p>
17	<p>此处有 946 字相似</p> <p>回滚操作，使计算机恢复到执行可疑程序之前的状态。近年来病毒检测厂商已经将沙盒技术应用与商业的查杀工具中并进行了推广。</p> <p>(4) 基于语义的检测技术</p> <p>基于语义的检测技术是现在研究的热点。因为混淆技术</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.一旦检测到所处的环境是虚拟机便会执行一些其他的行为来躲避检测。对于这样的恶意代码动态检测技术无能为力。(7) 基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃</p>



<p>只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，抽象程序指令的行为并建立其行为模型，使得该模型既描述恶意程序的基本行为，又具有很强的泛化能力。这样因其有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。</p> <p>M.Christodorescu[8][9]提出了一套抽象理论和语义框架，使用自动机描述程序的行为，通过抽象理论描述程序的行为建立抽象模式库，并将其作为自动机的符号表，最终经恶意行为描述为自动机表述的模板，最后通过模型检测方法检测样本是否含有恶意行为。模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成恶意代码的检测。</p> <p>( 5 ) 基于机器学习的检测技术</p> <p>基于机器学习和数据挖掘的检测方法。随着检测技术的不断发展，机器学习和数据挖掘的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。</p> <p>恶意代码检测与反检测技术总是相互促进，相辅相成。检测技术的进步也带动了反检测技术的发展，当前出现了各种各样的恶意代码反检测技术，现总结如下：</p> <p>( 1 ) 加壳技术</p> <p>恶意代码</p> <p>作者为了防止自己的程序被检测软件发现，利用一些软件技术给恶意代码加外壳，可以是利用算法将自己伪装成正常程序，或者利用特殊</p>	<p>圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，抽象程序指令的行为并建立其行为模型，使得该模型既描述恶意程序的基本行为，又具有很强的泛化能力。这样因其有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。M.Christodorescu[11,12]提出了一套抽象理论和语义框架，使用自动机描述程序的行为，通过抽象理论描述程序的行为建立抽象模式库，并将其作为自动机的符号表，最终经恶意行为描述为自动机表述的模板，最后通过模型检测方法检测样本是否含有恶意行为。模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出-11-哈尔滨工业大学工学硕士学位论文迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，通过</p> <p>2.统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出-11-哈尔滨工业大学工学硕士学位论文迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成恶意代码的检测。从图 2-2 中可以看出，首先，要根据语义分析的理论分析程序，并提取行为特征建立语义模型最终形成一个语义模板库。其</p> <p>3.环境，一旦检测到所处的环境是虚拟机便会执行一些其他的行为来躲避检测。对于这样的恶意代码动态检测技术无能为力。( 7 ) 基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程</p> <p>4.建立，这还需要后人不断的努力探索。图 2-2 基于语义的恶意代码检测的模型( 8 ) 基于机器学习和数据挖掘的检测方法。随着检测技术的不断发展，机器学习和数据挖掘的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种-12-哈尔滨工业大学工学硕士学位论文特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提</p> <p>5.的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种-12-哈尔滨工业大学工学硕士学位</p>
---	---

论文特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下图详细介绍了现在应用机器学习实现恶意代码检测的几种方法。从图 2-3 中可以看出，现在已经有很多机器学习技术应用

6.中 加载脚本一些木马、病毒嵌入网页 网页浏览 人工一些恶意网站等2.2 恶意代码的检测技术随着**恶意代码检测技术和反检测技术的对抗发展**，现在已经有了很多检测技术。恶意代码检测技术主要分为静态和动态检测。静态检测不执行代码，通过代码的内容和结构信息实现检测；

BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)

1.对系统造成影响，要通过系统恢复模块会复发到运行恶意代码之前的状态。图2-2 基于行为的基本原理图  
2.2.7 **基于语义的检测方法**基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，通过抽象程序指令的行为，建立其行为模型，使得该模型既描述恶意程序的基本行为，又要具有很强的泛化能力。这样因其具有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。M. Christodorescu[11,12]提出了一套抽象理论和语义框架，使用自动机描述程序的行为，通过抽象理论描述程序的行为建立抽象模式库，并将其作为自动机的符号表，最终经恶意行为描述为自动机表述的模板，最后通过模型检测方法检测样本是否含有恶意行为。模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，并通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成**恶意代码的检测**。从图2-3中可以看出，首先，要根据语义分析的理论分析程序，并提取行为特征建立语义模型最终形成一个语义模板库。其次对于要检

2.立，这还需要后人不断的努力探索。图2-3 基于语义的恶意代码检测的模型2.2.8**基于机器学习和数据挖掘的检测方法**随着检测技术的不断发展，数据挖掘和机器学习的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及聚类等多种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。

其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定要提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下图详细介绍了现在应用机器学习实现恶意代码的几种方法。从2-4图中可以看出，现在已经有很多机器学习技术应用到了恶意

3.会给人类的生活带来诸多不便，而且也会使用户及企业蒙受巨大的经济损失，有些甚至能危害到国家信息的安全。随着恶意代码的检测技术和反检测技术的不断对抗发展，每天产生的大量新的恶意代码给分析人员带来巨大的压力和严峻的挑战。传统的恶意代码检测能力已经远远不能满足需求。恶意代码检测技术分为静态和动态检测。静

BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)

1.，则会对系统造成影响，要通过系统恢复模块会恢复到运行恶意代码之前的状态。图2-2 基于行为的基本原理图（7）基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，通过抽象程序指令的行为，建立其行为模型，使得该模型既描述恶意程序的基本行为，又要具有很强的泛化能力。这样因其有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。M. Christodorescu[11,12]提出了一套抽象理论和语义框架，使用自动机描述程序的行为，通过抽象理论描述程序的行为建立抽象模式库，并将其作为自动机的符号表，最终经恶意行为描述为自动机表述的模板，最后通过模型检测方法检测样本是否含有恶意行为。模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，并通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成恶意代码的检测。从图2-3中可以看出，首先，要根据语义分析的理论分析程序，并提取行为特征建立语义模型最终形成一个语义模板库。其次对于要检

2.型建立，这还需要后人不断的努力探索。图2-3 基于语义的恶意代码检测的模型（8）基于机器学习和数据挖掘的检测方法。随着检测技术的不断发展，机器学习和数据挖掘的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及

聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定要提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下图详细介绍了现在应用机器学习实现恶意代码的几种方法。从2-4图中可以看出，现在已经有机器学习技术应用到了恶意

3.会给人类的生活带来诸多不便，而且也会使用户及企业蒙受巨大的经济损失，有些甚至能危害到国家信息的安全。随着恶意代码的检测技术和反检测技术的不断对抗发展，每天产生的大量新的恶意代码给分析人员带来巨大的压力和严峻的挑战。传统的恶意代码检测能力已经远远不能满足需求。恶意代码检测技术分为静态和动态检测。静

10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 (是否引证：否)

1.，则会对系统造成影响，要通过系统恢复模块会复发到运行恶意代码之前的状态。图2-2 基于行为的基本原理图（7）基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，通过抽象程序指令的行为，建立其行为模型，使得该模型既描述恶意程序的基本行为，又要具有很强的泛化能力。这样因其有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。M. Christodorescu[11,12]提出了一套抽象理论和语义框架，使用自动机描述程序的行为，通过抽象理论描述程序的行为建立抽象模式库，并将其作为自动机的符号表，最终经恶意行为描述为自动机表述的模板，最后通过模型检测方法检测样本是否含有恶意行为。模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，并通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成恶意代码的检测。从图2-3中可以看出，首先，要根据语义分析的理论分析程序，并提取行为特征建立语义模型最终形成一个语义模板库。其次对于要检

2.型建立，这还需要后人不断的努力探索。图2-3 基于语义的恶意代码检测的模型（8）基于机器学习和数据



		<p>挖掘的检测方法。随着检测技术的不断发展，机器学习和数据挖掘的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定要提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下图详细介绍了现在应用机器学习实现恶意代码的几种方法。从2-4图中可以看出，现在已经有机器学习技术应用到了恶意</p> <p>3.会给人类的生活带来诸多不便，而且也会使用户及企业蒙受巨大的经济损失，有些甚至能危害到国家信息的安全。随着恶意代码的检测技术和反检测技术的不断对抗发展，每天产生的大量新的恶意代码给分析人员带来巨大的压力和严峻的挑战。传统的恶意代码检测能力已经远远不能满足需求。恶意代码检测技术分为静态和动态检测。静</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)</p> <p>1.习和数据挖掘的检测方法机器学习、数据挖掘的方法随着恶意代码检测技术的进步，已经开始在恶意代码检测过程中使用。主要应用分类、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下面详细介绍了现在应用机器学习实现恶意代码检测的几种方法[21-23,25]。首先，在特征表示方面有基于</p> <p>2.增长、技术不断发展恶意代码给人们的生活带来了很大困扰，也导致了个人及企业的经济损失，甚至威胁到国家安全。随着恶意代码的检测技术和反检测技术的不断对抗发展，日益增多的恶意代码给分析人员带来巨大的压力和严峻的挑战。★本文研究了基于深度置信网络的恶意代码检测技术，深度置信网络采用半监督</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p> <p>1.机器学习和数据挖掘的检测方法机器学习、数据挖掘的方法随着恶意代码检测技术的进步，已经开始在恶意代码检测过程中使用。主要应用分类、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分</p>
--	--	---

	<p>类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下面详细介绍了现在应用机器学习实现恶意代码检测的几种方法。首先，在特征的表示方面有基于n-gram算法提取的字节序</p> <p>2.段。数目不断增长、技术不断发展恶意代码给人们的生活带来了很大困扰，也导致了个人及企业的经济损失，甚至威胁到国家安全。随着<b>恶意代码的检测技术和反检测技术的不断对抗发展</b>，每天产生大量新的恶意代码给分析人员带来巨大的压力和严峻的挑战。本文研究了基于深度置信网络的恶意代码检测技术，深度置</p>
	<p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.机器学习和数据挖掘的检测方法机器学习、数据挖掘的方法随着恶意代码检测技术的进步，已经开始在恶意代码检测过程中使用。<b>主要应用分类、序列模式分析以及聚类</b>等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。下面详细介绍了现在应用机器学习实现恶意代码检测的几种方法。首先，在特征的表示方面有基于n-gram算法提取的字节序</p> <p>2.段。数目不断增长、技术不断发展恶意代码给人们的生活带来了很大困扰，也导致了个人及企业的经济损失，甚至威胁到国家安全。随着<b>恶意代码的检测技术和反检测技术的不断对抗发展</b>，每天产生大量新的恶意代码给分析人员带来巨大的压力和严峻的挑战。本文研究了基于深度置信网络的恶意代码检测技术，深度置</p>
	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 -《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.程中特定“模式”，利用这些“模式”，再结合动态分析，可以提高检测效率，取得较好效果。4<b>恶意代码反检测技术当前出现了各种各样的恶意代码反检测技术，现总结如下。代码迷惑技术(对抗静态分析)</b> 恶意代码迷惑技术是指通过某种程序代码变换，恶意代码P传播到目标后</p>
	<p>094616006_冯本慧 冯本慧 -《学术论文联合比对库》- 2013-09-23 (是否引证：否)</p> <p>1.性的检测方法。各种不同检测技术有各自的优点，也有各自的局限，因此往往将各种检测技术进行结合，取长补短，提高恶意代码<b>检测的精度。(4) 基于语义的</b></p>

		<p><b>检测技术基于语义的检测</b>技术是通过静态分析或动态分析（如动态污点分析抽象恶意代码的静态或动态行为语义，从恶意代码的逻辑关系出发利用形式化</p>
18	<p>此处有 39 字相似</p> <p>术的进步也带动了反检测技术的发展，当前出现了各种各样的恶意代码反检测技术，现总结如下：</p> <p>（1）加壳技术</p> <p>恶意代码作者</p> <p>为了防止自己的程序被检测软件发现，利用一些软件技术给恶意代码加外壳，可以是利用</p> <p>算法将自己伪装成正常程序，或者利用特殊的算法将自己压缩或加密，使得检测软件很难检测。但是这些“壳”都有一个特点就是，他们</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01（是否引证：否）</p> <p>1.行的进程是分离的。因为并不是所有的随机值都可以使攻击正常发生。加壳技术 病毒编写者<b>为了防止自己病毒被杀毒软件截获[135]</b>，利用一些软件技术给病毒加外壳，可以利用特殊的压缩算法将病毒进行压缩或加密，也可以利用算法将自己伪装成驱动程序等。但是这些“壳”都有一个特性就是他们</p> <p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01（是否引证：否）</p> <p>1.法来逃避检测[3]。由于编写新的恶意代码需要编写者有很高的技术，所以现实中涌现了很多恶意代码的编写工具。人们<b>为了追求更多的利益，利用混淆技术使得恶意代码爆炸式增长</b>。每天产生大量新的恶意代码给分析人员带来庞大的工作量和巨大的压力。传统的恶意代码检测能力已经远远不</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04（是否引证：否）</p> <p>1.新分配等方法来逃避检测[3]。由于编写新的恶意代码需要编写者有很高的技术，所以现实中涌现了很多恶意代码的编写工具。人们<b>为了追求更多的利益，利用混淆技术使得恶意代码成爆炸式增长</b>。每天产生的大量新的恶意代码给分析人员带来庞大的工作量和巨大的压力。传统的恶意代码检测能力已经远远不能满足反</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11（是否引证：否）</p> <p>1.新分配等方法来逃避检测[3]。由于编写新的恶意代码需要编写者有很高的技术，所以现实中涌现了很多恶意代码的编写工具。人们<b>为了追求更多的利益，利用混淆技术使得恶意代码成爆炸式增长</b>。每天产生大量新的恶意代码给分析人员带来庞大的工作量和巨大的压力。传统的恶意代码检测能力已经远远不能满足反恶</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12（是否引证：否）</p> <p>1.新分配等方法来逃避检测[3]。由于编写新的恶意代码需要编写者有很高的技术，所以现实中涌现了很多恶意代码的编写工具。人们<b>为了追求更多的利益，利用混淆技术使得恶意代码成爆炸式增长</b>。每天产生大量新的恶意代码给分析人员带来庞大的工作量和巨大的压力。传统的恶意代码检测能力已经远远不能满足反恶</p>
19	<p>此处有 30 字相似</p> <p>软件技术给恶意代码加外壳，可以是利用算法将自己伪装成正常程序，或者利用特殊的算法将自己压缩或加密，使得检测软件很难检测。</p> <p>但是这些“壳”都有一个特点就是，他们先于程序获得执</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01（是否引证：否）</p> <p>1.给病毒加外壳，可以利用特殊的压缩算法将病毒进行压缩或加密，也可以利用算法将自己伪装成驱动程序等。但是这些“壳”都有一个特性就是他们先于原始程序运</p>

	<p>行控制权，然后把伪装后的程序还原，再把执行权交还给原始代码：是一类自修改代码。</p> <p>(2) 反虚拟执行技术</p> <p>不可否认，虚拟执行的系统</p>	<p>行，接着壳会把压缩、加密后的代码还原成原始程序代码，然后再把执行权交还给原始代码;是一类自修改代码 (self-m</p>
20	<p>此处有 181 字相似</p> <p>，再把执行权交还给原始代码：是一类自修改代码。</p> <p>(2) 反虚拟执行技术</p> <p>不可否认，虚拟执行的系统和真实系统或多或少存在差异[41]。比如，硬件上，调试器总是会设置硬件断点，而虚拟机总是在模拟硬件，这和真实的硬件是有差别的；执行环境，内核地址空间，对于虚拟机和真正的机器是不同的，还有调试器必须挂靠某些进程来插桩进程用于监控；应用程序，虚拟机和调试器都有外部应用程序，对进程可见，用于检查运行环境。一些指令在虚拟机环境中，执行时间总是远远长于真实环境，一个经常执行此类指令的程序能够指示它在虚拟机环境中运行。</p> <p>(3) 代码迷惑技术</p> <p>恶意代码迷惑技术是指通过某种程序代码变换，改变自身在空间和时间上的结</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1. 我们利用这个最基本的假设，虚拟执行的系统和实际执行的系统，总是会存在差异[36]。1)硬件上的差异。例如，调试器总是会设置硬件断点，而虚拟机总是在模拟硬件，这和真正的硬件是不同的。2)执行环境。例如，内核地址空间，对于虚拟机和真正机器是不同的。另外，调试器必须挂靠某些进程或者有时候的内核地址空间，来插桩进程用于监视。3)应用程序。虚拟机和调试器都有外部应用程序，对进</p> <p>2.核地址空间，对于虚拟机和真正机器是不同的。另外，调试器必须挂靠某些进程或者有时候的内核地址空间，来插桩进程用于监视。3)应用程序。虚拟机和调试器都有外部应用程序，对进程可见，用于检查运行环境。4)行为。例如，当单步执行一个进程时，两个事件点之间的间隔将会戏剧性的增加，相比较于一个自然的执行。一些指</p> <p>3.行为。例如，当单步执行一个进程时，两个事件点之间的间隔将会戏剧性的增加，相比较于一个自然的执行。一些指令在虚拟机中，执行时间要远远长于正常的环境中，一个经常执行此类指令的程序能够告诉我们，它是在虚拟机环境中执行。2统计机器学习相关技术简介</p> <p>2.1核心思想 机</p>
21	<p>此处有 35 字相似</p> <p>码迷惑技术</p> <p>恶意代码迷惑技术是指通过某种程序代码变换，改变自身在空间和时间上的结构，但是完成相同的逻辑功能。恶意代码在</p> <p>进行迷惑处理之后，使得逆向工程分析变得难以进行。迷惑技术本身是一种保护</p> <p>软件的手段，但是常常被用来对抗分析和检测。恶意代码的迷惑技术可以有效的对抗恶意代码的静态分析技术和动态反汇编技术。目前主</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.上演化成新的恶意代码P'，P和P'完成相同的逻辑功能[132，33]。恶意代码经过模糊变换处理后，使得代码难以进行上述的逆向工程分析。模糊变换本身是一种软件保护的手段，但也常常成为恶意代码对抗分析和检测的有效手段。恶意代码迷惑技术可以有效地对抗恶意代码的静态分析技</p>
22	<p>此处有 29 字相似</p> <p>变得难以进行。迷惑技术本身是一种保护软件的手段，但是常常被用来对抗分析和检测。恶意代码的迷惑技术可以有效的对抗恶意代码的</p> <p>静态分析技术和动态反汇编技术。目前主要有基于加密的迷惑技术</p> <p>和基于代码变换的迷惑术。其中代码变换主要指在源程序中，利用等价指令替换、指令位置交换、添加新指令</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》- 2010-05-01 (是否引证：否)</p> <p>1.是一种软件保护的手段，但也常常成为恶意代码对抗分析和检测的有效手段。恶意代码迷惑技术可以有效地对抗恶意代码的静态分析技术和动态反汇编技术。目前基于模糊策略的隐蔽技术主要有两种实现方式:基于加密的迷惑技术的和基于代码变换的迷惑术。其中代码变换主要指的是从一个源程</p>



	等手段改变程序形式，但逻辑功能保持不变。	
23	<p>此处有 110 字相似</p> <p>令替换、指令位置交换、添加新指令等手段改变程序形式，但逻辑功能保持不变。</p> <p>恶意代码分析是确定恶意代码意图的过程，是实行恶意代码检测的必要前提。恶意代码分析的直接结果是用于实现恶意行为建模的元数据信息，如指令流、API调用序列等，为后续恶意代码的检测工作提供必要的支持。</p> <p>恶意代码的分析技术一般可分为静态分析和动态分析。</p> <p>(1) 静态分析技术</p> <p>静态分析技术是指对被测软件的源程序或者二进制码进行扫描，从语法、语义的层面去理解程序的行为，以期望获取程序在运行</p>	<p>恶意代码行为挖掘关键技术研究 解培岱 - 《国防科学技术大学博士论文》 - 2013-10-01 (是否引证：否)</p> <p>1.一种恶意行为模型。2.5恶意代码分析方法恶意代码分析是确定恶意代码的功能和目的的过程,是开发有效的恶意代码去除工具、实现<b>恶意代码检测的必要前提</b> [17,82]。恶意代码分析的直接结果是用于实现恶意行为建模的元数据信息,如指令流、API调用序列等,为后续对恶意代码行为的分析和实现恶意代码检测提供必要支持。恶意代码分析技术一般可分为静态分析和动态分析。静态分析在不经实际执行被分析恶意代码的情况下实现对样本的分析,抽取其程序属性;而动态分析则是在可控分析环境中实际执行本分析样本</p> <p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》 - 2010-05-01 (是否引证：否)</p> <p>1.进行分析。整个系统分为恶意代码特征提取流程和待检测迷惑代码分析流程两个过程。其中恶意代码归纳流程用于提取每族<b>恶意代码的特征</b>。<b>待检测恶意代码通过分析流程，与恶意代码知识库中特征进行匹配</b>，对检测结果加权投票后判决该恶意代码家族或判定为非恶意代码。其中恶意代码特征能够提取</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)</p> <p>1.弥补了动态检测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 <b>恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因</b>。目前已有很多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于启发式等等。</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)</p> <p>1.弥补了动态检测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 <b>恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因</b>。目前已有很多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于启发式等等。</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)</p> <p>1.测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 <b>恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因</b>。目前已有很多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于</p>
24	<p>此处有 30 字相似</p> <p>恶意代码的分析技术一般可分为静态分析和动态分析。</p> <p>(1) 静态分析技术</p>	<p>基于污点跟踪的固件漏洞定位研究 戴忠华;费永康;赵波;王婷 - 《山东大学学报(理学版)》 - 2016-04-18 1 (是否引证：否)</p> <p>1.7]的分析方法目前主要有静态分析和动态分析两种手</p>

	<p>静态分析技术是指对被测软件的源程序或者二进制码进行扫描，从语法、语义的层面去理解程序的行为，以期期望获取程序</p> <p>在运行过程中的信息，而不需要运行程序。</p> <p>要进行恶意代码的静态分析，首先需要对恶意程序进行反汇编，常用的反汇编工具有：W</p>	<p>段[8]。静态分析技术一般不实际执行代码,在静态环境下对源程序或目标程序进行分析,从语法、语义层面理解程序的行为,直接分析程序代码的特性[6]。静态分析技术可以比较全面地获得程序的各类信息,如函数调用信息、语义信息、逻辑关系等。动态分析的分析对象</p>
25	<p>此处有 141 字相似</p> <p>望获取程序在运行过程中的信息，而不需要运行程序。</p> <p>要进行恶意代码的静态分析，首先需要对恶意程序进行反汇编，常用的反汇编</p> <p>工具有：W32DASM、objdump、PEid、HIEW、IDA Pro等。</p> <p>静态分析技术由于不会运行程序，因此不会对计算机系统造成任何伤害，其分析效率相对动态分析而言较高，同时由于静态分析技术从程序本身入手，因此可以获得程序的全部信息，分析结果较为全面。但是由于静态分析技术的</p> <p>前提条件是对程序进行正确的反汇编，现如今很多恶意代码编写者常常会对恶意代码进行加壳、加密或者压缩使得恶意程序很难被正确的</p>	<p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 - 《中南大学硕士论文》 - 2013-09-01 (是否引证：否)</p> <p>1.行反汇编或者反编译，不过，一般而言反编译是很困难的，所以静态分析基本上都是以反汇编的结果为基础。常用的反汇编工具有：W32DASM、PEid、HIEW、IDAPro等。静态分析技术由于无须运行恶意代码，因此不会对操作系统造成任何损害，其分析效率相对动态分析技术而言相对较高，同时由于静态分析技术是从恶意代码程序本身入手的，因此可以获得恶意代码的全部信息，分析结果相对全面。不过由于反汇编是静态分析的基础，因此静态分析技术效果的好坏完全取决于反汇编的结果正确与否，所以很多恶意代码为了干扰静态分析的过程，常常会对程序</p> <p>094616006_冯本慧 冯本慧 - 《学术论文联合比对库》 - 2013-09-23 (是否引证：否)</p> <p>1.先必须对其进行反汇编或者反编译，不过，一般而言反编译是很困难的，所以静态分析基本上都是以反汇编的结果为基础。常用的反汇编工具有：W32DASM、PEid、HIEW、IDA Pro等。静态分析技术由于无须运行恶意代码，因此不会对操作系统造成任何损害，其分析效率相对动态分析技术而言相对较高，同时由于静态分析技术是从恶意代码程序本身入手的，因此可以获得恶意代码的全部信息，分析结果相对全面。不过由于反汇编是静态分析的基础，因此静态分析技术效果的好坏完全取决于反汇编的结果正确与否，所以很多恶意代码为了干扰静态分析的过程，常常会对程序进行加壳、</p>
26	<p>此处有 144 字相似</p> <p>失效。</p> <p>(2) 动态分析技术</p> <p>动态分析技术是指在可控环境下实际运行程序，监控执行过程中的程序行为，记录程序执行的信息。</p> <p>由于动态分析需要先运行程序，所以为了防止恶意代码对当前环境的破坏，系统在普或者恶意代码相关信息之后，会自动恢复到恶意代码执行前的最初状态，防止影响下一次的分析结果，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有效地解决静态分析中譬如加壳、加密的干扰。</p> <p>当前最流行的动态分析技术</p> <p>是动态污点分析技术[38]，它的基本原理是将一切不信任的外部数据标记为污点，然后跟踪标记为污点的数据的传播情况，并记录相</p>	<p>094616006_冯本慧 冯本慧 - 《学术论文联合比对库》 - 2013-09-23 (是否引证：否)</p> <p>1.捕获恶意代码的与系统交互的相关行为信息，如：关键系统调用序列，对进程、磁盘、注册表、网络的操作等等，以此来检测恶意代码，由于动态分析先执行恶意代码，然后捕获行为，所以为了防止恶意代码对当前实验环境的破坏，系统在捕获到恶意代码的相关行为后，会自动恢复到恶意代码执行前的最初状态，防止影响下一次的分析结果，因此动态分析技术相比静态分析而言开销较大，这也是动态分析的一个缺陷，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有效地解决静态</p> <p>2.初始状态，防止影响下一次的分析结果，因此动态分析技术相比静态分析而言开销较大，这也是动态分析的一个缺陷，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有效地解决静态分析中譬如加壳、加密等问题的干扰。动态分析技术中比较有代表性的动态分析系统由Willems[]的CWSandbox系统和Bayer的</p>

		<p>TTAnalyze[]系统，前者是</p> <p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 - 《中南大学硕士论文》 - 2013-09-01 (是否引证：否)</p> <p>1.磁盘、注册表、网络的操作等等，以此来检测恶意代码，由于动态分析先执行恶意代码,然后捕获行为，所以为了防止恶意代码对当前实验环境的破坏，系统在捕获到恶意代码的相关行为后，会自动恢复到恶意代码执行前的最初始状态，防止影响下一次的分析结果，因此动态分析技术相比静态分析而言开销较大，这也是动态分析的一个缺陷，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有</p> <p>2.态，防止影响下一次的分析结果，因此动态分析技术相比静态分析而言开销较大，这也是动态分析的一个缺陷，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有效的解决静态分析中譬如加壳、加密等问题的干扰。动态分析技术中比较有代表性的动态分析系统由Wmems[37]的CWSandbox系统和Bayer的TTA3nalyze[8]系</p>
<p>27</p>	<p>此处有 375 字相似</p> <p>态分析技术能够获得恶意代码执行时的真实信息，可以有效地解决静态分析中譬如加壳、加密的干扰。当前最流行的动态分析技术是</p> <p>动态污点分析技术[38]，它的基本原理是将一切不信任的外部数据标记为污点，然后跟踪标记为污点的数据的传播情况，并记录相关的系统调用或者指令执行等相关信息，然后以此信息进行检测。动态污点分析能够记录恶意代码更细粒度的精确特征，是当前非常热门的恶意代码检测技术。</p> <p>动态分析技术也存在缺陷，比如开销大，一次只能分析一条路径，恶法应对恶意代码存在多路径的问题，同时由于恶法模拟出一个完全真实的计算机环境，对某些环境敏感的恶意代码无法进行有效的检测，因为恶意代码能够检测到虚拟机或者仿真机存在的情况，从而隐藏自身的真实行为，也无法知道某些恶意代码何时才会触发，动态分析技术也会受到行为层的混淆技术的干扰[39]如等价行为替换、模拟序列或者混淆序列等。于是有学者开始尝试静态分析与动态分析相结合[40]的方式进行恶意代码的检测，充分利用两种分析技术的优点。</p> <p>两种分析技术各有优缺点，静态分析技术开销小，关注的是恶意程序本身的语法或者结构特征，动态分析技术开销大，关注的是恶意代码</p>	<p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 - 《中南大学硕士论文》 - 2013-09-01 (是否引证：否)</p> <p>1.者则基于仿真器QEMU，他们都是基于监控恶意代码调用系统API的方式分析恶意代码。当前最流行的动态分析技术是动态污点分析技术[39]，它的基本原理是将一切来自外部不信任的数据标记为污点，然后实时跟踪污点数据的传播情况，并记录相关的系统调用或者指令执行等相关信息，然后以此信息进行检测。动态污点分析能够记录恶意代码更细粒度的精确特征，是当前非常热门的恶意代码检测技术。当然动态分析也存在缺陷，如开销大，一次只能分析一条路径无法应对恶意代码存在多路径的问题，同时由于无法模拟出一个完全真实的计算机环境，对某些环境敏感的恶意代码无法检测，因为这类恶意带能够检测到虚拟机或者仿真机的存在从而隐藏自身的恶意行为，也无法知道某些恶意代码何时才会触发，动态分析技术也会受到行为层的混淆技术的干扰[49]如等价行为替换、混淆序列、模拟序列等。于是有学者开始尝试动静结合的混合方式[41]进行恶意代码检测，充分利用两种检测技术的优缺点。总之，两种分析技术各自有各自的优势与缺点，静态分析技术无须运行恶意代码，开销小，关注的是恶意代码程序本身语法或结</p> <p>094616006_冯本慧 冯本慧 - 《学术论文联合比对库》 - 2013-09-23 (是否引证：否)</p> <p>1.e，后者则基于仿真器QEMU，他们都是基于监控恶意代码调用系统API的方式分析恶意代码。当前最流行的动态分析技术是动态污点分析技术[]，它的基本原理是将一切来自外部不信任的数据标记为污点，然后实时跟踪污点数据的传播情况，并记录相关的系统调用或者指令执行等相关信息，然后以此信息进行检测。动态污点分析能够记录恶意代码更细粒度的精确特征，是当前非常热门的恶意代码检测技术。当然动态分析也存在缺陷，如开销大，一次只能分析一条路径无法应对恶意代</p>



		<p>码存在多路径的问题，同时由于无法模拟出一个完全真实的计算机环境，对某些环境敏感的恶意代码无法检测，因为这类恶意带能够检测到虚拟机或者仿真机的存在从而隐藏自身的恶意行为，也无法知道某些恶意代码何时才会触发，动态分析技术也会受到行为层的混淆技术的干扰[如等价行为替换、混淆序列、模拟序列等。于是有学者开始尝试动静结合的混合方式]进行恶意代码检测，充分利用两种检测技术的优缺点。总之，两种分析技术各自有各自的优点与缺点，静态分析技术无须运行恶意代码，开销小，关注的是恶意代码程序的本身语法或结构特征</p>
28	<p>此处有 58 字相似</p> <p>序列等。于是有学者开始尝试静态分析与动态分析相结合[40]的方式进行恶意代码的检测，充分利用两种分析技术的优点。</p> <p>两种</p> <p>分析技术各有优缺点，静态分析技术开销小，关注的是恶意程序本身的语法或者结构特征，动态分析技术开销大，关注的是恶意代码</p> <p>的行为特征，各有侧重点。静态分析技术分析全面，可获得恶意代码的全部信息，但获取特征的方式一般都是无导向的，因此可能包含大</p>	<p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 -《中南大学硕士论文》- 2013-09-01 (是否引证：否)</p> <p>1.充分利用两种检测技术的优缺点。总之，两种分析技术各自有各自的优点与缺点，静态分析技术无须运行恶意代码，开销小，关注的是恶意代码程序的本身语法或结构特征，动态分析技术需要运行恶意代码，开销大，关注的是恶意代码的行为特征，各自的侧重点不同。静态分析技术分析全面，可获得恶意代码的所有信息，但获取的特征</p> <p>094616006_冯本慧 冯本慧 -《学术论文联合比对库》- 2013-09-23 (是否引证：否)</p> <p>1.意代码检测，充分利用两种检测技术的优缺点。总之，两种分析技术各自有各自的优点与缺点，静态分析技术无须运行恶意代码，开销小，关注的是恶意代码程序的本身语法或结构特征，动态分析技术需要运行恶意代码，开销大，关注的是恶意代码的行为特征，各自的侧重点不同。静态分析技术分析全面，可获得恶意代码的所有信息，但获取的特征的方式</p> <p>基于沙箱技术的恶意代码行为自动化检测方法 李志勇 -《华中科技大学硕士论文》- 2015-05-01 (是否引证：否)</p> <p>1.imeLine图 2.1 本文方法的整体结构图2.2 恶意代码静态分析方法 静态分析技术指的是在不执行恶意程序的情况下对其进行分析，是恶意代码分析中用到的基本手段，帮助分析人员对恶意代码有个初步的认识。恶意代码通常是二进</p>
29	<p>此处有 129 字相似</p> <p>种分析技术各有优缺点，静态分析技术开销小，关注的是恶意程序本身的语法或者结构特征，动态分析技术开销大，关注的是恶意代码的</p> <p>行为特征，各有侧重点。静态分析技术分析全面，可获得恶意代码的全部信息，但获取特征的方式一般都是无导向的，因此可能包含大量无用信息，也易受代码迷惑技术的影响，动态分析技术能够获得程序的真实行为信息，但一次只能获得一种行为并且与当前的检测环境相关，信息不够全面。</p> <p>当前应用最为广泛的技术还是静态分析技术。总之，无论是静态分析还是动态分析，都需要借助恶意代码分析技术和监控技术获得恶意代</p>	<p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 -《中南大学硕士论文》- 2013-09-01 (是否引证：否)</p> <p>1.代码，开销小，关注的是恶意代码程序的本身语法或结构特征，动态分析技术需要运行恶意代码，开销大，关注的是恶意代码的行为特征，各自的侧重点不同。静态分析技术分析全面，可获得恶意代码的所有信息，但获取的特征的方式一般都是无导向的，因此可能包含大量的冗余信息，也容易受到代码迷惑技术的影响，动态分析技术能获得恶意代码的真实行为，但一次只能获得一种行为并且与当前的检测环境相关，信息不够全面，动态分析技术最大的问题是受到恶意代码不可判定性的影响。14硕士学位论文2恶意代码相关分析与检测技术</p> <p>094616006_冯本慧 冯本慧 -《学术论文联合比对库》- 2013-09-23 (是否引证：否)</p>



		<p>1.运行恶意代码，开销小，关注的是恶意代码程序的本身语法或结构特征，动态分析技术需要运行恶意代码，开销大，关注的是恶意代码的<b>行为特征</b>，各自的侧重点不同。静态分析技术分析全面，可获得恶意代码的所有信息，但获取的特征的方式一般都是无导向的，因此可能包含大量的冗余信息，也容易受到代码迷惑技术的影响，动态分析技术能获得恶意代码的真实行为，但一次只能获得一种行为并且与当前的检测环境相关，信息不够全面，动态分析技术最大的问题是受到恶意代码不可判定性的影响。2.3 恶意代码的检测技术恶意代码的分析技术侧重的是恶</p>
30	<p>此处有 74 字相似</p> <p>一次只能获得一种行为并且与当前的检测环境相关，信息不够全面。当前应用最为广泛的技术还是静态分析技术。总之，无论是静态分析还是动态分析，都需要借助恶意代码分析技术和监控技术获得恶意代码的基本属性和执行信息，以便深入理解恶意代码的功能，进一步实现恶意代码的检测和抑制。</p> <p>本章首先介绍了恶意代码的定义以及恶意代码的分类，具体介绍了病毒、特洛伊木马、蠕虫等恶意代码的特征和危害。其次，介绍了恶意</p>	<p>恶意代码行为挖掘关键技术研究 解培岱 -《国防科学技术大学博士论文》- 2013-10-01 (是否引证：否)</p> <p>1.分析路径覆盖技术关键技术敏感调用迹指令执行序列系统事件序列??前向/后向切片图2.2恶意代码分析的一般过程无论是静态分析还是动态分析,都需要借助恶意代码分析和监控技术获得恶意代码的基本属性和执行信息,以便深入理解恶意代码的功能,进一步实现恶意代码的检测和抑制。2.5.1基础分析技术(一)程序属性分析对于二进制恶意代码来说,反汇编技术是静态分析技术最主要的分析手段,典型的静态反汇</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1.弥补了动态检测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因。目前已有许多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于启发式等等。近几</p> <p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.弥补了动态检测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因。目前已有许多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于启发式等等。近几</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证：否)</p> <p>1.测分析不全面的不足。这样的特征不仅仅能很好的诠释样本，更能很好的排除混淆技术给分类带来的干扰。2.2 恶意代码检测技术随着恶意代码爆炸性的增长，恶意代码成为导致个人、企业信息泄露最大的原因。目前已有许多成熟的恶意代码检测技术，比如基于签名、基于特征码、基于启发</p>
31	<p>此处有 40 字相似</p> <p>代码分析技术和监控技术获得恶意代码的基本属性和执行信息，以便深入理解恶意代码的功能，进一步实现恶意代码的检测和抑制。</p> <p>本章首先介绍了恶意代码的定义以及恶意代码的分类</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.的基于机器学习的恶意代码检测方法的简单总结-13-哈尔滨工业大学工学硕士学位论文2.3 本章小结本章首先介绍了恶意代码的定义并详细介绍了恶意代码的分类</p>

	<p>，具体介绍了病毒、特洛伊木马、蠕虫等恶意代码的特征和危害。其次，介绍了恶意代码的检测技术，详细阐述了基于特征码的检测技术、基于行为的检测技术、基于启发式的检测技术、基于启发式的</p>	<p>，分别介绍了病毒、木马、蠕虫、后门、Rootkit、间谍软件、广告软件及网页恶意脚本等恶意代码的特征和危害。其次，将每一种分类通过从特征、传播方</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.种方法是人们现阶段研究的热点。图2-4 现有的基于机器学习的恶意代码检测方法的简单总结2.3 本章小结 本章首先介绍了恶意代码的定义并详细介绍了恶意代码的分类，分别介绍了病毒、木马、蠕虫、后门、Rootkit、间谍软件、广告软件及网页恶意脚本等恶意代码的特征和危害。其次，将每一种分类通过从特征、传播方式、存</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.种方法是人们现阶段研究的热点。图2-4 现有的基于机器学习的恶意代码检测方法的简单总结2.3 本章小结 本章首先介绍了恶意代码的定义并详细介绍了恶意代码的分类，分别介绍了病毒、木马、蠕虫、后门、Rootkit、间谍软件、广告软件及网页恶意脚本等恶意代码的特征和危害。其次，将每一种分类通过从特征、传播方式、存</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.种方法是人们现阶段研究的热点。图2-4 现有的基于机器学习的恶意代码检测方法的简单总结2.3 本章小结 本章首先介绍了恶意代码的定义并详细介绍了恶意代码的分类，分别介绍了病毒、木马、蠕虫、后门、Rootkit、间谍软件、广告软件及网页恶意脚本等恶意代码的特征和危害。其次，将每一种分类通过从特征、传播方式、存</p>
32	<p>此处有 81 字相似</p> <p>先介绍了恶意代码的定义以及恶意代码的分类，具体介绍了病毒、特洛伊木马、蠕虫等恶意代码的特征和危害。其次，介绍了恶意代码的检测技术，详细阐述了基于特征码的检测技术、基于行为的检测技术、基于启发式的检测技术、基于语义的检测技术和基于机器学习的检测技术。最后，针对目前主流的恶意代码反检测手段以及分析技术做了详细说明。</p> <p>本章对汇编操作码的特点进行分析，研究了众多的汇编码抽象方式后，改进了一种基于机器学习算</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) 特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检</p> <p>2.一旦检测到所处的环境是虚拟机会执行一些其他的行为来躲避检测。对于这样的恶意代码动态检测技术无能为力。(7) 基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。2.2.1 特征码</p>

		<p>检测技术基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通</p> <p>2.对系统造成影响，要通过系统恢复模块会复发到运行恶意代码之前的状态。图2-2 基于行为的基本原理图</p> <p>2.2.7 基于语义的检测方法基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) 特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通</p> <p>2.，则会对系统造成影响，要通过系统恢复模块会复发到运行恶意代码之前的状态。图2-2 基于行为的基本原理图(7) 基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.静态检测不执行代码，通过代码的内容和结构信息实现检测；动态检测是在虚拟环境中通过代码执行的行为来判断。下面将详细介绍几种检测技术。(1) 特征码检测方法。基于特征码的检测技术，是通过对恶意代码的文本内容进行分析，提取二进制、字符串、字节序列、文件名等特征，将这些特征码存入特征库。当检测样本时，通</p> <p>2.，则会对系统造成影响，要通过系统恢复模块会复发到运行恶意代码之前的状态。图2-2 基于行为的基本原理图(7) 基于语义的检测方法。基于语义的检测技术是现在研究的热点。因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意</p> <p>基于数据挖掘与机器学习的恶意代码检测技术研究 冯本慧 - 《中南大学硕士论文》- 2013-09-01 (是否引证：否)</p> <p>1.为基于网络的检测技术和基于主机的检测检测技术两类，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码检测技术主要有基于特征码的检测、基于完整性的检测、基于启发式和基于语义的检测等方法，但这些检测方法存在过于依赖特征码、过于依赖分析人员的经验、无法检测未知恶意代码等问题，面对</p>
--	--	---

	<p>2.模型，又可以有效的提高分类算法的精度与泛化能力，因此具有很好的研究意义。1.2国内外的研究现状  恶意代码<b>检测技术主要分为基于网络的检测技术和基于主机的检测检测技术两类</b>，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码检测技术主要有基于特征码的检测、基于完整性的检测、</p>
	<p>094616006_冯本慧 冯本慧 -《学术论文联合比对库》- 2013-09-23 (是否引证：否)</p> <p>1.主要分为基于网络的检测技术和基于主机的检测检测技术两类，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码<b>检测技术主要有基于特征码的检测、基于完整性的检测、基于启发式</b>和基于语义的检测等方法，但这些检测方法存在过于依赖特征码、过于依赖分析人员的经验、无法检测未知恶意代码等问题，面对当前数</p> <p>2.分类模型，又可以有效的提高分类算法的精度与泛化能力，因此具有很好的研究意义。1.2 国内外的研究现状  恶意代码<b>检测技术主要分为基于网络的检测技术和基于主机的检测检测技术两类</b>，本文主要关注基于主机的恶意代码检测技术。基于主机的恶意代码检测技术主要有基于特征码的检测、基于完整性的检测、基于启</p>
	<p>恶意代码检测中若干关键技术研究 陈良 -《扬州大学硕士论文》- 2012-04-01 (是否引证：否)</p> <p>1.分析原理的A6 扬州大学硕士学位论文不同对现有的恶意代码检测方法进行分类主要分为基于特征码的检测方法、<b>基于函数特征的检测方法、启发式检测方法、基于行为的检测方法等</b>。1.2.1基于特征码的检测法基于特征码的检测方法[7]是使用最广泛和最古老的方法，这种方法通过密罐系统提取恶</p>
	<p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1、基于启发式等等。近几年利用机器学习和数据挖掘的方法进行恶意代码检测的技术为当前热点。下面阐述几种比较成熟常见的恶意代码<b>检测技术</b>。<b>2.2.1基于签名的检测基于签名的检测技术</b>是一种采用匹配的恶意代码检测方法，对每种恶意软件生成一个标记，然后利用标记构建恶意数据库。对恶意代码生成的签名标记包含多</p>
	<p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1、基于启发式等等。近几年利用机器学习和数据挖掘的方法进行恶意代码检测的技术为当前热点。下面阐述几种比较成熟常见的恶意代码<b>检测技术</b>。<b>2.2.1基于签名的检测基于签名的检测技术</b>是一种采用匹配的恶意代码检测方法，对每种恶意软件生成一个标记，然后利用标记构建恶意数据库。对恶意代码生成的签名标记包含多</p>
	<p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》- 2013-12-01 (是否引证：否)</p> <p>1.启发式等等。近几年利用机器学习和数据挖掘的方法进行恶意代码检测的技术为当前热点。下面阐述几种比较成熟常见的恶意代码<b>检测技术</b>。<b>2.2.1 基于签名的检</b></p>



		测基于签名的检测技术[15-18]是一种采用匹配的恶意代码检测方法，对每种恶意软件生成一个标记，然后利用标记构建恶意数据库。对恶意代码
33	<p>此处有 45 字相似</p> <p>细介绍，其中概率矩阵的生成方法是本章的核心，也是本方法与其他传统方法的主要区别。最后对本文方法进行了实验仿真，证明本方法</p> <p>对于恶意代码的检测具有实际意义。</p> <p>大部分的基于机器学习算法的恶意代码静态检测技术都会使用</p> <p>汇编码进行分类模型构建，汇编操作码有以下特点：</p> <p>第一，汇编语言是面向机器的程序语言，是一种用文字助记符来表示机器指令的</p>	<p>结合语义的统计机器学习方法在代码安全中应用研究 孔德光 - 《中国科学技术大学博士论文》 - 2010-05-01 (是否引证：否)</p> <p>1.序列或者行为序列，从而进一步提升迷惑恶意代码检测的准确率。4.6相关工作比较 对于迷惑恶意代码的静态检测，主要有两种方式，一是基于语义的恶意代码检测技术，利用迷惑代码语义变换前后的等价性和相似性，检测具有一定特征的恶意代码;例如5.Jha 【“习提出一种基于语义模板</p>
34	<p>此处有 37 字相似</p> <p>意义。</p> <p>大部分的基于机器学习算法的恶意代码静态检测技术都会使用汇编码进行分类模型构建，汇编操作码有以下特点：</p> <p>第一，</p> <p>汇编语言是面向机器的程序语言，是一种用文字助记符来表示机器指令的符号语言，</p> <p>它是目前所有编程语言中最接近机器码的一种语言。它的针对性特别强，需要对机器硬件进行精确的控制，所以它的每一条指令都是极致</p>	<p>液压支架电液控制系统的研究与实现 郭科伟 - 《重庆大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.择的是 C8051F 系列的单片机，编程语言既可以使用 C 语言，也可以使用汇编语言，这两种编程方式各有其优缺点；汇编语言是底层编程语言，是一种用文字助记符来表示机器指令的符号语言，是最接近机器码的一种语言，所以其优点就是占用 CPU 资源少，程序执行效率很高，但是当程序复杂时，由于其代码</p>
35	<p>此处有 38 字相似</p> <p>不过可以肯定的是，任何一种指令抽象方式都有其适用范围，那么如何在复杂的互联网环境中保证检测系统始终选择最适合抽象化方式是</p> <p>本章要解决的问题。</p> <p>基于机器学习算法的恶意代码检测方法一般流程如图3-1。</p> <p>首先对样本进行预处理，该阶段包括对样本进行查壳脱壳以及对样本反汇编得到其对应的汇编文本；然后对汇编文本直接使用n-gra</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 (是否引证：否)</p> <p>1.背景；国内外的研究现状；概括本文的主要研究内容；对论文的整体结构进行概述。第二章机器学习与恶意代码特征表示。本章主要对机器学习的概况和恶意代码的检测流程进行简单的描述，并重点对现有的恶意代码检测的特征表示方法，如n元序列、Opcode、系统调用图、系统调用等方法进行</p>
36	<p>此处有 56 字相似</p> <p>析可知，汇编操作码直接应用于恶意代码的检测时会有一定的局限性，所以本文改进了一种恶意代码的检测方法，主要流程如图3-2。</p> <p>首先是样本预处理，该步骤包括对所有的样本进行查壳脱壳处理，并对脱壳后的样本进行反汇编，得到其汇编文本文件；其次，</p> <p>根据汇编文本文件提取汇编操作码，并根据不同的抽象方式对汇编操作码进行抽象，得到对应的中间码序列；然后使用n-gram算法</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)</p> <p>1.检测方法，通过结合程序的控制流程图分析每条可能的执行路径的操作码序列来实现恶意代码的检测。研究方法的步骤是：首先，数据预处理，对恶意代码查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析；其次，按顺序提取基于文本内容的操作码的序列并提取基于控制流程图的操作码序列；然后，分析恶意代码和正常代码的操作码序列</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证：否)</p> <p>1.列的恶意代码检测方法，通过结合程序的控制流程图</p>

		分析每条可能的执行路径的操作码序列来实现恶意代码的检测。研究方法的步骤是：首先，数据预处理，对恶意代码查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析；其次，按顺序提取基于文本内容的操作码的序列并提取基于控制流程图的操作码序列；然后，分析恶意代码和正常代码的操作码序列的特征，根
		10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证：否)
		1.列的恶意代码检测方法，通过结合程序的控制流程图分析每条可能的执行路径的操作码序列来实现恶意代码的检测。研究方法的步骤是：首先，数据预处理，对恶意代码查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析；其次，按顺序提取基于文本内容的操作码的序列并提取基于控制流程图的操作码序列；然后，分析恶意代码和正常代码的操作码序列的特征，根
		BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 (是否引证：否)
		1.列的恶意代码检测方法，通过结合程序的控制流程图分析每条可能的执行路径的操作码序列来实现恶意代码的检测。研究方法的步骤是：首先，数据预处理，对恶意代码查壳和脱壳处理并对脱壳后的恶意代码进行反汇编分析，分析恶意代码和正常代码的操作码序列的特征；其次，按顺序提取基于文本内容的操作码的序列并提取基于控制流程图的操作码序列；然
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证：否)
37	<p>此处有 37 字相似</p> <p>列；然后使用n-gram算法提取样本的中间码特征，接着对各中间码特征序列进行频繁项集的相似度分析，选择相似度最小的中间码特征序列，作为生成概率矩阵的依据；最后，使用机器学习算法完成分类检测。</p> <p>图3-2 本文算法主要流程</p> <p>数据预处理阶段和一般检测算法并没有多大区别，包括恶意代码的查壳与脱壳、反汇编。</p>	1.论基础和系统实现，并将深度学习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证：否)
		1.论基础和系统实现，并将深度学习算法应用恶意代码检测系统之中。本文的主要贡献为以下几点：(1) 本文完成了恶意程序的查壳、脱壳工作，并对脱壳后的恶意代码进行反汇编，分析反汇编后的文本，提取操作码数据集4000。(2)将恶意代码检测与深度置信网络模型相结合，一方面，提高了恶意代码检测的准确性和稳定性
		基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 (是否引证：否)
		1.出了一种新的静态恶意代码检测方法，主要是结合程序的控制流程图提取了恶意代码反汇编后的操作码序列。这种方法结合了基于特征码和行为的共同特点，最终使用机器学习的分类技术实现检测。这样既克服了动态检测的不完整性，又解决了混淆技术带来的困惑。通过对恶意代码的了解，首先，对恶意代码进行脱壳处
		BH2009921397 - 《学术论文联合比对库》 - 2012-12-

	<p>壳是指在</p>	<p>04 (是否引证：否)</p> <p>1.要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制流程图提取了恶意代码反汇编后的操作码序列。这种方法结合了基于<b>特征码和行为的共同特点</b>，<b>最终使用机器学习的分类技术实现检测</b>。这样既克服了动态检测的不完整性，又解决了混淆技术带来的困惑。通过对恶意代码的了解，首先，我们对恶意代码进行脱壳处理，因为</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制流程图提取了恶意代码反汇编后的操作码序列。这种方法结合了基于<b>特征码和行为的共同特点</b>，<b>最终使用机器学习的分类技术实现检测</b>。这样既克服了动态检测的不完整性，又解决了混淆技术带来的困惑。通过对恶意代码的了解，首先，我们对恶意代码进行脱壳处理，因为</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.要是提出了一种新的静态恶意代码检测方法。主要是结合程序的控制流程图提取了恶意代码反汇编后的操作码序列。这种方法结合了基于<b>特征码和行为的共同特点</b>，<b>最终使用机器学习的分类技术实现检测</b>。这样既克服了动态检测的不完整性，又解决了混淆技术带来的困惑。通过对恶意代码的了解，首先，我们对恶意代码进行脱壳处理，因为</p>
38	<p>此处有 60 字相似</p> <p>图3-2 本文算法主要流程</p> <p>数据预处理阶段和一般检测算法并没有多大区别，包括恶意代码的查壳与脱壳、反汇编。</p> <p>壳是指在</p> <p>计算机软件中一段专门保护软件不被非法修改或反编译的程序。它们一般都先于程序运行，拿到控制权，然后完成保护软件的功能。</p> <p>壳通常分为两类：压缩壳和加密壳。压缩壳出现较早，可追溯到DOS时代，使用压缩壳可以帮助缩减PE文件大小，隐藏PE文件内部</p>	<p>浅谈病毒与反病毒 梅净缘; - 《电脑与信息技术》- 2012-04-15 (是否引证：否)</p> <p>1.病毒“加壳”躲避查杀在自然界中,大家对壳这东西应该都不会陌生的,植物用它来保护种子,动物用它来保护身体等等。同样,在一些<b>计算机软件里也有一段专门负责保护软件不被非法修改或反编译的壳</b>。它们一般都是先于<b>程序运行,拿到控制权,然后做保护软件的工作</b>。动植物的壳一般都是在身体外面同样软件的壳也是如此,但后来也出现了所谓的“壳中带籽”的壳。而“加壳”指的是对编译好的EXE</p>
39	<p>此处有 231 字相似</p> <p>的加密壳有：ASProtector、Armadillo、EXECryptor、Themida、VMProtect。</p> <p>现阶段主要的查壳手段有两种：<b>基于特征码和基于信息熵</b>。</p> <p><b>基于特征码</b>：同样的加壳方法会使得加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。</p> <p><b>基于信息熵</b>：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.后重定位跳转到原程序入口点，把控制权转交给原程序。为了更好的分析恶意代码原程序，首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：<b>基于特征码和基于信息熵</b>。<b>基于特征码</b>：同样的加壳方法会使得加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。<b>基于信息熵</b>：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩</p>

<p>变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的指标，这样经过加壳的PE文件的熵一般会大于源PE文件的熵。</p> <p>目前有很多加壳工具，常用的有ASPACK、PEPACK、UPX、PECOMPACT 等。相应的也有一些查壳软件有PEI</p>	<p>会使 PE 文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的 PE 文件的熵一般会大于源文件 PE文件的熵。表 3-1 为本文选取的恶意代码样本加壳与脱壳后的熵值的比较，表中的结果都是通过PEID软件查壳和脱壳并计算其信息</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 ( 是否引证：否 )</p> <p>1.最后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，我们首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得的加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构的形式。然而，熵是用来衡量不确定性的，这样经过加壳的PE文件的熵一般会大于源文件PE文件的熵。表3-1为本文选取的恶意代码样加壳与脱壳后的熵值的比较，表中的结果都是通过PEID软件查壳和脱壳并计算其信息熵获得。从表</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.最后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，我们首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得的加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的PE文件的熵一般会大于源文件PE文件的熵。表3-1为本文选取的恶意代码样加壳与脱壳后的熵值的比较，表中的结果都是通过PEID软件查壳和脱壳并计算其信息熵获得。从表</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.最后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，我们首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得的加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的PE文件的熵一般会大于源文件PE文件的熵。表3-1为本文选取的恶意代码样加壳与脱壳后的熵值的比较，表中的结果都</p>
---	---



		<p>是通过PEID软件查壳和脱壳并计算其信息熵获得。从表</p>
		<p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》 - 2013-11-14 ( 是否引证：否 )</p> <p>1.序，最后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的PE文件的熵一般会大于源文件PE文件的熵。表2-1为本文选取的恶意代码样本加壳与脱壳后的熵值的比较，表中的结果都是通过PEID软件查壳、脱壳并计算其信息熵获得。从</p>
		<p>11S051036-陈晨-丁宇新 陈晨 -《学术论文联合比对库》 - 2013-11-27 ( 是否引证：否 )</p> <p>1.序，最后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的PE文件的熵一般会大于源文件PE文件的熵。表2-1为本文选取的恶意代码样本加壳与脱壳后的熵值的比较，表中的结果都是通过PEID软件查壳、脱壳并计算其信息熵获得。从</p>
		<p>基于操作码行为深度学习的恶意代码检测方法 陈晨 -《哈尔滨工业大学硕士论文》 - 2013-12-01 ( 是否引证：否 )</p> <p>1.后重定位跳转到原程序入口点，把控制权转交给原程序。为了很好的分析恶意代码原程序，首先要做的是查壳与脱壳工作。现阶段的查壳技术主要有两种：基于特征码和基于信息熵。基于特征码：同样的加壳方法会使得加壳后的 PE 文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。基于信息熵：经过加壳的 PE 文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使 PE 文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的，这样经过加壳的 PE 文件的熵一般会大于源文件 PE 文件的熵。表 2-1 为本文选取的恶意代码样本加壳与脱壳后的熵值的比较，表中的结果都是通过 PEID 软件查壳、脱壳并计算其</p>
40	<p>此处有 105 字相似</p> <p>件变成随机性更大的无结构形式。然而，熵是用来衡量</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否</p>

<p>不确定性的指标，这样经过加壳的PE文件的熵一般会大于源PE文件的熵。</p> <p>目前有很多加壳工具，常用的有ASPACK、PEPACK、UPX、PECOMPACT等。相应的也有一些查壳软件有PEID、FILEINFO、FILE SCANNER等，本文主要是用PEID的批量模式查壳。</p> <p>反汇编指将机器代码转换为汇编代码、低级转高级的意思，常用于逆向工程领域。目前网络上许多“免费软件”，PSP、PS、NDS</p>	<p>)</p> <p>1.Viking.dz.exe FSG 7.90 6.25Alcoula.a.exe UPX 7.62 2.59目前有很多加壳工具，常用的有 ASPACK、PEPACK、UPX、PECOMPACT等。相应的也有一些查壳软件有 PEID、FILEINFO、FILE SCANNER 等，本文主要是用 PEID 的批量模式查壳。脱壳与加壳是相反的过程，为了更好的分析恶意程序的真实意图，首先要去掉外面的保护壳，找到原程序的真正入口点。脱壳分为</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 ( 是否引证：否 )</p> <p>1.信息熵大于7，而不加壳的程序一般都小于6.5，但是也有某些特别的样本。表3-1恶意代码加壳和脱壳后熵值的对比目前有很多加壳工具，常用的有ASPACK、PE-PACK、UPX、PECOMPACT等。相应的也有一些查壳软件有PEID、DIE、FILEINFO、FILE SCANNER等，本文主要是用PEID的批量模式查壳。脱壳与加壳是相反的过程，为了更好的分析恶意程序的真实意图，我们首先要去掉外面的保护壳，找到原程序的真正入口点。脱壳分为手</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.信息熵大于7，而不加壳的程序一般都小于6.5，但是也有某些特别的样本。表3-1恶意代码加壳和脱壳后熵值的对比目前有很多加壳工具，常用的有ASPACK、PE-PACK、UPX、PECOMPACT等。相应的也有一些查壳软件有PEID、DIE、FILEINFO、FILE SCANNER等，本文主要是用PEID的批量模式查壳。脱壳与加壳是相反的过程，为了更好的分析恶意程序的真实意图，我们首先要去掉外面的保护壳，找到原程序的真正入口点。脱壳分为手</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.信息熵大于7，而不加壳的程序一般都小于6.5，但是也有某些特别的样本。表3-1恶意代码加壳和脱壳后熵值的对比目前有很多加壳工具，常用的有ASPACK、PE-PACK、UPX、PECOMPACT等。相应的也有一些查壳软件有PEID、DIE、FILEINFO、FILE SCANNER等，本文主要是用PEID的批量模式查壳。脱壳与加壳是相反的过程，为了更好的分析恶意程序的真实意图，我们首先要去掉外面的保护壳，找到原程序的真正入口点。脱壳分为手</p> <p>基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》- 2013-12-01 ( 是否引证：否 )</p> <p>1.熵做了统计，大部分加壳的恶意程序的信息熵大于7，而不加壳的程序一般都小于 6.5，但是也有某些特别的样本。目前常用的加壳工具有 ASPACK、PECOMPACT 等，对应的查壳工具有PEID、FILE SCANNER。本文主要是用 PEID 的批量模式查壳。脱壳与加壳是相反的过程，为了更好的分析恶意程序的真实意图，首先要去掉外面的保护壳，找到原程序的真正入口点。脱壳分为</p>
--	---

		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 ( 是否引证 : 否 )
		1.加壳的程序一般都小于6.5,但是也有某些特别的样本。目前常用的加壳工具有ASPACK、PECOMPACT等,对应的查壳工具有PEID、FILE SCANNER。本文主要是用PEID的批量模式查壳。脱壳与加壳是相反的过程,为了更好的分析恶意程序的真实意图,首先要去掉外面的保护壳,找到原程序的真正入口点。脱壳分为手动和
		11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 ( 是否引证 : 否 )
		1.加壳的程序一般都小于6.5,但是也有某些特别的样本。目前常用的加壳工具有ASPACK、PECOMPACT等,对应的查壳工具有PEID、FILE SCANNER。本文主要是用PEID的批量模式查壳。脱壳与加壳是相反的过程,为了更好的分析恶意程序的真实意图,首先要去掉外面的保护壳,找到原程序的真正入口点。脱壳分为手动和
41	<p>此处有 43 字相似</p> <p>常用于逆向工程领域。目前网络上许多“免费软件”,PSP、PS、NDS游戏机的破解和苹果系统的越狱都跟反汇编息息相关。</p> <p>基本的反汇编算法有两种:线性扫描反汇编和递归下降反汇编。</p> <p>线性扫描反汇编算法从程序的</p> <p>机器指令进行全盘扫描,从程序的第一个指令字节开始,对所有检测到的指令以线性模式逐条反汇编,顺序的把所有字节码转换成指令,</p>	基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证 : 否 )
		1.行反汇编。反汇编是指在对可执行代码的二进制文件进行分析得到汇编代码程序,通过分析汇编代码获得程序行为的方法。基本的反汇编算法有两种:线性扫描反汇编和递归下降反汇编。线性扫描反汇编算法从代码段的第一个字节开始,以线性模式逐条反汇编每条指令。其优点在于:它能够扫描所有代码段。缺点:对数据和代码混合的情况
		BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证 : 否 )
		1.执行代码进行反汇编。反汇编是指在对可执行代码的二进制文件进行分析得到汇编代码程序,通过分析汇编代码获得程序的功能的方法。基本的反汇编算法有两种:线性扫描反汇编和递归下降反汇编。线性扫描反汇编算法采用一种非常直接的方法来确定需要反汇编指令的位置:一条指令结束、另一条指令开始的地方。其优点在于:它能够完全覆盖程序的所
		BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 ( 是否引证 : 否 )
		1.可执行代码进行反汇编。反汇编是指在对可执行代码的二进制文件进行分析得到汇编代码程序,通过分析汇编代码获得程序行为的方法。基本的反汇编算法有两种:线性扫描反汇编和递归下降反汇编。线性扫描反汇编算法从代码段的第一个字节开始,以线性模式逐条反汇编每条指令。其优点在于:它能够扫描所有代码段。缺点:对数据和代码混合的情况没有考虑。
		10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 ( 是否引证 : 否 )
		1.可执行代码进行反汇编。反汇编是指在对可执行代码的二进制文件进行分析得到汇编代码程序,通过分析汇编代码获得程序行为的方法。基本的反汇编算法有两种

		：线性扫描反汇编和递归下降反汇编。线性扫描反汇编算法从代码段的第一个字节开始，以线性模式逐条反汇编每条指令。其优点在于：它能够扫描所有代码段。缺点：对数据和代码混合的情况没有考虑。
42	<p>此处有 50 字相似</p> <p>位置作为一个分岔点，然后顺序反汇编每一个分支，每一个分支中如果又有分岔点，做同样的递归处理，直到遇到程序的终结点。</p> <p>反汇编工具有很多，如有IDA Pro、C32Asm、W32DASM、花指令清除器1.2 等等。本文使用objdump工具对恶意代码样本进行批量反汇编，主要使用objdump -d指令，配合PowerShell脚本完成恶意代</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.效的代码。缺点在于：它无法处理间接代码路径[44]。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm 等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA采用递归下降反汇编算法并结合启发式技术来弥补递归下降</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.接代码路径，如利用指针来查找目标地址的跳转和调用[44]。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA是一种递归下降反汇编器，但是为了克服递归下降的缺点，IDA应用大量的</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.及其无效的代码。缺点在于：它无法处理间接代码路径[44]。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA采用递归下降反汇编算法并结合启发式技术来弥补递归下降算法的不足。它提</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.及其无效的代码。缺点在于：它无法处理间接代码路径[44]。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA采用递归下降反汇编算法并结合启发式技术来弥补递归下降算法的不足。它提</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-14 (是否引证：否)</p> <p>1.能够扫描所有代码段。缺点：对数据和代码混合的情况没有考虑。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA采用递归下降反汇编算法并结合启发式技术来弥补递归下降算法的不足。它提</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》- 2013-11-27 (是否引证：否)</p> <p>1.能够扫描所有代码段。缺点：对数据和代码混合的情况没有考虑。反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有IDA Pro、W32Dasm、C32Asm等。本文使用了强大的反汇编工</p>



	具IDA Pro，简称IDA。IDA采用递归下降反汇编算法并结合启发式技术来弥补递归下降算法的不足。它提
	基于操作码行为深度学习的恶意代码检测方法 陈晨 - 《哈尔滨工业大学硕士论文》 - 2013-12-01 (是否引证：否)
	1.况没有考虑。7哈尔滨工业大学硕士学位论文反汇编技术非常重要，应用也越来越广泛，随之也产生了很多反汇编的工具，主要有 IDA Pro、W32Dasm、C32Asm 等。本文使用了强大的反汇编工具IDA Pro，简称IDA。IDA 采用递归下降反汇编算法并结合启发式技术来弥补递归下降

指 标

疑似剽窃观点

1. 还是动态分析，都需要借助恶意代码分析技术和监控技术获得恶意代码的基本属性和执行信息，以便深入理解恶意代码的功能，进一步实现恶意代码的检测和抑制。

疑似剽窃文字表述

1. 后门可以使攻击者远程控制目标主机，危害无穷。后门提供的通道有几种类型：本地权限提升、远程命令行访问、单命令远程执行、远程控制等。
2. 通常攻击者通过后门获取管理权限，并使用 Rootkit 维持管理权限使的恶意代码能隐藏在目标系统中。Rootkit 分为用户模式和内核模式。用户模式通过通道插入恶意代码、覆盖文件、API 钩子和 DLL 注入等方式达到目的。而内核模式通过安装恶意的设备驱动程序、打补丁、修改内存中运行的内核以及虚拟伪造系统的方式实现。
3. 它是在未经授权的情况下窃取用户的信息并通过网络发送给攻击者的一种恶意代码。这种恶意代码不仅仅能泄露目标主机的数据信息，还可以提供恶意代码的植入接口使得被侵系统受到更加严重的破坏。
4. 它是指在未经用户授权的情况下和别的程序捆绑在一起，以便经常弹出一些用户不想接受的广告。这种恶意程序目的是通过这种强制的方式做商业宣传。一些广告插件的安装会降低主机的性能。广告软件主要的危害是弹出一些色情或者恶意的广告，这会给用户带来很大的困扰。
5. 它是指在网页中嵌入一些用脚本语言编写的有恶意行为的代码。当用户点击带恶意脚本的网站后，脚本通过修改目标系统的注册表、下载病毒或者加载木马程序等方式对被侵系统实施破坏行为。
6. 软件中，一般采用的都是基于“特征码”的检测技术，基本思想是，当新的恶意代码被发现后，对其进行采集取样，分析代码构成，提取有用的特征码，然后将新的特征码加入已有数据库中，用户更新病毒库之后，
7. 特征库去匹配恶意代码，如果匹配成功则进行相应的处理。但是特征码检测技术的缺点是只能对已知恶意代码进行有效的将测，对
8. 恶意代码检测领域提出了启发式检测算法来预防和检测新的恶意代码。根据对恶意代码分析原理的不同对现有恶意代码检测方法进行分类，主要分为基于特征码的检测技术、基于行为的检测技术、基于启发式的检测技术、基于语义的检测方法和基于机器学习算法的检测技术等。
9. Symantec等多有著名病毒检测厂商所使用，是目前已知的所有恶意代码检测方法中最简单、开销最小的方法，广泛用于文件类型的病毒检测中。
10. 匹配，如果匹成功，则判断该程序是恶意的。此技术的关键在于如何选取最能代表恶意程序的特征值。
11. 还有，这种方法使得特征库不断增加，这需要用户经常更新特征库，随着时间的流逝，特征库会越来越庞大，这会影响检测的速度和系统的性能。
12. 当程序运行时，监控其行为，如果发现了异常行为，则立即报警。一般用于检测恶意代码的行为特征
13. 执行时，就要将自身代码附加在感染文件中，可以监控是否有异常写操作。
2) 监控系统调用序列：某些系统调用序列可以体现某种程度的程序语义。系统调用是用户
14. 系统调用接口。
3) 修改内存总量：恶意代码为了完成特定的恶意意图，经常会常驻在内存中，并且不能被覆盖，那么将会减少系统内存的总量，使得该段内存不受系统
15. 例如一些恶意代码都会固定的对一些内核函数进行调用，通常这些调用的顺序是有一定的规律，因此利用对内核函数的名称和调用次数进行分析，可以构建一个恶意代码
16. 启发式方法属于主动防御技术，对未知的恶意代码检测具有明显的效果，因此，这种方法在现如今的商业开发被重点应用。启发式检测可分为静态启发和动态启发。
静态启发方法其实是对传统的特征识别方法的一种扩展，通过分析程序对系统API的调用序列作为特征，有领域专家根据
17. 研究总结出某些恶意代码的行为特征，当对行为进行监控时，此类特征一旦被发现，就立即报警并做相应的处理。这种

- 方法能够有效的检测出已知的恶意代码，并发现部分未知的恶意代码，但在发现恶意代码的时候，系统往往已经被感染。另外，
18. 动态启发式技术主要的工作原理是在计算机系统中划分出一各独立的虚拟环境，当发现可疑程序时，并不立即停止，而是让其继续运行。“沙盒”技术就是动态启发式技术的一种，沙盒会对可疑程序的行为进行记录，直到恶意代码完全暴露后，它在执行回滚操作，使计算机恢复到执行可疑程序之前的状态。
19. 因为混淆技术只是通过插入垃圾指令、改变指令顺序及寄存器重新分配等方法来改变程序，但是程序的基本语义是等价的。通过分析恶意程序，抽象程序指令的行为并建立其行为模型，使得该模型既描述恶意程序的基本行为，又具有很强的泛化能力。这样因其有很强的泛化能力，使得检测恶意程序的变种更加方便快捷。除此之外，也可以检测未知类型的恶意代码。现阶段基于语义的检测方法分为基于内存和函数调用的方法。
20. 模型检测是通过遍历系统所有状态空间，看其中是否有一条符合的路径状态。之后，他还提出迹语义这一概念，将迹语义作为程序的基本语义，并定义了等价的条件，通过抽象解释的方法给出了近似的检测算法。抽象解释理论为解决不可判定和复杂问题的逼近求解提供可很好的构造方法。基于函数调用的方法是将程序中使用的函数提取出来，并结合程序的控制流程图，通过图的同构、线性时态逻辑、计算逻辑树、有穷状态机及下推自动机等方法描述恶意行为，最终通过模型检测完成恶意代码的检测。
21. 随着检测技术的不断发展，机器学习和数据挖掘的方法已经被开始应用在恶意代码检测的领域。主要应用分类、关联规则挖掘、序列模式分析以及聚类等几种技术。主要思想是利用数据挖掘技术从现有的数据中挖掘一些有意义的模式，用机器学习技术归纳出已有样本的特征，然后根据特征的相似性等完成分类的任务。其中，最主要的是选择好的特征和有效的分类器。检测步骤如下：首先，要分析样本确定提取哪种特征或者特征序列；其次，根据特征的特性选择合适的特征选择方法从所有提取的特征中选择一些分类效果好的特征；最后，根据实际情况选择较好的模型实现分类。
- 恶意代码检测与反检测技术总是相互促进，相辅相成。
22. 比如，硬件上，调试器总是会设置硬件断点，而虚拟机总是在模拟硬件，这和真实的硬件是有差别的；执行环境，内核地址空间，对于虚拟机和真正的机器是不同的，还有调试器必须挂靠某些进程来插桩进程用于监控；应用程序，虚拟机和调试器都有外部应用程序，对进程可见，用于检查运行环境。一些指令在虚拟机环境中，执行时间总是远远长于真实环境，一个经常执行此类指令的程序能够
23. 恶意代码检测的必要前提。恶意代码分析的直接结果是用于实现恶意行为建模的元数据信息，如指令流、API调用序列等，为后续恶意代码的检测工作提供必要的支持。
- 恶意代码的分析技术一般可分为静态分析和动态分析。
24. 工具有：W32DASM、objdump、PEid、HIEW、IDA Pro等。
- 静态分析技术由于不会运行程序，因此不会对计算机系统造成任何伤害，其分析效率相对动态分析而言较高，同时由于静态分析技术从程序本身入手，因此可以获得程序的全部信息，分析结果较为全面。但是由于静态分析技术的
25. 由于动态分析需要先运行程序，所以为了防止恶意代码对当前环境的破坏，系统在普或者恶意代码相关信息之后，会自动恢复到恶意代码执行前的最初状态，防止影响下一次的分析结果，但是动态分析技术能够获得恶意代码执行时的真实信息，可以有效地解决静态分析中譬如加壳、加密的干扰。
26. 动态污点分析能够记录恶意代码更细粒度的精确特征，是当前非常热门的恶意代码检测技术。
27. 分析技术各有优缺点，静态分析技术开销小，关注的是恶意程序本身的语法或者结构特征，动态分析技术开销大，关注的是恶意代码
28. 行为特征，各有侧重点。静态分析技术分析全面，可获得恶意代码的全部信息，但获取特征的方式一般都是无导向的，因此可能包含大量无用信息，也易受代码迷惑技术的影响，动态分析技术能够获得程序的真实行为信息，但一次只能获得一种行为并且与当前的检测环境相关，信息不够全面。
29. 本章首先介绍了恶意代码的定义以及恶意代码的分类，具体介绍了病毒、特洛伊木马、蠕虫
30. 检测技术，详细阐述了基于特征码的检测技术、基于行为的检测技术、基于启发式的检测技术、基于语义的检测技术和基于机器学习的检测技术。最后，针对目前主流的恶意代码反检测
31. 对于恶意代码的检测具有实际意义。
- 大部分的基于机器学习算法的恶意代码静态检测技术都会使用
32. 首先是样本预处理，该步骤包括对所有的样本进行查壳脱壳处理，并对脱壳后的样本进行反汇编，得到其汇编文本文件；其次，
33. 计算机软件中一段专门保护软件不被非法修改或反编译的程序。它们一般都先于程序运行，拿到控制权，然后完成保护软件的功能。
34. 现阶段主要的查壳手段有两种：基于特征码和基于信息熵。
- 基于特征码：同样的加壳方法会使得加壳后的PE文件在特定的位置有相同的字节序列。相同的序列即为该加壳方式的特征，然而这种基于特征码的查壳方法，只能检测出已有的并且加入到特征库的加壳方法。
- 基于信息熵：经过加壳的PE文件，其结构会产生变化。壳一般分为加密壳和压缩壳，加密和压缩会使PE文件变成随机性更大的无结构形式。然而，熵是用来衡量不确定性的指标，这样经过加壳的PE文件的熵一般会大于源PE文件的熵。目前有很多加壳工具，常用的有ASPACK、PE PACK、UPX、PECOMPACT等。相应的也有一些查壳软件有PEID、FILEINFO、FILE SCANNER等，本文主要是用PEID的批量模式查壳。

35. 基本的反汇编算法有两种：线性扫描反汇编和递归下降反汇编。  
线性扫描反汇编算法从程序的
36. 汇编工具有很多，如有IDA Pro、C32Asm、W32DASM、花指令清除器1.2 等等。本文使用

16. 80022497643220302\_基于动态手势的智能终端身份识别技术研究\_第16部分 总字数：9907

相似文献列表 文字复制比：4.7%(468) 疑似剽窃观点：(0)

1	10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12	4.0% ( 393 ) 是否引证：否
2	BH2009925453 - 《学术论文联合比对库》 - 2012-12-11	4.0% ( 393 ) 是否引证：否
3	基于操作码序列的静态恶意代码检测方法的研究 卢占军(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2012-12-01	3.4% ( 335 ) 是否引证：否
4	BH2009921397 - 《学术论文联合比对库》 - 2012-12-04	3.2% ( 319 ) 是否引证：否
5	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27	0.8% ( 80 ) 是否引证：否
6	11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14	0.8% ( 80 ) 是否引证：否
7	基于深度表征的网络异常检测模型研究 薛成龙(导师：丁宇新) - 《哈尔滨工业大学硕士论文》 - 2014-12-01	0.6% ( 64 ) 是否引证：否
8	12s051010-薛成龙-丁宇新 薛成龙 - 《学术论文联合比对库》 - 2014-12-05	0.6% ( 64 ) 是否引证：否
9	基于深度表征的入侵异常检测研究_薛成龙 薛成龙 - 《学术论文联合比对库》 - 2014-11-06	0.6% ( 64 ) 是否引证：否
10	基于深度表征的网络异常检测模型研究_基于深度表征的入侵异常检测研究 薛成龙 - 《学术论文联合比对库》 - 2014-11-05	0.6% ( 64 ) 是否引证：否
11	入侵检测中神经网络融合学习方法的研究 吴静(导师：刘衍珩) - 《吉林大学博士论文》 - 2010-06-01	0.4% ( 38 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 75 字相似</p> <p>码表示，最后得到N种中间码序列，N表示配置文件中抽象方式的个数。</p> <p>n-gram是计算机语言学和概率论范畴内的概念，目前n-gram被广泛应用于自然语言的自动分类功能。该模型基于第n个词出现只和其前n-1个词相关的假设，对于处理依靠序列关系分类的问题具有很强的优势。</p> <p>为避免多次读写文件影响系统性能，本文在得到中间码序列之后，继续在内存中使用n-gram算法得到中间码特征序列，并将中间码</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》 - 2012-12-01 ( 是否引证：否 )</p> <p>1.043 0.09864High 97.8883 1037.2673.4.3 n-gram 算法提取特征n-gram模型被广泛应用在自然语言处理，语音识别等领域。该模型基于第n个词出现只与前n-1个词相关的假设，对于处理依靠序列关系分类的问题具有很强的优势。本文应用 n-gram[45]的思想是：将获得的操作码序列，根据大小为 n的滑动窗口，获得一系列的长度为 n 的序</p>
		<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证：否 )</p> <p>1.表示的情况如图3-9所示。图3-9 反汇编后使用中间语言表示的示例3.4.2 n-gram算法提取特征n-gram模型被广泛应用在自然语言处理，语音识别等领域。该模型基于第n个词出现只与前n-1个词相关的假设，对于处理依靠序列关系分类的问题具有很强的优势。在本文中，我们应用n-gram[45] 的思想是：将获得的操作码序列，根据大小为n的滑动窗口，获得一系列的长度为n的序列</p>
		<p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 ( 是否引证：否 )</p> <p>1.成正比的。表3-4所有样本的特征序列文件和源文件的大小比值的平均值3.4.3 n-gram算法提取特征n-gram模型被广泛应用在自然语言处理，语音识别等领域。该模</p>



		<p>型基于第n个词出现只与前n-1个词相关的假设，对于处理依靠序列关系分类的问题有很强的优势。在本文中，我们应用n-gram[45]的思想是：将获得的操作码序列，根据大小为n的滑动窗口，获得一系列的长度为n的序列</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.成正比的。表3-4所有样本的特征序列文件和源文件的大小比值的平均值3.4.3 n-gram算法提取特征n-gram模型被广泛应用在自然语言处理，语音识别等领域。该模型基于第n个词出现只与前n-1个词相关的假设，对于处理依靠序列关系分类的问题有很强的优势。在本文中，我们应用n-gram[45]的思想是：将获得的操作码序列，根据大小为n的滑动窗口，获得一系列的长度为n的序列</p>
2	<p>此处有 39 字相似</p> <p>SVM算法是最初设计是为解决二分类问题的，但是本此实验是一个多分类问题，所以需要SVM算法做一些改动。目前针对SVM</p> <p>多分类问题主要有两种解决方案：直接法和间接法。</p> <p>直接法：即直接修改目标函数，</p> <p>但是这种算法复杂度较高，很难实现，只适合解决规模较小的问题。</p> <p>间接法：通过组合多个二分类器实现多分类器功能，常见的方法</p>	<p>入侵检测中神经网络融合学习方法的研究 吴静 -《吉林大学博士论文》- 2010-06-01 (是否引证：否)</p> <p>1.性使得对入侵检测数据进行简单的二类划分并不精确，需要找到一种有效的多分类划分方法。SVM解决多分类方法通常有两种：直接法和组合法。直接法是在目标函数上加入吉林大学博士学位论文 43 参数求解并合并实现最优超平面的求解；而组合法则是通</p>
3	<p>此处有 35 字相似</p> <p>NN) 算法属于一种无参惰性学习方法。无参类算法不会对数据的分布做任何的假设，而惰性学习方法则不求算法具备显性学习过程。</p> <p>算法的主要思想是：通过对样本进行建模，使得可以用一个向量表示一个样本，</p> <p>然后采用一种度量方式，度量方式一般使用欧式距离或者曼哈顿距离，计算出离被测样本最近的K个已知类别的样本，接着统计这K个样</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.-30-哈尔滨工业大学工学硕士学位论文3.5.2 KNNKNN ( K-Nearest Neighbor ) 算法的主要思想是：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，计算出与被测样本最近的 K 个样本 ( K 一般</p> <p>BH2009921397 -《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.性，将D划分为V个子集，为属性A的信息增益。3.5.2 KNNKNN ( K-Nearest Neighbor ) 算法的主要思想是：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，计算出与被测样本最近的 K个样本 ( K一般选择奇数 )，</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.性，将D划分为V个子集，为属性A的信息增益。3.5.2 KNNKNN ( K-Nearest Neighbor ) 算法的主要思想是：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，计算出与被测样本最近的 K个样本 ( K一般选择奇数 )，</p>



		<p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.性，将D划分为V个子集，为属性A的信息增益。3.5.2 KNNKNN ( K-Nearest Neighbor ) <b>算法的主要思想是：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，计算出与被测样本最近的K个样本 ( K一般选择奇数 )，</b></p>
4	<p>此处有 59 字相似</p> <p>想是：通过对样本进行建模，使得可以用一个向量表示一个样本，然后采用一种度量方式，度量方式一般使用欧式距离或者曼哈顿距离，  <b>计算出离被测样本最近的K个已知类别的样本，接着统计这K个样本的类别次数，则判定被测样本的类别与出现次数最多的类别相同，</b>          为了避免出现平票的情况，K一般选择为奇数。          KNN算法步骤如下：          ( 1 ) 首先对特征向量进行归一化处理，本文采用线性</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )</p> <p>1.预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，<b>计算出与被测样本最近的 K 个样本 ( K一般选择奇数 )，然后统计这 K 个样本中每个类别出现的次数，被测样本的类别与出现次数最多的类别相同。</b>如图 3-13 所示，在实线圆圈内 K=3，测试样本圆点被分类为三角形，而在虚线圆圈内 K=5，测试样本被分类为正</p> <p>BH2009921397 -《学术论文联合比对库》- 2012-12-04 ( 是否引证：否 )</p> <p>1.：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，<b>计算出与被测样本最近的K个样本 ( K一般选择奇数 )，然后统计这K个样本中每个类别出现的次数，被测样本的类别与出现次数最多的类别相同。</b>如图3-12所示，在实线圆圈内K=3，测试样本圆点被分类为三角形，而在虚线圆圈内K=5，测试样本被分类为正方形。图</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，<b>计算出与被测样本最近的K个样本 ( K一般选择奇数 )，然后统计这K个样本中每个类别出现的次数，被测样本的类别与出现次数最多的类别相同。</b>如图3-13所示，在实线圆圈内K=3，测试样本圆点被分类为三角形，而在虚线圆圈内K=5，测试样本被分类为正方形。图</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.：通过对数据预处理，得到每一个样本用一个向量表示，向量的最后一维代表样本的类别。采用一种度量的方法计算样本之间的相似度，<b>计算出与被测样本最近的K个样本 ( K一般选择奇数 )，然后统计这K个样本中每个类别出现的次数，被测样本的类别与出现次数最多的类别相同。</b>如图3-13所示，在实线圆圈内K=3，测试样本圆点被分类为三角形，而在虚线圆圈内K=5，测试样本被分类为正方形。图</p>
5	<p>此处有 40 字相似</p>	<p>BH2009921397 -《学术论文联合比对库》- 2012-12-</p>

	<p>前K个点出现频率最高的类别作为当前点的预测类别，则目标函数为：</p> $f_v = \arg \max_{c \in C_i} 1 - k \delta(c, f(v_i))$ <p>(3-4)</p> <p>其中，函数<math>f_v</math>表示特征向量<math>v</math>的类别，如果<math>f(v_i)</math>的类别和<math>c</math>相同，</p> <p>则<math>\delta(c, f_v) = 1</math>，否则<math>\delta(c, f_v) = 0</math>。</p> <p>本文使用KNN算法来做实验仿真分析，通过实验选择分类较好的K值为5。</p>	<p>04 (是否引证：否)</p> <p>1.方式，本文中我们用的是欧氏距离：(3-9)选择距离最近的K个样本，则目标函数为：(3-10)其中，表示特征向量<math>v</math>的类别，如果的类别值与<math>c</math>类别相同则，否则。</p> <p>K近邻是一种简单的算法，被广泛应用于文本分类，它对于K的值依赖很大，K一般选择奇数。没有一个标准的方法来选择</p> <p>BH2009925453 - 《学术论文联合比对库》- 2012-12-11 (是否引证：否)</p> <p>1.方式，本文中我们用的是欧氏距离：(3-9)选择距离最近的K个样本，则目标函数为：(3-10)其中，表示特征向量<math>v</math>的类别，如果的类别值与<math>c</math>类别相同则，否则。</p> <p>K近邻是一种简单的算法，被广泛应用于文本分类，它对于K的值依赖很大，K一般选择奇数。没有一个标准的方法来选择</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》- 2012-12-12 (是否引证：否)</p> <p>1.方式，本文中我们用的是欧氏距离：(3-9)选择距离最近的K个样本，则目标函数为：(3-10)其中，表示特征向量<math>v</math>的类别，如果的类别值与<math>c</math>类别相同则，否则。</p> <p>K近邻是一种简单的算法，被广泛应用于文本分类，它对于K的值依赖很大，K一般选择奇数。没有一个标准的方法来选择</p>
6	<p>此处有 138 字相似</p> <p>证后，就能够根据n次检验的平均正确率实现对模型的真实评估。本次实验n取值3，k取值为10，即就是3重10折交叉验证。</p> <p>对于恶意代码的检测系统，一般采用准确率 (Accuracy)，精确率 (Precision) 和召回率 (Recall) 来评价其结果。用TP表示恶意样本被正确分类的数量；FN表示恶意样本被判定为正常样本的数量；TN表示正常样本被正确分类的数量；FP表示正常样本被判定为恶意样本的数量。</p> <p>对于多元分类模型，准确率、精确率和召回率的计算公式如下，其中I表示分类个数：</p> $\text{Average Accuracy} = \frac{1}{I} \sum_{i=1}^I \text{Accuracy}_i$	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 - 《哈尔滨工业大学硕士论文》- 2012-12-01 (是否引证：否)</p> <p>1.恶意样本和正常样本分别分成 5 份，然后再组合起来以达到样本划分均匀的目的，最后计算分类结果的平均值。对于恶意代码的检测系统，一般采用准确率 (Accuracy)，误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。用 TP 表示，恶意样本被正确分类的数量；FN 表示恶意样本被分为正常样本的数量；TN 表示正常样本被正确分类的数量；FP 表示正常样本被分为恶意样本的数量。则计算公式如下：Accuracy=TP+TN/TP+FP+TP+TN(4-1)-35-哈尔滨</p> <p>BH2009921397 - 《学术论文联合比对库》- 2012-12-04 (是否引证：否)</p> <p>1.5重交叉验证，并且把恶意样本和正常样本分别分成5份，然后再组合起来以达到样本划分均匀，最后计算了分类结果的平均值。对于恶意代码的检测系统，一般采用准确率 (Accuracy)，误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。我们用TP表示，恶意样本被正确分类的数量；FN表示恶意样本被分为正常样本</p> <p>2.对于恶意代码的检测系统，一般采用准确率 (Accuracy)，误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。我们用TP表示，恶意样本被正确分类的数量；FN表示恶意样本被分为正常样本的数量；TN表示正常样本被正确分类的数量；FP表示正常样本被分为恶意样本的数量。则计算公式如下：(4-1)(4-2)(4-3)4.3</p> <p>实验数据本文所使用的实验数据包括57</p>

		<p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 (是否引证: 否)</p> <p>1.5重交叉验证, 并且把恶意样本和正常样本分别分成5份, 然后再组合起来以达到样本划分均匀, 最后计算了分类结果的平均值。对于恶意代码的检测系统, 一般采用准确率 (Accuracy), 误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。我们用TP表示, 恶意样本被正确分类的数量; FN表示恶意样本被分为正常样本</p> <p>2.对于恶意代码的检测系统, 一般采用准确率 (Accuracy), 误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。我们用TP表示, 恶意样本被正确分类的数量; FN表示恶意样本被分为正常样本的数量; TN表示正常样本被正确分类的数量; FP表示正常样本被分为恶意样本的数量。则计算公式如下: (4-1) (4-2) (4-3)4.3 实验数据本文所使用的实验数据包</p> <p>10S051019-卢占军-丁宇新 卢占军 - 《学术论文联合比对库》 - 2012-12-12 (是否引证: 否)</p> <p>1.5重交叉验证, 并且把恶意样本和正常样本分别分成5份, 然后再组合起来以达到样本划分均匀, 最后计算了分类结果的平均值。对于恶意代码的检测系统, 一般采用准确率 (Accuracy), 误报率 (FPR) 以及漏报率 (FNR) 来评价其效果。我们用TP表示, 恶意样本被正确分类的数量; FN表示恶意样本被分为正常样本的数量; TN表示正常样本被正确分类的数量; FP表示正常样本被分为恶意样本的数量。则计算公式如下: (4-1) (4-2) (4-3)4.3 实验数据本文所使用的实验数据包</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-14 (是否引证: 否)</p> <p>1.2) 评价方法对于恶意代码的检测系统, 一般采用准确率(Accuracy),误报率(FPR)以及漏报率(FNR)来评价其效果。恶意样本被正确分类的数量用TP表示, 恶意样本被分为正常样本的数量用FN表示, 正常样本被正确分类的数量用TN表示, 正常样本被分为恶意样本的数量用FP表示。则计算公式如下: (4-1) (4-2) (4-3) 其中, Accuracy为准确率, FPR为正常</p> <p>11S051036-陈晨-丁宇新 陈晨 - 《学术论文联合比对库》 - 2013-11-27 (是否引证: 否)</p> <p>1.2) 评价方法对于恶意代码的检测系统, 一般采用准确率(Accuracy),误判率(FPR)以及漏报率(FNR)来评价其效果。恶意样本被正确分类的数量用TP表示, 恶意样本被分为正常样本的数量用FN表示, 正常样本被正确分类的数量用TN表示, 正常样本被分为恶意样本的数量用FP表示。则计算公式如下: (4-1) (4-2) (4-3) 其中, Accuracy为准确率, FPR为正常</p> <p>基于深度表征的网络异常检测模型研究_基于深度表征的入</p>
--	--	--

		<p>侵异常检测研究 薛成龙 - 《学术论文联合比对库》 - 2014-11-05 ( 是否引证：否 )</p> <p>1.中，被正确分类的样本(包括正常样本以及异常样本)所占整体样本的比例为准确率 ( Accuracy ) ，用Acc表示；恶意样本被正确分类的数量用TP 表示，恶意样本被分为正常样本的数量用FN 表示，正常样本被正确分类的数量用TN 表示，正常样本被分为恶意样本的数量用FP 表示。则计算公式如下：? ( 4-1 ) ( 4-2 ) ( 4-3 ) ?其中，Accuracy 为准确率，FP</p> <p>基于深度表征的入侵异常检测研究 薛成龙 薛成龙 - 《学术论文联合比对库》 - 2014-11-06 ( 是否引证：否 )</p> <p>1.中，被正确分类的样本(包括正常样本以及异常样本)所占整体样本的比例为准确率 ( Accuracy ) ，用Acc表示；恶意样本被正确分类的数量用TP 表示，恶意样本被分为正常样本的数量用FN 表示，正常样本被正确分类的数量用TN 表示，正常样本被分为恶意样本的数量用FP 表示。则计算公式如下：? ( 4-1 ) ( 4-2 ) ( 4-3 ) ?其中，Accuracy 为准确率，FPR</p> <p>基于深度表征的网络异常检测模型研究 薛成龙 - 《哈尔滨工业大学硕士论文》 - 2014-12-01 ( 是否引证：否 )</p> <p>1.中,被正确分类的样本(包括正常样本以及异常样本)所占整体样本的比例为准确率(Accuracy),用Acc表示;恶意样本被正确分类的数量用TP表示,恶意样本被分为正常样本的数量用FN表示,正常样本被正确分类的数量用TN表示,正常样本被分为恶意样本的数量用FP表示。则计算公式如下:Accuracy=TP TNTP FN TN FP????(4-1)FPFRFP TN??</p> <p>12s051010-薛成龙-丁宇新 薛成龙 - 《学术论文联合比对库》 - 2014-12-05 ( 是否引证：否 )</p> <p>1.中，被正确分类的样本(包括正常样本以及异常样本)所占整体样本的比例为准确率 ( Accuracy ) ，用Acc表示；恶意样本被正确分类的数量用TP 表示，恶意样本被分为正常样本的数量用FN 表示，正常样本被正确分类的数量用TN 表示，正常样本被分为恶意样本的数量用FP 表示。则计算公式如下：? ( 4-1 ) ( 4-2 ) ( 4-3 ) ?其中，Accuracy 为准确率，FP</p>
7	<p>此处有 43 字相似</p> <p>较不符合，除了SVM算法的精确率在Abs2时大于Abs1，随机森林算法和KNN算法都不符合特征分析结果。</p> <p>图3-12</p> <p>n=3时精确率对比图</p> <p>对比图3-13的结果，当n=4时可以得出如下结论：</p> <p>( 1 )</p> <p>平均精确率：KNN&gt;SVM&gt;随机森林算法。</p> <p>( 2 ) 算法稳定性：KNN&gt;随机森林算法&gt; SVM。</p> <p>( 3 ) 是否符合特</p>	<p>BH2009921397 - 《学术论文联合比对库》 - 2012-12-04 ( 是否引证：否 )</p> <p>1.在基于信息增益的特征选择方法特征数为60时，基于文本的SVM算法的误报率低于基于控制流程图的。图4-5 当n=3时误报率的对比图通过对比图4-6中结果，当n=4时我们可以得出如下几点结论：( 1 ) 从图中的曲线可以看出，增加控制流程图的特征提取方法的误报率低于基于文本的方法的误报率，只有个别的点的结果相反，误报率</p> <p>BH2009925453 - 《学术论文联合比对库》 - 2012-12-11 ( 是否引证：否 )</p> <p>1.在基于信息增益的特征选择方法特征数为60时，基于文本的SVM算法的误报率低于基于控制流程图的。图4-5 当n=3时误报率的对比图通过对比图4-6中结果，当n=4时我们可以得出如下几点结论：( 1 ) 从图中的曲线</p>



		<p>可以看出，增加控制流程图的特征提取方法的误报率低于基于文本的方法的误报率，只有个别的点的结果相反，误报率</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.在基于信息增益的特征选择方法特征数为60时，基于文本的SVM算法的误报率低于基于控制流程图的。图4-5 当n=3时误报率的对比图通过对比图4-6中结果，当n=4时我们可以得出如下几点结论：( 1 ) 从图中的曲线可以看出，增加控制流程图的特征提取方法的误报率低于基于文本的方法的误报率，只有个别的点的结果相反，误报率</p>
8	<p>此处有 39 字相似</p> <p>NN&gt; SVM。</p> <p>( 3 ) 是否符合特征分析结果：符合，三种机器学习算法在Abs2时召回率都大于Abs1时的召回率。</p> <p>4. 实验结果分析</p> <p>通过对实验结果的对比分析，可以得出如下结论：</p> <p>( 1 )</p> <p>本文提出的方法相比于传统方法有一定的优越性。在2-gram算法中，本文提出的方法将会选择Abs1对应的中间码特征序列生成</p>	<p>基于操作码序列的静态恶意代码检测方法的研究 卢占军 -《哈尔滨工业大学硕士论文》- 2012-12-01 ( 是否引证：否 )</p> <p>1.有关，同一准确率，误报率低则漏报率就会相对高。- 43-哈尔滨工业大学工学硕士学位论文4.4.4 实验结果分析通过上述实验结果的对比，可以得出如下结论：( 1 ) 结合程序控制流程图的操作码序列提取方法的准确率、误报率及漏报率的效果要好于基于文本的。其中主要的原因是结合程序</p> <p>BH2009925453 -《学术论文联合比对库》- 2012-12-11 ( 是否引证：否 )</p> <p>1.报率低则漏报率就会相对高，另外实验中恶意样本比正常样本相对少一些，可能这也是造成漏报率较高的一个原因。4.4.3 实验结果分析通过上述实验结果的对比，可以得出如下结论：( 1 ) 结合程序控制流程图的操作码序列提取方法的准确率、误报率及漏报率的效果要好于基于文本的操作码序列提取方法。其中主要的原</p> <p>10S051019-卢占军-丁宇新 卢占军 -《学术论文联合比对库》- 2012-12-12 ( 是否引证：否 )</p> <p>1.率较高。整个实验的漏报率相对较高，因为误报率和漏报率有相关性，同一准确率，误报率低则漏报率就会相对高。4.4.3 实验结果分析通过上述实验结果的对比，可以得出如下结论：( 1 ) 结合程序控制流程图的操作码序列提取方法的准确率、误报率及漏报率的效果要好于基于文本的操作码序列提取方法。其中主要的原</p>

指 标		
疑似剽窃文字表述		
<p>1. n-gram被广泛应用于自然语言的自动分类功能。该模型基于第n个词出现只和其前n-1个词相关的假设，对于处理依靠序列关系分类的问题具有很强的优势。</p> <p>2. 计算出离被测样本最近的K个已知类别的样本，接着统计这K个样本的类别次数，则判定被测样本的类别与出现次数最多的类别相同，</p> <p>3. 用TP表示恶意样本被正确分类的数量；FN表示恶意样本被判定为正常样本的数量；TN表示正常样本被正确分类的数量；FP表示正常样本被判定为恶意样本的数量。</p>		
17. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第17部分 总字数：2377		
相似文献列表 文字复制比：0%(0) 疑似剽窃观点：(0)		
18. 80022497643220302_基于动态手势的智能终端身份识别技术研究是实现_第18部分 总字数：1174		

相似文献列表 文字复制比：13.1%(154) 疑似剽窃观点：(0)		
1	恶意代码检测中若干关键技术研究 陈良(导师：李斌;陈斌) - 《扬州大学硕士论文》 - 2012-04-01	13.1% ( 154 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 54 字相似</p> <p>第十四章朱鹏博</p> <p>计算机的普及和高速发展的互联网技术带给人们方便的同时，用户信息的安全性受到的威胁也越来越严重，加强互联网安全</p> <p>已经成为了国家的一项重要战略。在众多的网络安全事件中，尤以恶意代码的危害最大，给个人、企业甚至政府造成了巨大的损失，所以</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 ( 是否引证：否 )</p> <p>1.分类器的性能及其特性。陈良：恶意代码检测中若干关键问题研究 555总结与展望5.1本文工作总结随着计算机的普及和互联网技术的高速发展，用户信息的安全性受到的威胁也越来越严重。信息安全成为人们研究的重要领域之一，恶意代码的产生及其成指数级的增多给许多的包括商业、政府、及普通的用户造成巨大的损失，</p>
2	<p>此处有 100 字相似</p> <p>越严重，加强互联网安全已经成为了国家的一项重要战略。在众多的网络安全事件中，尤以恶意代码的危害最大，给个人、企业甚至政府造成了巨大的损失，所以恶意代码检测技术成为了信息安全领域重要的研究方向。本文主要研究了基于机器学习算法的恶意代码检测技术，介绍了信息安全和机器学习领域的相关理论，分析了恶意代码检测技术的国内外研究现状</p> <p>，在研究分析了大量恶意代码以及相关机器学习算法之后，提出了一种基于机器学习算法的恶意代码检测方法，通过实验证明了本文提出</p>	<p>恶意代码检测中若干关键技术研究 陈良 - 《扬州大学硕士论文》 - 2012-04-01 ( 是否引证：否 )</p> <p>1.严重。信息安全成为人们研究的重要领域之一，恶意代码的产生及其成指数级的增多给许多的包括商业、政府、及普通的用户造成巨大的损失，继而恶意代码检测技术成为信息安全领域重要的研究方向。本文主要研究基于机器学习方法的恶意代码检测技术，介绍了机器学习领域的相关理论，分析了恶意代码检测技术的国内外研究现状，和当前的特征表示方法和特征选择方法，对当前已经存在的特征表示方法进行梳理归类，对当前常用的特征选择方法进行</p>

指 标	
疑似剽窃文字表述	
<p>1. 计算机的普及和高速发展的互联网技术带给人们方便的同时，用户信息的安全性受到的威胁也越来越严重，加强互联网安全</p> <p>2. 造成了巨大的损失，所以恶意代码检测技术成为了信息安全领域重要的研究方向。本文主要研究了基于机器学习算法的恶意代码检测技术，介绍了信息安全和机器学习领域的相关理论，分析了恶意代码检测技术的国内外研究现状</p>	

19. 80022497643220302\_基于动态手势的智能终端身份识别技术研究\_第19部分 总字数：7731

相似文献列表 文字复制比：6.3%(488) 疑似剽窃观点：(0)		
1	基于Intent的SDN北向接口研究综述 张岩;张忠平;张曼君; - 《信息通信技术》 - 2016-02-15	1.2% ( 96 ) 是否引证：否
2	在OpenFlow网络中提供路由服务功能的设计与实现 李鹏飞(导师：王文东) - 《北京邮电大学硕士论文》 - 2013-01-10	1.2% ( 95 ) 是否引证：否
3	专题地图总体设计智能化理论与方法研究 马俊(导师：王光霞) - 《解放军信息工程大学博士论文》 - 2013-04-16	1.1% ( 85 ) 是否引证：否
4	BNF(巴科斯范式) - Hefee的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2017	1.1% ( 85 ) 是否引证：否
5	基于物联网仿真平台的动态可配置网络系统的设计与实现 陈博(导师：张雷) - 《北京邮电大学硕士论文》 - 2014-12-25	0.9% ( 67 ) 是否引证：否
6	盘锦职业技术学院网络性能管理系统的设计与实现 孙伟(导师：卢光辉;王菊) - 《电子科技大学硕士论文》 - 2012-09-01	0.8% ( 61 ) 是否引证：否
7	基于SDN Controller的交换保护研究 安博(导师：窦军) - 《西南交通大学硕士论文》 - 2015-04-20	0.5% ( 39 ) 是否引证：否
8	基于园区网的SDN控制器设计研究	0.4% ( 31 ) 是否引证：否

9	基于SDN的异构无线网络负载均衡研究 汤文强(导师：廖青) - 《北京邮电大学硕士论文》 - 2015-01-15	0.4% ( 31 ) 是否引证：否
10	基于SDN的无线传感器网络流量异常检测机制研究 姬生生(导师：韩志杰) - 《河南大学硕士论文》 - 2015-05-01	0.4% ( 29 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 30 字相似</p> <p>将设备中的控制权放在集中的控制器中进行管理，减轻了网络设备的负担[8]。同时集中式的控制屏蔽了底层的网络设备的差异。而且</p> <p>采用可编程的方式开发控制器，用户可以针对自己的需求对网络进行</p> <p>按需配置，更加灵活和自由。而当SDN与网络虚拟化相集合时，逻辑网络可以根据业务需求进行配置和迁移，不受物理网络的限制[9]</p>	<p>盘锦职业技术学院网络性能管理系统的设计与实现 孙伟 - 《电子科技大学硕士论文》 - 2012-09-01 ( 是否引证：否 )</p> <p>1.监测网络的利用率和出错率及相应的阈值等，进行网络阈值报警，保障网络的可用性，为用户提供较好的网络通信服务。本系统以采用分层设计的方式为主，按照网络的不同功能对网络的情况进行有效的划分，并以此结合有关面向对象程序设计的知识，将本系统分为三大功能模块，具体图 4-4 所示。系统设计部</p>
2	<p>此处有 45 字相似</p> <p>的网络设备的差异。而且采用可编程的方式开发控制器，用户可以针对自己的需求对网络进行按需配置，更加灵活和自由。而当SDN与</p> <p>网络虚拟化相集合时，逻辑网络可以根据业务需求进行配置和迁移，不受物理网络的限制[9]。</p> <p>15.1.1 SDN基础架构</p> <p>SDN的架构方式有很多种,其中ONF的SDN架构图最为经典。如图2-1所示，SDN的典型</p>	<p>基于SDN Controller的交换保护研究 安博 - 《西南交通大学硕士论文》 - 2015-04-20 ( 是否引证：否 )</p> <p>1.化:SDN网络通过统一和开放的南向接口,忽略了底层物理交换设备上的硬件差异,从而实现了对上层应用的透明化。逻辑网络和物理网络分离后,逻辑网络可以根据业务需求进行配置、迁移,不再受物理网络的限制。1.1.3 SDN网络的演变SDN概念在2006年在美国斯坦福大学 GENI项目资助的项目中诞生,斯坦福大学的Nick M</p>
3	<p>此处有 35 字相似</p> <p>实现上层控制器与底层OpenFlow网络设备之间的通信。控制器通过向网络设备发送OpenFlow支持的消息来对网络设备</p> <p>进行控制，网络设备通过Openflow协议将自身的信息反馈到控制器中。</p> <p>从而实现了控制器对于底层设备的掌控。</p> <p>OpenFlow交换机主要由三部分组成：安全通道，交换机通过安全通道与SDN控制</p>	<p>在OpenFlow网络中提供路由服务功能的设计与实现 李鹏飞 - 《北京邮电大学硕士论文》 - 2013-01-10 ( 是否引证：否 )</p> <p>1.ow具有以下不足：1.整个OpenFlow网络的路由表由传统路由协议交互形成，并没有使用控制器对路由表的生成进行控制，没有体现OpenFlow集中控制的优势。</p> <p>2.控制器中需要为每台底层的OpenFlow交换机维护一个对应的虚拟机，对于控制器来说开销太大，对性能产生消极的影响。</p>
4	<p>此处有 67 字相似</p> <p>通过安全通道与SDN控制器建立连接；多级流表，用来匹配交换机收到的报文；组表，用来对流的动作与策略进行定义[11]。</p> <p>OpenFlow协议支持的消息类型一共有三种，分别是 Controller-to-Switch，Asynchronous和 Symmetric[12]。控制器通过这三种类型的消息与交换机进行信息交互，实现对外层交换机的整体控制，其中包括建立连接，消息处理</p>	<p>基于物联网仿真平台的动态可配置网络系统的设计与实现 陈博 - 《北京邮电大学硕士论文》 - 2014-12-25 ( 是否引证：否 )</p> <p>1.送[24]。Openflow协议是SDN中应用最广泛的通信协议，其重要用于控制平面与数据转发平面之间通信[25]，OpenFlow协议支持三种报文类型，分别是 ControUer-to-Switch，Asynchronous 和 Symmetirc,每一种报文类型都包含不止一种子类型。Controller-to-Switch报文是控制器向交换机发起，其目的是为</p> <p>基于SDN Controller的交换保护研究 安博 - 《西南交通大学硕士论文》 - 2015-04-20 ( 是否引证：否 )</p> <p>1.行操作,同时更新Counters。匹配失败则将数据包转发给Controller。2.1.3.3 OpenFlow通信通道OpenFlow协</p>



		议支持三种不同的消息类型:Controller-to-Switch,Asynchronous(异步)和Symmetric(同步),每一类消息又有多个子消息类型。Controller-to-
5	<p>此处有 32 字相似</p> <p>场景 现今SDN的主要应用场景主要集中在各运营商、数据中心,还有一些政企客户和大型的互联网公司。 15.1.3.1 <b>数据中心</b> <b>数据中心中的常见网络需求主要表现在其存在大量的租户、</b> 流的转发路径复杂且多样化、各类虚拟机需要智能迁移和部署、网络需要整体集中化的管理,另外还包含一些绿色工程等方面的工作[1]</p>	<p>基于SDN的异构无线网络负载均衡研究 汤文强 - 《北京邮电大学硕士论文》 - 2015-01-15 (是否引证:否)</p> <p>1.线接入网负载均衡架构3.1 SDN的主要应用场景目前SDN技术主要应用在数据中心网络和数据中心互联中。<b>数据中心网络的需要主要表现在海量的虚拟租户、</b>多路径转发、VM的智能部署和迁移、网络集中自动化管理、绿色节能、数据中心能力开放等方面的需求。SDN控制逻辑</p>
6	<p>此处有 32 字相似</p> <p>发分离,逻辑集中的特点使得网络的控制权可以集中于一点,分布式的进行控制,满足了其对网络的集中化管理的需求。而利用SDN的<b>网络开发能力以及网络虚拟化的特性,可以通过构建虚拟网络的方式进行</b> 智能机的部署,在进行网络迁移时也不用考虑底层的物理结构[16]。 在SDN的部署可行性方面,数据中心的运营和维护具有相</p>	<p>在OpenFlow网络中提供路由服务功能的设计与实现 李鹏飞 - 《北京邮电大学硕士论文》 - 2013-01-10 (是否引证:否)</p> <p>1.信协议主要包括ICMP协议,ARP协议,TCP/UDP协议等。同时,由于L3routing将OpenFlow网络虚拟为一个大的路由器,数据包可以通过这个虚拟出来的路由器进行传输,但是并没有实现OpenFlow网络内部交换机相连的主机与传统主机之间的通信。所以并未真正意义上的实现Open</p>
7	<p>此处有 96 字相似</p> <p>且用一种通用的语言对网络接口进行描述。在此背景下,多个企业组织开始设计与研发一种新的SDN北向应用程序接口[21]。该种<b>应用程序只需要指定业务需求,而无需关注底层的网络细节,这种新的应用程序接口被称为“基于Intent的北向接口模型(Intent NBI)”。</b>在这种应用程序接口模型中,由控制器决定了租户Intent被映射到底层网络设备中的具体方式,使得网络最后呈现为租户期望的状态[22]。租户在利用这种模型时不需要考虑底层的实现</p>	<p>基于Intent的SDN北向接口研究综述 张岩;张忠平;张曼君; - 《信息通信技术》 - 2016-02-15 (是否引证:否)</p> <p>1.的北向接口研究中,研究者希望用一种通用的语言描述网络控制接口,多个组织开始在SDN北向接口推动一种新的应用程序接口,这种<b>应用程序只需要制定通信需求,并且独立于底层网络的各种细节,这个新的应用程序接口称为“基于Intent的北向接口模型(Intent NBI)”。</b>在这个模型中,SDN控制器决定了如何将用户Intent解析到网络基础设施上,使网络表现为用户期望的方式。Intent NBI是一种与网络实现技术无关的北向接口,不需要看到</p>
8	<p>此处有 29 字相似</p> <p>15.3.1 OpenStack上的Congress Congress的面向对象是云平台的管理用户,它对租户提供对<b>网络整体进行管控的策略和服务,使得租户可以利用它对网络设备</b> 的资源进行策略下发和安全性监控等功能,并且以此来保障上层的应用业务被合理自动部署[6]。 在以前进行策略的配置需要进行</p>	<p>基于SDN的无线传感器网络流量异常检测机制研究 姬生生 - 《河南大学硕士论文》 - 2015-05-01 (是否引证:否)</p> <p>1.项目Ethane启发,Casado M和他的导师McKeown N教授,最先使用OpenFlow的概念[22]。出于方便<b>网络管理员管控网络的安全控制策略,并将策略应用于网络设备</b>,设计了集中式的控制器来实现对整个网络通信的管理。紧接着,2008年,McKeown N等人在ACM SIGCOMM发表了</p>



9	<p>此处有 32 字相似</p> <p>EMO模型。</p> <p>2.NEMO引擎。NEMO还提供NEMO引擎以能够控制计算、存储和网络。NEMO引擎主要由四部分组成租户管理模块、语法规义解析模块、虚拟网络构建模块和物理网络管理模块。</p> <p>通过这些模块它可以实现的功能是：</p> <p>(1) 为每个租户创建VN Space，并防止VN Spaces之间的冲突</p> <p>(2) 检</p>	<p>盘锦职业技术学院网络性能管理系统的设计与实现 孙伟 - 《电子科技大学硕士论文》- 2012-09-01 (是否引证：否)</p> <p>1.益的准备。系统的设计与实现主要就是围绕这网络管理的五大功能模块而进行的，在设计中主要划分为性能管理模块、故障管理模块、配置管理模块、安全管理模块及计费管理模块五个子模块，在五个模块中有部分需要重复采集的数据信息或是数据处理过程相似的内容集中到一个公共模块中处理，从而</p> <p>基于园区网的SDN控制器设计研究 桂兴亮;张晓如;程伟; - 《信息技术》- 2017-02-25 (是否引证：否)</p> <p>1.DN控制器模块分析基于园区网的SDN控制器基于模块化设计,其模块结构如图4所示。主要包含公共模块、网络通信模块、物理网络管理模块、虚拟网络管理模块、信令管理模块、应用管理模块六大模块。2.2.1公共模块该模块包含了日志、通用工具、文件操作、线程操作、并发控制、定时器和容器七个子模块</p> <p>在OpenFlow网络中提供路由服务功能的设计与实现 李鹏飞 - 《北京邮电大学硕士论文》- 2013-01-10 (是否引证：否)</p> <p>1.们可以看出，L3routing的控制平面是在现有的OpenFlow控制器上开发了新的路由组件并且将路由组件分为三个模块：路由解析模块、路由选择模块和接口管理模块。其中接口管理模块的主要功能是维护着底层边缘交换机端口的一个列表，在这个列表中有每一个端口在整个Urouting系统</p>
10	<p>此处有 90 字相似</p> <p>,Replica的概念，分别表示server实例，多个node的集合，数据分片和shard备份。</p> <p>15.4.2 BNF 范式</p> <p>巴科斯范式(BNF: Backus-Naur Form 的缩写)是由 John Backus 和 Peter Naur 首次引入一种形式化符号来描述给定语言的语法[29]。</p> <p>它不仅能严格地表示语法规则，而且所描述的语法是与上下文无关的。它具有语法简单，表示明确，便于语法分析和编译的特点。BNF</p>	<p>BNF(巴科斯范式) - Hefee的专栏 - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - (是否引证：否)</p> <p>1.巴科斯范式(BNF: Backus-Naur Form的缩写)是由 John Backus 和 Peter Naur首次引入一种形式化符号来描述给定语言的语法. 式: 示例：用BNF来定义C语言中的for语句：FOR_STATEMENT ::= "for" "(" ( vari</p> <p>专题地图总体设计智能化理论与方法研究 马俊 - 《解放军信息工程大学博士论文》- 2013-04-16 (是否引证：否)</p> <p>1.个对知识表示法兼容性强、融合性好的平台是专题地图设计领域知识表达能达到指标的基础，具有非常重要的意义。巴科斯范式 ( BNF: Backus-Naur Form 的缩写 ) 是由 John Backus 和 Peter Naur 首次引入一种形式化符号来描述给定语言的语法，并且现在几乎每一位新编程语言书籍的作者都使用巴科斯范式来定义编程语言的语法规则。巴克斯范式的规则如下：使用""中的</p>

说明：1.总文字复制比：被检测论文总重合字数在总字数中所占的比例

2.去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例

3.去除本人已发表文献复制比：去除作者本人已发表文献后，计算出来的重合字数在总字数中所占的比例

4.单篇最大文字复制比：被检测文献与所有相似文献比对后，重合字数占总字数的比例最大的那一篇文献的文字复制比

5.指标是由系统根据《学术论文不端行为的界定标准》自动生成的

6.红色文字表示文字复制部分;绿色文字表示引用部分

7.本报告单仅对您所选择比对资源范围内检测结果负责



CNKI科研诚信管理系统研究中心