Hadoop 分布式架构下大数据集的并行挖掘

吕婉琪,钟 诚,唐印浒,陈志朕
(广西大学 计算机与电子信息学院,广西 南宁 530004)

摘 要:基于 Hadoop 分布式计算平台,给出一种适用于大数据集的并行挖掘算法。该算法对非结构化的原始大数据集以及中间结果文件进行垂直划分以确保能够获得完整的频繁项集,将各个垂直分块数据分配给不同的 Hadoop 计算节点进行处理,以减少各个计算节点的存储数据,进而减少各个计算节点执行交集操作的次数,提高并行挖掘效率。实验结果表明,给出的并行挖掘算法解决了大数据集挖掘过程中产生的大量数据通信、中间数据以及执行大量交集操作的问题,算法高效、可扩展。

关键词:数据挖掘;大数据集;并行算法;Hadoop
中图分类号:TP311.133.2;TP338.6 **文献标识码**:A **文章编号**:1673-629X(2014)01-0022-04
doi:10.3969/j.issn.1673-629X.2014.01.

Parallel Mining of Large Dataset in Hadoop Distributed Computing Framework

LÜ Wan-qi,ZHONG Cheng,TANG Yin-hu,CHEN Zhi-zhen
(School of Computer and Electronics and Information,Guangxi University,Nanning 530004,China)

Abstract:Based on Hadoop distributed computing framework,propose a parallel algorithm for mining the large dataset. The presented algorithm divides the original large non-structured dataset and large middle result files into several smaller-scale data blocks by vertical partitioning pattern in order to ensure the completeness of the frequent item set. The algorithm can reduce the size of the data to be stored in each computing node and decrease the execution times that each computing node calculates the intersection operations by distributing the data blocks to the computing nodes to parallel mining in Hadoop distributed computing environment,and it can improve the efficiency of parallel mining. The experimental results show that the presented parallel mining algorithm can solve the problem that the mining large dataset will generate large amount of data communication and large number of operations for calculating intersection,and it is efficient and scalable.

Key words:data mining;large dataset;parallel algorithm;Hadoop

0 引 言

诸如 Apriori 算法和 FP-Tree 算法这样的数据挖掘算法基于水平数据格式进行挖掘[1]。Zaki 等人在文献[2]中提出采用垂直数据格式的串行数据挖掘算法 Eclat 和 MaxEclat,这两个算法均采用基于前缀的等价类技术,不同的是 Eclat 算法采用自底向上的搜索策略,而 MaxEclat 算法采用了混合搜索策略,实验表明这两个算法的效率优于 Apriori 算法。基于概念格理论,文献[3]对 Eclat 串行算法进行改进,改进后的算法将项集划分成独立的块(子概念格),各子概念格采用自底向上的搜索方法独立产生频繁项集,当数据集包含较多稠密模式和长模式时,该算法可以有效降低挖掘时间。文献[4]对串行 Eclat 算法进行改进,给出的 Diffset 算法在求交集过程中求出的不再是记录项集的所有事务的标识符,而是记录两个项集的事务标识符集合之差,这样减小了中间数据的规模,降低了系统内存消耗,提高了计算效率。文献[5]改进了 Eclat 串行算法,它的思想是将数据库划分成非重叠的小数据块,对这些数据块一个接一个地挖掘,对某个小数据块挖掘出频繁 k-项集后,依据其前几个小数据块所得出的频繁 k-项集按照一定的约束,对当前小数据块的频繁项集进行删减,以加快挖掘速度;该算法对水平子

收稿日期:2013-03-18 **修回日期**:2013-06-23 **网络出版时间**:2013-11-12
基金项目:广西自然科学基金(2011GXNSFA018152);广西研究生教育创新计划项目(YCSZ2012007)
作者简介:吕婉琪(1989-),女,硕士生,研究方向为并行数据挖掘、网络信息安全;钟 诚,博士,教授,博士生导师,CCF 高级会员,研究方向为并行分布计算、网络信息安全。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20131112.1650.047.html

数据块进行处理,虽然减少了数据块内交集操作次数,但是会造成频繁项集缺失。文献[6]在集群环境下实现基于概念格遍历策略的 Eclat 并行算法 Par-Eclat,它假定原始数据库本身是垂直分布的,每个子数据块均包含相关的项集事务列表信息,将子数据块分配到各个处理器进行项集支持度的计算,此算法适用于分散的数据源的挖掘。

Hadoop[7] 分布式架构是实现云计算的平台之一。Hadoop 分布式架构的文件系统(HDFS)以流式数据访问模式来存储超大文件[8]。文献[9]研究了如何利用 MapReduce 编程模型来实现大数据的处理。文献[10]介绍了基于云计算的 Hadoop 集群框架和数据挖掘技术中的 SPRINT 分类算法,描述了 MapReduce 编程模型上的 SPRINT 并行算法的执行流程。文献[11]探讨应用 MapReduce 编程模型实现了 Eclat 挖掘算法,但是它没有深入研究数据集的分块处理和降低数据通信开销的方法,而且在挖掘过程中存在频繁项集缺失现象。文献[12]利用 MapReduce 编程模型对 Apriori 算法进行改进,给出了在 Hadoop 分布式架构下实现数据挖掘的过程。

文中的主要贡献是:给出一种 Hadoop 分布式架构下并行挖掘频繁项集的算法 HDEclat,该算法对非结构化的原始数据以及中间结果文件进行垂直划分,以确保能够获得完整的频繁项集,通过将互不重叠的数据块分配给不同的云计算节点进行处理,以增大并行粒度,减小交集操作的规模,提高并行挖掘效率,解决了当原始数据集规模较大时挖掘过程中产生大量数据通信、中间数据以及执行大量交集操作的问题。

1　Hadoop 分布式架构下大数据集的并行挖掘算法

串行 Eclat 算法将原始数据集转换成垂直数据表示格式进行处理,当事务数据量很大时,项目 item 的事务标识符集合会非常的长,在求项集的支持度而执行交操作时会产生较多比较次数,相当耗时,挖掘效率较低。

文中基于 Hadoop 分布式架构,研究通过垂直划分大数据集的方式对 Eclat 挖掘算法并行化,设计实现获得完整的频繁项集、高效和可扩展的大数据集挖掘并行算法。

Hadoop 系统默认将数据文件划分成大小为 64 MB 的分块,这里的分块是传统的水平分块。水平分块会导致项目分布在不同的子块中,计算出的频繁项集不完整。下面举例分析说明。将表 1 的水平格式的原始数据集转换成垂直数据格式,如表 2 所示。

假设数据集的最小支持度 minsup 为 0.4,则频繁 1 - 项集为 $\{I_1, I_2, I_5, I_6, I_7\}$,水平分块划分为 $\{I_1, I_2\}$ 和 $\{I_5, I_6, I_7\}$。2 - 项候选集的事务标识符集合 Tidset 通过计算各分块的频繁 1 - 项集的交集得到,只能得到集合 $\{I_1 \cup I_2, I_5 \cup I_6, I_5 \cup I_7, I_6 \cup I_7\}$ 中元素的支持度,两个数据块之间数据不能相交,造成频繁项集缺失。

表 1　水平数据格式表示的原始数据集

事务 ID	项目 Item 列表
T_1	I_1　I_2　I_3　I_7
T_2	I_2　I_4　I_5　I_6　I_9
T_3	I_1　I_4　I_5　I_7　I_8
T_4	I_1　I_6　I_8　I_9
T_5	I_1　I_7　I_9
T_6	I_1　I_2　I_5　I_7
T_7	I_3　I_4　I_5　I_6
T_8	I_2　I_5　I_6　I_7
T_9	I_1　I_2　I_4　I_5
T_{10}	I_1　I_2　I_5
T_{11}	I_2　I_5　I_7

表 2　原始数据集的垂直数据格式

项目 Item	TID 集
I_1	T_1　T_3　T_4　T_5　T_6
I_2	T_1　T_2　T_6　T_8
I_3	T_1　T_7
I_4	T_2　T_3　T_7
I_5	T_2　T_3　T_6　T_7　T_8　T_9　T_{10}
I_6	T_2　T_4　T_7　T_8
I_7	T_1　T_3　T_5　T_6　T_8　T_{11}
I_8	T_3　T_4

为了获得完整的频繁项集,文中采取垂直划分数据块的方式,垂直划分是指按照项目所在的事务 ID 范围进行划分,该例中每个项目的事务 ID 取值范围为 T_1 到 T_{11},可以划分成三个块 $\{T_1, \cdots, T_4\}$、$\{T_5, \cdots, T_8\}$、$\{T_9, \cdots, T_{11}\}$,最后结果如表 3 所示。

表 3　垂直划分成块

项目 Item	TID_1 集	TID_2 集	TID_3 集
I_1	T_1　T_3　T_4	T_5　T_6	null
I_2	T_1　T_2	T_6 T_8	null
I_3	T_1	T_7	null
I_4	T_2　T_3	T_7	null
I_5	T_2　T_3	T_6　T_7　T_8	T_9 T_{10}
I_6	T_2 T_4	T_7　T_8	null
I_7	T_1　T_3	T_5　T_6　T_8	T_{11}
I_8	T_3　T_4	null	null

文中基于“分而治之”思想,运用 MapReduce 将产生的大数据文件进行划分。由于水平分块不适用于采用垂直数据格式的 Eclat 算法并行化,所以文中不采用 Hadoop 默认的分块方式,而是按照每个项目的事务标识符集合中数据的取值范围对数据集进行垂直切割,划分成互不重叠的数据块。然后,将包含有项目信息的子块分配给各个节点,每个 map 进程对一个子块进行处理。

算法中利用 Apriori 性质(“频繁项集的所有非空子集也必须是频繁的”)对候选项集进行剪枝,减少对额外的候选集进行支持度计数。在并行计算过程中,利用 Hadoop 架构的本身负载均衡的特性,使得每个计算节点处理完一个数据块后请求下一个数据块,这样计算能力强的节点处理更多的数据,缓解可能出现数据分配不均的现象。

假设大数据文件 F 的规模为 N,事务个数为 TN,HDFS 的数据块 block 的大小为 M,Hadoop 将大数据文件 F 划分成$\lceil N/M \rceil$个分块数据;并设 Hadoop 系统中最大并行 map 任务个数为 maxmap。当$\lceil N/M \rceil$ > maxmap 时,首先给每个 map 任务分配一个数据块,若有的 map 任务执行结束则请求下一个数据块,计算每个项目出现在哪些事务中以及支持度计数,删除支持度小于 TN × minsup 的 1 - 项集,同时将数据转换成垂直表示格式。输出时按照垂直划分的方式将不同范围的事务标识符存储到不同的文件中,然后进行计算频繁 2 - 项集。

Hadoop 分布式架构下大数据集挖掘并行算法描述如下:

算法 1 :Hadoop 分布式架构下大数据集挖掘并行算法 HDEclat。

输入:大数据文件 F,最小支持度阈值 minsup;

输出:频繁项集 L。

Begin

(1) 主程序将大小为 N 的大数据集划分成$\lceil N/M \rceil$个数据块,将数据块分发给每个计算节点中的 map 进程进行并行处理,每个 map 任务的输出结果是包含 < item:TID > 数据的一个临时文件;

(2) 为了减少节点间的通信量,编写 Combiner 将 map 阶段的本地输出结果进行合并,减少输出中间结果的数据量,其中每行的格式是 < item:TidSet >,TidSet 是包含此项目的事务集合;

(3)reduce 函数将各个计算节点的临时文件整合成垂直1 - 项集,计算项的支持度,若大于 TN × minsup 则是频繁 1 - 项集,将该项目的 TidSet 分块输出到不同的文件中,为了使每个垂直分块数据与 HDFS 的数据块的规模尽可能相同,这里将每个1 - 项集的 TidSet 分成$\lceil N/M \rceil$个数据块,其中前$\lceil N/M \rceil$个文件中包含的事务个数为 TN/$\lceil N/M \rceil$,最后一个文件中包含的事务个数为 TN - TN × ($\lceil N/M \rceil$ - 1)/$\lceil N/M \rceil$,由此可得到事务 i 在第 j 个文件中,其中 j = $\begin{cases} \lceil i/(\mathrm{TN}/\lceil N/M \rceil) \rceil, & \text{若 } i \leqslant \mathrm{TN} \times (\lceil N/M \rceil - 1)/\lceil N/M \rceil \\ \lceil N/M \rceil, & \text{若 } i > \mathrm{TN} \times (\lceil N/M \rceil - 1)/\lceil N/M \rceil \end{cases}$;

(4)$k \leftarrow 2$;

(5) 将项目的 TidSet 分块输出分发给 map 进程,各个进程并行计算候选 k - 项集支持度,reduce 进程负责将各个 map 进程的输出结果汇总起来,得到全局频繁 k - 项集。利用步骤(3) 中的分块思想,将全局频繁 k - 项集分块输出到不同的文件中;

(6)$k \leftarrow k + 1$;

(7) 重复执行步骤(5) ~ (7),直到没有包含更多频繁项集的数据输出为止。

End

基于 Hadoop 分布式架构的 HDEclat 并行算法在对大数据集进行并行挖掘的过程中,它通过移动程序而不是移动数据执行分布式并行挖掘程序,并且各个计算节点运用第(2)步将自己挖掘的结果数据进行汇聚以减少中间结果的数据量,这样大大减少了数据的通信量,明显减少了并行挖掘过程中所需要的通信代价。该算法通过将大数据集垂直划分成互不重叠的数据块的方法,使得基于两两结合进行交运算的 HDEclat 算法得以有效并行执行,增大了可运算的数据规模,从而实现对大数据集的有效并行挖掘。

2 实 验

2.1 实验环境和实验数据

使用6 台计算机互连组成的 Hadoop 集群系统进行实验。实验将其中一台计算机作为名字节点(NameNode),它是整个 HDFS 的核心,不参与运算;其余 5 台计算机为数据节点(DataNode),用来存储和计算数据。运行的操作系统均为 Red Hat Enterprise Linux 5。采用 MapReduce 编程模型及 Java 语言编程实现算法。实验测试数据是扩充的 mushroom. dat,该数据集含有 120 个项目(item)。实验使用的数据如下:数据集 A -24 MB,包含 324 960 条记录数据;数据集 B -96 MB,包含有 1299 840 条记录数据;数据集 C -384 MB,包含 5199 360 条记录数据。为了充分利用 Hadoop 集群中的计算节点,增大并行粒度,文中修改了 Hadoop 架构的默认分块大小,对不同的数据规模设置不同的分块大小。

2.2 实验结果与分析

图 1 ~ 图 3 分别给出了当最小支持度为 0.6、0.7 和 0.8,Hadoop 集群中运行的计算节点逐步增加时,

HDEclat 并行算法并行挖掘数据集 *A*、*B* 和 *C* 所需要的时间。图 4 给出了当 Hadoop 集群运行计算节点个数为 4,最小支持度变化时,HDEclat 并行挖掘算法对 24 MB 数据集、96 MB 数据集和 384 MB 数据集进行并行挖掘所需的时间。

图 1 ~图 3 的实验结果表明:无论最小支持度是 0.6 或者 0.7 或者 0.8,对于 384 MB 这样的大数据集,当 Hadoop 集群中运行的计算节点逐步增多时,HDEclat 算法并行挖掘所需的运行时间均明显减少;当要处理的数据集相对较小(24 MB、96 MB)时,Hadoop 集群中运行计算节点的增加对并行挖掘所需的时间均影响不大,即对于较小规模数据集的并行挖掘,增加 Hadoop 集群运行的计算节点难以获得大的挖掘效率提升。这是因为 HDEclat 算法的主要运算时间由迭代求 k-频繁项集所花费的时间组成,每次迭代均需要重新调度 Hadoop 的作业任务,无论是大规模数据集的并行挖掘、还是中等规模数据集的并行挖掘抑或较小规模数据集的并行挖掘,均需要一定的调度任务时间开销。因此,Hadoop 集群下 HDEclat 并行算法挖掘大数据集更有明显优势。这表明,Hadoop 分布式架构下 HDEclat 并行挖掘算法实现了高效地挖掘大数据集,具有良好的可扩展性。

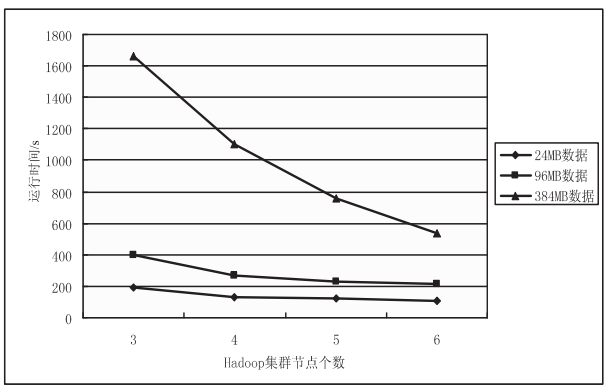

图 1　最小支持度为 0.6 时 HDEclat 算法并行挖掘时间

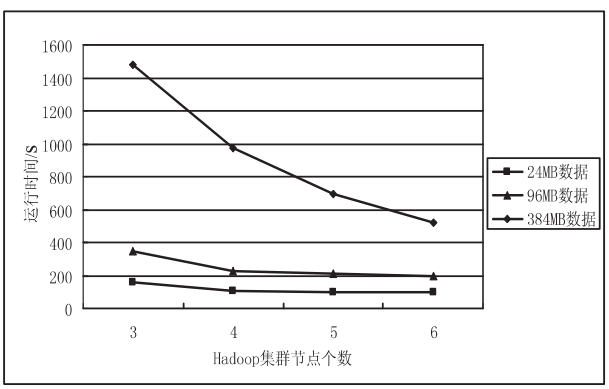

图 2　最小支持度为 0.7 时 HDEclat 算法并行挖掘时间

由图 4 的结果可以看到,对于三种规模的数据集,当最小支持度从 0.5 逐步增加到 0.9 时,HDEclat 并行挖掘算法挖掘数据所需的运行时间逐步减少,这说明最小支持度的大小对 Hadoop 分布式架构下 HDEclat 并行挖掘算法的效率有明显的影响。这是因为当最小支持度较大时,在每次迭代处理过程中,并行挖掘算法就过滤掉更多不符合条件的数据记录,这样以后迭代处理的数据量逐步减少,整个并行挖掘需要的时间就减少;尤其是对于大数据集的挖掘,减少的时间更加明显。

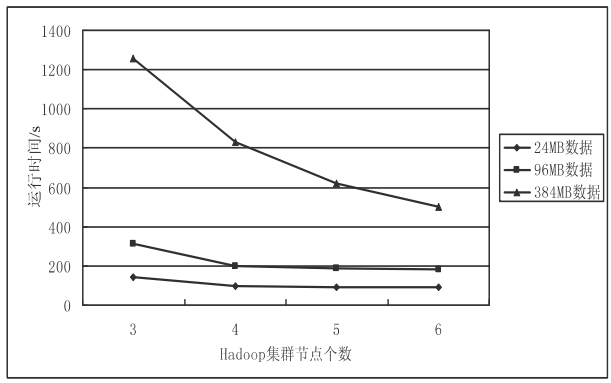

图 3　最小支持度为 0.8 时 HDEclat 算法并行挖掘时间

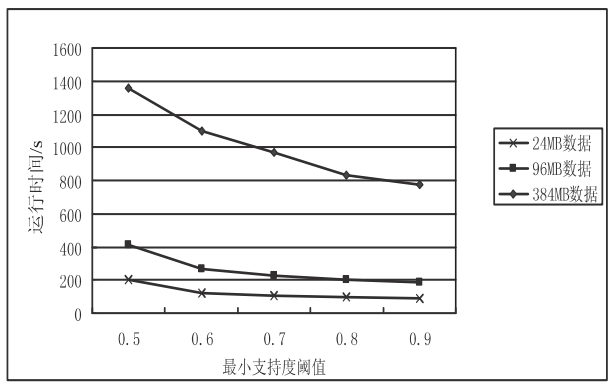

图 4　最小支持度变化时 HDEclat 算法并行挖掘时间

另一方面,从图 4 中还可以看到,当数据集规模从 24 MB 增大到 384 MB(数据集规模增大 16 倍)时,HDEclat 并行挖掘算法所需的运行时间并没有相应地增加 16 倍,而只是增加了 5 倍左右,这说明 Hadoop 分布式架构下 HDEclat 并行挖掘算法处理大数据集时可以提高并发度。

3　结束语

文中利用 MapReduce 并行编程模式,设计实现的 Hadoop 架构下 HDEclat 并行挖掘算法获得完整的频繁项集,解决了挖掘垂直数据格式的大数据集时遇到的求交集操作消耗大量时间的问题。该算法将原始数据集垂直划分并将分块数据分发给不同的 map 进程并行计算,有效利用系统的计算和存储资源,减少了每个数据分块求交集的操作次数,各个计算节点运用 MapReduce 将自己挖掘的结果数据进行汇聚以减少中间结果的数据量,显著地减少并行挖掘过程中所需的通

(下转第 30 页)

6 结束语

通过已经描述的边界片段模型对物体类别的检测实例表明其性能与所提的一些文献相比有所提高。实验可以发现对于一些部分边界问题可以用边缘检测器的方法解决。BFM 在结合外观和边界的情况下对于那些纹理可变的物体(如瓶子,杯子)其性能可能会更优越。实验结果表明文中所提算法在基于形状局部特征的物体检测方面具有很大的发展潜力。

参考文献:

[1] Leibe B, Leonardis A, Schiele B. Combined object categorization and segmentation with an implicit shape model[C]//Proc of workshop on stat. learning in computer vision. [s. l.]:[s. n.],2004:17-32.

[2] Leibe B, Schiele B. Scale-invariant object categorization using a scale-adaptive mean-shift search[C]//Proc of DAGM'04. [s. l.]:[s. n.],2004:145-153.

[3] Kumar M, Torr P, Zisserman A. Extending pictural structures for object recognition[C]//Proc of BMVC. [s. l.]:[s. n.], 2004.

[4] Opelt A, Pinz A, Zisserman A. A boundary-fragment-model for object detection[C]//Proc of ECCV. [s. l.]:[s. n.], 2006:575-588.

[5] 郝文欣,高立宁. 基于视觉注意机制与局部描述子的物体检测[J]. 计算机工程与设计,2012,33(5):1918-1922.

[6] Borgefors G. Hierarchical chamfer matching: A parametric edge matching algorithm[J]. IEEE PAMI,1988,10(6):849-865.

[7] Chia A Y S, Rajan D, Leung M K, et al. Object recognition by discriminative combinations of line segments, ellipses, and appearance features[J]. IEEE transactions on pattern analysis and machine intelligence,2012,34(9):1758-1772.

[8] Chan D Y, Hsu R C, Chiu T Y, et al. Fast robust object segmentation by progressive shape-anchor selecting and adaptive thresholding piecewise linking[J]. International journal of innovative computing information and control, 2012, 8(10a): 6903-6919.

[9] 李梦亮,翁正新. 基于改进区域生长法和霍夫变换的车道分割法[J]. 计算机应用与软件,2011,28(12):246-248.

[10] 黄 琦,张国基,唐向京. 基于霍夫变换的图像运动模糊角度识别法的改进[J]. 计算机应用,2008,28(1):211-213.

[11] Duda R O, Hart P E. Use of Hough transformation to detect line sand curves in pictures[J]. Communications of the ACM, 1972,15(1):11-15.

[12] 李泽宇. 浅谈训练样本对 Adaboost 算法的影响[J]. 信息通信,2012(5):10-11.

[13] 胡国胜. 基于加权支持向量机与 AdaBoost 集成的预测模型研究[J]. 计算机应用与软件,2012,29(12):280-281.

[14] Opelt A, Pinz A, Zisserman A. Learning an alphabet of shape and appearance for multi-class object detection[J]. International journal of computer vision,2008,80(1):16-44.

[15] Comaniciu D, Meer P. Mean shift: A robust approach towards feature space analysis[J]. PAMI,2002,24(5):603-619.

[16] 张维泽. 基于简单局部特征学习的物体检测算法[D]. 杭州:浙江大学,2010.

[17] Gao Y S, Leung M K H. Line segment Hausdorff distance on face matching[J]. Pattern recognition, 2002, 35(2): 361-371.

(上接第 25 页)

信时间开销。Hadoop 架构下的 HDEclat 并行挖掘算法能够高效地挖掘大数据集,具有良好的扩展性。下一步工作将研究大数据集的数据挖掘隐私保护技术。

参考文献:

[1] Han Jiawei, Kamber M. 数据挖掘:概念与技术[M]. 范明,孟小峰,译. 北京:机械工业出版社,2008.

[2] Zaki M J, Parthasarathy S, Ogihara M, et al. New algorithms for fast discovery of association rules[C]//Proc of 3rd intl conf on knowledge discovery and data mining. Palo Alto, California: AAAI Press,1997:283-286.

[3] Zaki M J. Scalable algorithms for association mining[J]. IEEE transactions on knowledge and data engineering, 2000, 12(3):372-390.

[4] Zaki M J. Fast vertical mining using diffsets[R]. New York, USA: Rensselaer Polytechnic Institute,2001.

[5] 张玉芳,熊忠阳,耿晓斐,等. Eclat 算法的分析及改进[J]. 计算机工程,2010,36(23):28-30.

[6] Zaki M J, Parthasarathy S, Ogihara M, et al. Parallel algorithms for discovery of association rules[J]. Data mining and knowledge discovery,1997,1(4):343-373.

[7] Apache Hadoop[EB/OL]. 2012-12-03. http://hadoop.apache.org/.

[8] Shvachko K, Kuang Hairong, Radia S, et al. The Hadoop distributed file system[C]//Proc of 26th IEEE symposium on mass storage systems and technologies. Piscataway, NJ: IEEE Press,2010:1-10.

[9] Dean J, Ghemawat S. MapReduce: Simplified data processing on large clusters[J]. Communications of the ACM,2008,51(1):107-113.

[10] 王 鄂,李 铭. 云计算下的海量数据挖掘研究[J]. 现代计算机(专业版),2009(11):22-25.

[11] 李伟卫,赵 航,张 阳,等. 基于 MapReduce 的海量数据挖掘技术研究[J/OL]. 2012. http://www.cnki.net/kcms/detail/11.2127.TP.20120601.1457.016.html.

[12] 李玲娟,张 敏. 云计算环境下关联规则挖掘算法的研究[J]. 计算机技术与发展,2011,21(2):43-46.

Hadoop分布式架构下大数据集的并行挖掘

作者： 吕婉琪， 钟诚， 唐印浒， 陈志朕， L Wan-qi， ZHONG Cheng， TANG Yin-hu， CHEN Zhi-zhen

作者单位： 广西大学 计算机与电子信息学院,广西 南宁,530004

刊名： 计算机技术与发展

ISTIC

英文刊名： Computer Technology and Development

年，卷(期)： 2014(1)