

# 水平分布数据库的正负关联规则挖掘

吴青,傅秀芬

(广东工业大学 计算机学院,广东 广州 510060)

**摘要:**目前,正负关联规则的挖掘受到越来越多的关注,在现实运用中也越来越广泛。随着信息技术和经济全球化的发展,许多企业都在全国甚至全世界范围内拥有自己的数据中心。企业通过分析这些数据的关联来为他们的战略抉择和政策制定服务。而直接在这些庞大的数据中寻找数据之间的关联不是一件容易的事。而且,在这些大量的数据中,不是所有的数据都是分析中所需要的。文中通过在文献[4]中所提的方法中引入对不同数据库赋予不同权重值的方式,使得在分布式数据库中挖掘正负关联规则更加高效。经过测试,这一改进是有效的。

**关键词:**数据挖掘;正负关联规则;水平分布式数据库

**中图分类号:**TP311

**文献标识码:**A

**文章编号:**1673-629X(2010)06-0113-05

## Positive and Negative Association Rules Mining on Horizontally Partitioned Database

WU Qing, FU Xiu-fen

(School of Computer and Science, Guangdong University of Technology, Guangzhou 510060, China)

**Abstract:** Recently, positive and negative association rules mining has received some attention and been proved to be useful in real world. As the developing of information technology and economy's globalization, many enterprises and companies have established their data centers around the country, even around the world. They usually mine useful information in their distributed databases to help their decisions making and policies establishing. But, there are thousands of items in each database and it will be expensive if directly mining association rules in such a large database. What's more, most of the items are not interesting. Extend the algorithm mentioned in reference[4] by place different weight on different database to make it more efficient to mine both positive and negative association on horizontally partitioned data. Through testing, this optimization is effective.

**Key words:** data mining; positive and negative association rules; horizontally partitioned database

## 0 引言

数据挖掘(Data Mining),就是从大量数据中获取有效的、新颖的、潜在有用的、最终可理解的模式的非平凡过程。数据挖掘的广义观点:就是从存放在数据库、数据仓库或其他信息库中的大量的数据中“挖掘”有趣知识的过程。通常数据挖掘可以做以下的事情:分类、估值、相关性分组或关联规则、聚集、描述和可视化以及复杂数据类型(Tex、Web、图形图像、视频、音频等)。笔者将研究的正是这些应用中的相关性分组或关联规则方面。

数据关联是数据库中存在的—类重要的可被发现

的知识。若两个或多个变量的取值之间存在某种规律性,就称为关联。关联可分为简单关联、时序关联、因果关联。关联分析的目的是找出数据库中隐藏的关联网。有时并不知道数据库中数据的关联函数,即使知道也是不确定的,因此关联分析生成的规则带有可信度。关联规则挖掘发现大量数据中项集之间有趣的关联或相关联系。关联规则的挖掘是一种重要的知识表示。它揭示了大型事务中各个项目之间潜在的关系。关联规则挖掘(ARM)最早出现在 Agrawal 的文献<sup>[1]</sup>中。Agrawal 等首先提出了挖掘顾客交易数据库中项集间的关联规则问题,以后诸多的研究人员(文献[2]等)对关联规则的挖掘问题进行了大量的研究。他们的工作包括对原有的算法进行优化,如引入随机采样、并行的思想等,以提高算法挖掘规则的效率;对关联规则的应用进行推广。关联规则挖掘在数据挖掘中是一个重要的课题,最近几年已被业界所广泛研究。在电信网络、市场分析、风险评估、仓库控制以及诊断抉择

收稿日期:2009-10-02;修回日期:2010-01-10

基金项目:广东省自然科学基金项目(07001802)

作者简介:吴青(1981-),男,四川绵阳人,硕士,研究方向为数据库技术与应用;傅秀芬,博士,教授,研究方向为多媒体网络软件、数据库技术与应用、中间件技术。

支持中关联规则挖掘都有广泛的应用。

传统的管理按规则挖掘主要关注的是项目间的正关联。而最近越来越多的关注开始投向负关联的挖掘。已经提出了许多算法来解决知名的和代价可估算的关联规则的挖掘。

随着信息技术和经济全球化的发展,许多企业都在全国甚至全世界范围内拥有自己的数据中心。他们通过分析这些数据的关联来为他们的战略抉择和政策制定服务。而直接在这些庞大的数据中寻找数据之间的关联不是一件容易的事。而且,在这些大量的数据中,不是所有的数据都是分析中所需要的。根据调查,许多分布式数据库都是水平划分的(按事务分组的),比方说超市和连锁市场的数据库。而且,在商业意义上讲,所有的分布式数据库的重要性都是不一样的。那些拥有大宗交易数量的分支理当在全局关联规则挖掘中具有更重要的意义。虽然之前有人提出过负关联的挖掘算法,但是他们主要是针对单一数据库提出的,并没有给出分布式数据库中正负关联规则的挖掘算法。在此,将给出一个在水平分布数据库中挖掘正负关联规则的有效算法。由于这些水平分布数据库在全局关联规则挖掘中具有不同的重要性,将采用类似于文献[3]中的算法来处理数据,给不同数据库赋予不同的权重系数。最后将改进文献[4]的算法来实现水平分布数据库中正负关联规则的挖掘。

## 1 相关工作

首先,介绍管理按规则挖掘中的基本概念和定义。然后说明一下文献[1],将用文献[1]中的方法来剔除关联规则挖掘中那些无意义的关联规则。接着将介绍一下关系系数(correlation coefficient)<sup>[4]</sup>。然后将把使用权重和相关系数的方法用于分布式数据库的正负关联规则的挖掘中。在描述有关关联规则的一些细节之前,先来看一个有趣的故事:“尿布与啤酒”的故事。

在一家超市里,有一个有趣的现象:尿布和啤酒赫然摆在一起出售。但是这个奇怪的举措却使尿布和啤酒的销量双双增加了。这不是一个笑话,而是发生在美国沃尔玛连锁店超市的真实案例,并一直为商家所津津乐道。沃尔玛拥有世界上最大的数据仓库系统,为了能够准确了解顾客在其门店的购买习惯,沃尔玛对其顾客的购物行为进行购物篮分析,想知道顾客经常一起购买的商品有哪些。沃尔玛数据仓库里集中了其各门店的详细原始交易数据。在这些原始交易数据的基础上,沃尔玛利用数据挖掘方法对这些数据进行分析和挖掘。一个意外的发现是:“跟尿布一起购买最多的商品竟是啤酒!经过大量实际调查和分析,揭示

了一个隐藏在“尿布与啤酒”背后的美国人的一种行为模式:在美国,一些年轻的父亲下班后经常要到超市去买婴儿尿布,而他们中有30%~40%的人同时也为自己买一些啤酒。产生这一现象的原因是:美国的太太们常叮嘱她们的丈夫下班后为小孩买尿布,而丈夫们在买尿布后又随手带回了他们喜欢的啤酒。

按常规思维,尿布与啤酒风马牛不相及,若不是借助数据挖掘技术对大量交易数据进行挖掘分析,沃尔玛是不可能发现数据内在这一有价值的规律的。

### 1.1 概念和定义

设  $I = \{i_1, i_2, \dots, i_n\}$  是  $n$  个项目的集合,称  $I$  为项目集。设  $DB$  是事务的集合,各个事务包含若干项目。每个事务都具有一个独一无二的标识符 TID。设  $A$  是项目的集合叫做项目集。项目集中项目的个数叫做项目集的长度(length)。长度为  $k$  的项目集表示为  $k$ -itemset。如果  $A \subset T$  就说事务  $T$  包含项目集  $A$ 。所谓关联规则是指  $A \Rightarrow B$ , 并且  $A \subset I, B \subset I, A \cap B = \emptyset$ 。称  $A$  为规则的前项,  $B$  称作规则的后项。

定义1 如果在数据库  $DB$  中有  $s\%$  的事务包含  $A \cup B$ , 则说规则  $A \Rightarrow B$  的支持度为  $s$ 。换句话说,支持度就是指现存案例中同时包含  $A$  和  $B$  的概率。

$$\text{supp}(A \Rightarrow B) = \text{supp}(A \cup B) = P(A \cup B) \quad (1)$$

定义2 如果数据库  $DB$  中包含  $A$  的事务中有  $c\%$  的事务同时也包含  $B$ , 就说规则  $A \Rightarrow B$  的自信度是  $c$ 。换句话说,就是在前项  $A$  成立的条件下后项  $B$  为真的条件概率。

$$\text{conf}(A \Rightarrow B) = P(B | A) = \frac{\text{supp}(A \cup B)}{\text{supp}(A)} \quad (2)$$

### 1.2 经典算法

关联规则挖掘就是发掘那些对抉择制定和政策建立有帮助作用的关联规则的过程。因此,希望只挖掘那些感兴趣的项目。但是怎么判断某个项目是不是感兴趣的呢?关联规则是有趣的,如果满足最小支持度阈值和最小置信度阈值。这些阈值由用户或者专家设定。比较经典的算法是关联规则挖掘的支持度-自信度框架<sup>[1,5]</sup>。它一般分为如下两个步骤:

(1) 生成所有的频繁集:找到所有支持度大于等于用户给定的最小支持度  $ms$  的项目集。

(2) 生成所有具有用户指定的最小自信度  $mc$  的规则:对所有的频繁项目集  $X$  和  $B \subset X$ , 设  $A = X - B$ 。如果规则  $A \Rightarrow B$  具有  $mc$ , 那么这就是有效的规则。

举一个简单的例子:表1是顾客购买记录数据库  $D$ , 包括6个事务。项目集  $I = \{\text{火腿肠}, \text{红酒}, \text{西瓜}, \text{面包}\}$ 。假设给定最小支持度  $ms = 50\%$ 。容易看出事务1,2,3,4,5,6 包含火腿肠,  $\text{supp}(\text{火腿肠}) = 6/6 =$

100%, 事务 2, 4, 5, 6 包含红酒  $\text{supp}(\text{红酒}) = 4/6 = 66.7\%$  均为有兴趣的项。而  $\text{supp}(\text{西瓜}) = \text{supp}(\text{面包}) = 2/6 = 33.3\%$  均为无兴趣的项。考虑关联规则: 火腿肠  $\Rightarrow$  红酒。事务 1, 2, 3, 4, 5, 6 包含火腿肠, 事务 2, 4, 5, 6 同时包含火腿肠和红酒。支持度分别为:

$$\text{supp}(\text{火腿肠}) = 6/6 = 100\%$$

$$\text{supp}(\text{火腿肠} \Rightarrow \text{红酒}) = 4/6 = 66.7\%$$

火腿肠  $\Rightarrow$  红酒的自信度为:

$$\text{conf}(\text{火腿肠} \Rightarrow \text{红酒}) = \frac{\text{supp}(\text{火腿肠} \cup \text{红酒})}{\text{supp}(\text{火腿肠})} =$$

$$\frac{4/6}{6/6} = 66.7\%$$

若给定的最小支持度为  $\text{mc} = 50\%$ , 则关联规则: 火腿肠  $\Rightarrow$  红酒是有趣的, 认为购买火腿肠和购买红酒之间存在关系。

表1 关联规则的简单例子

TID	火腿肠	红酒	西瓜	面包
1	1	0	0	1
2	1	1	1	0
3	1	0	1	0
4	1	1	0	0
5	1	1	0	1
6	1	1	0	0

### 1.3 负关联规则

设项目集  $A$  是负的, 是指不存在项目集  $A$ , 用  $\neg A$  表示。把形如  $A \Rightarrow B$  的规则称为正关联规则, 而形如  $A \Rightarrow \neg B$ ,  $\neg A \Rightarrow B$ , and  $\neg A \Rightarrow \neg B$  的规则称为负关联规则。

负关联规则的支持度和自信度可以利用正关联规则的支持度和自信度表示<sup>[6-8]</sup>。支持度的表达如下:

$$\text{supp}(\neg A) = 1 - \text{supp}(A) \quad (3)$$

$$\text{supp}(A \cup \neg B) = \text{supp}(A) - \text{supp}(A \cup B) \quad (4)$$

$$\text{supp}(\neg A \cup B) = \text{supp}(B) - \text{supp}(A \cup B) \quad (5)$$

$$\text{supp}(\neg A \cup \neg B) = 1 - \text{supp}(A) - \text{supp}(B) + \text{supp}(A \cup B) \quad (6)$$

自信度的表达如下:

$$\text{conf}(A \Rightarrow \neg B) = \frac{\text{supp}(A) - \text{supp}(A \cup B)}{\text{supp}(A)} \quad (7)$$

$$\text{conf}(\neg A \Rightarrow B) = \frac{\text{supp}(B) - \text{supp}(A \cup B)}{1 - \text{supp}(A)} \quad (8)$$

$$\text{conf}(\neg A \Rightarrow \neg B) =$$

$$\frac{1 - \text{supp}(A) - \text{supp}(B) + \text{supp}(A \cup B)}{1 - \text{supp}(A)} \quad (9)$$

负关联规则的挖掘就是查找支持度和自信度大于等于用户指定的  $\text{ms}$  和  $\text{mc}$  的这三种形式的规则。这些规则通常称作有用的负关联规则。

### 1.4 相关系数

目前已经存在许多关联规则挖掘的方法。但是在此提及的方法是一种可以同时挖掘正负关联规则的方法。

在进行正负关联规则挖掘的过程中, 发现存在着一个正关联规则  $A \Rightarrow B$  和若干个负关联规则将被挖掘。例如, 形如  $A \Rightarrow \neg B$  和  $\neg A \Rightarrow B$  的关联规则(在实际应用中形如  $\neg A \Rightarrow \neg B$  的关联规则通常不会被考虑)。为了判断所挖掘的关联规则的类型, 引入相关系数  $(\text{corr}_{A,B})^{[3,9]}$  这一概念。假设存在着  $A$  和  $B$  两个项目集, 就可以通过相关系数来发现这两个项目集之间的关联。形式如下:

$$\text{corr}_{A,B} = \frac{\text{supp}(A \cup B)}{\text{supp}(A)\text{supp}(B)} \quad (10)$$

相关系数的取值有如下三种可能:

(1)  $\text{corr}_{A,B} > 1$ , 此时说  $A$  和  $B$  正相关。也就是说有越多的  $A$  存在于某个事务中, 就可能有越多的  $B$  存在于此事务中, 反之亦然。

(2)  $\text{corr}_{A,B} = 1$ , 此时说  $A$  和  $B$  无关。也就是说  $B$  是否存在于事务中与此事务中是否包含  $A$  无关。

(3)  $\text{corr}_{A,B} < 1$ , 此时说  $A$  和  $B$  负相关。也就是说有越多的  $A$  存在于某个事务中,  $B$  存在于此事务中的概率就越小, 反之亦然。

通过定义相关系数, 可以得到如下推论:

推论1 如果项目集  $A$  和  $B$  是正相关的, 那么形如  $A \Rightarrow B$  和  $\neg A \Rightarrow \neg B$  的规则将被挖掘。

推论2 如果项目集  $A$  和  $B$  是负相关的, 那么形如  $A \Rightarrow \neg B$  和  $\neg A \Rightarrow B$  的规则将被挖掘。

### 1.5 权重相关系数

实际生活中的水平分布数据库可能在关联规则挖掘中具有不同的重要性。为了让关联规则挖掘更真实, 引入文献[3, 10, 11]中所提到的方法来决定水平分布数据库的重要性。

假设  $DB_1, DB_2, \dots, DB_n$  是水平分布数据库系统中的  $n$  个数据库。 $DB_{\text{size}(i)s}$  是  $DB_i$  中的事务数,  $DB_i$  的权重表示为:

$$\omega DB_i = \frac{DB_{\text{size}(i)s}}{\sum_{i=1}^n DB_{\text{size}(i)s}} \quad (11)$$

这样就可以得到所有数据库的权重, 分别用  $\omega_1, \omega_2, \dots, \omega_n$  表示数据库  $DB_1, DB_2, \dots, DB_n$  的权重。对一个频繁集  $X$ , 合成  $X$  的权重支持度如下:

$$\text{supp}_\omega(X) = \omega_1 \times \text{supp}_1(X) + \omega_2 \times \text{supp}_2(X) + \dots + \omega_n \times \text{supp}_n(X) \quad (12)$$

现在, 可以延伸公式(10)得到:

$$\text{corr}_\omega(A, B) = \frac{\text{supp}_\omega(A \cup B)}{\text{supp}_\omega(A) \text{supp}_\omega(B)} \quad (13)$$

其中,  $\text{supp}_\omega(A \cup B)$ ,  $\text{supp}_\omega(A)$ ,  $\text{supp}_\omega(B)$  是频繁集  $AB$ ,  $A$ ,  $B$  扩展后的支持度。同文献[3]的推论一样, 可以得到如下结论:

(1)  $\text{corr}_\omega(A, B) > 1$ , 此时说  $A$  和  $B$  正相关。也就是说有越多的  $A$  存在于某个事务中, 就可能有越多的  $B$  存在于此事务中, 反之亦然。

(2)  $\text{corr}_\omega(A, B) = 1$ , 此时说  $A$  和  $B$  无关。也就是说  $B$  是否存在于事务中与此事务中是否包含  $A$  无关。

(3)  $\text{corr}_\omega(A, B) < 1$ , 此时说  $A$  和  $B$  负相关。也就是说有越多的  $A$  存在于某个事务中,  $B$  存在于此事务中的概率就越小, 反之亦然。

## 1.6 算法设计

Algorithm: PNAR\_on Weighted CORR measure

Input:  $FS_1, FS_2, \dots, FS_n$ : the frequent itemsets of database  $DB_1, DB_2, \dots, DB_n$

$\text{minsupp}_\omega$ : the threshold of the globe support

$\text{minconf}_\omega$ : the threshold of the confidence

$\text{difminsupp}_\omega$ : different minimum support

$\text{difminconf}_\omega$ : different minimum confidence

$\omega_1, \omega_2, \dots, \omega_n$  are the weight of  $DB_1, DB_2, \dots, DB_n$

Output: PARs and NARs:

(1)  $FS \leftarrow \{FS_1, FS_2, \dots, FS_n\}$ ; PARs =  $\emptyset$ ; NARs =  $\emptyset$

(2) For each frequent itemset  $X$  in  $FS$  do

$\text{supp}_\omega(X) = \omega_1 \times \text{supp}_1(X) + \omega_2 \times \text{supp}_2(X) + \dots + \omega_n \times \text{supp}_n(X)$

If  $\text{supp}_\omega(X) < \text{minsupp}_\omega$

$FS \leftarrow FS - \{X\}$

(3)  $\text{corr}_\omega(A, B) = \frac{\text{supp}_\omega(A \cup B)}{\text{supp}_\omega(A) \text{supp}_\omega(B)}$

For each synthesized frequent itemset  $X$  in  $FS$  do  
for and itemset

$A \cup B = X$  and  $A \cap B = \emptyset$  do {

If  $\text{corr}_{\omega(A, B)} > 1$  then {

If  $\text{conf}_\omega(A \Rightarrow B) \geq \text{minconf}_\omega$  then

PARs = PARs  $\cup \{A \Rightarrow B\}$ ;

If  $\text{conf}_\omega(\neg A \Rightarrow \neg B) \geq \text{minconf}_\omega$  then

NARs = NARs  $\cup \{\neg A \Rightarrow \neg B\}$ ;

}

If  $\text{corr}_{\omega(A, B)} < 1$  then {

If  $\text{conf}_\omega(A \Rightarrow \neg B) \geq \text{difminconf}_\omega$

and  $\text{supp}_\omega(A \Rightarrow \neg B) \geq \text{difminsupp}_\omega$  then

NARs = NARs  $\cup \{A \Rightarrow \neg B\}$ ;

If  $\text{conf}_\omega(\neg A \Rightarrow B) \geq \text{minconf}_\omega$

and  $\text{supp}_\omega(\neg A \Rightarrow B) \geq \text{difminsupp}_\omega$  then

NARs = NARs  $\cup \{\neg A \Rightarrow B\}$ ;

}

}

Output PARs and NARs .

## 1.7 试验

为了说明算法是有效的, 进行如下试验:

假设有 3 个数据库  $D_1, D_2, D_3$ :

$D_1 = \{(A, C, D); (B, C)\}$

$D_2 = \{(A, B, F); (A, C); (A, C, D)\}$

$D_3 = \{(A, B, E,); (E); (B, C, F); (A, D); (B, F)\}$

可以算出各个数据库的权重系数为  $\omega_1 = 0.2, \omega_2 = 0.3, \omega_3 = 0.5$ , 同时假设  $\text{minsupp}_\omega = 0.3$ ,  $\text{minconf} = 0.3$ ,  $\text{difminsupp} = 0.4$ ,  $\text{difminconf} = 0.6$ 。利用各数据库的权重系数来合成通过 Aprior 算法得到的频繁集得到  $FS = \{A, B, C, D, F\}$ 。再经过算法处理, 在算法的第二步得到  $FS = \{A, B, C\}$ 。在表 2 中列出了各自的  $\text{supp}_\omega$  和  $\text{corr}_\omega$ 。从表 2 中可以看出  $A$  和  $C$  之间没有关联关系,  $A$  和  $B$ ,  $B$  和  $C$  之间存在负关联关系。由于  $\text{difminsupp} = 0.4$ ,  $\text{difminconf} = 0.6$ , 从表 3 中知道只有  $A \Rightarrow \neg B$  有意义。从这个简单的例子可以看出算法是有效果的, 而且还可以通过调整  $\text{minsupp}_\omega$ ,  $\text{minconf}$ ,  $\text{difminsupp}$ ,  $\text{difminconf}$  来决定哪些项目是感兴趣的。

表 2 项目的  $\text{supp}_\omega$  和  $\text{corr}_\omega$

item	$\text{supp}_\omega$	$\text{corr}_\omega$
A	0.6	—
B	0.5	—
C	0.5	—
AB	0.2	0.7
AC	0.3	1.0
BC	0.2	0.8

表 3 项目的  $\text{conf}_\omega$  和  $\text{supp}_\omega$

item	$\text{conf}_\omega$	$\text{supp}_\omega$
$A \neg B$	0.67	0.4
$\neg AB$	0.75	0.3
$\neg BC$	0.6	0.3
$B \neg C$	0.6	0.3

## 2 结束语

随着科学技术的发展以及计算机化管理的普遍应

用,各领域用户都拥有自己的数据库。而随着经济的全球化,很多企业的数据库都被设计为分布式的。而且这些数据库都存储了大量的业务信息数据需要他们通过分析了解这些数据的关联,来为他们的战略抉择和政策制定服务。而直接在这些庞大的数据中寻找数据之间的关联不是一件容易的事。而且,在这些大量的数据中,不是所有的数据都是分析中所需要的。为了让分析更有效、更迅速,在此修改扩展了文献[4]的算法来挖掘水平分布数据库系统中的正负关联规则。另外,考虑到实际应用中数据库系统中的数据库在我们的关联规则挖掘中可能具有不同的重要性。还引入了文献[3]中的方法来改进算法。通过试验,可以看出算法是有效的。

#### 参考文献:

- [1] Agrawal R, Imielinski T, Swami A N. Mining association rules between sets of items in large databases[C]//Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. Washington, DC: [s. n.], 1993: 207-216.
  - [2] 刘兴涛, 石冰. 挖掘关联规则中 Apriori 算法的一种改进[J]. 山东大学学报, 2008(11): 1-5.
  - [3] Ramkumar T, Srinivasan R. Modified algorithms for synthesizing high-frequency rules from different data sources[M]. London: Springer-Verlag, 2008: 313-334.
  - [4] Zhu Honglei, Xu Zhigang. An Effective Algorithm for Mining Positive and Negative Association Rules[C]//Proceedings of the 2008 International Conference on Computer Science and Software Engineering. Washington, DC, USA: IEEE Computer Society, 2008: 455-458.
  - [5] 李志云, 周国祥. 一种基于 MFP 树的快速关联规则挖掘算法[J]. 计算机技术与发展, 2007, 17(6): 94-96.
  - [6] Dong X, Wang S, Song H, et al. Study on Negative Association Rules[J]. Transactions of Beijing Institute of Technology, 2004, 24(11): 978-981.
  - [7] Yang Jingrong, Zhao Chunyu. Study on the Data Mining Algorithm Based on Positive and Negative Association Rules[J]. Computer and Information Science, 2009(2): 103-106.
  - [8] Cornelis C, Yan Peng, Zhang Xing. Mining Positive and Negative Association Rules from Large Databases[J]. Computer and Information Science, 2006(6): 1-6.
  - [9] 张毅驰, 朱巧明. 改进的关联规则算法及其应用[J]. 计算机系统应用, 2007(10): 80-84.
  - [10] Jiang He, Zhao Yuanyuan, Dong Xiangjun. Mining Positive and Negative Weighted Association Rules from Frequent Itemsets Based on Interest [C]//2008 International Symposium on Computational Intelligence and Design. [s. l.]: [s. n.], 2008: 242-245.
  - [11] Dong X, Song H, Jiang H, et al. Minimum Interestingness Based on Method for Discovering Positive and Negative Association Rules[J]. Computer Project and Application, 2004, 40(27): 24-31.
- 
- (上接第 112 页)
- of the 11th Int. Conf. on Data Engineering. Taipei: [s. n.], 1995: 3-14.
  - [2] Pei Jian, Han Jiawei. Mining Sequential Patterns by Pattern-growth: The PrefixSpan Approach[J]. IEEE Transactions on Knowledge and Data Engineering, 2004, 6(10): 1-17.
  - [3] Cheung D W, Han J, Ng V T. A fast distributed algorithm for mining association rules[C]//Proceedings of the 4th International Conference on Parallel and Distributed Information Systems. Los Alamitos Cal, USA: [s. n.], 1996: 31-44.
  - [4] Guralnik V, Garg N, Karypis G. Parallel tree projection algorithm for sequence mining[J]. Lecture Notes in Computer Science, 2001, 2150: 310-320.
  - [5] Savasere A, Omiecinski E, Navathe S. An efficient algorithm for mining association rules in large database[C]//Proc. 21st Int. Conf. Very Large Data Bases. Switzerland: [s. n.], 1995.
  - [6] Chen Tzung-Shi, Hsua Shih-Chun. Mining frequent tree-like patterns in large datasets[J]. Data & Knowledge Engineering, 2007, 62(1): 65-83.
  - [7] Yun U. A new framework for detecting weighted sequential patterns in large sequence databases[J]. Knowledge-Based Systems, 2008, 21(2): 110-122.
  - [8] Amador J J. Sequential clustering by statistical methodology[J]. Pattern Recognition Letters, 2005, 26: 2152-2163.
  - [9] Kuo R J, Chao C M, Liu C Y. Integration of K-means algorithm and AprioriSome algorithm for fuzzy sequential pattern mining[J]. Applied Soft Computing, 2009, 9(1): 85-93.
  - [10] Wu Shaochun, Wu Gengfeng, Jin Shenjie. Pre-clustering based sequential pattern mining [C]//Proceedings of the Fourth International Conference on Computer and Information Technology. [s. l.]: [s. n.], 2004: 1008-1013.
  - [11] Li Yanjun, Chung Soon. Parallel bisecting k-means with prediction clustering algorithm[J]. The Journal of Supercomputing, 2007, 39: 19-37.
  - [12] Nguyen S, Orłowska M. A Partition-Based Approach for Sequential Patterns Mining[C]//2007 IEEE International Conference on Research, Innovation and Vision for the Future. [s. l.]: [s. n.], 2007: 200-205.