面向语义的精简化多关系频繁模式发现方法

杨炳儒，张　伟，钱　榕
（北京科技大学信息工程学院，北京 100083）

［摘要］ 多关系频繁模式发现能够直接从复杂结构化数据中发现涉及多个关系的复杂频繁模式，避免了传统方法的局限。有别于主流基于归纳逻辑程序设计技术的方法，提出了基于合取查询包含关系的面向语义的精简化多关系频繁模式发现方法，具有理论与技术基础的新颖性，解决了两种语义冗余问题。实验表明，该方法在可理解性、功能、效率以及可扩展性方面具有优势。

［关键词］ 多关系数据挖掘；频繁模式发现；合取查询；精简化模式
［中图分类号］ TP182 ［文献标识码］ A ［文章编号］ 1009－1742(2008)09－0047－07

1 前言

数据挖掘致力于发现海量数据中隐藏的模式。频繁模式发现是数据挖掘的重要任务之一，早期的研究成果包括 Apriori 算法[1]及其变体[2]。这类方法的知识表示方式主要是命题逻辑形式系统，并且只能从单一关系中发现模式。但是，大多数现实关系数据库中的信息存储于多个关系中，在多关系数据中发现模式时，模式自然的要涉及多个关系，若使用这类经典数据挖掘方法，应把数据先从多个关系中纳入一个单关系中，然后才能进行挖掘。这不仅需要大量的预处理工作和谨慎的设计，并且可能导致信息丢失、语义偏差以及效率降低等问题，此外许多复杂模式难以用命题逻辑语言表示。另一类频繁模式发现方法来自于多关系数据挖掘领域[3]。多关系频繁模式发现方法，能够发现关系数据库中涉及多个关系的复杂模式，并且直接在多个关系上分析数据而无需向单一数据表转换。

当先验背景知识存在的情况下，如果不考虑这类知识的存在，会导致结果集中太多语义冗余模式。语义冗余模式分为两种情况：一种是模式内部存在语义冗余成分，另一种是模式在结果集中存在其他语义的等价模式。语义冗余的存在一方面会给数据挖掘结果使用者带来理解上的困难；另一方面会导致候选模式集合规模过于庞大，从而影响评估阶段的效率和扩展性。一般情况下，候选模式评估阶段的时间消耗经常占据系统整体时间消耗的 85 % 以上。因此，有效的消除语义冗余成为多关系频繁模式发现方法研究的重点之一，这类研究称为面向语义的精简化多关系频繁模式发现方法研究。

在多关系频繁模式发现研究中，最为知名的方法是 WARMR[4]和 FARMER[5]，这两种方法并没有考虑先验知识的存在；C－ARMR[6]在 WARMR 基础上考虑到了先验知识的存在，解决了上述第一类语义冗余问题，但没有解决第二类语义冗余问题。另一方面 C－ARMR 作为基于归纳逻辑程序设计技术（inductive logic programming，ILP）的方法存在如下问题。

1）ILP 技术是一种机器学习技术，其底层实现 Prolog 引擎都是面向演绎推理的，因而在面向海量数据的数据挖掘应用过程中，在效率和可扩展行方面尚有较大的研究空间。

2）ILP 技术要求在数据挖掘前，原始关系数据库必须花费大量时间预处理为一阶逻辑程序表示的

［收稿日期］ 2007－01－29；修回日期 2007－06－14
［基金项目］ 国家自然科学基金资助项目（60675030）；国家科技成果重点推广计划资助项目（2003EC000001）
［作者简介］ 杨炳儒（1943－），男，天津市人，北京科技大学教授，博士生导师，主要研究方向为知识发现与推理机制、柔性建模与集成技术

知识库。数据挖掘的预处理工作在大多数实践的情况下,在整个知识发现过程中时间占用了太大的比例,如何解决预处理阶段的时间消耗已经是一个重要的研究内容。

3) 数据挖掘算法在 Prolog 引擎上执行,系统与数据库系统不再耦合,无法有效利用关系数据库已有的查询优化和有效的数据组织策略;再次,数据挖掘算法的搜索空间构造、模式评估方法对于大多数关系数据库分析与管理专家都比较陌生。

上述因素阻碍了基于 ILP 技术的多关系频繁模式发现方法的广泛有效应用。如何在提高多关系数据挖掘效率的同时能够使用关系数据库理论和技术表达和实现挖掘算法,从而在理论和实践层面与关系数据库系统紧密集成,已经成为于国际研究共同体内认为的多关系数据挖掘将来发展的重要挑战与开放问题之一。

为了解决上述问题,提出了明显区别于上述方法的全新的 CMRFPDA 方法,具体体现在如下几方面。

1)根据关系数据库理论与技术定义了面向语义的精简化多关系频繁模式发现方法,从而使得 CMRFPDA 方法能够更有效地为与关系数据库有效集成奠定基础;原始数据无需向表示方式方面转化,节省了预处理时间;方法能够为数据库研究和使用者较为方便地理解和使用,为方法的有效实践和广泛使用创造了条件。

2)基于合取查询及其包含关系概念与方法,实现了消除两种语义冗余的功能。

3)使用了一个优化的精化算子构建搜索空间,精化算子一方面有效地利用了关系数据库隐含的数据模式特征,从而能够自然地构建有趣形态的模式,并且避免了模式语法等价变体的产生,减少了模式产生阶段语义冗余测试的计算量;使用了一个候选模式评估共享计算策略,从而降低了方法评估阶段的时间复杂性。实验表明,方法整体上具有良好的效率和可扩展性。

2 任务定义

2.1 关系数据库基本概念和理论

多关系频繁模式发现的挖掘对象是关系型数据库,为了描述挖掘对象和定义多关系频繁模式发现任务以及方法,先介绍关系数据库的有关概念和理论[7]。

设 $\Omega = \{A_1, A_2, \cdots, A_n\}$ 是属性集合,Dom (A_i)是 A_i 的定义域,即 A_i 得所有可能取值的集合。Ω 上的一个关系,记为 $R[\Omega]$,是这些定义域的笛卡儿集上的一个子集合。

令 $D_i = \text{Dom}(A_i)$, $i = 1, 2, \cdots, n$,它们的笛卡儿积记为 $D_1 \times D_2 \times \cdots \times D_n$,$R[\Omega] \subseteq D_1 \times D_2 \times \cdots \times D_n$,称 n 为关系 $R[\Omega]$的度,一个关系就是一个表,每个行称为一个 tuple,每列对应一个属性,一个关系的属性集合称为关系图式。$D_1 \times D_2 \times \cdots \times D_n$ 是所有 n – tuple $(a_1, a_2, \cdots, a_n)$的集合,其中 $a_i \in D_i$。Tuple $(a_1, a_2, \cdots, a_n)$有 n 个分量,第 i 个分量是 a_i,通常用 $a_1\, a_2 \cdots\, a_n$ 表示$(a_1, a_2, \cdots, a_n)$,或者简写为 a。

设 Ω是属性集合,$\Omega_i \subseteq \Omega$, $i = 1, 2, \cdots, m$,如果 $\forall i$, $\Omega_i \neq \emptyset$,且 $\cup_{i=1}^{m} \Omega_i = \Omega$,则称 $D = \{\Omega_1, \Omega_2, \cdots, \Omega_n\}$是 Ω 上的一个数据库图式(Schema),若 R_i 是 Ω_i 上的一个关系,则称 $R = \{R_1, R_2, \cdots, R_m\}$是在 D 上的一个数据库。

定义 1 笛卡儿积。设 R 和 S 是度分别为 k_1 和 k_2 的两个关系,则 R 和 S 的笛卡儿积 $R \times S$ 是前 k_1 个分量为 R 的一个 tuple 和后 k_2 个分量为 S 的一个 tuple 而形成的$(k_1 + k_2)$ – tuple 的集合。

定义 2 投影。对一个关系 R,移去某些分量和重新排列剩下的分量,如果剩余的分量为 K,则投影表示为 π_k。

定义 3 选择。设 F 公式涉及下列算子:

1) 常数,分支数;

2) 算术比较算子 $<$, $=$, $>$, $\leqslant$, $\neq$, $\geqslant$;

3) 逻辑算子 $\wedge$, $\vee$, $\neg$。

$\sigma_F(R)$是 R 中的具有下述性质的 tuple t 的集合:当对所有 i,在 F 中的任意出现的 i 都用 t 的第 i 个分量代替,公式变为真。

定义 4 合取查询的定义。合取查询是仅使用选择、投影和笛卡儿积算子形成的数据库查询。

合取查询是最典型的商业关系数据库使用的查询类型, 既可以用 SQL 语句表示也可以用 DATALOG 语言表示。因为 DATALOG 语言比 SQL 语句在表达模式方面更便于理解,采用 DATALOG 表示合取查询,并使用 DATALOG 表示的合取查询作为模式描述语言。

定义 5 合取查询包含关系和等价关系的定义。

如果对于数据库 D 的模式的任意数据库实例,一个合取查询 Q_1 的结果集总是另一个合取查询 Q_2

结果集的子集，则称 Q_1 包含于 Q_2，记为 $Q_1 \subseteq Q_2$。如果 $Q_1 \subseteq Q_2$ 并且 $Q_2 \subseteq Q_1$，则称 Q_1 和 Q_2 是等价的，记为 $Q_1 \equiv Q_2$。

检测合取查询 Q_1 包含于 Q_2 的方法与算法有很多，选择使用基于规范数据库的方法[8]：

1）建立查询 Q_1 的规范数据库集合 CDBS（Q_1）；

2）提交 Q_1 和 Q_2 到所有的规范数据库 $d \in$ CDBS（Q_1）。如果对于每一个 $d \in$ CDBS（Q_1），Q_1 头部谓词对应的关系总是 Q_2 头部谓词对应的关系的子集，那么 $Q_1 \subseteq Q_2$。

2.2 发现任务定义

笔者的目的在于发现一个多关系数据库中的全部多关系频繁模式。文献[9]给出了一个频繁模式发现问题一般定义，根据该定义，形式化定义多关系频繁模式发现任务如下。

定义 6　给定一个多关系数据库 D，一个多关系模式表示语言 L，一个选择谓词 q，多关系频繁模式发现任务即为发现一个关于 D，L，q 的理论 Th(L, D, q)，使得 Th(L, D, q) = $\{Q \in L \mid q(D, Q)$ 为真$\}$。$q(D, Q)$ 为真当且仅当 Q 在 D 中的频度不小于频度阈值。频度也被称为支持度。Th(L, D, q)中的模式称为多关系频繁模式。

定义 7　面向语义的精简化的多关系频繁模式任务定义。当存在以物化视图集合 V 表达的先验背景知识时，发现定义 6 中的 Th(L, D, q)，同时对于 Th(L, D, q)中的模式 Q 应符合以下 2 个条件的多关系频繁模式：

1）Q 中不得存在这样的 V 中的视图关系文字 v：如果去掉 v 所得的合取查询与原始合取查询等价（消除模式内部存在语义冗余成分）。

2）Th(L, D, q)中不存在与 Q 合取查询等价的其他多关系频繁模式（消除结果集中存在的冗余等价模式）。

对于这一形式化定义，需要进一步对于多关系模式表示语言 L 进行说明，并给出频度定义。另外，模式的搜索空间需要精化算子予以构建，还需对精华算子进行具体说明。

2.2.1　多关系模式语言

使用一阶逻辑语言表示多关系频繁模式。首先给出一阶逻辑语言作为知识表示方式的几种类型。

一阶数据库。一个一阶数据库可以被看作为一个单一的一阶逻辑闭公式。

子句数据库。一个子句数据库是一个一阶数据库，并且由子句的合取组成。

规范数据库。一个规范数据库是一个子句数据库，其中的子句最多有一个头原子，子句体部可能也含有负文字。

确定性数据库。一个确定性数据库是一个规范数据库，其中的子句体部只有正文字没有负文字。

DATALOG 数据库。一个 DATALOG 数据库是一个确定性数据库，其中涉及的词项不含有函数。

关系数据库。一个关系数据库是一个 DATALOG 数据库，其中不含有递归规则。关系数据库逻辑上可以表示为不含递归的 DATALOG 数据库，所以使用 DATALOG 表示的合取查询作为模式描述语言。具体来说，一个多关系频繁模式是一个一阶逻辑谓词合取式，其中每个文字的参数不包含函数，即要么是常数要么是变量。为了描述涉及多个关系的模式，语言必须能够描述关系（relation）之间的关系（relationships），同时能够描述对于各关系属性的值约束，因此，模式语言有两种一阶逻辑文字。

第一种类型文字为 m 元谓词。一个 m 元谓词表示多关系数据库 D 中的一个同名 m 元关系，谓词的参数代表相应关系的属性。m 元谓词通过相互之间的共享变量表示相应关系之间的联系，这种联系规定为关系在数据库定义中的主外键联系或其他关系间等值联系（为了叙述方便，在不引起歧义的情况下，以下把主外键关系或其他关系间等值联系统称为主外键关系），从而连接文字可以在模式中引入关系和主外键联系。称这种谓词为连接文字。

第二种文字是变量与变量之间或变量与常数值之间的算术比较式。称这种文字为赋值文字。赋值文字表达连接文字表示的关系属性上的约束。

为了计算多关系模式的频率，必须确定在多关系数据库中哪一个关系是核心考虑对象。称这种关系为目标关系，表示目标对象的文字为键文字，统一用 key 予以表示。因此，一个多关系模式至少包含一个文字 key。

假定关系数据库 D 由关系集合$\{r_1, r_2, r_3, r_4\}$组成，目标关系为 r_1，各关系之间的主外键关系为 $F = \{r_1[2] - r_2[1], r_1[3] - r_3[1]\}$。一个合法的多关系模式应为 $Q = r_1(X, Y, Z), r_2(Y, U), r_3(Z, R), R = \text{medium}$。

2.2.2　多关系模式频度

一个多关系模式的覆盖是满足该模式描述的在

目标关系中的非重复元组集合,其中每个目标关系中元组由其主键值代表。用如下关系代数操作符形式定义一个多关系模式 Q 的覆盖。

定义 8　多关系模式覆盖的定义。给定一个多关系模式 Q ,其覆盖为

$$\mathrm{cov}(Q) = \pi_K(\sigma_c(r_1 \times r_2 \times \cdots \times r_n)) \quad (1)$$

其中,作为笛卡儿积参数的 $r_i(1 \leqslant i \leqslant n)$ 表示模式中由连接文字引入的同名关系,选择操作符 σ_c 的条件 c 由连接文字表示的连接关系和赋值文字表示的值约束组成。在投影操作符中的 K 参数表示目标关系的主键。

基于上述覆盖的定义,给出多关系模式的频度定义如下:

定义 9　假定 D 是一个多关系数据库,r_1 是目标关系,Q 是一个多关系模式,则 Q 的频度为

$$\mathrm{freq}(Q, D, r_1) = |\pi_K(\sigma_c(r_1 \times r_2 \times \cdots \times r_n))|/|(_K(r_1)| \quad (2)$$

一个多关系模式的频度也称为支持度。一个多关系频繁模式是其支持度不小于用户给定的最小支持度阈值的多关系模式。

2.2.3　优化的精化算子

一般性地使用上述语言,不仅会导致无限大的搜索空间,另外还会产生大量无趣的模式。这一问题已经在机器学习领域的声明性语言偏置研究领域得以较好地解决。这里,改编了来自多关系子组发现系统的 Midos [10] 的声明性语言偏置及其精化算子,来指导限制搜索空间。应该指出的是,主流多关系频繁模式发现算法使用的是来自多关系决策树方法 TILDE [11] 的声明性语言偏置及其相应的精化算子。这一偏置下,搜索空间会产生语法等价的冗余模式,从而导致等价测试巨大的时间消耗。FARMER 算法对于该语言偏置作出了进一步的约束,从而避免等价的冗余模式的产生,但是却带来了模式表达能力的问题。使用的 Midos 语言偏置能够避免上述问题,实验结果部分,验证了这一结果。

假设多关系数据库 D 有关系集合 $R = \{r_1, r_2, \cdots, r_m\}$ 以及 R 中的主外键关系 $F = \{f_1, f_2, \cdots, f_n\}$。如果由关系 r 开始存在多个主外键关系,假定这些连接是有序的,每个这样的连接附以顺序值 o_r。

定义 10　变量的相容性。设 $C = \{L_1, L_2, \cdots, L_n\}$ 是文字的集合。V 是首次出现在 L_i 中的一个变量,其相应关系名为 r,表示在 r 的位置为 p_i 的属性。U 是首次出现在 L_j 中的一个变量,其相应关系名为 s,表示在 s 的位置为 p_j 的属性。V 和 U 是关于 C 和连接集合 F ($V \infty_{C,F} U$) 相容的,当且仅当 $r[p_i] - s[p_j] \in F$。对两个变量集合 V 和 U, 进一步定义 $V \infty_{C,F} U := \{(V, U) \in V \times U \mid V \infty_{C,F} U\}$。如果从上下文可以清楚符号的具体含义,将忽略下标。

定义 11　L 的精化算子定义。设以 $V := \mathrm{vars}(Q)$ 为变量的 $Q := L_1, \cdots, L_n$ 是将要特殊化的模式,精化算子 ρ 可以定义如下:

1) 对任意 Q 中的 L_1, Q': $= L_1, \cdots, L_{i-1}, \rho(L_i), \cdots, L_n$ 属于 $\rho(Q)$, $[o = i, 1]$;

2) h 中任意的 $L_i = r(V_1, \cdots, V_{a(r)})$,使得 $r[m] - s[k] \in F$, 设 $U = \{U_1, \cdots, U_{k-1}, U_{k+1}, \cdots, U_{a(s)}\}$ 一个新变量的集合,即 $\mathrm{vars}(Q) \cap U = \emptyset$;那么 $Q' := L_1, \cdots, L_n, s(U_1, \cdots, U_{k-1}, U_{k+1}, \cdots, U_{a(s)})$,

$\Lambda_{U \in U} \mathrm{any}(U)\ \Lambda_{(V,U) \in V \times U} \mathrm{any}(V, U)$,

属于 $\rho(Q)$, $[o = i, 2, o_r(r[m] - s[k])]$。

定义 12　L 的单一文字精化算子定义。

1) 如果 $L = \mathrm{any}(X)$, X 具有有序值 $\{v_1, v_2, \cdots, v_{n-1}, v_n\}$。若 $n = 2$, $X = v_1$ 与 $X = v_2$ 属于 $\rho(Q)$, 否则 $X \geqslant v_2$,且 $X \leqslant v_{n-1}$ 属于 $\rho(Q)$。

2) 如果 $L = X \geqslant v$ ($X \leqslant v$), X 具有有序值 $\{\cdots, u, v, w, \cdots\}$。若 $w = v_n$ 且 $u = v_1$, 则 $X = v$ 与 $X = w$ ($X = u$) 属于 $\rho(Q)$, 否则 $X = v$,且 $X \geqslant w$ 与 $X \leqslant u$) 属于 $\rho(Q)$。

3) 如果 $L = \mathrm{any}(X)$,X 是树状结构的, 且在 X 的值树中, b_0 是值数的根结点,则 $X = b_0$ 属于 $\rho(Q)$。

4) 如果 $L = X = a$, X 是树状结构的, 且在 X 的值树中, b_1 到 b_n 是 a 的子结点。那么 $X = b_1, \cdots, b_n$ 属于 $\rho(Q)$。

5) 如果 $L = \mathrm{any}(X, Y)$,则 $X = Y$ 属于 $\rho(Q)$。

在方括号中的 o 表示 ρ 中的精化索引数值。如果进一步要求精化算子符合条件

$$\rho_{\mathrm{opt}}(Q) := \{Q' \in \rho(Q) \mid o(Q') > o(Q)\},$$

那么上述精化算子 ρ_{opt} 是一个优化的精化算子,通过这样的精化算子可以保证一个模式整个精化过程中仅出现一次。

定理 1　$\rho_{\mathrm{opt}}(Q) := \{Q' \in \rho(Q) \mid o(Q') > o(Q)\}$ 是一个优化的精化算子。

证明:

1）考虑搜索空间中的一个节点，有两条不同的精化路径 path1 和 path2 由其开始。如果 path2 选择一个比 path1 更为向右的文字予以精化，那么 path2 从不会返回到由 path1 精化的文字。因为它的精化索引参数高于 path1 的精化参数。

2）因为所有的新文字引入新变量，从不同已存在文字通过主外键联系引入的文字是不同的。

3）如果 path1 和 path2 的区别在于前者精化现存文字，而后者引入新文字，那么 path2 永远不会再次精化当前已存在的文字，因为它的精化索引参数值不允许这种情况发生。

3 面向语义的精简化多关系频繁模式发现算法

3.1 主体算法与候选模式产生算法

基于上述的一系列定义提出了面向语义的精简化多关系频繁模式发现算法 CMRFPDA，算法实行的是由一般到特殊的逐层展开、宽度优先的搜索策略。算法的主体流程如下。

输入：Multi-relational database D; Object relation r_1 and corresponding Literal key, the minimum support minfreq

输出：All conjunctive queries $Q \in L$ with frq(Q,D,n) ≥minfreq

1. Initialize level d: =1;
2. Initialize the set of candidate queries C_1: = key;
3. Initialize the set of frequent queries F: =Ø;
4. While C_d not empty
5. for each $Q \in C_d$
6. Count frq (Q,D,n);
7. if frq(Q,D,n) ≥minfreq then
8. F: = $F \cup Q$;
9. $C_{d+1} = C_{d+1} \cup$ GEN(Q);
10. end if
11. d: = d +1;
12. Return F;

在 CMRFPDA 输入中的多关系数据库 D 包含有表示背景知识的物化的试图及其定义。步骤 9 使用候选模式产生算法 GEN 只精化频繁模式生成下一层的候选模式，算法 GEN 如下。

输入：Multi-relational database D, congjunctive query Q

输出：all conjunctive queries that are refinded by p_{opt} and condensed

1. p: = p_{opt}(Q);
2. for each $Q' \in p$
3. if Q' is intra redundant or inter redundant that
4. p: = p: − {Q'};
5. end if
6. Return p;

在算法 GEN 中，步骤 1 利用上一节定义的优化的精化算子 ρ_{opt} 产生语法非冗余模式，步骤 3 和步骤 4 根据合取查询等价算法消除语义冗余模式。

3.2 候选模式评估的共享计算策略

CMRFPDA 的主体算法在步骤 6 对候选模式频度进行评估，为了提高评估本身的性能，一方面要求所有试图物化；另一方面使用了如下共享计算策略。

给定一些多关系模式 $Q \in L$，设 R 为 Q 的任意精化。在候选评估阶段，R 的频率的计算涉及众多笛卡尔积、选择和投影的算子。计算的复杂度随着 R 的复杂度而增加。平凡地实现这类计算，既费时，又不具可扩展性。由于 R 与 Q 在结构内容上相似，二者的频度计算也相近。提高效率和可扩展性的最好的方法之一是去掉计算冗余，在评估 R 的过程中尽量多使用 Q 的计算结果。

对任意多关系频繁模式 Q，用目标关系和非目标关系所包含的全部对象构建一个关系。将这种关系称为外延关系，将 Q 的外延关系记为 I_Q。I_Q 的具体定义为 $I_Q = \pi_{k_1,k_2,\cdots,k_n}(\sigma_c(r_1 \times r_2 \cdots \times r_n))$。投影算子 $\pi_{k_1,k_2,\cdots,k_n}$ 中的属性 k_i($1 \leqslant i \leqslant n$) 为关系 r_i($1 \leqslant i \leqslant n$) 的主键。将 I_Q 中每个视为模式 Q 的一个外延。

从 I_Q 可以容易地计算出模式 Q 的频率为 $|\pi_K(IQ)|/|\pi_K(r_1)|$，其中 K 为目标关系的主键。

对 Q 的任意精化 R，基于外延关系 I_Q，可以容易地得到关系 I_R 的外延。详细方法如下。

1）设 R 是通过对 Q 中的赋值文字 S 精化而得到的，且 S 不是 any(X, Y) 类型的，该文字中相关的变量第一次出现在连接文字 r_a 中。设 S' 表示 S 的精化，根据外延关系的定义，I_R 可被认为是 I_Q 中外延的集合，且每个外延中来自关系 r_a 的每个对象

应该满足在关系 r_a 上由 S'确定的条件。这样，I_R 可通过 $\pi_{k_1,k_2,\cdots,k_n}(\sigma_j(I_Q \times \sigma_s(r_a)))$计算得到，其中 s 表示 S'确定的条件，j 表示$(I_Q K_a = r_a K_a)$。

2）设 R 是通过对 Q 中的赋值文字 S 精化而得到的，且 S 是 any(X, Y)类型的，文字中的两个变量第一次出现在连接文字 r_a和 r_b，设 S'表示 S 的精化。I_R 可被认为是 I_Q 中外延的集合且每个外延来自 r_a 中的对象与 r_b 中的对象可以通过 S'确定的连接条件进行连接。这样，I_R 可通过 $\pi_{k_1,k_2,\cdots,k_n}(\sigma_j(I_Q \times \sigma'_j(r_a \times r_b)))$计算得到，其中 j 表示$(I_Q K_a = r_a K_a) \wedge (I_Q K_b = r_b K_b)$，$j'$表示$(I_Q K_a = r_a K_a)$。

3）设 R 是通过在 Q 中添加一个新的连接文字 r_b，r_b 与 Q 中的连接文字 r_a 有共同的变量。这样，I_R 可通过 $\pi_{k_1,\cdots,k_n,k_b}(\sigma_j(I_Q \times \sigma'_j(r_a \times r_b)))$计算得到，其中 j 表示 $I_Q K_a = r_{a'} K_a$，j'表示 $r_{a'} K_a = r_{b'} K_b$，K_b 表示关系 r_b 的主键。

将与多关系模式的频率的定义相关的原始计算与上面描述的计算进行对比，可以发现，效率和可伸缩性的提高是明显的。对一个具有 n 个连接字的关系 R，这里的计算至多需要执行 3 次连接和 1 次选择，而平凡化方法至少需要 $n-1$ 次连接和 1 次选择。

4 实验结果

使用 C 语言实现了 CMRFPDA，原始关系数据库使用 Oracle 8.0。所有的实验运行平台具有以下参数的 PC：

- P4 1.7 GHz CPU ×2；
- 512 MB RAM；
- Windows 2000。

使用了 2 个多关系数据挖掘领域知名的基准测试数据集。第一个是描述分子结构的 Mutagenesis 数据集[12]，数据集中每一个关系的属性与元组数量在表 1 中描述，其中 Molecule 关系是目标关系。第二个是描述基因组的 Saccharomyces Cerevisiae 数据集[13]，对数据集的描述在表 2 中，其中 Gene 是目标关系。在每个测试集合都加入了 7 个视图表示先验背景知识。

表 1 Mutagenesis 数据集说明

Table 1 The specification of mutagenesis dataset

Relation name	# tuples	# attributes
Molecule	188	5
Atom	4 893	4
Molecule – Atom	4 893	2
Bond	5 244	3

表 2 Saccharomyces Cerevisiae 数据集说明

Table 2 The specification of saccharomyces cerevisiae dataset

Relation name	# tuples	# attributes
Gene	4 053	4
Hasfunction	12 839	2
Funcat	1 307	7
Homology	1 044 816	3
Eval	3 618 919	3
Swissprot	46 163	4
Databaseref	196 535	2
Keyword	14 270	2
Class	3 105	2
Secondar	384 165	4
Ssea	8 211 378	4

Mutagenesis 数据集是一个相对较小规模的数据集合，使用该测试集主要是测试方法的效率。Saccharomyces Cerevisiae 数据集是一个相对较大规模的数据集合，使用该测试集主要是测试方法的可扩展性。性能比较的试验结果分别见图 1a 与图 1b，图 1 中的刻度比例是对数化的。结果表明 CMRFPDA不论是在效率还是可扩展性方面都优于经典算法 C – ARMR。

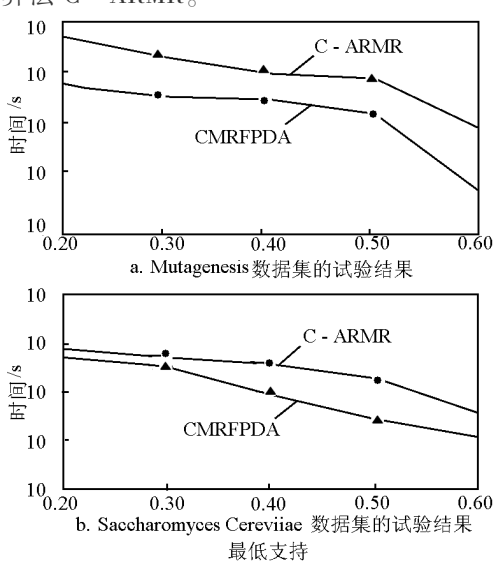

图 1 试验结果

Fig. 1 Experiment results

5 结语

笔者提出了有别于主流基于归纳逻辑程序设计的面向语义的精简化多关系频繁模式发现方法（CMRFPDA），其优势表现在：

1）根据关系数据库理论与技术定义了面向语义的精简化多关系频繁模式发现方法，从而使得 CMRFPDA 方法能够更有效的为与关系数据库有效集成奠定了基础；原始数据无须向表示方式方面转

化,节省了预处理时间;方法能够为数据库研究者和使用者较方便地理解和使用,为方法的有效实践和广泛使用创造了条件。

2)基于合取查询及其包含关系概念与方法,实现了消除两种语义冗余的功能。

3)使用了一个优化的精化算子构建搜索空间,这一精化算子一方面有效的利用了关系数据库隐含的数据模式特征,从而能够自然地构建有趣形态的模式,并且避免了模式语法等价变体的产生,减少了模式产生阶段的计算量;使用了一个候选模式评估共享计算策略,从而降低了方法评估阶段的时间复杂性。试验表明,方法整体上具有良好的效率和可扩展性。

参考文献

[1] Agrawal R, Srikant R. Fast algorithms for mining association rules in large databases [A]. Proceedings of the 20th International Conference on Very Large Data Bases [C]. Morgan Kaufmann Publishers, Inc, San Francisco, CA, USA, 1994. 487 - 499

[2] Hipp J, Guntzer U, Nakaeizadeh G. Algorithms for association rule mining - a general survey and comparison [J]. ACM SIGKDD Explorations, 2000, 2(1): 58 - 64

[3] Dzeroski S, Lavrac N. Relational Data Mining [M]. Springer, Berlin, 2001

[4] Dehaspe L, Toivonen H. Discovery of frequent datalog patterns [J]. Data Mining and Knowledge Discovery, 1999, 3(1): 7 - 36

[5] Nijssen S, Kok J N. Faster association rules for multiple relations [A]. Proceedings of the 17th International Joint Conference on Artificial Intelligence [C]. Morgan Kaufmann Publishers, Inc, Seattle, USA, 2001. 891 - 896

[6] De Raedt L, Ramon J. Condensed representations for inductive logic programming [A]. Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning [C]. AAAI Press, USA, 2004. 438 - 446

[7] Ullman J. Principles of Database and Knowledge - base Systems, Volume 1 [M]. Computer Science Press, USA, 1988

[8] Miguel R, Nieves R. A general procedure to check conjunctive query containment [J]. Artificial Intelligence, 2002, 38(7): 489 - 529

[9] Mannila H, Toivonen H. Levelwise search and borders of theories in knowledge discovery [J]. Data Mining and Knowledge Discovery, 1997, 1(3): 241 - 258

[10] Wrobel S. An algorithm for multi - relational discovery of subgroups [A]. Proceedings of the First European Symposium on Principles of Data Mining and Knowledge Discovery [C]. Springer, Berlin, Germany, 1997. 78 - 87

[11] Blockeel H, Raedt L D. Top - down induction of first - order logical decision trees [J]. Artificial Intelligence, 1998, 101(1 — 2): 285 - 297

[12] King R, Muggleton S, Srinivasan A, et al. Structure - activity relationships derived by machine learning: the use of atoms and bonds and their connectivities to predict mutagenicity in inductive learning programming [A]. Proceedings of the National Academy of Sciences [C]. USA, 1996;93(1) 438 - 442

[13] ILP 2005 Challenge, Bonn, Germany [EB/OL]. http://www.protein - logic.com/data.html, 2005

Semantically condensed multi-relational frequent pattern discovery based on conjunctive query containment

Yang Bingru, Zhang Wei, Qian Rong

(*School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China*)

[**Abstract**] Multi-relational data mining is one of rapidly developing subfields of data mining. Multi-relational frequent pattern discovery approaches directly look for frequent patterns that involve multiple relations from a relational database. While the state-of-the-art of multi-relational frequent pattern discovery approaches is based on the inductive logical programming techniques, we propose an approach to semantically condensed multi-relational frequent pattern discovery based on conjunctive query containment in terms of the theory and technique of relational database. With the novelty of the groundwork, the proposed approach deals with two kinds of semantically redundant problems. In theory and experiments, it shows that our approach improve the understandability, function, efficiency and scalability of the state-of-the-art of multi-relational frequent pattern discovery approaches.

[**Key words**] multi-relational data mining; frequent pattern discovery; conjunctive query; condensed pattern