基于项集依赖的最小关联规则挖掘

孟 军 王 蓬 张 静 王秀坤
(大连理工大学计算机科学与技术学院 大连 116024)

摘 要 传统关联规则挖掘可能会得到大量的、杂乱的规则,它们对用户来说是不相关的或不感兴趣的。提出最小关联规则集和项集强依赖关系的概念,以实现基于项集依赖的最小关联规则挖掘算法。其不仅可以避免验证某一频繁项集下的所有非空真子集是否可形成关联规则,还可以通过删除那些过于复杂、有重复信息的规则来进一步简化传统规则集合。通过最小关联规则集可推导得到大多数冗余规则的支持度和置信度,实现了传统规则集的一种近似无损表述。采用UCI机器学习库中数据集进行实验,结果表明提出的方法得到的规则数量明显减少,且规则更加简短、无重复信息,为最小关联规则挖掘提供了更好的方法。

关键词 最小关联规则,项集依赖,冗余规则
中图法分类号 TP181 **文献标识码** A

Minimal Association Rules Mining Based on Itemset Dependency

MENG Jun WANG Peng ZHANG Jing WANG Xiu-kun
(School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China)

Abstract There are excessive and messy rules produced by traditional association rule mining, many of which are not relevant to users' interest. Minimal association rules mining algorithm was represented based on the concept of minimal association rules-set and strong dependency between items. Not only avoiding checking whether every non-empty subset of one frequent itemset can form an association rule, but also simplifying the traditional rules set by deleting those excessively complex and reduplicative rules. The support and confidence degree of most redundant rules can be derived from minimal association rules set, which achieves a nearly lossless representation of the traditional rules set. The results based on four benchmark data sets from UCI repository show that the number of rules generated by proposed method is reduced greatly and those rules in rules set are more briefly without reduplicative information. This provides a better way to find minimal association rules.

Keywords Minimal association rules, Itemset dependency, Redundant rules

1 引言

关联规则挖掘是在大型数据集中找到潜在的、有趣的关联和规律。这种起源于超级市场购物篮分析的数据处理模式,自1993年提出以来,在商业、电信、金融等领域有着广泛的应用,一直是数据挖掘领域中一个十分活跃的研究分支。

传统的关联规则挖掘一般会找出所有满足最小支持度和最小置信度阈值的关联规则。为了实现该目的,Agrawal等人提出了Apriori算法[1],它将关联规则挖掘分为两个子问题:(1)根据支持度,产生频繁项集;(2)根据置信度,产生强关联规则。但多次扫描数据库和产生庞大的候选集是它两个致命的性能瓶颈;并且,关联规则挖掘的效率主要由频繁项集挖掘的多少和快慢决定,通过不同手段提高Apriori算法的执行效率成为关联规则挖掘的重点研究内容。为了有效地减少I/O操作时间,在每次扫描数据库的过程中同时获取多个不同大小的频繁项集[2]和单次扫描数据库是两种主要的改进措施;为了提高算法的计算性能,可改变传统的集合操作为快速的位操作[3];为了节省内存空间,通常将数据按水平方向和垂直方向压缩进位表内[4]。此外,还有大量研究集中在如何改进Apriori方法获取频繁项集的逻辑策略[5]和如何利用并行思想加速Apriori方法的执行速度[6]等问题上。

上述方法尽管大大提高了算法的执行效率,但是针对大规模数据时仍会产生大量的规则,引发规则爆炸问题。因此,学者们致力于对第二个子问题的深入研究,分析如何减少规则数量和表示规则,从而使挖掘出的规则便于决策者理解和运用。根据改进方法得到的规则集与原始规则集之间的关系,可分为有损表述模型和无损表述模型[7]。有损表述模型需要用户的领域知识来限制最后的结果,兴趣度约束[8]和概率约束[9]是其中典型的方法。前者通过增加"规则兴趣度"标准去掉用户不感兴趣的规则,后者利用概率知识约束关联规则,两种方法都提高了规则的可用度,使结果集更能满足用户的需求。而无损表述模型由于不受应用背景的限制,往往更实用。基本关联规则[10]是一种典型的近似无损表述模型,通过较少的规则可推出全部冗余规则的置信度和部分冗余规则

到稿日期:2012-03-29 返修日期:2012-06-25 本文受国家自然科学基金(60973068,61173162)资助。
孟 军(1964—),女,硕士,副教授,主要研究方向为数据库和决策支持系统、机器学习与数据挖掘,E-mail:mengjun@dlut.edu.cn;王 蓬(1987—),男,硕士生,主要研究方向为数据挖掘与粗糙集理论;张 静(1989—),女,硕士生,主要研究方向为粗糙集与粒计算理论。

的支持度。简单关联规则[11]和最小非冗余关联规则[12]是两种完全无损的表述模型，前者只筛选后件项数为1的规则，后者采用伽罗瓦闭集理论去除冗余规则。不难看出，以上方法都试图重新描述和表示关联规则集，采用这些策略获得的规则数量大大减少，这充分体现出了规则集表述的多样性。

目前研究虽然大多致力于如何选择更有意义的规则或如何获取更少的规则，但仍存在以下两个问题：(1)对于冗余规则的定义过于严格，规则数量的减少是相对有限的[7,11]；(2)包含项数过多的频繁项集产生的关联规则往往十分复杂，会包含大量冗余的、重复的信息，不便于用户使用。为了解决以上问题，本文通过探讨项集之间的依赖关系，提出最小关联规则集的概念，并给出相关挖掘算法。该算法得到的规则集是传统规则集的一种近似无损表述，保证了信息的完整性；并且，规则数量大大减少，可显著提高用户从规则中提取有效信息并做出正确决策的效率。

2 问题定义及描述

2.1 初步描述

令 $I=\{i_1,i_2,\cdots,i_m\}$ 为事务数据库 D 中所有项的集合，那么每个事务 t 由不同的项构成，且满足 $t\subseteq I$。一条关联规则 r 可表示为 $X\Rightarrow Y$ 的形式，$X\subset I$，$Y\subset I$ 且 $X\cap Y=\emptyset$。令 $\mathrm{Sup}(X)$ 表示 D 中包含 X 事务的个数，那么关联规则 $r:X\Rightarrow Y$ 的支持度定义为：$\mathrm{Sup}(r)=\mathrm{Sup}(X\cup Y)$，置信度定义为：$\mathrm{Conf}(r)=\mathrm{Sup}(X\cup Y)/\mathrm{Sup}(X)$。关联规则挖掘就是要找出支持度和置信度分别大于预先设定的最小支持度阈值 ε 和最小置信度阈值 η 的规则。本文用 $\mathcal{R}$ 表示挖掘到的原始规则集合，即 $\mathcal{R}=\{r:X\Rightarrow Y\mid \mathrm{Sup}(r)\geqslant\varepsilon,\mathrm{Conf}(r)\geqslant\eta,X\subset I,Y\subset I\text{ 且 }X\cap Y=\emptyset\}$，$r$ 的长度记为 $L(r)=|X|+|Y|$。

2.2 最小关联规则

本文通过引入“项集依赖度阈值 λ”来控制规则的简化和规则集合的去冗，从而获取最小关联规则集。最小关联规则要求删除条件项集和结论项集中信息表达重复的项，力图实现用最少的项数来表达规则和用最少的规则来表述原始规则集的目的。当 $\mathrm{Conf}(r:X\Rightarrow Y)\geqslant\lambda$（一般情况下 $\eta<\lambda\leqslant 1$，且人为设定为等于或者十分接近于1）时，认为规则 r 的决策项集 Y 同条件项集 X 之间存在一种很强的依赖关系，多数情况下它们不应同时出现在长度更大的规则中。这样本文获得的规则集合是 $\mathcal{R}$ 的一个子集，在剔除大量冗余规则后包含与 $\mathcal{R}$ 几乎相同的信息，实现了原始规则集的一种近似无损的表述方式。

定义 1 对 $\forall r:M\Rightarrow N$，若 $\mathrm{Conf}(r)\geqslant\lambda$，则称项集 N 与项集 M 存在强依赖关系，记作 $r_d:M\xrightarrow{D}N$，所有强依赖关系构成强依赖关系集 $\mathcal{R}_D$。用 $\mathcal{R}_D^m$ 表示 $\mathcal{R}_D$ 中所有长度为 m 的强依赖关系，则 $\mathcal{R}_D^m=\{r_d\mid r_d\in\mathcal{R}_D\text{ 且 }L(r_d)=m\}$，其中 $m=2,\cdots,n_1$，n_1 为 $\mathcal{R}_D$ 中最长依赖关系的长度。

不难看出，$\mathcal{R}_D$ 中是置信度不小于 λ 的关联规则。

定义 2 对 $\forall r:X\Rightarrow Y\in\mathcal{R}$，若 $\exists r_d:M\xrightarrow{D}N\in\mathcal{R}_D$ 且满足 $L(r_d)<L(r)$，使得如下3种情况：(1)$M\cup N\subseteq X$；(2)$M\cup N\subseteq Y$；(3)$M\subseteq X$ 且 $N\subseteq Y$ 有1种成立，则认为 r 是一条冗余关联规则。

定理 1 已知 $M,N,X,Y\subset I$，项集两两的交为空且存在 $M\xrightarrow{D}N\in\mathcal{R}_D$，那么 $\mathcal{R}$ 中形如：$MNX\Rightarrow Y$，$X\Rightarrow MNY$，$MX\Rightarrow NY$ 和 $MX\Rightarrow N$ 规则的支持度和置信度分别可由形如 $MX\Rightarrow Y$，$X\Rightarrow MY$，$MX\Rightarrow Y$ 和 $M\Rightarrow N$ 更加简短的规则的支持度和置信度导出。

证明：由于 $M\xrightarrow{D}N$ 存在，那么 $\mathrm{Conf}(M\Rightarrow N)=\mathrm{Sup}(M\cup N)/\mathrm{Sup}(M)\geqslant\lambda$，且 $\lambda\cong 1$，‘$\cong$’表示等于或十分接近于。因此，$\mathrm{Sup}(M\cup N)\cong\mathrm{Sup}(M)$。(1)对于规则 $r:MNX\Rightarrow Y$，$\mathrm{Sup}(r)=\mathrm{Sup}(M\cup N\cup X\cup Y)\cong\mathrm{Sup}(M\cup X\cup Y)=\mathrm{Sup}(MX\Rightarrow Y)$；$\mathrm{Conf}(r)=\mathrm{Sup}(M\cup N\cup X\cup Y)/\mathrm{Sup}(M\cup N\cup X)\cong\mathrm{Sup}(M\cup X\cup Y)/\mathrm{Sup}(M\cup X)=\mathrm{Conf}(MX\Rightarrow Y)$。即，若 $M\xrightarrow{D}N$ 存在，那么比 $MX\Rightarrow Y$ 长度更大、形如 $MNX\Rightarrow Y$ 的规则是冗余的，其支持度和置信度可推导得到。规则 $X\Rightarrow MNY$ 和 $MX\Rightarrow NY$ 的冗余性证明同上。(2)对于规则 $MX\Rightarrow N$，$\varepsilon\leqslant\mathrm{Sup}(MX\Rightarrow N)=\mathrm{Sup}(M\cup X\cup N)\leqslant\mathrm{Sup}(M\Rightarrow N)=\mathrm{Sup}(M\cup N)$；$\mathrm{Conf}(MX\Rightarrow N)=\mathrm{Sup}(M\cup X\cup N)/\mathrm{Sup}(M\cup X)\cong\mathrm{Sup}(M\cup X)/\mathrm{Sup}(M\cup X)=1$，即形如 $MX\Rightarrow N$ 的规则必然会发生，其支持度和置信度可被推导得到（支持度获取的是一个范围数值），但其由于支持度较规则 $M\Rightarrow N$ 的小，因此相对实用意义不大，即是冗余的。

在上述证明中，若 λ 等于1，则 $\mathrm{Sup}(M\cup N)=\mathrm{Sup}(M)$；若 λ 小于接近1，则 $\mathrm{Sup}(M\cup N)\approx\mathrm{Sup}(M)$，这时，由经验告诉挖掘者允许一定的偏差存在，且仍可认为 M 和 N 同时出现。以上策略虽然会损失部分冗余规则支持度的精确值，但却可以用一些长度较小、更加简洁的规则推导出那些长度大、复杂的规则，实现了原始规则集合的一种近似无损表述。

定义 3 已知 $\mathcal{R}$ 和 $\mathcal{R}_D$，令 $\mathcal{R}_S$ 表示最小关联规则集，那么 $\mathcal{R}_S=\{r:X\Rightarrow Y\mid r\in\mathcal{R},\text{且对}\forall r_d:M\xrightarrow{D}N\in\mathcal{R}_D,\text{满足 }MN\not\subset X\text{ 且 }MN\not\subset Y\text{ 且}\{M\not\subset X\&N\not\subset Y\}\}$。用 $\mathcal{R}_S^m$ 表示 $\mathcal{R}_S$ 中所有 m 长度的规则，则 $\mathcal{R}_S^m=\{r\mid r\in\mathcal{R}_S\text{ 且 }L(r)=m\}$，其中 $m=2,\cdots,n_2$，n_2 为 $\mathcal{R}_S$ 中最长规则的长度。

定义 4 令 R 是一个关联规则集合。如果 $r\in R$ 或者 r 的支持度和置信度可由 R 中规则的支持度和置信度以定理1的形式推导出来，则称 r 是由 R 可推导的，表示为：$r\overleftarrow{\in}_dR$。

定理 2 基于特定的数据库给定最小支持度 ε、最小置信度 η 和项集依赖度 λ，(1)如果 $r\in\mathcal{R}$，那么 $r\overleftarrow{\in}_d\mathcal{R}_S$；(2)$\{r\mid r\overleftarrow{\in}_d\mathcal{R}_S,\mathrm{Sup}(r)\geqslant\varepsilon,\mathrm{Conf}(r)\geqslant\eta\}=\mathcal{R}$。

证明：(1)对于 $\mathcal{R}$ 中任意长度为2的规则 $r:X\Rightarrow Y$，其中 $X,Y\in I$，很明显 $r\in\mathcal{R}_S$。对于 $\mathcal{R}$ 中任意长度大于2的规则 r，若 r 不可被简化，那么 $r\in\mathcal{R}_S$。否则，由定理1知，若 $\exists r_d:M\xrightarrow{D}N\in\mathcal{R}_D$，则对于 $\mathcal{R}$ 中任意长度大于 $L(r_d)$ 的规则 r：形如 $MX\Rightarrow N$ 的，$\mathrm{Conf}(r)\cong 1$，$\varepsilon\leqslant\mathrm{Sup}(r)\leqslant\mathrm{Sup}(r_d:M\xrightarrow{D}N)$，且 $r_d\in\mathcal{R}_S$，所以 $r\overleftarrow{\in}_d\mathcal{R}_S$；其它形如 $MNX\Rightarrow Y$，$X\Rightarrow MNY$ 和 $MX\Rightarrow NY$ 的，其支持度和置信度可分别由 $\mathcal{R}_S$ 中形如 $MX\Rightarrow Y$，$X\Rightarrow MY$ 和 $MX\Rightarrow Y$ 的规则推导得到。所以，对 $\forall r\in\mathcal{R}$，其支持度和置信度都可以由 $\mathcal{R}_S$ 集合推导获得，即 $r\overleftarrow{\in}_d\mathcal{R}_S$。(2)一方面，如果 $r\in\mathcal{R}$，那么 $\mathrm{Sup}(r)\geqslant\varepsilon$，$\mathrm{Conf}(r)\geqslant\eta$，则根据(1)知，$r\overleftarrow{\in}_d\mathcal{R}_S$。另一方面，如果 $r\overleftarrow{\in}_d\mathcal{R}_S$，且 $\mathrm{Sup}(r)\geqslant\varepsilon$，$\mathrm{Conf}(r)\geqslant\eta$，那么必定有 $r\in\mathcal{R}$。因此，$\{r\mid r\overleftarrow{\in}_d\mathcal{R}_S,\mathrm{Sup}(r)\geqslant\varepsilon,\mathrm{Conf}(r)\geqslant\eta\}=\mathcal{R}$。

从定理2不难发现，$\mathcal{R}$ 中所有的规则均可由 $\mathcal{R}_S$ 中规则推导得到。

2.3 举例说明

采用文献[11]的例子对最小关联规则、规则数量及规则

集的近似无损表示进行说明。表1是一个交易记录。

表1 交易记录

交易号	产品项
1	A,D
2	B,E
3	A,B,D,E
4	B,D,E
5	B,C,D,E
6	A,B,E
7	A,B,C,D,E

设最小支持度阈值ε、最小置信度阈值η和项集依赖度阈值λ分别为3/7,65%和100%。如表2所列,常规仅依赖ε和η的方法输出18条规则(#1～#18),SAR方法[11]输出14条规则(#1～#14),而本文仅输出9条最小关联规则(#1～#9)。'*'表示该条规则同时存在强依赖关系,#10～#18为冗余关联规则,并在最后一列标出冗余规则可由哪些规则推导获得。不难看出,最小关联规则集在数量上相对SAR方法获得的规则更少,并且信息是基本无损的。

表2 规则集$\mathcal{R}$中的最小规则和冗余规则

序号	最小规则	序号	冗余规则		来源于
#1	A⇒B	#10	AB⇒E*	≪	#6 B⇒E
#2	A⇒D	#11	AE⇒B*	≪	#7 E⇒B
#3	A⇒E	#12	BD⇒E*	≪	#6 B⇒E
#4	B⇒D	#13	BE⇒D	≪	#4 B⇒D 或 #9 E⇒D
#5	D⇒B	#14	DE⇒B*	≪	#7 E⇒B
#6	B⇒E*	#15	A⇒BE	≪	#1 A⇒B 或 #3 A⇒E
#7	E⇒B*	#16	B⇒ED	≪	#4 B⇒D
#8	D⇒E	#17	D⇒BE	≪	#5 D⇒B 或 #8 D⇒E
#9	E⇒D	#18	E⇒BD	≪	#9 E⇒D

3 最小关联规则挖掘算法

基于最小关联规则的定义,将详细介绍如何进行规则的挖掘,并提出最小关联规则挖掘算法(Generating Minimum Association Rules,GMAR)。

3.1 基本思想

在描述算法前,先给出相关引理和定理。

引理1 若l^k是一个频繁k-项集,则对于l^k的任意非空真子集也是频繁的。

证明:假设项集l^m满足$\mathrm{Sup}(l^m)<\varepsilon$,则$l^m$不是频繁的。若将非空项集$l^n$添加到项集$l^m$使得$(l^n\cup l^m)=l^k$,则$l^k$不可能比$l^m$更频繁出现。因此,$\mathrm{Sup}(l^k)<\varepsilon$,即$l^k$也不可能是频繁的。由逆否命题性质知,引理1显然成立。

引理2[11] 令$X,Y,Z\subset I$,且两两相交不为空集,则(1)$\mathrm{Sup}(X\Rightarrow YZ)=\mathrm{Sup}(XY\Rightarrow Z)=\mathrm{Sup}(XZ\Rightarrow Y)$;(2)$\mathrm{Conf}(X\Rightarrow YZ)=\mathrm{Conf}(X\Rightarrow Y)*\mathrm{Conf}(XY\Rightarrow Z)=\mathrm{Conf}(X\Rightarrow Z)*\mathrm{Conf}(XZ\Rightarrow Y)$。

证明:(1)据支持度的定义,很明显$\mathrm{Sup}(X\Rightarrow YZ)=\mathrm{Sup}(X\cup Y\cup Z)=\mathrm{Sup}(XY\Rightarrow Z)=\mathrm{Sup}(XZ\Rightarrow Y)$;(2)据置信度定义,$\mathrm{Conf}(X\Rightarrow YZ)=\mathrm{Sup}(X\cup Y\cup Z)/\mathrm{Sup}(X)=(\mathrm{Sup}(X\cup Y)/\mathrm{Sup}(X))*(\mathrm{Sup}(X\cup Y\cup Z)/\mathrm{Sup}(X\cup Y))=\mathrm{Conf}(X\Rightarrow Y)*\mathrm{Conf}(XY\Rightarrow Z)$,将$Y,Z$互调可证$\mathrm{Conf}(X\Rightarrow YZ)=\mathrm{Conf}(X\Rightarrow Z)*\mathrm{Conf}(XZ\Rightarrow Y)$。

定理3 已知l^k是一个频繁k-项集($k\geqslant 3$),判断l^k形成的最小关联规则集R'时:(1)若对于$\mathcal{R}_S\neq\varnothing$,其中$m=2,\cdots,k-1$,则$R'=\{r:l\Rightarrow\{l^k-l\}\mid l\neq\varnothing,l\subset l^k$且$Conf(r)\geqslant\eta\}$;(2)若$\exists\mathcal{R}_S=\varnothing$,且对于任意的$n\in\{m+1,\cdots,k-1\}$,$\mathcal{R}_S\neq\varnothing$,那么$R'=\{r:l\Rightarrow\{l^k-l\}\mid l\subset l^k,m\leqslant|l|\leqslant k-1$且$\mathrm{Conf}(r)\geqslant\eta\}$。

证明:对于情况(1),不能排除任何子集的判断,同传统子集搜索方法相同,无需证明。情况(2)需分条件讨论:(a)存在强依赖项,放弃判断的子集:由于$\mathcal{R}_S=\varnothing$,令$\forall l^m\subset l^k$,则由引理1知$l^m$是频繁的;且对任意由$l^m$产生的$m$长度规则必然根据冗余性被删除,即$\forall r:l'\Rightarrow\{l^m-l'\}\in\mathcal{R},l'\subset l^m$规则冗余,那么一定存在$l^+\xrightarrow{D}l^-\in\mathcal{R}_D$使得$l^+l^-\subseteq l'$或$l^+l^-\subseteq\{l^m-l'\}$或$l^+\subseteq l'\&l^-\subseteq\{l^m-l'\}$成立。则对于$\{r:l\Rightarrow\{l^k-l\}\mid l\neq\varnothing$且$l\subset l^k\}$中的规则,由于存在$l^m$使得$l'\subset l$且$\{l^m-l'\}\subset\{l^k-l\}$,即使得$l^+l^-\subset l$或$l^+l^-\subset\{l^k-l\}$或$l^+\subset l\&l^-\subset\{l^k-l\}$存在,则$r$必然冗余,不用判断;(b)置信度小于$\eta$,放弃判断的子集:对集合$\{r:l\Rightarrow\{l^k-l\}\mid l\neq\varnothing,l\subset l^k$且$|l|<m\}$中的规则,据引理2可知:$\mathrm{Conf}(l\Rightarrow\{l^k-l\})=\mathrm{Conf}(l\Rightarrow l')*\mathrm{Conf}(\{ll'\}\Rightarrow\{l^k-l-l'\})$,令$|l|+|l'|=m$,则$\mathrm{Conf}(l\Rightarrow l')<\eta$,故$\mathrm{Conf}(l\Rightarrow\{l^k-l\})<\eta$,无需判断这样的子集;对于集合$\{r:l\Rightarrow\{l^k-l\}\mid l\neq\varnothing,l\subset l^k$且$|l|\geqslant m\}$中的规则,无法论证出$\mathrm{Conf}(r)$同$\eta$的大小关系,故需要判断包含项数不小于$m$的真子集是否可以形成关联规则。

例1 已知$\mathcal{R}_S\neq\varnothing,\mathcal{R}_S\neq\varnothing,\mathcal{R}_S=\varnothing$且$ABCDE$是一个频繁5-项集。传统方法需要判断左侧子集项个数不同的形如:$A\Rightarrow BCDE,AB\Rightarrow CDE,ABC\Rightarrow DE$和$ABCD\Rightarrow E$的规则是否可形成关联规则,GMAR方法则不需要。由于冗余性质使得$\mathcal{R}_S=\varnothing$,假设$A\Rightarrow BCD$是由$A\xrightarrow{D}B$导致的冗余规则,那么根据定义2知,上述形式规则均不用判断,都是冗余的;由于规则置信度过小导致$\mathcal{R}_S=\varnothing$,则对于频繁4-项集(如$ABCD$)形成的任何一条规则$r$,$\mathrm{Conf}(r)<\eta$,那么$\mathrm{Conf}(A\Rightarrow BCDE)=\mathrm{Conf}(A\Rightarrow BCD)*\mathrm{Conf}(ABCD\Rightarrow E)<\eta$。规则$AB\Rightarrow CDE$和$ABC\Rightarrow DE$同理可证。但对规则$ABCD\Rightarrow E$,不能判断其置信度与$\eta$的关系,如定理3所说的那样,只需判断包含项数不小于4的真子集即可。

基于以上定义及定理,GMAR算法采用Apriori方法思想,将关联规则挖掘分为两个子任务:

(1)找出所有满足最小支持度阈值ε的频繁项集。本文利用文献[13]提出的位运算和粒计算思想改进Apriori算法,不仅实现了单次扫描数据库,避免了繁琐的剪枝操作,而且利用位运算的快速性,提高了频繁项集的挖掘效率。

(2)找出最小关联规则集。对于每一个频繁k-项集l^k,利用定理3判断任意一条潜在规则r的冗余性和置信度。若r是非冗余且$\mathrm{Conf}(r)\geqslant\eta$的,则认为$r$是一条最小关联规则。

3.2 GMAR算法描述

GMAR算法的执行过程描述如下:

输入:事务数据库D,最小支持度阈值ε、最小置信度阈值η和依赖关系阈值λ。

输出:最小关联规则集。

```
Begin
(1) m=1; //设置子集搜索范围,长度从1开始
(2) L1={l1j | l1j(I,Sup(l1j))≥ε,1≤j≤|I| }; //频繁1-项集
(3) for(k=2;Lk-1≠∅;k++) {
(4)   Ck=BitGrC_apriori_gen(Lk-1); //候选k-项集
(5)   Lk={c∈Ck| Sup(c)≥ε}; //频繁k-项集集合
(6)   RSk=MinAR(Lk,η); //获长度为k的最小关联规则集RSk
```

(7) for each r∈$\mathcal{R}_S$

(8) if(Conf(r)≥λ) then add r to $\mathcal{R}_D$; //获取依赖关系

(9) if ($\mathcal{R}_S$==∅) then m=k; //将子集搜索范围设置到从长度 k 开始

(10) $\mathcal{R}_D=\bigcup_k\mathcal{R}_D$,$\mathcal{R}_S=\bigcup_k\mathcal{R}_S$;//获得所有的强依赖项关系最小规则

End

其中,BitGrC_apriori_gen 方法是一种利用粒表示和二进制位运算加速常规 Apriori_gen 的过程,用来处理频繁项集挖掘的子问题,这非本文研究的重点,将不做详细描述;MinAR 方法利用定理 3 搜索需要判断的子集,从而可在一定程度上提高规则发现子问题的效率,详细描述如下:

MinAR (L_k,η) //获长度为 k 的最小关联规则集 $\mathcal{R}_S$

(1) for each $l^k\in L_k$ {

(2) if(m==1) then

(3) T={ l | l≠∅,l⊂l^k}; //获得 l^k 的所有非空真子集

(4) else then

(5) T={ l | l≠∅,l⊂l^k,|l|≥m};//长度不小于 m 的非空真子集

(6) for each l∈T

(7) r:l⇒{l^k-l}; //获得一条候选规则

(8) if(Conf(r)≥η) then //满足置信度要求

(9) if (IsMinimum(r,$\mathcal{R}_D$)==true) then add r to $\mathcal{R}_S$;//冗余判断

(10) return $\mathcal{R}_S$;

MinAR 方法中的 IsMinimum 方法利用项集之间的强依赖关系,从冗余性判断角度出发,分析该条关联规则是否是冗余的,具体如下:

IsMinimum (r:l⇒{l^k-l},$\mathcal{R}_D$) //判断 r 的冗余性

(1) flag=true; //标示变量:true-最小的,false-非最小的

(2) for each $r_d:l^+\xrightarrow{D}l^-\in\mathcal{R}_D$

(3) if ($l^+l^-\subseteq l$) or ($l^+l^-\subseteq\{l^k-l\}$) or ($l^+\subseteq l$ & $l^-\subseteq\{l^k-l\}$) then

(4) flag=false,break; //标记冗余

(5) return flag;

通过对 GMAR 算法在规则归纳子问题执行过程的详细描述不难看出,该算法不仅可简化规则长度,而且可以删除规则集中冗余的规则,从而实现最小关联规则的挖掘。

3.3 算法性能分析

假定在 D 中,N 为事务总数,w 为事务的平均宽度,则 GMAR 算法的时间复杂度主要由以下 5 个因素决定:第一次扫描数据库的时间开销为 t_1:$O(N*w)$;其后每次循环中,由 L_{k-1} 生成 C_k 需要 $|L_{k-1}^2|$ 次连接操作,每次连接最多需 $k-2$ 次相等比较和 1 次不等比较,则连接的时间开销为 t_2:$O((k-1)*|L_{k-1}^2|)$;候选集支持度判断的时间开销为 t_3:$O(C_k)$;最小规则处理时,对于 L_k 中的任意频繁 k-项集 l^k,需根据定理 3 和非冗余性生成最小规则,该步的时间开销为 t_4:$O(\sum_{i=m}^{k-1}C_k^i*|L_{k-1}|*|\mathcal{R}_D|)$,其中 m 为开始判断的子集长度;获取强依赖关系的时间开销为 t_5:$O(|\mathcal{R}_S|)$。因此,GMAR 算法的总体时间复杂度为:$O(t_1+\sum_{k\geq2}(t_2+t_3+t_4+t_5))$。

实际应用中,长度较小的频繁项集及候选集的数量庞大,且 $|\mathcal{R}_D|$ 和 $|\mathcal{R}_S|$ 分别为常量级别,这导致 t_2+t_3 要远远大于 t_4+t_5。因此,GMAR 方法的效率主要由频繁项集挖掘的子问题决定,这与现存大多数基于 Apriori 方法、扫描一遍数据库的改进算法的时间复杂度同属一个数量级,即为 $O(t_1+\sum_{k\geq2}(t_2+t_3))\approx O(t_1+\sum_{k\geq2}(t_2))$,没有带来时间上过多的消耗。但 GMAR 算法在规则提取子问题中,不必验证 l^k 所有的子集,且 MinAR 方法在最好和最坏情况下的时间复杂度分别为 $O(C_k^{k-1}*|L_{k-1}|*|\mathcal{R}_D|)$ 和 $O(\sum_{i=1}^{k-1}C_k^i*|L_{k-1}|*|\mathcal{R}_D|)$,$\mathcal{R}_S$ 为空出现的越早,方法性能提升就会越大,不但得出的规则更加简短,而且可以排除大量的冗余规则。

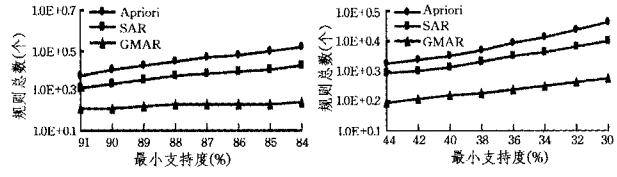
4 实验结果与分析

本文所有实验是在 Windows XP 系统、2.19GHz 酷睿处理器和 2G 内存环境下进行的,并根据粒表示和位运算结合的方式采用 Java 实现 Apriori、SAR 和 GMAR 算法。选取 UCI 中的 4 个数据集进行测试,如表 3 所列。其中,对于 adult 数据集,由于数据中存在很多未经离散化的属性,因此本文选取 Occupation 等 11 个离散型属性,并选择前 27,039 条数据。

表 3 数据集性质

Database	#Items	Avg. Record Length	#Records
mushroom	120	23	8,124
chess	76	37	3,196
Nursery	32	8	12,960
Adult	≥100	11	27,039

图 1 比较了基于粒表示和位运算的 Apriori、SAR 和 GMAR 方法在不同支持度下生成规则的数目。其中 chess、mushroom 和 adult 均是在 $\eta=0.8$,$\lambda=0.95$ 的情况下计算的。而对 Nursery 数据集在 η 过高时,Apriori 方法挖掘出的规则个数和 SAR 方法挖掘出的规则个数几乎相同,为了易于区分,故采用较低的置信度阈值,即 $\eta=0.5$,$\lambda=0.9$。



(a) Chess 数据集 (b) Mushroom 数据集

(c) Nursery 数据集 (d) Adult 数据集

图 1 Apriori、SAR 和 GMAR 方法规则个数比较

为了更加详细地展现 GMAR 方法对不同长度规则的约简效果,表 4 列出了 Adult 数据集在两个支持度下不同长度规则的详细数目。

表 4 Adult 规则个数详细比对

Min_sup	Method	Number of rules with n length						
		2	3	4	5	6	7	Total
7%	Apriori	84	439	891	964	576	161	3115
	SAR	84	404	684	561	231	43	2007
	GMAR	84	358	535	352	91	0	1420
30%	Apriori	19	46	50	22	∅	∅	118
	SAR	19	36	24	5	∅	∅	84
	GMAR	19	30	16	0	∅	∅	65

(下转第 217 页)

参 考 文 献

[1] Dempster A P. Studies in Fuzziness and Soft Computing [M]. Berlin: Springer, 2008: 73-104

[2] Gottlob G, Leone N, Scarello F. A Comparison of Structural CSP Decomposition Methods [C]// Proc of the 6th Int Joint Conf on Artificial Intelligence. California: Morgan Kaufmann, 1999: 394-399

[3] 季晓慧,张健. 约束问题求解[J]. 自动化学报, 2007, 2(33): 17-23

[4] Pearl J. Causality: Models, Reasoning, and Inference [M]. Cambridge: Cambridge University Press, 2000

[5] Butz C J, Yao H, Hua S. A join tree probability propagation architecture for semantic modeling [J]. Journal of Intelligent Information Systems, 2009, 33(2): 147-158

[6] Dechter R. Bucket elimination: A Unifying Framework for Reasoning [J]. Artificial Intelligence, 1999, 113(2): 41-85

[7] Lepar V, Shenoy P P. A comparison of Lauritzen-Spiegelhalter, Hugin and Shenoy-Shafer architectures for computing marginals of probability distributions [C]// Proc of the 14th Conf on Uncertainty in Artificial Intelligence. California: Morgan Kaufmann, 1998: 328-337

[8] Park J D, Darwiche A. Morphing the Hugin and Shenoy-Shafer Architectures [C]// Proc of the 7th European conference on symbolic and quantitative approaches to reasoning with uncertainty. LNCS 2711, Berlin: Springer, 2003: 149-160

[9] Fargier H, Vilarem M. Compiling CSPs into Tree-Driven Automata for Interactive Solving[J]. Constraints, 2004, 9(4), 263-287

[10] 田凤占,张宏伟,陆玉昌,等. 多模块贝叶斯网络中推理的简化[J]. 计算机研究与发展, 2003, 40(8): 1230-1237

[11] 汪荣贵. bayes网络理论及其在目标检测中应用研究[D]. 合肥: 合肥工业大学, 2004

[12] Cozman F G. Generalizing variable-elimination in bayesian networks[C]// Proc of the Workshop on Prob reasoning in Bayesian networks at SBIA/Iberamia. 2000: 21-26

[13] Kask K, Dechter R. A general scheme for automatic generation of search heuristics from specification dependencies [J]. Artificial Intelligence, 2001, 129(1/2): 91-131

[14] Kask K, Dechter R, Larrosa J, et al. Unifying Tree Decompositions for Reasoning in Graphical Models [J]. Artificial Intelligence, 2005, 166(1/2): 165-193

[15] http://www.norsys.com

[16] Ide J S. BNGnerator, Version 0.3[EB/OL]. http://www.pmr.poli.usp.br/ltd/Software/BNGenerator/, 2010-07-17

(上接第 186 页)

表 4 中,当最小支持度为 7%时,最大频繁项集长度为 7,故有 2 到 7 长度的规则;当最小支持度为 30%时,最大频繁项集长度为 5。'Ø'表示不存在该长度的频繁项集,'0'表示该长度下频繁项集产生的最小关联规则个数为零。

从以上实验不难看出,本文提出的 GMAR 方法能大大减少规则数目。相比 SAR 方法,若挖掘出的规则中后件包含一项的过多或者前件项数很多,且包含大量的冗余、重复项,那么 SAR 方法并不能有效地减少规则数目和进行规则项集的简化。GMAR 方法却克服了这一点,始终会输出最简的、无冗余的规则集合。

结束语 本文通过分析传统关联规则挖掘方法在规则提取子问题中的不足,提出基于项集依赖关系的最小关联规则挖掘方法。该方法用较少的规则实现了原始规则集的一种近似无损表述;并且,较 SAR 方法而言,本文提出的方法能更进一步地减少规则数目,从而找到更加简洁的、规则的前件和后件均不包含冗余项的关联规则。然而,由于目前生成的频繁项集仍然过多,接下来将着重研究该方法在闭包频繁项集挖掘模式下的应用和如何利用最小关联规则集推演原始规则集,实现规则集的无损表述等问题。

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 ACM SIGMOD International Conference on Management of Date. Washington DC, 1993: 207-216

[2] Loo K K, Yip C-I, Kao Ben, et al. A lattice-based approach for I/O efficient association rule mining [J]. Information Systems, 2002, 27(1): 41-74

[3] Lin T Y, Hu Xiao-hua, Louie E. A fast association rule algorithm based on bitmap and granular computing[C]// Proceedings of IEEE International Conference on Fuzzy Systems. 2003: 678-683

[4] 任永功,宋奎勇,寇香霞. 一种结合散列与位表挖掘频繁项目集算法[J]. 计算机科学, 2010, 37(12): 145-148

[5] Stanisic P, Tomovic S. Apriori multiple algorithm for mining association rules[J]. Information Technology and Control, 2008, 37(4): 311-320

[6] Aouad L M, Le-Khac N A, Kechadi T M. Performance study of distributed Apriori-like frequent itemsets mining[J]. Knowledge and Information Systems, 2010, 23(1): 55-72

[7] 陈茵,闪四清,刘鲁,等. 最小冗余的无损关联规则集表述[J]. 自动化学报, 2008, 34(12): 1490-1496

[8] Geng Li-qiang, Hamilton H J. Interestingness measures for data mining: a survey[J]. ACM Computing Surveys, 2006, 38(3): 1-33

[9] Christian H W. Statistical mining of interesting association rules [J]. Statistics and computing, 2008, 18(2): 185-194

[10] Li Gui-chong, Hamiltion H J. Basic association rules [C]// Proceeding of the 4th SIAM International Conference on Data Mining. Orlando, USA, 2004: 166-177

[11] Chen Guo-qing, Wei Qiang, Liu De, et al. Simple association rules(SAR) and the SAR-based rule discovery[J]. Computers and Industrial Engineering, 2002, 43(4): 721-733

[12] Mohammed J Z. Non-Redundant Association Rules [J]. Data Mining and Knowledge Discovery, 2004, 9(3): 223-248

[13] Louie E, Lin T Y. Finding association rules using fast bit computation: machine-oriented modeling[C]// Proceedings of International Symposium ISMIS2000. Charlotte, North, Lecture Notes in Artificial Intelligence, 2000: 486-494