

<<大数据>>

图书基本信息

书名：<<大数据>>

13位ISBN编号：9787115291318

10位ISBN编号：7115291314

出版时间：2012-9

出版时间：人民邮电出版社

作者：Anand Rajaraman, Jeffrey David Ullman

译者：王斌

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<大数据>>

前言

本书是在Anand Rajaraman和Jeff Ullman于斯坦福大学教授多年的一门季度课程的材料基础上总结而成的。

该课程名为“Web挖掘”(编号CS345A)，尽管它已经成为高年级本科生能接受并感兴趣的课程之一，但其原本是一门为高年级研究生设计的课程。

本书内容简单来说，本书是关于数据挖掘的。

但是，本书主要关注极大规模数据的挖掘，也就是说这些数据大到无法在内存中存放。

由于重点强调数据的规模，所以本书的例子大都来自Web本身或者Web上导出的数据。

另外，本书从算法的角度来看待数据挖掘，即数据挖掘是将算法应用于数据，而不是使用数据来“训练”某种类型的机器学习引擎。

本书的主要内容包括：(1) 分布式文件系统以及已成功应用于大规模数据集并行算法构建的Map-Reduce工具；(2) 相似性搜索，包括最小哈希和局部敏感哈希的关键技术；(3) 数据流处理以及面对快速到达、须立即处理、易丢失的数据的专用处理算法；(4) 搜索引擎技术，包括谷歌的PageRank、链接作弊检测及计算网页导航度(hub)和权威度(authority)的HITS方法；(5) 频繁项集挖掘，包括关联规则挖掘、购物篮分析、A-Priori及其改进算法；(6) 大规模高维数据集的聚类算法；(7) Web应用中的两个关键问题：广告管理及推荐系统。

先修课程尽管从编号CS345A看，本课程属于高年级研究生课程，但是我们发现高年级本科生和低年级硕士生也能接受该课程。

该课程将来可能会分配一个介于高年级研究生和低年级硕士生水平之间的编号。

CS345A的先修课程包括：(1) 数据库系统的首期课程，包括基于SQL及其他数据库相关语言(如XQuery)的应用编程；(2) 大二的数据结构、算法及离散数学课程；(3) 大二的软件系统、软件工程及编程语言课程。

习题本书包含大量的习题，基本每节都有对应习题。

较难的习题或其中较难的部分都用惊叹号“!”

来标记，而最难的习题则标有双惊叹号“!!”

!

来标记。

致谢本书封面由Scott Ullman设计。

感谢Foto Afrati和Arun Marathe精心阅读本书初稿并提出建设性的意见。

感谢Leland Chen、Shrey Gupta、Xie Ke、Haewoon Kwak、Brad Penoff、Philips Kokoh Prasetyo、Mark Storus、Tim Triche Jr.及Roshan Sumbaly指出了本书中的部分错误。

当然，剩余错误均由我们负责。

A. R. J. D. U. 加利福尼亚州帕洛阿尔托2011年6月

<<大数据>>

内容概要

《大数据：互联网大规模数据挖掘与分布式处理》源自作者在斯坦福大学教授多年的“Web挖掘”课程材料，主要关注大数据环境下数据挖掘的实际算法。书中分析了海量数据集数据挖掘常用的算法，介绍了目前Web应用的许多重要话题。主要内容包括：分布式文件系统以及Map-Reduce工具；相似性搜索；数据流处理以及针对易丢失数据等特殊情况的专用处理算法；搜索引擎技术，如谷歌的PageRank；频繁项集挖掘；大规模高维数据集的聚类算法；Web应用中的关键问题：广告管理和推荐系统。

作者简介

Jeffrey David Ullman 斯坦福大学计算机科学系Stanford W. Ascherman教授，数据库技术专家。

他独立或与人合作出版了15本著作，发表了170多篇技术论文。

他的研究兴趣包括数据库理论、数据库集成、数据挖掘和利用信息基础设施进行教育。

他是美国国家工程院成员，曾获得Knuth奖、SIGMOD贡献奖、Karlstrom杰出教育家奖和Edgar F. Codd发明奖。

Anand Rajaraman 数据库和Web技术领域权威,1创业投资基金Cambrian联合创始人,斯坦福大学计算机科学系助理教授。

Rajaraman职业生涯非常成功：1996年创办Jungle公司,两年后该公司被亚马逊以2.

5亿美元收购,Rajaraman被聘为亚马逊技术总监,推动亚马逊从一个零售商转型为零售平台；2000年与人合创Cambrian,孵化出几个后来被谷歌收购的公司；2005年创办Kosmix公司并任CEO,该公司2011年被沃尔玛集团收购。

Rajaraman生于印度,在斯坦福大学获得计算机科学硕士和博士学位。

求学期间与人合著的一篇文章荣列近20年来被引用次数最多的论文之一。

书籍目录

第1章数据挖掘基本概念 1.1数据挖掘的定义 1.1.1统计建模 1.1.2机器学习 1.1.3建模的计算方法 1.1.4数据汇总 1.1.5特征抽取 1.2数据挖掘的统计限制 1.2.1整体情报预警 1.2.2邦弗朗尼原理 1.2.3邦弗朗尼原理的一个例子 1.2.4习题 1.3相关知识 1.3.1词语在文档中的重要性 1.3.2哈希函数 1.3.3索引 1.3.4二级存储器 1.3.5自然对数的底e 1.3.6幂定律 1.3.7习题 1.4本书概要 1.5小结 1.6参考文献 第2章大规模文件系统及Map - Reduce 2.1分布式文件系统 2.1.1计算节点的物理结构 2.1.2大规模文件系统的结构 2.2Map - Reduce 2.2.1Map任务 2.2.2分组和聚合 2.2.3Reduce任务 2.2.4组合器 2.2.5Map - Reduce的执行细节 2.2.6节点失效的处理 2.3使用Map - Reduce的算法 2.3.1基于Map - Reduce的矩阵 - 向量乘法实现 2.3.2向量v无法放入内存时的处理 2.3.3关系代数运算 2.3.4基于Map - Reduce的选择运算 2.3.5基于Map - Reduce的投影运算 2.3.6基于Map - Reduce的并、交和差运算 2.3.7基于Map - Reduce的自然连接运算 2.3.8一般性的连接算法 2.3.9基于Map - Reduce的分组和聚合运算 2.3.10矩阵乘法 2.3.11基于单步Map - Reduce的矩阵乘法 2.3.12习题 2.4Map - Reduce的扩展 2.4.1工作流系统 2.4.2Map - Reduce的递归扩展版本 2.4.3Pregel系统 2.4.4习题 2.5集群计算算法的效率问题 2.5.1集群计算的通信开销模型 2.5.2实耗通信开销 2.5.3多路连接 2.5.4习题 2.6小结 2.7参考文献 第3章相似项发现 3.1近邻搜索的应用 3.1.1集合的Jaccard相似度 3.1.2文档的相似度 3.1.3协同过滤——一个集合相似问题 3.1.4习题 3.2文档的shingling 3.2.1k - Shingle 3.2.2shingle大小的选择 3.2.3对shingle进行哈希 3.2.4基于词的shingle 3.2.5习题 3.3保持相似度的集合摘要表示 3.3.1集合的矩阵表示 3.3.2最小哈希 3.3.3最小哈希及Jaccard相似度 3.3.4最小哈希签名 3.3.5最小哈希签名的计算 3.3.6习题 3.4文档的局部敏感哈希算法 3.4.1面向最小哈希签名的LSH 3.4.2行条化策略的分析 3.4.3上述技术的综合 3.4.4习题 3.5距离测度 3.5.1距离测度的定义 3.5.2欧氏距离 3.5.3Jaccard距离 3.5.4余弦距离 3.5.5编辑距离 3.5.6海明距离 3.5.7习题 3.6局部敏感函数理论 3.6.1局部敏感函数 3.6.2面向Jaccard距离的局部敏感函数族 3.6.3局部敏感函数族的放大处理 3.6.4习题 3.7面向其他距离测度的LSH函数族 3.7.1面向海明距离的LSH函数族 3.7.2随机超平面和余弦距离 3.7.3梗概 3.7.4面向欧氏距离的LSH函数族 3.7.5面向欧氏空间的更多LSH函数族 3.7.6习题 3.8LSH函数的应用 3.8.1实体关联 3.8.2一个实体关联的例子 3.8.3记录匹配的验证 3.8.4指纹匹配 3.8.5适用于指纹匹配的LSH函数族 3.8.6相似新闻报道检测 3.8.7习题 3.9面向高相似度的方法 3.9.1相等项发现 3.9.2集合的字符串表示方法 3.9.3基于长度的过滤 ~ 3.9.4前缀索引 3.9.5位置信息的使用 3.9.6使用位置和长度信息的索引 3.9.7习题 3.10小结 3.11参考文献 第4章数据流挖掘 4.1流数据模型 4.1.1一个数据流管理系统 4.1.2流数据源的例子 4.1.3流查询 4.1.4流处理中的若干问题 4.2流当中的数据抽样 4.2.1一个富于启发性的例子 4.2.2代表性样本的获取 4.2.3一般的抽样问题 4.2.4样本规模的变化 4.2.5习题 4.3流过滤 4.3.1一个例子 4.3.2布隆过滤器 4.3.3布隆过滤方法的分析 4.3.4习题 4.4流中独立元素的数目统计 4.4.1独立元素计数问题 4.4.2FM算法 4.4.3组合估计 4.4.4空间需求 4.4.5习题 4.5矩估计 4.5.1矩定义 4.5.2二阶矩估计的AMS算法 4.5.3AMS算法有效的原因 4.5.4更高阶矩的估计 4.5.5无限流的处理 4.5.6习题 4.6窗口内的计数问题 4.6.1精确计数的开销 4.6.2DGIM算法 4.6.3DGIM算法的存储需求 4.6.4DGIM算法中的查询应答 4.6.5DGIM条件的保持 4.6.6降低错误率 4.6.7窗口内计数问题的扩展 4.6.8习题 4.7衰减窗口 4.7.1最常见元素问题 4.7.2衰减窗口的定义 4.7.3最流行元素的发现 4.8小结 4.9参考文献 第5章链接分析 5.1PageRank 5.1.1早期的搜索引擎及词项作弊 5.1.2PageRank的定义 5.1.3Web结构 5.1.4避免终止点 5.1.5采集器陷阱及“抽税”法 5.1.6PageRank在搜索引擎中的使用 5.1.7习题 5.2PageRank的快速计算 5.2.1转移矩阵的表示 5.2.2基于Map - Reduce的PageRank迭代计算 5.2.3结果向量合并时的组合器使用 5.2.4转移矩阵中块的表示 5.2.5其他高效的PageRank迭代方法 5.2.6习题 5.3面向主题的PageRank 5.3.1动机 5.3.2有偏的随机游走模型 5.3.3面向主题的PageRank的使用 5.3.4基于词汇的主题推断 5.3.5习题 5.4链接作弊 5.4.1垃圾农场的架构 5.4.2垃圾农场的分析 5.4.3与链接作弊的斗争 5.4.4TrustRank 5.4.5垃圾质量 5.4.6习题 5.5导航页和权威页 5.5.1HITS的直观意义 5.5.2导航度和权威度的形式化 5.5.3习题 5.6小结 5.7参考文献 第6章频繁项集 6.1购物篮模型 6.1.1频繁项集的定义 6.1.2频繁项集的应用 6.1.3关联规则 6.1.4高可信度关联规则的发现 6.1.5习题 6.2购物篮及A - Priori算法 6.2.1购物篮数据的表示 6.2.2项集计数中的内存使用 6.2.3项集的单调性 6.2.4二元组计数 6.2.5A - Priori算法 6.2.6所有频繁项集上的A - Priori算法 6.2.7习题 6.3更大数据集在内存中的处理 6.3.1PCY算法 6.3.2多阶段算法 6.3.3多哈希算法 6.3.4习题 6.4有限扫描算法 6.4.1简单的随机化算法 6.4.2抽样算法中的错误规避 6.4.3SON算法 6.4.4SON算法和Map

<<大数据>>

- Reduce 6.4.5Toivonen算法 6.4.6Toivonen算法的有效性分析 6.4.7习题 6.5流中的频繁项计数 6.5.1流的抽样方法 6.5.2衰减窗口中的频繁项集 6.5.3混合方法 6.5.4习题 6.6小结 6.7参考文献 第7章聚类 7.1聚类技术介绍 7.1.1点、空间和距离 7.1.2聚类策略 7.1.3维数灾难 7.1.4习题 7.2层次聚类 7.2.1欧氏空间下的层次聚类 7.2.2层次聚类算法的效率 7.2.3控制层次聚类的其他规则 7.2.4非欧空间下的层次聚类 7.2.5习题 7.3k - 均值算法 7.3.1k - 均值算法基本知识 7.3.2k - 均值算法的簇初始化 7.3.3选择七的正确值 7.3.4BFR算法 7.3.5BFR算法中的数据处理 7.3.6习题 7.4CURE算法 7.4.1CURE算法的初始化 7.4.2CURE算法的完成 7.4.3习题 7.5非欧空间下的聚类 7.5.1GRGPF算法中的簇表示 7.5.2簇表示树的初始化 7.5.3GRGPF算法中的点加入 7.5.4簇的分裂及合并 7.5.5习题 7.6流聚类及并行化 7.6.1流计算模型 7.6.2 - 个流聚类算法 7.6.3桶的初始化 7.6.4桶合并 7.6.5查询应答 7.6.6并行环境下的聚类 7.6.7习题 7.7小结 7.8参考文献 第8章Web广告 8.1在线广告相关问题 8.1.1广告机会 8.1.2直投广告 8.1.3展示广告的相关问题 8.2在线算法 8.2.1在线和离线算法 8.2.2贪心算法 8.2.3竞争率 8.2.4习题 8.3广告匹配问题 8.3.1匹配及完美匹配 8.3.2最大匹配贪心算法 8.3.3贪心匹配算法的竞争率 8.3.4习题 8.4Adwords问题 8.4.1搜索广告的历史 8.4.2Adwords问题的定义 8.4.3Adwords问题的贪心方法 8.4.4Balance算法 8.4.5Balance算法竞争率的一个下界 8.4.6多投标者的Balance算法 8.4.7 - 般性的Balance算法 8.4.8Adwords问题的最后论述 8.4.9习题 8.5Adwords的实现 8.5.1投标和搜索查询的匹配 8.5.2更复杂的匹配问题 8.5.3文档和投标之间的匹配算法 8.6小结 8.7参考文献 第9章推荐系统 9.1一个推荐系统的模型 9.1.1效用矩阵 9.1.2长尾现象 9.1.3推荐系统的应用 9.1.4效用矩阵的填充 9.2基于内容的推荐 9.2.1项模型 9.2.2文档的特征发现 9.2.3基于Tag的项特征获取 9.2.4项模型表示 9.2.5用户模型 9.2.6基于内容的项推荐 9.2.7分类算法 9.2.8习题 9.3协同过滤 9.3.1相似度计算 9.3.2相似度对偶性 9.3.3用户聚类和项聚类 9.3.4习题 9.4降维处理 9.4.1UrV分解 9.4.2RMSE 9.4.3UV分解的增量式计算 9.4.4对任一元素的优化 9.4.5一个完整UV分解算法的构建 9.4.6习题 9.5NetFlix竞赛 9.6小结 9.7参考文献 索引

章节摘录

版权页：插图：然而，当项对的数目太多而无法在内存中对所有的项对计数时，上述简单的方法就不再可行。

A-Priori算法被设计成能够减少必须计数的项对数目，当然其代价是要对数据做两遍而不是一遍扫描。

1.A-Priori算法的第一遍扫描 第一遍扫描中，我们要建立两张表。

如有必要，第一张表要将项的名称转换为1到n之间的整数（参考6.2.2节中的描述）。

另一张表则是一个计数数组，第i个数组元素是上述第i个项的出现次数。

这些所有项的计数值的初始值都是0。

在读取购物篮时，我们检查购物篮中的每个项并将其名称转换为一个整数。

然后，将该整数作为计数数组的下标找到对应的数组元素，最后，对该数组元素加1。

2.A-Priori算法两遍扫描之间的处理 第一遍扫描之后，我们检查所有项的计数值，以确定哪些项构成单元素频繁项集。

我们可能会看到，大部分单元素项集都是不频繁的。

这一点可能会有点出人意料。

但是，前面提到，我们常常将阈值s设置得足够高以保证频繁集不会太多。

一个典型的s值为所有购物篮数目的1%。

想象一下自己到超市购物的情况，我们购买某些商品的次数肯定会超过总次数的1%，这些商品可能是牛奶、面包、可口可乐或百事可乐什么的。

我们甚至相信，虽然我们不购买尿布，但是会有1%的顾客会购买尿布。

然而，货架上的大部分商品的顾客购买比例肯定都不会超过1%，比如奶油凯撒沙拉汁。

对于A-Priori算法的第二遍扫描，我们会只给频繁项重新编号，编号范围是1到m。

此时的表格是一个下标为1到n的数组，如果第i项不频繁，则对应的第|A|数组元素为0，否则为1到m之间的一个唯一整数。

我们应将此表格称为频繁项表格。

3.A-Priori算法的第二遍扫描 在第二遍扫描中，我们对两个频繁项组成的所有项对计数。

从6.2.3节的讨论可知，除非一个项对中的两个项都频繁，否则这个项对也不可能是频繁的。

因此，在扫描过程中我们不可能丢掉任何频繁项对。

如果采用前面提到的三角矩阵方法来计数的话，则第二遍扫描所需的空间是 $2n^2$ 而不是 $2n$ 。

需要注意的是，如果要使用一个大小正确的三角矩阵，那么就一定要只对频繁项进行重新编号处理。

第一遍和第二遍扫描中所使用的完整内存结构集合如图6-3所示。

需要注意的另外一点是，上述非频繁项去除的好处会被放大：如果只有一半的项是频繁项，那么在计数过程中仅需要原来空间的1/4。

类似地，如果使用三元组方式，我们只需要对至少出现在一个购物篮中的两个频繁项组成的项对进行计数。

第二遍扫描的技术细节如下：（1）对每个购物篮，在频繁项集表中检查哪些项是频繁的；（2）通过一个双重循环生成所有的频繁项对；（3）对每个频繁项对，在存储计数值的数据结构中相应的计数值上加1；最后，在第二遍扫描结束时，检查计数值结构以确定哪些项对是频繁项对。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>