文章编号:1001 - 9081(2009)05 - 1405 - 04

基于粗糙集的多维关联规则挖掘方法

陶多秀[1],吕跃进[2],邓春燕[2,3]
(1. 广西大学 电气工程学院, 南宁 530004; 2. 广西大学 数学与信息科学学院, 南宁 530004;
3. 广西河池学院 计算机与信息科学系,广西 宜州 546300)
(tdx220@163.com)

摘 要:海量的数据使得关联规则挖掘非常耗时,而并非所有的规则都是用户感兴趣的,应用传统的挖掘方法会挖掘出许多无关信息。此外,目前大部分算法是针对单维规则的。因此,定义了一种挖掘语言使得用户可以指定感兴趣的项以及关联规则的参数(如支持度,置信度等),并提出一种基于粗糙集理论的多维关联规则挖掘方法,动态生成频繁集和多维关联规则,减少频繁项集的生成搜索空间。实例分析验证该算法的可行性与有效性。

关键词:关联规则;多维关联规则;频繁集;粗糙集
中图分类号: TP301.6 **文献标志码**:A

Method based on rough set for mining multi-dimensional association rules

TAO Duo-xiu[1], LV Yue-jin[2], DENG Chun-yan[2,3]
(1. *College of Electrical Engineering, Guangxi University, Nanning Guangxi* 530004, *China*;
2. *College of Mathematics and Information Science, Guangxi University, Nanning Guangxi* 530004, *China*;
3. *Department of Computer and Information Science, Hechi University, Yizhou Guangxi* 546300, *China*)

Abstract: It is very time-consuming to discover association rules from the mass of data, and not all the rules are attractive to the user, so a lot of irrelevant information to the user's requirements may be generated when traditional mining methods are applied. In addition, most of the existing algorithms are for discovering one-dimensional association rules. Therefore, the authors defined a mining language which allowed users to specify items of interest to the association rules, as well as the parameters (for example, support, confidence, etc.). A method based on rough set theory for multi-dimensional association rules mining was also proposed, which dynamically generated frequent item sets and multi-dimensional association rules, and reduced the search space to generate frequent item sets. Finally, an example verifies the feasibility and effectiveness of the method.

Key words: association rule; multi-dimensional association rules; frequent item sets; rough set

0 引言

关联规则是指大量数据中项集之间有趣的关联或相关联系[1],已在许多领域得到了广泛应用。然而海量的事务数据使得关联规则挖掘非常耗时,而且并不是所有的规则都是有价值或者用户感兴趣的,用户对关联规则参数的设置需要也不尽相同,应用传统的挖掘方法会挖掘出许多与用户需要无关的信息。因此,本文定义了一种挖掘语言使得用户可以指定感兴趣的项以及规则的参数(如支持度,置信度等)。

文献[2]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及 Apriori 算法。随后人们相继对 Apriori 算法进行了一些优化,如文献[3-4]。然而 Apriori 系列方法可能产生大量的频繁集,如当长度为 1 的频繁集有 1 000 个时,长度为 2 的候选集将会超过 1 000 万。计算频繁集及规则的代价巨大。针对这一固有缺陷,文献[5]提出了不产生候选挖掘频繁项集的方法——FP-growth 方法。该方法在经过第一遍扫描之后,把数据库中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,然后再对条件库分别进行挖掘。该方法降低了搜索开销,提高了效率,但当数据库很大时,构造基于内存的 FP-tree 也是不现实的。此外,还有其他一些基于约束的挖掘频繁模式的方法也被提了出来,如文献[6]提出了一种转换约束,并对挖掘频繁模式的约束方法做了总结,分成简洁、反单调性、单调性和转换约束,将约束推进和频繁模式挖掘集成为一个统一的框架。本文则基于用户需求的约束来挖掘频繁模式。

文献[7]提出一种类 SQL 的操作用于抽取关联规则。但这种类 SQL 的操作不能完全表达特定项与其他所有项之间的关联,且执行效率不高。文献[8]介绍了一种用户友好的挖掘语言,建立一个关联图,并通过遍历这关联图生成所有的频繁集。但该方法需要大量的内存空间来存储相关信息。本文借鉴该方法的优点,定义了一种可以让用户参与到挖掘过程中的语言,使用户可以指定感兴趣的项集和规则参数,从而挖掘满足用户需要的规则。

目前,大部分关联规则挖掘算法是针对挖掘 2 值属性(即布尔值属性)的数据库或挖掘单维关联规则研究的,而多值属性或多维规则更具普遍性。单维关联规则中仅包含多次出现的单个谓词,而多维规则中涉及两个或多个谓词。虽然一些单维关联规则的方法可扩展为多维关联规则挖掘,但应用于大规模数据库存在效率低的问题[9]。

收稿日期:2008 - 11 - 17;**修回日期**:2009 - 02 - 19。
基金项目:国家自然科学基金项目(70861001);广西研究生科研创新项目(2008105930701M51);广西自然科学基金资助项目(桂科自 099102)。
作者简介:陶多秀(1985 -),女,广西桂林人,硕士研究生,主要研究方向:数据挖掘、粗糙集; 吕跃进(1958 -),男,广东龙川人,教授,主要研究方向:数据挖掘、粗糙集、概念格、运筹与控制; 邓春燕(1971 -),女,广西河池人,讲师,硕士研究生,主要研究方向:数据挖掘、粗糙集。

针对上述问题，即基于约束提高频繁模式挖掘的效率，使用户的个性化需要融入到规则的挖掘中，多维情况下的关联规则挖掘等，本文定义了一种挖掘语言使得用户可以指定感兴趣的项以及拟挖掘的关联规则的参数（如支持度，置信度等），并提出了一种新的基于粗糙集理论的多维关联规则挖掘技术，以减少频繁项集的生成搜索空间，提高算法的效率。

1 多维关联规则的基本问题描述

1.1 基本概念[1,10]

定义1 令 $AI = \{I_1, I_2, \cdots, I_m\}$ 是所有项的集合，其中 $I_k(k = 1,2,\cdots,m)$ 称为项。项的集合称为项集 I（或模式），$I \subseteq AI$。包含 k 个项的项集称为 k 项集，k 表示项长，即 $k = |I|$。

定义2 一个事务（或称交易）$T = \langle TID, I\rangle$ 是一个元组，其中 TID 为事务的标识，$I \subseteq AI$。

事务数据库 TDB 是事务的集合，$TDB = \{T1, T2, \cdots Tn\}$。设 X 是一个项集，事务 $T = \langle TID, I\rangle$ 支持项集 X（或者说 X 包含于 T）当且仅当 $X \subseteq T$。项集 X 的支持计数为 TDB 中包含 X 的事务数，则 X 的支持度等于支持计数与 TDB 中事务总数的百分比。

定义3 频繁项集。给定一个支持度阈值 $minsup$，$1 \leqslant minsup \leqslant |TDB|$，则支持度大于等于 $minsup$ 的项集 X 称为频繁项集，简称频繁集，也称大项集。

定义4 关联规则是如下的一种蕴含式：$X \Rightarrow Y$，其中 $X \subset T, Y \subset T$，且 $X \cap Y = \varnothing$。

多维关联规则可以表示成形如 $X \Rightarrow Y$ 的蕴含式，其中的 X、Y 分别表示规则合取范式构成的逻辑公式，$X \cap Y = \varnothing$。X 和 Y 分别称为规则的前件和后件。

若一个规则仅描述项是否出现在某种情况间的联系，那么它就是一个单维布尔关联规则。例如规则：买牛奶 ⇒ 买面包（支持度 support = 2%，置信度 = 60%）。若一个规则涉及到两个或更多个维度，诸如年龄、收入、职业和是否购买电脑等，那么它就是一个多维关联规则。例如规则：年龄（x，“30 ~ 34”）∧收入（x，“42K ~ 48K”）⇒ 购买（x，“computer”），其中 x 代表顾客变量。

定义5 关联规则 $X \Rightarrow Y$ 的支持度指事务集 TDB 中同时支持项集 X 和 Y 的事务（即包含 $X \cup Y$ —— X 和 Y 二者）所占的百分比。支持度说明了该条规则在所有事务中具有多大的代表性，是对关联规则重要性的衡量。显然支持度越大，规则越重要，其数学形式为：

$$support(X \Rightarrow Y) = P(X \cup Y) = \frac{|\{T | T \in TDB \wedge (X \cup Y) \subseteq T\}|}{|TDB|}$$

定义6 置信度。规则 $X \Rightarrow Y$ 的置信度指的是事务集 TDB 中支持项集 X 的事务里，同时也支持 Y 的事务所占的百分比。简而言之，置信度就是指在出现了 X 的事务中，Y 也同时出现的概率。置信度是对关联规则准确度的衡量，其数学形式为：

$$confidence(X \Rightarrow Y) = \frac{|\{T | T \in TDB \wedge (X \cup Y) \subseteq T\}|}{|\{T | T \in TDB \wedge X \subseteq T\}|} = support(X \cup Y)/support(X)$$

满足支持度和置信度阈值的规则称为强规则。

1.2 多维关联规则问题描述

给定一个事务数据库 TDB（如表1，其中 CH 表示 CHEQUE，CA 表示 CASH），支持度阈值 $minsup$，置信度阈值 $minconf$。多维关联规则挖掘就是要找出支持度和置信度分别大于 $minsup$ 和 $minconf$ 的多维关联规则。通常的思路是分解为以下两个子问题：1）从 TDB 中找出所有频繁集；2）利用频繁集生成满足 $minconf$ 的关联规则。

表1 多维事务数据库示例[11]

TID	付款方式	性别	收入/元	年龄	啤酒	鱼	…
39808	CH	M	27 000	46	F	F	…
63762	CA	F	30 000	28	F	T	…
10872	CA	M	13 200	36	T	F	…
26748	CA	F	12 200	26	T	T	…
91609	CH	M	11 000	24	F	F	…
…	…	…	…	…	…	…	…

2 粗糙集基本理论

粗糙集（Rough Set，RS）理论[12]是一种处理不确定性和含糊性的数据分析理论。近年来，该理论受到了世界各国学者越来越多的关注，已在决策分析、人工智能、专家系统、模式识别、故障检测、归纳推理、机器学习和知识发现等许多领域得到了相当广泛的应用[13-16]。下面先介绍 RS 的几个关键概念[12]。

定义7 信息系统（属性—值系统）$S = \langle U, A, V, f\rangle$，其中 $U = \{u_1, u_2, \cdots, u_{|U|}\}$ 是对象的有限非空子集；$A = \{a_1, a_2, \cdots, a_{|A|}\}$ 是非空的属性子集；$V = \{V_{a_1}, \cdots, V_{a_{|A|}}\}$ 是属性值的集合，其中 $V_{a_j} = \{V_{a_{j1}}, \cdots, V_{a_{jk}}\}$ 是属性 a_j 的值域。对于每个 $a \in A$，存在一个映射：$f(u,a): U \times A \rightarrow V_a$。特别地，若 $A = C \cup D, C \cap D = \varnothing$，则称信息系统为一个决策系统，其中 C 中的属性称为条件属性，D 中的属性称为决策属性。

定义8 对于属性 $a \in A$，对象 $u, v \in U$ 称在 U 上具有等价关系当且仅当 $f(u,a) = f(v,a)$。一个等价关系给出了 U 上关于 a 的一组划分 $U/a = \{W_1, \cdots, W_r\}$，满足：任意两个对象 $u, v \in U$ 在同一个等价类 X_i 当且仅当 $f(u,a) = f(v,a)$；$W_i \subseteq U, W_i \neq \varnothing, \cup W_i = U$，且 $W_i \cap W_j = \varnothing, i \neq j, i, j = 1, \cdots, r$。

定义9 $B \subseteq A$ 的不可区分关系为 $ind(B) = \{(x,y) \in U \times U | \forall a \in B, f(x,a) = f(y,a)\}$。

显然 $ind(B)$ 是 B 中所有不可区分关系的交，也是一个等价关系。因此，$ind(B)$ 对应一个等价划分 $U/ind(B)$，简写成 U/B。U/B 中的每个元素即 $ind(B)$ 的等价类，称之为基本集或基本概念。

定义10 属性约简定义为不包含多余属性且保证分类正确的最小属性子集。设 $B \subseteq A$，若 B 满足下列两个性质，就称 B 为 A 的约简：1）$ind(B) = ind(A)$；2）不存在 $B' \subset B$ 使得 $ind(B') = ind(A)$。

3 挖掘多维关联规则的粗糙集方法

3.1 多关联规则挖掘的粗糙集方法原理分析

基于粗糙集模型，有：事务数据库 $TDB = \langle T, I, V, f\rangle$。

定义11 $I^* \subseteq I$ 的不可区分关系 $ind(I^*)$ 在 T 上形成一个划分 U/I^*，令 $[X]$ 为其中的一个等价类，若 $|I^*| = k$，$|I^*| = k, I^* = \{I_1, I_1, \cdots, I_k\}$ 则称 $[X]$ 为 k 项基本集，记为 $[X^k]$，$[X^k] = \{T_i | f(T_i, I_1) = i_{1'} \wedge \cdots \wedge f(T_i, I_k) = i_{k'}\}$。$X^k$ 为 k 项描述 $X^k = (I_1 = a_{1'}) \wedge \cdots \wedge (I_k = a_{k'})$。其中的任一范式 $I_t = i_{t'}$ 称为描述子句，$i_{t'} \in V(I_t)$。将出现在规则后件的

基本集称为概念,出现在规则前件的基本集称为特征类。

定义 12　k 项基本集 $[X^k]$ 的支持度 $sup\_count([X^k]) = card([X^k])/card(T)$. 其中 $card(X)$ 为集合 X 的基数,也可写成 $|X|$。令 $minsup$ 为最小支持度,则 $sup\_count([X^k]) \geqslant minsup$ 的 $[X^k]$ 称为 k 项频繁基本集,或简称 k 项频集。记 k 项基本集的集合为 C^k,k 项频集的集合为 L^k,则 C^k 为 L^k 的候选集。

为使形式统一,将多维关联规则的前件和后件中的项分别用 X_i, Y_j 来表示。则由描述子句 $(X_1 = x_{1'}) \wedge \cdots (X_k = x_{k'})$ 和 $(Y_1 = y_{1'}) \wedge \cdots \wedge (Y_k = y_{k'})$ 可抽取项集 X 和 Y 之间的关联规则 $X \Rightarrow Y$。其支持度和置信度分别为:

$$s\% = \frac{card([(X_1 = x_{1'}) \wedge \cdots \wedge (X_k = x_{k'})] \cap [(Y_1 = x_{1'}) \wedge \cdots \wedge (Y_k = x_{k'})])}{card(U)} \times 100\%$$

$$c\% = \frac{card([(X_1 = x_{1'}) \wedge \cdots \wedge (X_k = x_{k'})] \cap [(Y_1 = x_{1'}) \wedge \cdots \wedge (Y_k = x_{k'})])}{card([X])} \times 100\%$$

根据 RS 中属性约简的概念可定义冗余规则如下。

定义 13　冗余规则:若 $X_1 \Rightarrow Y, X_2 \Rightarrow Y$ 为导出同一概念描述 Y 的两条规则,其中 $X' = (X_1 = x_1) \wedge (X_2 = x_2) \wedge \cdots \wedge (X_k = x_k)$,$X'' = (X_1 = x_1) \wedge (X_2 = x_2) \wedge \cdots \wedge (X_k = x_k) \wedge (X_{k+1} = x_{k+1}) \wedge \cdots \wedge (X_{k+m} = x_{k+m})$,即 $X_2 = X_1 \wedge (X_{k+1} = x_{k+1}) \wedge \cdots \wedge (X_{k+m} = x_{k+m})$,则 $X'' \Rightarrow Y$ 是一条冗余规则。

根据 Apriori 性质[1],任何频繁集的子集都是频繁集,任何非频繁集的超集都是非频繁集,可得以下定理。

定理 1　k 项基本集成为 k 项频集的必要条件是其包含的所有 j 项基本集都是频集,$1 \leqslant j \leqslant k-1$。

综上所述,多关联规则挖掘的粗糙集方法主要包括以下两个运算:

1)根据定理 1,通过基本集的交运算,由 L^k 生成 C^{k+1}:求出 C^k,选择 C^k 中的支持计数超过最小支持计数的特征类,形成 L^k。再将 L^k 与 L^1 的特征类进行交运算,形成 C^{k+1}。

2)删除运算。根据定理 1,对 C^{k+1} 中的每个基本集 $[X^{k+1}]$ 的所有 k 项子描述进行检查,若有某 k 项子描述的基本集 $[X^k]$ 不在 L^k 中,则将 $[X^k]$ 从 C^k 中删除。

在 L^k 中,通过计算 $c\%$ 抽取满足置信度阈值的规则 $(X_1 = x_{1'}) \wedge \cdots \wedge (X_k = x_{k'}) \Rightarrow (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_k = y_{k'})$,并将该规则对应的基本集 $[(X_1 = x_{1'}) \wedge \cdots \wedge (X_k = x_{k'}) \wedge (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_k = y_{k'})]$ 从 L^k 中删除,从而避免产生冗余规则。

3.2　表达用户个性化需要的挖掘语言

通常用户只对某些项集感兴趣,如:购买电脑是否与职业有关。此外对参数(如支持度、置信度)的设置要求也比较灵活。

为了将用户的个性化需要融入到挖掘过程中,使得规则更具有针对性和实用性。本文定义了一种挖掘关联规则的语言个性化数据挖掘语言(Personalized Data Mining Language, PDML),用户可在其中指定感兴趣的项集、规则的相关参数。该语言定义如下:

Ming < interesting items > From < Transaction DB > with *minsup* $s\%$ and *minconf* $c\%$

其中 interesting items 指定要挖掘的项,Transaction DB 指定挖掘的数据源即事务数据库,*minsup* 和 *minconf* 分别指定挖掘的最小支持度为 $s\%$,最小置信度为 $c\%$。

3.3　基于 RS 和挖掘语言的挖掘算法

根据所定义的挖掘语言 PDML 和多关联规则挖掘的粗糙集方法原理分析,给出一个算法以挖掘满足用户需要的多维关联规则。该算法对数据格式的要求为项的取值是离散的。考虑到离散化后的数据保持原有的不可分辨关系,在本文实验中采用了 Semi Naive Scaler 离散化算法[17]。用户的获取需求可分成两种情形:Case1:用户指定规则后件项集 Y(项长为 s,可表示为 $Y = \{Y_1, \cdots, Y_s\}$)和前件项集 X(长度为 t,可表示为 $Y = \{X_1, \cdots, X_t\}$);Case2:用户只指定规则后项集 Y,$Y = \{Y_1, \cdots, Y_s\}$。算法描述如下:

算法:从事务数据库 *TDB* 中挖掘满足用户需求的多维关联规则

输入:事务数据库 *TDB*,最小支持度 *minsup*;

输出:满足用户指定需求的多维关联规则集 *RL*。

过程:

1)初始化 $RL = \varnothing$。

2)扫描 *TDB*,获取各单项的等价类,组成 1 项基本集的集合 C^1,并通过基本集的支持数与 $minsup \times |TDB|$ 比较,从 C^1 中找出 1 项频集的集合 L^1。

3)求取包含指定项的基本集(对于 Case1 为 $[(X_1 = x_{1'}) \wedge \cdots \wedge (X_t = x_{t'}) \wedge (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$,对于 Case2 为 $(Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$)及其支持度,若没有满足最小支持度的基本集,则挖掘结束;否则所有大于最小支持度的基本集构成满足用户指定项的频集的集合 L^k(这里的 k 指的是指定项的长度,对于 case1,$k = s + t$;对于 case2,$k = s$),对于 Case2 转 5)。其中 $[(X_1 = x_{1'}) \wedge \cdots \wedge (X_t = x_{t'}) \wedge (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$ 或 $(Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$,通过 L^1 中的基本集 $[X = x_{i'}]$ 与 $[Y = y_{j'}]$ 的交运算获得。

4)对于 Case1,在 L^k 中试图抽取符合用户指定需要的规则,若有满足置信度的规则 $[(X_1 = x_{1'}) \wedge \cdots \wedge (X_t = x_{t'}) \wedge (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$,则将其加入规则库,并从 L^k 中删除对应的基本集 $[(X_1 = x_{1'}) \wedge \cdots \wedge (X_t = x_{t'}) \wedge (Y_1 = y_{1'}) \wedge \cdots \wedge (Y_s = y_{s'})]$,否则转 5);

5)生成 $k+1$ 项或候选基本集 C^{k+1}。通过基本集的交运算,将 L^k 与 L^1 连接,生成 C^{k+1},再从中求出 L^{k+1}。

6)类似 4),在 L^{k+1} 中试图抽取符合前件包含 X,后件为 Y 的规则,如有则加入规则库,否则进行递推迭代,直至第 n 次 $L^{k+n} = \varnothing$ 时,计算终止,输出 *RL*。

算法时间复杂度:设事务数据库中共有 n 条记录(即 n 个事务),m 个单项,每个单项可取 t 个离散值;计算一个基本集支持度的时间复杂度为:$O(n)$。若用户指定的后项集 Y 项长为 L,即 $|Y| = L$,对应的有 h 个概念,则数据库中共有 $K = C^1_{m-L} \times t + C^2_{m-L} \times t^2 + \cdots + C^2_{m-L} \times t^{m-L}$ 个特征类,在最坏情况下计算某个概念 Y 的规则需要计算 K 次支持度。而将各单项取值离散映射的时间复杂度为 $n\log n$,因此算法的计算时间为 $h \times (Kn + n\log n)$,而 $m, t, h \ll n$,所以算法的时间复杂度为 $O(n\log n)$。此外,算法在求出 k 项频集后,若找到满足需要的规则,则将相应的基本集删除,避免冗余规则的同时也减少了生成 $k+1$ 项基本集的交运算,从而提高了算法的效率。

4 实例分析

数据库中有10个事务如表2所示，即$|TDB| = 10$。

表2 数据库实例

TID	A	B	C	D	E
T1	0	1	3	1	1
T2	0	2	0	1	1
T3	0	1	3	0	1
T4	1	2	3	1	1
T5	2	1	3	2	1
T6	1	2	1	1	1
T7	1	2	3	2	0
T8	0	2	0	1	0
T9	1	1	1	1	0
T10	2	1	3	1	0

假设用户现在需要找出B与E的关联，以E为后项集，求B对E的影响，要求规则最小支持度$minsup = 30\%$，最小置信度$minconf = 70\%$。

首先将这一问题转化为所提出的语言：

Mining (B), (E) From TDB with $minsup = 30\%$ and $minconf = 70\%$

利用所提出的算法，过程如下：

1)扫描TDB，获取各项的等价类，得出所有1项基本集：

$C^1 = \{[A = 0],[A = 1],[A = 2],[B = 1],[B = 2],[C = 0],[C = 1],[C = 3],[D = 0],[D = 1],[D = 2],[E = 0],[E = 1]\}$

2)求出所有的1项频集。如：$Sup\_count([A = 0]) = |\{T1,T2,T3,T8\}| = 4 > minsup \times |TDB|$，同理可求出其他特征类的支持计数，经与最小支持计数$minsup \times |TDB| = 30\% \times 10 = 3$比较，从而得：

$L^1 = \{[A = 0],[A = 1],[B = 1],[B = 2],[C = 3],[D = 1],[E = 0],[E = 1]\}$

3)求包含B与E的L^2，得$L^2 = \{[B = 1 \wedge E = 1],[B = 2 \wedge E = 1]\}$。

4)在L^1中求包含指定项的规则：

$B = 1 \Rightarrow E = 1$的置信度

$$conf(B = 1 \Rightarrow E = 1) = \frac{card([B = 1] \cap [E = 1])}{card([B = 1])} = \frac{|\{T1,T3,T5\}|}{\{T1,T3,T5,T9,T10\}} = 60\% < minconf$$

同理，计算出$B = 2 \Rightarrow E = 1$的置信度也为$60\% < minconf$，不满足用户要求，不能加入规则库。

5)将L^1与L^1中的基本集进行交运算，求出包含B与E的3项频集L^3。如$[B = 1 \wedge E = 1] \cap [C = 3] = [B = 1 \wedge C = 3 \wedge E = 1]$，其支持度为$= |\{T1,T3,T5\}| = 3 = minsup \times |TDB|$，可归入$L^3$。求得：$L^3 = \{[B = 1 \wedge C = 3 \wedge E = 1],[B = 2 \wedge D = 1 \wedge E = 1]\}$，支持计数均为3。

6)在L^3中抽取规则：$B = 1 \wedge C = 3 \Rightarrow E = 1$，置信度为$75\% > minconf$，$B = 2 \wedge D = 1 \Rightarrow E = 1$，置信度也为$75\% > minconf$，于是将这两个规则加入规则库$RL$，并从$L^3$中删除$[B = 1 \wedge C = 3 \wedge E = 1]$以及$[B = 2 \wedge D = 1 \wedge E = 1]$。

7)此时L^3已为空，于是输出RL中的规则，结束。

5 结语

本文定义了一种挖掘语言PDML以提高挖掘过程中与用户的交互程度，使得用户可以指定感兴趣的项集，设置支持度与置信度等参数。根据RS理论，将多维关联规则的信息转化为基本集的描述，前件对应特征类，后件对应概念，从而给出一种挖掘满足用户个性化需要的多维关联规则挖掘算法，并结合动态生成频繁集的思想，边生成频集边抽取规则，可以减少频集生成的搜索空间，降低复杂性，而且可以避免冗余规则。算法的不足之处在于需要先将项的取值离散化。

参考文献：

[1] (德)巴斯蒂安 M. 数据仓库与数据挖掘[M]. 武森，高学东，译. 北京：冶金工业出版社，2003.

[2] AGRAWAL R, IMIELINSKI T, SWAMI A. Mining associations between sets of items in massive databases [C]// Proceedings of 1993 ACM SIGKDD International Conference on Management of Data. New York: ACM Press, 1993: 207 - 216.

[3] AGRAWAL R, SRIKANT R. Fast algorithms for mining association rules[C]// Proceedings of the 20th International Conference on Very Large Data Bases: VLDB'94. San Francisco: Morgan Kaufmann Publishers, 1994: 487 - 499.

[4] SAVASERE A, OMIECINISKI E, NAVATHE S B. An efficient algorithm for mining association rules in large database [C]// Proceedings of the 21th International Conference on Very Large Data Bases: VLDB'95. San Francisco: Morgan Kaufmann Publishers, 1995: 432 - 443.

[5] HAN J, PEI J. Mining frequent patterns by pattern-growth: Methodology and implications[C]// Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2000: 30 - 36.

[6] PEI J, HAN J. Can we push more constraints into frequent pattern mining? [C]// Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2000: 350 - 354.

[7] MEO R, PSAILA G, CERI S. A new SQL-like operator for mining association rules[C]// Proceedings of the 22th International Conference on Very Large Data Bases. San Francisco: Morgan Kaufmann Publishers, 1996: 122 - 133.

[8] YEN S-J, CHEN A L P. An efficient data mining technique for discovering interesting association rules[C]// Proceedings of the 8th International Workshop on Database and Expert Systems Applications: DEXA. Washington, DC: IEEE Computer Society, 1997: 664 - 669.

[9] 黄勇，刘锋. 关系数据库中多维关联规则挖掘的一种新算法[J]. 计算机应用与软件，2008，24(10)：60 - 61，83.

[10] 朱明. 数据挖掘[M]. 合肥：中国科技大学出版社，2002.

[11] SPSS Clementine 11.0, DEMO [CP/DK]. SPSS Inc. 233 South Wacker Drive, 11th Floor, Chicago, IL 60606-6307, USA.

[12] 张文修. 粗糙集理论与方法[M]. 北京：科学出版社，2001.

[13] 吕跃进，刘南星，陈磊，一种基于并行遗传算法的粗糙集属性约简[J]. 计算机科学，2008，35(3)：219 - 221.

[14] 李金海，吕跃进. 一种基于关系矩阵的信息系统属性约简算法[J]. 计算机工程与应用，2008，44(9)：147 - 149，189.

[15] 陈鑫影. 李雄飞. 基于粗糙集理论的并行约简算法[J]. 计算机应用，2007，27(8)：1964 - 1966.

[16] 王家伟，黄大荣，雷鸣. 基于粗糙集和分形理论的交通流优化控制设计模型[J]. 计算机应用，2008，28(5)：1200 - 1203.

[17] 王国胤. Rough 集理论与知识获取[M]. 西安：西安交通大学出版社，2001.