综 合 述 评

关联规则挖掘算法综述

毕建欣，张岐山

（福州大学管理学院，福州 350002）

[摘要] 介绍了关联规则挖掘算法的基本原理，并按照挖掘中涉及到的变量数目（维数）、数据的抽象层次和处理变量的类别（布尔型和数值型），依次对关联规则挖掘算法的研究进行综述，并对一些典型的算法进行分析和比较，最后展望了关联规则挖掘算法的研究方向。

[关键词] 数据挖掘；关联规则；算法；综述

[中图分类号] TP311 [文献标识码] A [文章编号] 1009－1742（2005）04－0088－07

1 引言

数据挖掘是指从大型数据库或数据仓库中提取隐含的、先前未知的、对决策有潜在价值的知识和规则。它是人工智能和数据库发展相结合的产物，是国际上数据库和信息决策系统最前沿的研究方向之一。数据挖掘主要的算法有分类模式、关联规则、决策树、序列模式、聚类模式分析、神经网络算法等等。关联规则是数据挖掘领域中的一个非常重要的研究课题，广泛应用于各个领域，既可以检验行业内长期形成的知识模式，也能够发现隐藏的新规律。有效地发现、理解、运用关联规则是完成数据挖掘任务的重要手段，因此对关联规则的研究具有重要的理论价值和现实意义。

R. Agrawal 等人[1]于 1993 年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，其核心方法是基于频集理论的递推方法。此后人们对关联规则的挖掘问题进行了大量研究，包括对 Apriori 算法优化[2~19]、多层次关联规则算法[7, 20]、多值属性关联规则算法[21, 22]、其他关联规则算法[23~33]等，以提高算法挖掘规则的效率。

2 关联规则基本原理

设 $I=\{i_1, i_2, \cdots, i_m\}$ 是 m 个不同的项目组成的集合，给定一个事务数据库 D，其中的每一个事务 T 是 I 中一组项目的集合，即 $T\subset I$，T 有一个唯一的标志符 TID。若项集 $X\subset I$ 且 $X\subset T$，则事务集 T 包含项集 X。一条关联规则就是形如 $X\Rightarrow Y$ 的蕴涵式，其中 $X\subset I$，$Y\subset I$，$X\cap Y=\varnothing$。关联规则 $X\Rightarrow Y$ 成立的条件：**a.** 它具有支持度 s，即事务数据库 $\boldsymbol{D}$ 中至少有 s % 的事务包含 $\boldsymbol{X\cup Y}$。**b.** 它具有置信度 c，即在事务数据库 D 中包含 X 的事务至少有 c % 同时也包含 Y。

关联规则挖掘问题就是在事务数据库 D 中找出具有用户给定的最小支持度 minsup 和最小置信度 minconf 的关联规则。关联规则挖掘问题可以分解为以下 2 个子问题[1, 2]。

1）找出存在与事务数据库中的所有强项集 X 的支持度 support（X）不小于用户给定的最小支持度 minsup，则称 X 为强项集（large itemset）。

2）利用强项集生成关联规则。对于每个强项集 A，若 $B\subset A$，$B\neq\varnothing$，且 support（A）/support（B）$\geqslant$minconf，则有关联规则 $B\Rightarrow(A-B)$。

[收稿日期] 2004－04－24；修回日期 2004－06－02

[基金项目] 福建省自然科学基金资助项目（A0210013）；福建省教育厅资助项目（JA03006）

[作者简介] 毕建欣（1974－），女，吉林九台市人，福州大学硕士研究生

第 2 个子问题比较容易，其生成算法可参考文献［2］。目前大多数研究集中在第 1 个子问题上。

3　关联规则算法概述及典型算法分析

R. Agrawal 等提出了关联规则挖掘问题以后，一批有效的挖掘关联规则的算法在过去几年中得到了长足的发展。到目前为止，其主要研究方向有：基于规则中涉及到的数据维数的挖掘算法，基于规则中数据的抽象层次的挖掘算法，基于规则中处理变量类别的挖掘算法，其他关联规则算法等。

3.1　基于规则中涉及到的数据维数的挖掘算法

按照关联规则中涉及到的变量数目可以把关联规则分为单维关联规则和多维关联规则。单维关联规则只涉及数据的一个维度（即一个变量）；而多维关联规则要处理多维数据，涉及多个变量。

3.1.1　单维关联规则

1）经典频集方法

R. Agrawal 等提出的 AIS[1]，Apriori[2]算法。

在算法 AIS 中，候选强项集是在扫描数据库的过程中产生，即在对数据库进行第 k 次扫描时，候选强项集（其中每一个元素的元素个数不一定是 k 个，可以大于 k）是由第 $k-1$ 次扫描所产生的边界集（frontier set）通过增加当前事务中的项得到，同时计算候选强项集中的元素支持数，直到某一次扫描所产生的边界集为空时停止运算，第 k 次扫描所产生的边界要大于本次扫描生成的强项集，该算法的缺点在于生成的候选强项集太大。

算法 Apriori 利用“在给定的事务数据库 D 中，任意强项集的子集都是强项集；任意弱项集的超集都是弱项集”这一原理对事务数据库进行多次扫描，第一次扫描得出大 1－项集 L_1，第 k（$k>1$）次扫描前先利用第 $k-1$ 次扫描的结果（即大 $k-1$ 项集 L_{k-1}）和函数 Apriori-gen 产生候选大 k－项集 C_k，然后在扫描过程中确定 C_k 中每个元素的支持数，最后在每次扫描结束时计算出大 k－项集 L_k，算法在当候选大 k－项集 C_k 为空时结束。该算法所产生的候选强项集比算法 AIS 小得多，提高了算法效率。

2）DHP 算法

J. S. Park 等[3]提出的 DHP 算法，是利用哈西（Hashing）技术有效地改进了候选强项集的生成过程，产生了比前述算法更小的候选强项集（对大 2－候选集尤为明显），同时也缩减了事务数据库的大小，减小了 I/O 操作时间，其效率比算法 Apriori 有明显提高。

3）频集算法的优化方法

虽然 Apriori 算法自身已经进行了一定的优化，但是在实际的应用中，还是存在不令人满意的地方，于是人们相继提出了一些优化的方法。

a. 基于划分的方法　Savasere 等[4]设计了一个基于划分（partition）的算法。该算法分为两部分：在第一部分中，算法首先将要在其中发现关联规则的事务数据库 D 分为 n 个互不相交的事务数据库的$D^1, D^2, \cdots, D^n$，$D^i(i=1,2,\cdots,n)$的大小要求能够容纳在内存之中，然后将每个分事务数据库 D^i（$i=1,2,\cdots,n$）读入内存并发现其中的强项集 L^i，最后在第一部分结束时将所有分事务数据库的强项集合并成为一个在事务数据库 D 中的潜在强项集 $L_p=\bigcup_{i=1}^{n} L^i$；算法第二部分计算潜在强项集 L_p 在事务数据库 D 中的支持数，并得出强项集 L。该算法只对事务数据库 D 扫描 2 次，大大减少了 I/O 操作，从而提高了算法效率。

b. 基于采样的方法　基于对数据集前一遍扫描得到的信息，对此仔细地做组合分析，可以得到一个改进的算法。Mannila 等[5]先考虑了这一点，他们认为采样是发现规则的一个有效途径。随后又由 Toivonen 进一步发展了这个思想，先使用从数据库中抽取出来的采样得到一些在整个数据库中可能成立的规则，然后对数据库的剩余部分验证这个结果。Toivonen 的算法相当简单并显著地减少了 I/O 代价，但是一个很大的缺点是产生的结果不精确，即存在所谓的数据扭曲（data skew）。分布在同一页面上的数据时常是高度相关的，可能不能表示整个数据库中模式的分布，由此导致采样 5 % 的交易数据所花费的代价可能同扫描一遍数据库相近。Lin 和 Dunham 讨论了反扭曲（anti-skew）算法来挖掘关联规则，在那里他们引入的技术使扫描数据库的次数少于 2 次，算法使用了采样处理来收集有关数据的次数，以减少扫描遍数。

Brin 等[6]提出的算法使用比传统算法少的扫描遍数来发现频集，同时比基于采样的方法使用更少的候选集，改进了算法在低层的运行效率。具体的考虑是，在计算 k 项集时，一旦认为某个（$k+1$）项集可能是频集时，就并行地计算（$k+1$）项集的支持度，算法需要的总的扫描次数通常少于最大的频集的项数。这里，基于他们的工作基础也使

用了杂凑技术，并提出产生“相关规则”(correlation rules)的一个新方法。

c. 减少交易的个数 R. Agrawal 等人提出的 AprioriTid 和 AprioriHybrid[2] 算法。算法 AprioriTid 除了具有 Apriori 算法的特点外，还有另外一个特点，即仅在第一次扫描时用事务数据库 D 计算候选强项集的支持数，其他各次扫描用其上一次扫描生成的候选数据库 D' 来计算候选强项集的支持数。在最后的几次扫描中，D' 的大小要远远小于 D，减少 I/O 操作时间，提高了算法效率。算法 AprioriHybrid 是算法 Apriori 与算法 AprioriTid 的结合，当候选事务数据库 D' 不能完全容纳于内存时用算法 Apriori，当内存能够完全容纳候选事务数据库 D' 时，则用算法 AprioriTid。

d. 基于兴趣度的关联规则挖掘算法 现有的关联规则挖掘算法主要考虑可信度和支持度的阈值，如经典的 Apriori 算法和 DHP 算法。然而过去的一些应用发现，从一个数据库中很容易产生大量规则，但是其中的大部分对用户来说可能是不感兴趣的或者是没用的，甚至还可能引起误导。文献[7]给出了感兴趣规则的定义（R-interesting），文献[8]对感兴趣的规则定义做了改进，文献[9]定义了否定关联规则的兴趣度，文献[10]中提出一个基于差异思想的兴趣度定义，并提出了改进的定义以及使用兴趣度后对挖掘算法的修改等。

e. 基于约束的规则挖掘 基于约束的关联规则的主要目的是发现更有趣的、更实用的和更特别的关联规则。文献[11]研究了在提供布尔表达式约束情况下的关联规则发现问题。布尔表达式约束允许用户指定他所感兴趣的关联规则的集合，这种约束不仅可以对事务数据库进行预加工，而且可以把它集成在挖掘算法内部，从而提高算法的执行效率。根据这种继承方式的不同给出了3种不同的算法：MultipleJoins，Recorder，Direct。文献[12]提出并分析了用户所给出的2个对发现算法的剪枝步骤非常重要的属性：反单调性（anti-monotonicity）和简洁性（succinctness），提出了一个高效的基于约束的关联规则的挖掘算法 CAP。另一种类型的基于约束的关联规则挖掘方法是元模式制导的关联规则挖掘算法[13]。这种类型的发现算法首先由用户给定要发现的关联规则的元模式或模板，然后根据这种模式提出了2个相应的算法：大谓词增长算法（large predicate-growing）和直接P谓词测试算法（direct P-predicate testing）。

综上所述，基于约束的规则挖掘[14]约束的内容可以是：**a.** 数据约束。用户可以指定对哪些数据进行挖掘，而不一定是全部的数据。**b.** 指定挖掘的维和层次。用户可以指定对数据哪些维以及这些维上的哪些层次进行挖掘。**c.** 规则约束。可以指定哪些类型的规则是所需要的。引入一个模板(template)的概念，用户使用它来确定哪些规则是令人感兴趣的：如果一条规则匹配一个包含的模板(inclusive template)，则是令人感兴趣的，然而如果一条规则匹配一个限制的模板（rextrictive template），则被认为是缺乏兴趣的。

对于基于 Apriori 的频集方法，即使进行了优化，但一些固有的缺陷还是无法克服。Apriori 的算法及其优化算法可能产生大量的候选集。当长度为1的频集有10000个的时候，长度为2的候选集就会成指数倍增长，其候选集个数将会超过 10^7。如果要生成一个很长的规则时，要产生的中间元素也是巨大量的。对此可采用 FP 树算法解决。

FP 树算法采用了一种 FP-growth 的方法[15]。它采用了分而治之的策略：在对数据库进行第一次扫描后，把找到的频集压缩进一棵频繁模式树(FP-tree)，同时依然保留其中的关联信息。随后再将 FP-tree 分化成一些条件库，每个库和一个长度为1的频集相关。然后再对这些条件库分别进行挖掘。当原始数据量很大的时候，也可以结合划分的方法，使得一个 FP-tree 可以放入主存中。实验表明，FP-growth 对不同长度的规则都有很好的适应性，同时在效率上比 Apriori 算法有很大的提高。

4）增量式关联规则挖掘算法

关联规则增量更新主要有3个问题：**a.** 在给定的最小支持度和最小置信度下，当一个新的数据集 db 添加到旧的数据库 DB 中时，如何生成 db∪DB 中的关联规则。**b.** 在给定的最小支持度和最小置信度下，当一个数据集 db 从旧的数据库 DB 中删除时，如何生成 DB∪db 中的关联规则。**c.** 给定数据库 DB，在最小支持度和最小置信度发生变化时，如何生成数据库 DB 中的关联规则。

所有的关联规则更新问题都可以归结为以上3种情况或它们的组合。解决关联规则更新问题的最直观和最简单的方法就是运用挖掘算法对更新后的数据库重新进行挖掘。这种方法虽然实现简单，但是它没有充分利用已经得到的结果，效率较低。

D. W. Cheung 等[16]首先考虑了关联规则的高效更新问题，提出的增量式更新算法 FUP，FUP 的基本框架和 Apriori 是一致的。算法 FUP2 考虑了关联规则更新的第 1 个和第 2 个问题，给出了一个解决这 2 个问题较通用的算法。DELI 算法用来估计某一数据库发生变化前与发生变化后的关联规则变化的大小，若变化较大，则需要进行关联规则的更新；若变化较小，则认为不用进行关联规则更新，从而避免盲目的更新。冯玉才等[17]针对关联规则更新的第 3 个问题进行了研究，设计出了相应的 IUA 和 PIUA 算法。

算法 IUA 采用了一个独特的候选强项集生成算法 IUA－GEN，在每一次对数据库 DB 扫描之前生成较小的候选强项集，从而提高了算法的效率。它也要求上一次对数据库进行挖掘时发现的强项集 $L=\bigcup_{i=1}^{n}L_i$（n 为 L 中最大元素的元素个数）在本次挖掘时是可以得到的。因为人们在发现关联规则时，常常需要不断地调整最小支持度和最小可信度来聚集到那些真正令其感兴趣的关联规则上，因而该算法具有很重要的意义；在 IUA 算法中，将所有的频繁 k 项目集分成了互不相交的 3 类，这使得 IUA 算法能够很容易实现基于共享内存（shared-memory）多处理机结构的并行化，即 PIUA 算法。事实上，像 PIUA 这样的基于共享内存多处理机结构的并行算法特别有利于在限时应用中用来加快单个大顺序算法的计算。

单维关联规则算法除了以上介绍的之外，还有 J. Roberto 等[18]提出了一种在数据库中有效的挖掘长模式的 Max-Miner 算法。Max-Miner 算法采取了一种发现最长频集的策略：先从数据库中找出所有 1－项频集，由 1－项频繁项逐步形成 set-eumeration 树，然后采用宽度优先搜索策略，同时根据最小支持度进行候选集的剪枝工作，产生并生成候选集。R. Wille[19]于 1999 年首先提出基于概念格的挖掘算法，提供了将数据库中蕴涵的知识形式化成有用概念的一种有效工具。

3.1.2　*多维关联规则挖掘*　它指关联规则涉及 2 个或 2 个以上变量。根据是否允许同一个维重复出现，多维关联规则又可以细分为维间关联规则（不允许维重复出现）和混合维关联规则（允许维在规则的左右同时出现）。比如“年龄 20 至 30，喜欢郊游→喜欢游泳”就是混合维关联规则。维间关联规则和混合维关联规则的挖掘还要考虑不同的字段种类，即类别数据与数值数据。对于类别资料，一般关联规则算法都可以处理，而对数值型资料，就需要将这些资料转换成类别资料才可以处理。

处理数值型字段的方法基本上有以下几种：

1）数值字段被分成一些预定义的层次结构。这些区间都是由用户预先定义的。得出的规则也叫做静态数量关联规则。

2）数值字段根据数据的分布分成了一些布尔字段。每个布尔字段都表示一个数值字段的区间，落在其中则为 1，反之为 0。这种分法是动态的。得出的规则叫布尔数量关联规则。

3）数值字段被分成一些能体现它含义的区间。它考虑了数据之间的距离因素。得出的规则称基于距离的关联规则。

4）直接用数值字段中的原始数据进行分析。使用一些统计的方法对数值字段的值进行分析，并且结合多层关联规则的概念，在多个层次之间进行比较从而得出一些有用的规则。得出的规则称为多层数量关联规则。

3.2　基于规则中数据的抽象层次的挖掘算法

基于要挖掘的数据库中的概念层次和发现单一概念层次中的关联规则的算法，学者们提出了许多高效发现一般或多层关联规则的算法，主要有：Han 等的 ML－T2L1 及其变种 ML－T1LA，ML－TML1，ML－T2LA[20] 和 R. Srikant 等的 Coumulate，Stratify 及其变种 Estimate，EstMerge[7]等。

算法 ML－T2L1 的基本思想是首先根据要发现的任务从原事务数据库生成一个根据概念层次信息进行编码的事务数据库，利用这个具有概念层次信息的新生成的数据库，自顶向下逐层递近地在不同层次发现相应的关联规则。它实际上是算法 Apriori 在多概念层次环境中的扩展。根据在发现高层关联规则过程中所用的数据结构和所生成的中间结果的共享方式不同，算法 ML－T2L1 有三个变种：ML－T1LA，ML－TML1，ML－T2LA。

算法 Coumulate 的基本思想与 Apriori 完全一样，只是在扫描到事务数据库某一事务时，将此事务中所有项的祖先加入到本事务中。并加入 3 个优化：**a.** 对加入到事务中的祖先进行过滤。**b.** 预先计算概念关系 T 中的每一个项的祖先，得到项集与其祖先的对照表 T^*。**c.** 对既包含项集 X 又包

含 X 的祖先的项集进行剪枝。

算法 Stratify 基于“若项集 X 的父辈不是强项集，则 X 肯定不会是强项集”的事实进行设计。其基本思想为：在概念层次有向非循环图中，定义没有父辈的项集 X 的深度 depth $(X)=0$，其他项集的深度为：（max（{depth (X') /X'是 X 的父辈}）+1）。算法要对事务数据库进行多遍扫描，第 k（$k\geqslant 0$）次扫描计算深度为 k（$k\geqslant 0$）的所有项集 C_k 的支持数，并得出深度为 k（$k\geqslant 0$）的大项集 L_k。在第 k（$k\geqslant 1$）次扫描前，对 C_k 进行剪枝，即删除 C_k 中那些祖先包含在 $C_{k-1}-L_{k-1}$ 中的元素。围绕着决定某些深度较大的项集是否是强项集问题，文献［7］用抽样技术对算法 Stratify 进行扩展，形成算法 Estimate 和 EstMerge。

3.3 基于规则中处理变量类别的挖掘算法

根据变量的类别，关联规则分为布尔型关联规则和多值属性关联规则。多值属性又可分为数量属性(quantitative attribute)（如年龄，价格等）和类别属性(categorical attribute)（如品牌，制造商等）。

基于支持信任理论的关联规则挖掘布尔型描述的数据已经比较成熟。多值属性关联规则挖掘问题在文献［21］中首先提出。文献［21］中发现的多值属性关联规则的形式为：$x=q_x \Rightarrow y=q_y$，其前件和后件对应的都是单一的数值，而不是区间，所提出的算法比较简单，但当需要发现所有属性之间的关联规则时，将遇到属性组合的爆炸问题。

目前提出的挖掘多值属性关联规则的算法大多是将多值属性关联规则挖掘问题转化为布尔型关联规则挖掘问题，即将多值属性的值划分为多个区间，每个区间作为一个属性，将类别属性的每一个类别当作一个属性。然后针对这些属性应用布尔关联规则挖掘算法。

如何划分区段是实现多值关联规则挖掘到布尔型关联规则挖掘转变的关键。其中有 2 个互相牵制的问题：当区段的范围太窄时，则可能使每个区段对应的属性支持度很低，出现“最小支持度问题”；当区段的范围太宽时，则可能使每个区段对应的属性可信度很低，出现“最小可信度问题”。

给定一个数据库 D，多值关联规则挖掘问题就是发现所有支持度和置信度分别大于等于用户给定的最小支持度 minsup 和最小置信度 minconf 的多值关联规则。关联规则中的项目可以是数值或类别。

多值关联规则挖掘问题一般按照以下步骤完成：

1）在每个数值属性 x 的值域 $[l, u]$ 上确定划分的区间数及分割点，确定与属性 x 相关的原子集合 split (x)；

2）将每个 $\langle x, l_k, u_k\rangle \in$ split (x) 映射为一个逻辑属性 A，进而将所有数值属性及其值域区间映射为项目集；

3）产生支持度大于等于最小支持度 minsup 的频繁项集；

4）利用频繁项集产生置信度大于最小置信度 minconf 的关联规则；

5）从所有产生的规则中确定出有趣的规则。

文献［22］介绍将数值数据映射到二维空间，利用基于密度分布函数的聚类分析方法将数值属性区间分段，并在此基础上挖掘容易理解并且具有概括性和有效的数值属性关联规则。

3.4 其他关联规则算法

除了以上列举的关联规则算法之外，还有一些研究方向，如：发现关联规则的语言[23]，图像的关联规则发现算法[24]，加权关联规则算法[25, 26]，挖掘相关性和因果关系[27]，演变数据的动态的关联规则挖掘[28]，生成关联规则不同的衡量标准的算法研究[29, 30]，并行发现算法[31~33]等等。

4 总结与展望

目前，数据库关联规则挖掘已经取得了令人瞩目的成绩，但对下列问题进行研究将是具有挑战性的工作。

1）*开发更高效的挖掘算法* 随着数据库的尺寸不断增大，不仅增大了挖掘算法的搜索空间，而且也增加了盲目发现的可能性。因此必须利用领域知识去提取与发现任务有关的数据，删除无用的数据，有效地降低问题的维数，设计出更加有效的挖掘算法。在这方面，基于约束的关联规则挖掘具有广阔的前途。

2）*可视化挖掘* 设计一个灵活方便的用户界面，允许用户与挖掘系统进行交互，并对所挖掘的结果进行很好的可视化表示，使非领域专家也能进行挖掘。

3）*各种非结构化数据的挖掘* 目前大多数关联规则挖掘大多是基于关系数据库或事务数据库的算法，设计应用于其他类型数据库（如面向对象数

据库、数据仓库、文本数据、图形图像数据、多媒体数据等）关联规则挖掘算法也将是十分有意义的工作。

4）*并行关联规则数据挖掘*　随着数据挖掘中数据量的高速增长以及大规模并行计算在数据挖掘中的应用，由于挖掘系统本身的原因，并行数据挖掘过程更加趋向粗粒度的挖掘，无法实现任意程度的并行。目前在并行数据挖掘中尚有一些问题需要解决：数据量的不断增长，维数越来越高，数据定位问题，数据的不对称性，动态负载平衡，多表数据库的数据分布和索引方案，增量的方法，并行的数据库管理系统与文件系统。

5）*制定更为合理的关联规则衡量评价标准*　目前的关联规则衡量标准可能会发现一些冗余的、虚假的和非挖掘者关心的关联规则，因而有必要制定一些新的衡量标准，用来衡量关联规则挖掘算法的优劣，但这些标准的制定可能要具体问题具体分析。

6）*与其他系统的集成*　这里的集成包括与其他挖掘方法的集成和与其他系统（如专家系统、决策支持系统等）的集成。

7）*研究在网络环境下的关联规则挖掘技术*　特别是在 Internet 上建立 DM 服务器，与数据库服务器配合，实现数据挖掘。

参考文献

[1] Agrawal R, Srikant R. Mining association rules between sets of items in large databases [A]. Proc ACM SIGMOD Int'l Conf Management of data [C]. Washington DC, May 1993. 207～216

[2] Agrawal R, Srikant R. Fast algorithms for mining association rules [A]. Proc 20th Int'l Conf Very Large Database [C]. Santiago, Chile, Sept 1994. 487～499

[3] Park J S, Chen M S, Yu P S. An effective hash-based algorithm for mining association rules [A]. Proceedings of ACM SIGMOD International Conference On Management of Data [C]. San Jose, CA, May 1995. 175～186

[4] Savasere A, Omiecinski E, Navathe S. An efficient algorithm for mining association rules in large databases [A]. Proc of the 21th International Conference on Very Large Database [C]. Zurich, Switzerland, Sept 1995. 432～443

[5] Mannila H, Toivonen H, Verkamo A. Efficient algorithm for discovering association rules [A]. AAAI Workshop on Knowledge Discovery in Databases [C], 1994. 181～192

[6] Brin S, Motwani R, Silverstein C. Beyond market baskets generalizing association rules to correlations [A]. Proc of the 1997 ACM SIGMOD Int'l Conf on Management Of Data [C]. Tucson, Arizona, UAS: ACM Press, 1997. 265～276

[7] Srikant R, Agrawal R. Mining generalized association rules [A]. Proceedings of the 21th International Conference on Very Large Databases [C]. Zurich, Switzerland, Sept 1995. 407～419

[8] Srikant R, Agrawal R. Mining quantitative association rules in large relational tables [A]. Proc of the 1996 ACM SIGMOD Int'l Conf on Management Of Data [M]. Montreal, Quebee, Canada: ACM Press, 1996. 1～12

[9] Savasere A, Omiecinski E, Navathe S B. Mining for strong negative associations in a large database of customer transactions [A]. Proc of the 14th Int'l Conf on Data Engineering [M]. Orlando, Florida, USA: IEEE Computer Society Press, 1998. 494～502

[10] 周　欣，沙朝锋，朱扬勇，等. 兴趣度—关联规则的另一个阈值 [J]. 计算机研究与发展，2000，37(5)：627～633

[11] Srikant R, Agrawal R. Mining association rules with item constrains [A]. Proc of the 3rd Int'l Conference on Knowledge Discovery in Data Bases and Data Mining [C]. Newport Beach, California, August 1997. 67～73

[12] Ng R, Lakshmanan L V S, Han J, et al. Exploratory mining and pruning optimizations of constrained associations rules [A]. Proceedings of ACM SIGMOD International Conference on Management of Data [C]. Seattle, Washington, June 1998. 13～24

[13] Fu Y, Han J. Meta-rule-guided mining of association rules in relational databases [A]. Proc 1995 Int'l Workshop on Knowledge Discovery and Deductive and Object-Oriented Databases (KDOOD'95) [C]. Singapore, December 1995. 39～46

[14] Park J S, Chen M S, Yu P S. An effective hash-based algorithm for mining association rules [A]. Proceedings of ACM SIGMOD International Conference on Management of Data [C]. San Jose, CA, May 1995. 175～186

[15] Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation [A]. Proceedings of the ACM SIGMOD Internal Conference on Management of Data [M]. Dalas, Texas: ACM Press, 2000. 1～12

[16] Cheung D W, Han J, Ng R. Maintenance of discovered association rules in large databases: An incremental updating technique [A]. Proceedings of the 21th International Conference on Data Engineering [C]. New Orleans Louisiana, 1995. 106～114

[17] 冯玉才，冯剑琳. 关联规则的增量式更新算法 [J]. 软件学报，1998，9 (4)：301～306

[18] Roberto J, Bayardo Jr. Efficiently mining long patterns from Database [A]. Proc of the 1998 ACM SIGMOD Int' l Conf on Management of Data [C]. 1998. 85～93

[19] Ganter B, Wille R. Formal Concept Analysis: Mathematical Foundations [M]. Berlin: Springer 1999. 131～139

[20] Han J, Fu F. Discovery of multople-level association rules from large databases [A]. Proc of the 21th International Conference on Very Large Databases [C]. Zurich, Switzerland, Sept 1995. 420～431

[21] Shapiro G P. Discover, analysis, and presentation of strong rules [A]. Shapiro G P, Frawley W J. Knowledge Discovery in Database [M]. AAAI/MIT Press, 1991. 229～248

[22] 尹阿东，高学东，武 森，等. 基于数值属性的关联规则挖掘算法 [J]. 微机发展，2003，(4)：67～70

[23] Meo R, Psaila G, Ceri S. A new SQL-like operator for mining association rules [A]. Proc of the 22th Int' l Conf on Very Large Database [C]. Bombay, India, 1996. 122～133

[24] 颜雪松，蔡之华. 一种基于图像的关联规则发现算法的研究 [J]. 计算机工程与应用，2003，(2)：209～211

[25] Cai C H, Fu W C, Cheng C H, et al. Mining association rules with weighted items [A]. IEEE Int' l Database Engineering and Applications Symposium [C], Cardiff, 1998

[26] 陆建江. 加权关联规则挖掘算法的研究 [J]. 计算机研究与发展，2002，(10)：1281～1286

[27] Silverstein C, Brin S, Morwani R, et al. Scalable rechniques for mining causal strucrures [A]. Proc 1998 Int Conf Very Large Data Bases [C], New York, August 1998. 594～605

[28] 齐 雁，李石君，薛海峰. 对演变数据进行关联规则挖掘的新方法 [J]. 计算机工程，2002，(11)：126～128

[29] 罗 可，吴 杰. 关联规则衡量标准的研究 [J]. 控制与决策，2003，(5)：277～281

[30] 杨建林，邓三鸿，苏新宁. 关联规则兴趣度的衡量 [J]. 情报学报，2003，(8)：419～424

[31] Agrawal R. Parallel mining of association rules [J]. IEEE Transactions on Knowledge and Data Engineering, 1996, 8 (6): 926～969

[32] Park J S, Chen M S, Yu P S, et al. Efficient parallel data mining for association rules [A]. Proc Fourth Int' l Conf Information and Knowledge Management [C]. Baltimore, Nov 1995

[33] Cheung D W. Efficient mining of association rules in distributed databases [J]. IEEE Transactions on Knowledge and Data Engineering, 1996, 8 (6): 910～921

Survey of the Algorithms on Association Rule Mining

Bi Jianxin, Zhang Qishan

(*School of Management, Fuzhou University, Fuzhou* 350002, *China*)

[**Abstract**] In this paper the principle of the algorithms on association rule mining is introduced firstly, and researches of the algorithms on association rule mining are summarized in turn according to variable (dimension), abstract levels data and types of transacted variable (Boolean and Quantitative) in the process of data mining. At the same time some typical algorithms are analyzed and compared. At last, some future directions on association rule generation are viewed.

[**Key words**] data mining; association rule; algorithms; survey