一种多约束关联挖掘算法*

关 心[1,2], 李广原[3]

(1. 北京科技大学 自动化学院, 北京 100083; 2. 湛江师范学院 信息科学与技术学院, 广东 湛江 524048; 3. 广西师范学院 计算机与信息工程学院, 南宁 530023)

摘 要: 约束关联挖掘是在把项或项集限制在用户给定的某一条件或多个条件下的关联挖掘,是一种重要的关联挖掘类型,在现实中有着不少的应用。但由于大多数算法处理的约束条件类型单一,提出一种多约束关联挖掘算法。该算法以 FP-growth 为基础,创建项集的条件数据库。利用非单调性和单调性约束的性质,采用多种剪枝策略,快速寻找约束点。实验证明,该算法能有效地挖掘多约束条件下的关联规则,且可扩展性能很好。

关键词: 数据挖掘; 多约束关联挖掘; 单调性约束; 非单调性约束

中图分类号: TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2012)04-1294-03

doi:10.3969/j.issn.1001-3695.2012.04.025

Efficient algorithm for mining association rules with multiple constraints

GUAN Xin[1,2], LI Guang-yuan[3]

(1. *School of Automation, University of Science & Technology Beijing, Beijing* 100083, *China*; 2. *School of Information Science & Technology, Zhanjiang Normal University, Zhanjiang Guangdong* 524048, *China*; 3. *College of Computer & Information Engineering, Guangxi Teachers Education University, Nanning* 530023, *China*)

Abstract: Association rules mining with constraints is an important association mining method. It can mine the rules according to the given itemsets constraints. Because most of algorithms can only deal with single type of constraints, this paper proposed an efficient algorithm for mining association rules with multiple constraints. The algorithm was based on FP-growth algorithm, and generated the condition database of frequent itemsets. Making use of constraint characteristics of anti-monotone and monotone, moreover, using some prune techniques, to find the constraint checking points, the proposed algorithm was efficient for mining association rules with multiple constraints. Experimental results show that the proposed algorithm is efficient both in running time and scalability.

Key words: data mining; association rules mining with multiple constraints; monotone constraint; anti-monotone constraint

0 引言

关联挖掘是数据挖掘的一个重要内容,目前已提出了不少的挖掘算法。如经典的 Apriori 算法、基于划分的算法、基于抽样的算法、动态挖掘、并行与分布式挖掘算法等。约束挖掘是一种重要的关联挖掘类型,它根据用户提出的条件有针对性地挖掘隐藏在数据里的规则,但当数据量很大时,挖掘的规则通常都比较大,而且有些规则是无用的。通常采用支持度和置信度来挖掘规则生成的数目。但这种方法局限性比较大,因为它不能使用户直接参与到挖掘过程中,所以约束关联挖掘是解决这个问题的一种挖掘方法。在多约束条件下,挖掘难度比单约束要难得多,主要原因在于,当存在两个以上的约束时,在某个约束点,对于一个约束的子集,约束是成立的,而对另一个约束来说,恰恰相反。目前,已提出了一些有效的约束关联挖掘算法[1~8],但这些算法均是针对单约束或多个同一类型的约束的挖掘。本文给出一个多约束关联挖掘算法,该算法利用非单调性和单调性约束的性质,采用多种剪枝策略,有效地挖掘多约束条件下的规则。

1 问题描述

设 $I=\{i_1,i_2,\cdots,i_n\}$ 为 n 个不同项的集合,$X\subseteq I$ 称为项集,如果 X 包含 k 个项目,则称 X 为 k-项集。项集 X 的支持度记为 $\mathrm{supp}(X)$,是指包含 X 的事务数占数据集中所有事务数的百分比。如果 X 的支持度不小于用户设定的最小支持度阈值 s,即 $\mathrm{supp}(X)\geqslant s$,则 X 为频繁项集。

约束条件 C 可以表示为项 I 的一个谓词,$C:2^I\rightarrow\{\mathrm{true},\mathrm{false}\}$,一个项集 X 满足约束 C,当且仅当 $C(X)=\mathrm{true}$,数据集中所有满足约束 C 的项集可表示为 $SC(I)=\{S\mid S\subseteq I\wedge C(S)=\mathrm{true}\}$。

定义 1 多维约束是指作用于多维属性上的约束,约束可以表示为 n 个多维子约束的合取(析取),$n\geqslant 1$。

这里的多维属性是指其中的一个属性有两个属性项,如某商品有若干属性、价格、成本等。某商品 M 的约束可以表示为 $(M.\mathrm{cost}\leqslant p_1)\wedge(M.\mathrm{price}\leqslant p_2)$,$p_1$ 和 p_2 是设定值,$M.\mathrm{cost}\leqslant p_1$ 表示商品 M 的成本低于或等于 p_1。

定义 2 给定一个约束 C,对于任一项集 X,如果 $\forall Y\subseteq X$,

收稿日期: 2011-09-19; **修回日期:** 2011-10-28 **基金项目:** 国家自然科学基金资助项目(60875029);广西教育厅科研基金资助项目(201106LX302)

作者简介: 关心(1980-),女,辽宁辽阳人,博士研究生,主要研究方向为数据挖掘(wx162@163.com);李广原(1969-),男,副教授,博士研究生,主要研究方向为数据挖掘、智能信息处理.

当 $C(X)=\text{true}\Rightarrow C(Y)=\text{true}$,则约束 C 属于非单调性约束。

定义 3　给定一个约束 C,对于任一项集 X,如果 $\forall Y\supseteq X$,当 $C(X)=\text{true}\Rightarrow C(Y)=\text{true}$,则约束 C 属于单调性约束。

引理 1[7]　非单调性约束的合取也满足非单调性。

证明　设定 $C=C_1\wedge C_2\wedge\cdots\wedge C_k$,$C_i$ 是非单调性子约束。用反证法,假设 C 不是非单调性约束,则有:如果 $C(X)=\text{false}$,那么存在 X 的超集 X',使得 $C(X')=\text{true}$。一方面,因为 $C(X)=\text{false}$,所以至少有一个 $C_i(1\leqslant i\leqslant k)$,满足 $C(X_i)=\text{false}$;另一方面,因为 $C(X')=\text{true}$,根据非单调性约束定义,$C(X)=\text{true}$,与假设矛盾,引理得证。

引理 2　单调性约束的合取也满足单调性。

证明　设定 $C=C_1\wedge C_2\wedge\cdots\wedge C_k$,$C_i$ 是单调性子约束。用反证法,假设 C 不是单调性约束,则有:如果 $C(X)=\text{false}$,那么存在 X 的子集 Y,使得 $C(Y)=\text{true}$。一方面,因为 $C(X)=\text{false}$,所以至少有一个 $C_i(1\leqslant i\leqslant k)$,满足 $C(X_i)=\text{false}$;另一方面,因为 $C(Y)=\text{true}$,根据单调性约束定义,$C(X)=\text{true}$,与假设矛盾,引理得证。

引理 3　非单调性约束的析取满足非单调性,单调性约束的析取满足单调性。

引理 4　非单调约束的任何合取和析取的组合都是非单调的,单调性约束的任何合取和析取的组合都是单调的。

2　AMMC 算法

2.1　算法的基本思想

AMMC(association rules mining with multiple constraints)算法可以同时处理两类约束的析取或合取,即非单调性约束析取(合取)和单调性约束析取(合取)。表 1 为示例数据库,表 2 是有关数据库中项的属性描述。

表 1　事务数据库 T

TID	items
1	A,B,C
2	B,C,D
3	A,B,C,D
4	C,D

表 2　事务数据库 T 中的项和属性

itemID	cost	price
A	80	100
B	60	70
C	110	120
D	50	90

在给出的示例数据库中,本文考虑两类具体的约束一个是非单调约束 $C_1=C_{11}\vee C_{12}$(其中 $C_{11}=\max(S.\text{cost})\leqslant\min(S.\text{price})$,$C_{12}=S.\text{cost}\leqslant 60$),另一个是单调性约束 $C_2=C_{21}\wedge C_{22}$(其中 $C_{21}=\text{total}(S.\text{price})\geqslant 100$,$C_{22}=\text{total}(S.\text{cost})\geqslant 50$)。$S$ 是项集,S 中的每一个项有 cost 和 price 两个属性。算法的结果是输出符合约束条件 $C_1\wedge C_2$ 的数据库中的频繁项集。

定义 4[9]　a)设 pc 代表两个序列之间具有前缀关系的谓词,$pc(s_1,s_2)=\text{true}$,表示序列 s_1 是 s_2 的前缀。b)项集 β 称为事务集$\langle\text{tid},I_t\rangle$的最大 α 投影,当且仅当:(a)$\alpha\subseteq I_t\wedge\beta\subseteq I_t$;(b)$pc(\alpha,\beta)=\text{true}$;(c)不存在 β 的超集 γ,使得$\gamma\subseteq I_t\wedge pc(\alpha,\gamma)=\text{true}$。c)$\alpha$ 的条件数据库是包含 α 的事务集的最大 α 投影组成的集合。

这里,基于 FP-tree 结构的 α 条件数据库记做 $T|_\alpha$。

定义 5[9]　设 α 是一个频繁项集,λ 是 α 条件数据库中频繁项的集合,则 $\alpha\cup\lambda$ 就构成了 $T|_\alpha$ 中潜在的最大频繁项集。

为了对算法描述更清晰,先给出几个引理。

引理 5[9]　设 β 是 $T|_\alpha$ 中频繁项的集合,γ 是 $T|_{\{a\}\cup\alpha}$中的频繁项集合的子集,C 为非单调性约束,如果在 $T|_\alpha$ 中,$C(\alpha\cup\beta)=\text{true}$,则 $C(\{a\}\cup\alpha\cup\gamma)=\text{true}$,$a\in\beta$。

证明　由于$\{a\}\cup\alpha\cup\gamma\subseteq\alpha\cup\beta$,而 C 为非单调性约束,由非单调约束的性质马上可以得到结论。

引理 6　设 γ 是 $T|_{\{a\}\cup\alpha}$中频繁项集合中的子集,C 为非单调性约束,如果在 $T|_\alpha$ 中,$C(\{a\}\cup\alpha)=\text{false}$,$a$ 是 $T|_\alpha$ 中的一个频繁项,那么不需要生成 $T|_{\{a\}\cup\alpha}$,因为$\{a\}\cup\alpha\cup\gamma$ 包含$\{a\}\cup\alpha$,$C(\{a\}\cup\alpha\cup\gamma)=\text{false}$。

引理 6 的证明与引理 5 类似。

引理 7　设 β 是 $T|_\alpha$ 中频繁项的集合,γ 是 $T|_{\{a\}\cup\alpha}$中的频繁项集合的子集,C 为单调性约束,如果在 $T|_\alpha$ 中,$C(\alpha\cup\beta)=\text{false}$,则 $C(\{a\}\cup\alpha\cup\gamma)=\text{false}$,$a\in\beta$。

证明　根据单调性约束的性质直接得到结论。

引理 8　设 γ 是 $T|_{\{a\}\cup\alpha}$中频繁项集合中的子集,C 为单调性约束,如果在 $T|_\alpha$ 中,$C(\{a\}\cup\alpha)=\text{true}$,$a$ 是 $T|_\alpha$ 中的一个频繁项,那么不需要生成 $T|_{\{a\}\cup\alpha}$,因为$\{a\}\cup\alpha\cup\gamma$ 包含$\{a\}\cup\alpha$,$C(\{a\}\cup\alpha\cup\gamma)=\text{true}$。

证明与引理 7 类似。

引理 9　设 $X=\{x_1,x_2,\cdots,x_n\}$($n\geqslant 2$)为一项集,$C=C_1\rho C_2\rho\cdots\rho C_k$,$\rho\in\{\wedge,\vee\}$,$C_i(1\leqslant i\leqslant k)$为单调性约束,根据引理 4,$C$ 为单调性约束,且 $C_i(X)=C_i(x_1x_2\cdots x_n)=\text{true}$,设 x_1,$x_2,\cdots,x_n$ 是按其与 C 有关的属性值从大到小的顺序排列,如果每个约束 C_i 删除前 $k_i(k_i\geqslant 1)$个项,使得剩余项组成的项集不满足约束 C_i,那么对于任何 $\max(k_i)$个项,删除它们后,剩余项集不满足约束 C。

引理 9 作为算法的一个剪枝策略。

2.2　算法描述

算法 AMMC

输入:事务数据库 T;最小支持度 s;非单调约束 $C_1=C_{11}\vee C_{12}$,$C_{11}=\max(S.\text{cost})\leqslant\min(S.\text{price})$,$C_{12}=S.\text{cost}\leqslant 50$;单调性约束 $C_2=C_{21}\wedge C_{22}$,$C_{21}=\text{total}(S.\text{price})\geqslant 100$,$C_{22}=\text{total}(S.\text{cost})\geqslant 50$;初始化 C_{11}和 C_{12}的标志位 $\text{flag}_1=\text{flag}_2=0$;数据库中的事务保存为 FP-tree 结构。

输出:所有同时满足 C_1 和 C_2 的频繁项集,即$(C_{11}\vee C_{12})\wedge(C_{21}\wedge C_{22})$。

AMMC($\alpha,T|_\alpha,\text{flag}_1,\text{flag}_2$)

a)从 $T|\alpha$ 的 FP-tree 头表中得到频繁项集 L 以及它们的支持度。

b)$\beta=L$;if $C_{2i}(\beta\cup\alpha)=\text{false}$,$i=1,2$,then exit, there are no frequent itemsets that satisfy $C_1\wedge C_2$。

c)if $C_2(\beta\cup\alpha)=\text{true}$,应用引理 9 得出满足约束 C_2 的项集中项的个数 N。

d)for each $a\in\beta$,AMMC($\alpha,T|_\alpha,0,0$)

```
    if (|L|) < N//生成的频繁项体数小于 N
      continue
    else
    for each χ ∈ L
      if flag_q = 0, check , C_1q(χ∪a),
      if C_1q(χ∪a) = true then flag_q = 1
      //通过 flag_q 的取值决定C_1(χ∪a)
      if C_1(χ∪a) = true 生成 T_χ∪a
      if (|L|) > N//输出生成的频繁项集
    end for
  end for
```

3　实验分析

为了测试算法的性能,本文选择算法 FP-growth$^+$ 作为比较的对象,FP-growth$^+$ 采用 FP-growth 算法来找出所有频繁项集,然后再从中筛选出满足约束的频繁项集。测试程序运行在 Windows XP 系统上,CPU 为 PIV 2.4 GHz,内存为 1 GB,编程语

言为 C ++,采用 IBM 数据生成器[10]来生成测试数据集。数据集里的事务数为 20 ~ 100 K,项集的平均长度为 12,最大频繁项集的平均长度为 20。本文对每个项赋予两个属性,即 cost 和 price,属性值采用随机赋值。这里给出一个约束选择率的定义。所谓约束选择率,是指数据集中的频繁项集不满足约束占全部频繁项集的百分比。实验结果如图 1 ~3 所示。图 1 是测试约束选择率和运行时间的对应关系,事务数据为 80 K,图 2 是测试事务大小与可扩展性的相互关系,图 3 是支持度阈值大小和运行时间对应关系。

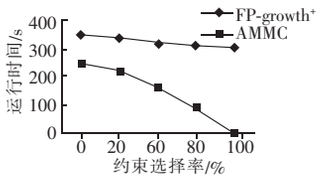

图 1 约束选择率运行结果

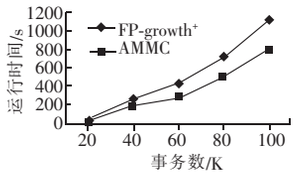

图 2 事务数与可扩展性的对应

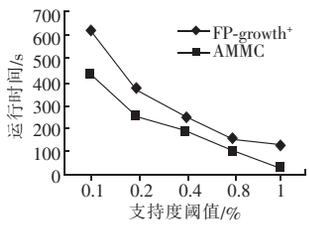

图 3 支持度阈值运行结果

从实验结果可以看出,AMMC 算法对多种约束关联是有效的,且扩展性能较好,充分将非单调性约束和单调性约束的性质及有效的剪枝技术相结合,可以在很大程度上减少不必要的数据扫描次数,从而使其无论是在运行时间还是在可扩展性方面均优于 FP-growth+ 算法。

4 结束语

约束关联挖掘最耗时的地方是在对约束点寻找和验证上,采用多种有效的剪枝技术是关键。本文给出一个基于单调性约束和非单调性约束的多约束关联挖掘算法,算法利用了这两类约束的性质,采用有效的剪枝技术,快速寻找约束点。通过实验证明,算法是有效的,可扩展性能较好。当前面向动态、多关系、分布式数据挖掘是数据挖掘的一个主流方向之一,面向此类数据的约束挖掘是笔者今后的一个研究方向。

参考文献:

[1] SRIKANT R, VU Q, AGRAWAL R. Mining association rules with item constraints[C]//Proc of the 3rd International Conference on Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1997:67-73.

[2] LAKSHMANAN L V S, NG R, HAN Jia-wei, *et al*. Optimization of constrained frequent set queries with 2-variable constraints[C]//Proc of ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 1999:157-168.

[3] BONCHI F, GIANNOTTI F, MAZZANTI A, *et al*. ExAMiner: optimized level-wise frequent pattern mining with monotone constraints [C]//Proc of the 3rd IEEE International Conference on Data Mining. Washington DC: IEEE Computer Society, 2003:11-18.

[4] BONCHI F, LUCCHESE C. On closed constrained frequent pattern mining[C]//Proc of the 4th IEEE International Conference on Data Mining. Washington DC: IEEE Computer Society, 2004:35-42.

[5] BONCHI F, LUCCHESE C. Pushing tougher constraints in frequent pattern mining [C]//Proc of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining. 2005:114-124.

[6] LAKSHMANAN L V S, LEUNG C K, NG R T. Efficient dynamic mining of constrained frequent sets[J]. ACM Trans on Database Systems, 2003, 28(4):337-389.

[7] 方刚. 一种快速挖掘约束性关联规则的算法[J]. 计算机应用与软件, 2009, 26(8):268-270, 280.

[8] 李英杰. 项约束频繁项集挖掘的新方法[J]. 计算机工程与应用, 2009, 45(3):161-164.

[9] LEE A J T, LIN Wan-chuen, WANG Chun-sheng. Mining association rules with multi-dimensional constraints [J]. Journal of Systems and Software, 2006, 79(1):79-92.

[10] AGRAWAL R, SRIKANT R. Fast algorithms for mining association rules in large database[C]//Proc of the 20th International Conference on Very Large Data Bases. San Francisco: Morgan Kaufmann, 1994: 487-489.

(上接第 1229 页)

[4] WIECHERT W. ^{13}C Metabolic flux analysis [J]. Metabolic Engineering, 2001, 3(3):195-206.

[5] CHRISTENSEN B, NIELSEN J. Metabolic network analysis of Penicillium chrysogenum using ^{13}C-labeled glucose[J]. Biotechnology and Bioengineering, 2000, 68(6):652-659.

[6] GOMBERT A K, Dos SANTOS M M, CHRISTENSEN B, *et al*. Network identification and flux quantification in the central metabolism of Saccharomyces cerevisiae under different conditions of glucose repression [J]. Journal of Bacteriology, 2001, 183(4):1441-1451.

[7] ZHAO Jiao, SHIMIZU K. Metabolic flux analysis of Escherichia coli K12 grown on ^{13}C-labeled acetate and glucose using GC-MS and powerful flux calculation method[J]. Journal of Bacteriology, 2003, 101(2):101-117.

[8] RIASCOS C A M, GOMBERT A K, PINTO J M. A global optimization approach for metabolic flux analysis based on labeling balances [J]. Computers & Chemical Engineering, 2005, 29(3):447-458.

[9] BHANDOHAL R K, SINGH M. Evolutionary programming method for optimization[J]. International Journal of Engineering and Information Technology, 2010, 2(1):48-54.

[10] SUN Jun, FANG Wei, XU Wen-bo. A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters [J]. IEEE Trans on Circuits and Systems II: Express Briefs, 2010, 57(2):141-145.

[11] 方伟, 孙俊, 谢振平, 等. 量子粒子群优化算法的收敛性分析及控制参数研究[J]. 物理学报, 2010, 59(6):3686-3694.

[12] 甘敏, 鹏辉. 一种新的自适应惩罚函数算法求解约束优化问题[J]. 信息与控制, 2009, 38(1):24-28.

[13] YANG Jing, WONGSA S, KADIRKAMANATHAN V, *et al*. Metabolic flux estimation: a self-adaptive evolutionary algorithm with singular value decomposition [J]. IEEE/ACM Trans on Computational Biology and Bioinformatics, 2007, 4(1):126-138.

[14] KADIRKAMANATHAN V, YANG Jing, BILLINGS S A, *et al*. Markov chain Monte Carlo algorithm based metabolic flux distribution analysis on corynebacterium glutamicum[J]. Bioinformatics, 2006, 22(12): 2681-2687.

[15] WIECHERT W, SIEFKE C, De GRAAF A A, *et al*. Bidirectional reaction steps in metabolic networks, part II: flux estimation and statistical analysis[J]. Biotechnology and Bioengineering, 1997, 55(1): 118-135.