引用格式:许普乐,纪允. 一种基于 Rymon 枚举树的快速挖掘无关集算法[J]. 轻工学报,2017,32(5):103 – 108.
中图分类号:TP311 **文献标识码**:A
DOI:10.3969/j.issn.2096 – 1553.2017.5.014
文章编号:2096 – 1553(2017)05 – 0103 – 06

一种基于 Rymon 枚举树的快速挖掘无关集算法

A fast algorithm for mining free sets based on Rymon setenumeration tree

许普乐[1],纪允[2]
XU Pu-le[1],JI Yun[2]

1. 芜湖职业技术学院 教务处,安徽 芜湖 241006;
2. 浙江出入境检验检疫局 信息化管理处,浙江 杭州 310016
1. *Teaching Affairs Office*, *Wuhu Institute of Technology*, *Wuhu* 241006, *China*;
2. *Informatization Management Department*, *Zhejiang Entry-Exit Inspection and Quarantine Bureau*, *Hangzhou* 310016, *China*

关键词:
数据挖掘;频繁项集;精简表示;δ 无关集;Rymon 枚举树;剪枝策略

Key words:
data mining; frequent itemsets; concise representation; δ free sets; Rymon setenumeration tree; pruning strategy

摘要:针对传统的挖掘算法在挖掘 δ 无关集时存在重复生成候选项集、遍历子项集等导致挖掘效率过低的问题,提出一个无关集判断定律,进而给出一种快速挖掘无关集算法 FMFS. 该算法利用 Rymon 枚举树作为搜索空间,结合一定的剪枝策略,再利用这个无关集判断定律对候选项集进行快速筛选. 实验结果表明,该算法不仅能够挖掘出所有的无关集,且挖掘过程中的时间消耗优于目前已有算法.

收稿日期:2017 – 06 – 14
基金项目:安徽高校自然科学研究重点项目(KJ2017A552);高校优秀青年人才支持计划重点项目(gxyqZD2016591)
作者简介:许普乐(1980—),男,安徽省芜湖市人,芜湖职业技术学院副教授,主要研究方向为数据挖掘、智能计算.

Abstract: In view that traditional mining δ free sets algorithm exist generating candidate itemsets, traversing all direct subsets, and so on leading to low efficiency, a free sets determine lemma and a fast mining free sets algorithm FMFS were proposed. FMFS used Rymon setenumeration tree as searching space and combined with pruning strategy, and then used the free sets determine lemma to quickly determine the candidate itemset. Experimental results showed FMFS algorithm not only mined all free sets, but also showed better performance than existed mining algorithm.

0　引言

数据挖掘能够在海量数据中挖掘出有用的重要信息，是目前大数据时代的研究重点之一. 其中，频繁项集[1-3]在诸多领域有着广泛的应用，例如分类、聚类、关联规则生成等. 但是从数据库中挖掘出来的频繁项集，数量庞大，特别是在数据密集型数据库中，问题尤为严重，因为它加重了计算机存储、I/O 和 CPU 等的负担；此外，庞大的频繁项集的传输对于网络带宽而言，也是一个极大的挑战. 这些都使得频繁项集的使用受到限制[4].

鉴于此，业界的学者们从不同的角度提出各种方法来降低频繁项集集合的规模. 研究发现，可以在大量的频繁项集中将相关频繁项集进行合并，使用一个项集作为代表，来压缩频繁项集的规模，并提出了生成器模型[5-6]、闭合项集模型[7]、最大频繁项集[8]等压缩模型. 但是这些压缩模型均存在一些问题：生成器模型和闭合项集模型虽然都能够判断一个项集是否频繁，也能给出该频繁项集准确的支持度，但是在实际使用过程中灵活性不足；最大频繁项集虽然能够很好地压缩频繁项集的规模，但是它只能判断一个项集是否频繁，却无法给出频繁项集的支持度.

在实际使用的过程中，随着数据量的急剧增加，频繁项集的数量也呈爆炸式增长，查询一个频繁项集的准确度显得非常困难. H. Mannila 等[9]提出一个创新的折中思想，在效率和准确率之间做一个平衡. 对于一个频繁项集，只需要知道该项集的大致支持度即可，而不需要其准确的支持度，即项集的支持度可以用在一定允许范围内变动的支持度表示. 同时还提出了一个概念，即大致接近支持度查询结果，也称为 ε 适当表示. ε 适当表示模型是替代另外一种同样类型查询答案的数据表示，但是这种精简表示模型会导致频繁项集损失一些精度，损失精度的最大值为 ε.

J. F. Boulicaut 等[10]将 ε 适当表示引入频繁项集的精简表示，提出了一种频繁项集有损压缩，即 δ 无关集. 项集支持度相差不大于 δ 的项集，可以合并归为一类项集. 如果一个项集是无关的，那么该项集中所有的项均不能组成一个精确规则. 针对无关集，他们还提出了 MINEX 算法，挖掘无关集和频繁负向边界，其挖掘过程主要采用类似 Apriori 算法的广度优先策略. 但是这种算法存在反复扫描数据库、重复生成候选项集、需要遍历子项集等诸多缺点，导致该算法效率不高.

针对 MINEX 算法挖掘过程中存在的问题，许普乐等[11]提出了利用 FP 树挖掘无关集的 FMINEX 算法，该算法利用每次迭代快速生成不重复的候选项集，并得到候选项集的支持度，但仍需要遍历所有直接子项集.

鉴于此，本文拟提出一个挖掘无关集的新算法 FMFS(fast mining free sets)，该算法利用 Rymon 枚举树[12-13]的深度递归思想，结合一定的剪枝策略和遍历路径，同时根据本文提出的无关集判断定律快速判断无关集，以期避免反复扫描数据库、重复生成候选项集、遍历所有直

接子项集,从而提高算法效率.

1 相关概念和理论

1.1 频繁项集

定义 1 交易数据库 G:所有项的集合构成 R 集合,一条交易记录 r 是 R 的子集,即 $r \subseteq R$,数据库 G 是多条 r 的组合,$G = \{r \mid r \subseteq R\}$.

定义 2 项集支持度:项集 I 的支持度是在数据库 G 中出现 I 的交易记录个数,记为 $Freq(I) = |\{R \mid I \subseteq R, R \in G\}|$.

定义 3 频繁项集:项集 I 支持度大于最小人为设置支持度 $minsup$,即 $Freq(I) \geqslant minsup$,则称为频繁项集,记为 $Freq(G, minsup)$.

定义 4 关联规则 U:R 是所有项的集合,基于项的集合 R 关联规则的表现形式为 $W \Rightarrow E$,其中 W, E 属于 R,即 $U = \{W \Rightarrow E \mid W \subseteq R, E \subseteq R, E \neq \varphi, W \cap E = \varphi\}$.

定义 5 δ 强规则:在数据库 D 中存在一个关联规则 $W \Rightarrow E$,其中 $Freq(W) - Freq(W \cup E) \leqslant \delta$,即项集 X 和项集 $X \cup Y$ 的支持度相差不超过 δ 行,该规则称为 δ 强规则.

定义 6 δ 无关集:当且仅当基于项集 I 没有 δ 强规则,则项集 I 为 δ 无关集,记为 $Free(G, I, \delta) = \{I \mid W \subseteq I,\ Freq(I) \geqslant minsup,\ \forall X \subseteq I,\ Freq(X) - Freq(I) > \delta\}$.

定义 7 频繁无关集:如果 δ 无关集是频繁的,则 δ 无关集是频繁无关集,记为 $FreqFree(G, minsup, \delta)$.

定义 8 无关集的反单调性:如果项集 X 是无关集,且 $Y \subseteq X$,则 Y 也是无关集;如果项集 X 是非无关集,且 $Y \supseteq X$,则 Y 也是非无关集.

定义 9 无关集的负边界:无关集的负向边界记为 $Bd^-(G, minsup, \delta) = \{I \mid I \subseteq R, I \notin FreqFree(G, minsup, \delta) \wedge (\forall X \subset I, X \in FreqFree(G, minsup, \delta))\}$. 如果项集是频繁的,则频繁无关集的负向边界记为 $FreeBd^-(G, minsup, \delta)$.

1.2 无关集精简表示模型

在无关集精简表示模型中,无关集主要由两个集合组成,频繁无关集和对应的支持度 $FreqFree(G, minsup, \delta)$,频繁无关集的负向边界 $FreeBd^-(G, minsup, \delta)$.

对于给定的项集 X,执行如下查询

If $(\exists Y \in FreeBd^-(G, minsup, \delta), Y \subseteq X)$ then $Freq(X) = 0$

Else $Freq(X) = \min\{Freq(Y) \mid Y \subseteq X, Y \in FreqFree(G, minsup, \delta)\}$

根据这样的判断方法,即可判断出一个项集是否是频繁的,如果是频繁项集,可以得到该频繁项集的精确支持度.

1.3 Rymon 枚举树

Rymon 枚举树是由学者 R. Rymon 等于 1992 年提出的,枚举树能够快速地枚举出集合中所有的组合情况. 枚举树在给定的序列中各元素之间设置一个偏序,这种偏序可以是人工设定的,也可以是任意设定的,以保证枚举树上各个节点之间的父—子关系. 枚举树可以进行深度优先搜索,也可以进行广度优先搜索. 由于枚举树能够快速地将一个给定集合中 n 个元素的所有不重复的 2^n 个组合完整地枚举出来,所以非常适合解决数据挖掘中的搜索问题.

枚举树的一个重要的特点就是分而治之,在树上进行搜索的时候,如果某个节点不满足相关条件的要求,即可对该分支进行剪枝,节约搜索时间,提高搜索效率. 结合 1.1 中的定义 8 可知,如果一个节点不能满足无关集的要求,那么在分支下面所有节点都不是无关集,可以全部剪除.

2 FMFS 算法

2.1 无关集判断定律

本文提出一个函数 f,定义如下:

在交易数据库 G 中,$f(X) = \{\{R \mid X \subseteq R,$

$R \in G\}\}$,即项集 X 在数据库 G 中的交易记录. 项集 X 的支持度为 $Freq(X) = |f(X)|$. 设 a 为单个项,且 $a \notin X$,则项集 $\{a \cup X\}$ 的支持度为 $|f(X) \cap f(a)|$.

无关集的判断定律:设项集 X,i 为单个项,且 $i \notin X$,如果不存在单个项 a,$a \in X$,使得 $|f(i)| - |f(i \cup a)| \leqslant \delta$ 或者 $|f(a)| - |f(i \cup a)| \leqslant \delta$,则项集 $X \cup i$ 是无关集,如果 $Freq(X \cup i) \geqslant minsup$,则 $X \cup i$ 是频繁无关集.

证明　设存在单个项 a,$a \in X$,使得 $|f(i)| - |f(i \cup a)| \leqslant \delta$,则意味着项集 i 和项集 $a \cup i$ 相差不超过 δ 行,根据定义 5 可知,项集 $a \cup i$ 不是无关集,而 $a \in X$,根据定义 8 可知,项集 $X \cup i$ 不是无关集. 同理可证,若 $|f(a)| - |f(i \cup a)| \leqslant \delta$,项集 $X \cup i$ 不是无关集.

因为不存在单个项 a,所以可证项集 $X \cup i$ 是无关集. 项集 $X \cup i$ 的支持度为 $|f(X) \cap f(a)|$,根据定义 3 可知 $X \cup i$ 为频繁无关集.

证毕.

2.2　FMFS 算法描述

根据无关集的判断定律和偏序思想,本文提出了一种新的挖掘无关集算法 FMFS. 该算法将数据库 G 中所有的项,利用 f 函数取得结果、保存结果,同时对数据库 G 中所有的项随机设置一个偏序. FMFS 算法描述如下.

输入:数据库 G,最小支持度 $minsup$,最大无关值 δ

输出:频繁无关集 FS 和负边界 BD^-

$FS = BD^- = \text{null}$

$Freeset = \text{null}$

//数据库中所有项,并设置偏序

$C_1 = \{1\text{ 项集}\}$

$POST = C_1$

$FS = \{\text{频繁 1 项集}\}$

$BD^- = \{\text{非频繁 1 项集}\}$

$FMFS(FS, BD^-, POST, minsup, \delta)$

Return FS and BD^-

其中,函数 $FMFS(FS, BD^-, POST, Freeset, minsup, \delta)$ 描述如下.

$FMFS(FS, BD^-, POST, Freeset, minsup, \delta)$

{

while $POST \neq$ NULL do

//在 $POST$ 集合中选择偏序最小的项

$a = \min(POST)$

$POST = POST \backslash i$

$NewFreeset = Freeset \cup a$

//判断候选项集 NewFreeset 是否是无关集

if($\forall i \in NewFreeset, |f(i)| - |f(i \cup a)| > \delta$ 或者 $|f(a)| - |f(i \cup a)| > \delta$)

$NewFreeset.Freq = |f(Freeset) \cap f(a)|$

if($NewFreeset.Freq \geqslant minsup$)

$FS = FS \cup NewFreeset$

$NEWPOST = POST$

$FMFS(FS, BD^-, NEWPOST, Freeset, minsup, \delta)$

Else

$BD^- = BD^- \cup NewFreeset$

endif

endif

endwhile

}

3　实验结果与分析

为了验证本文算法的性能,本文统一使用 C ++ 语言分别实现了 FMINEX 和 FMFS 算法. 运行的平台为 PC 机,内存为 DDR2 4G,CPU 为 Core i5-680,操作系统为 Windows 2008 R2/64 位,编译工具为 VC 6.0. 分别使用数据密集型数据集(chess,mushroom,pumsb,pumbs_star)和数据稀疏型数据集(T10I4D100K,T40I10D100K)进行实验.

实验分别考察了 δ 为 5,10,20 这 3 种情况

下无关集挖掘效果和时间消耗. 本文中 FMFS 和 FMINEX 挖掘出的结果与文献[10 – 11]中结果完全相同,能够挖掘出所有的无关集. FMFS 和 FMINEX 在数据密集型数据集上运行结果见图 1,在数据稀疏型数据集上运行结果见图 2.

从图 1 可以看出,FMFS 算法的速度是 FMINEX 的 2 倍以上,并且随着支持度的降低,FMFS 算法的优势更加明显. 从图 2 可以看出,在数据稀疏型数据集 T10I4D100K 和 T40I10D100K 上,FMFS 算法的速度比 FMINEX 快 40% 左右,并且随着支持度的降低,优势更加明显. 由于数据稀疏型数据集需要存储的 f 函数结果比较多,因此增加了检索的工作量,导致在稀疏型数据集上 FMFS 算法的时间优势没有在数据密集型数据集上那么明显,特别是在支

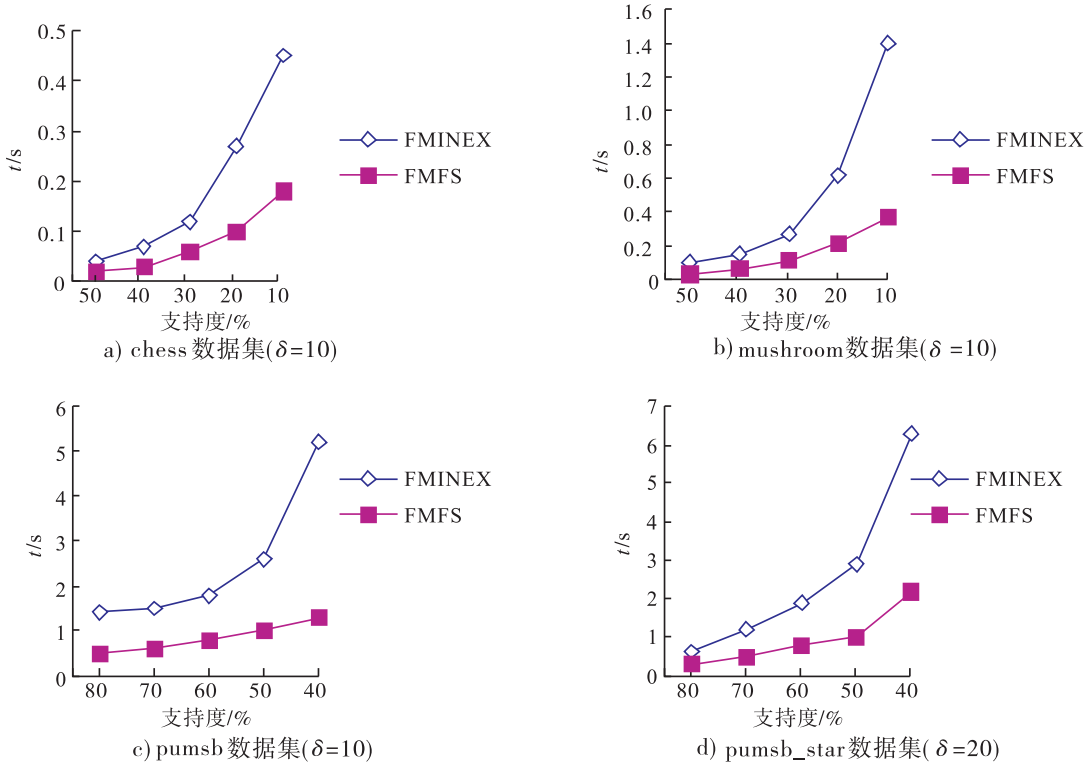

图 1 FMFS 和 FMINEX 算法在数据密集型数据集上的运行结果

Fig. 1 Results on intensive data sets of FMFS and FMINEX algorithms

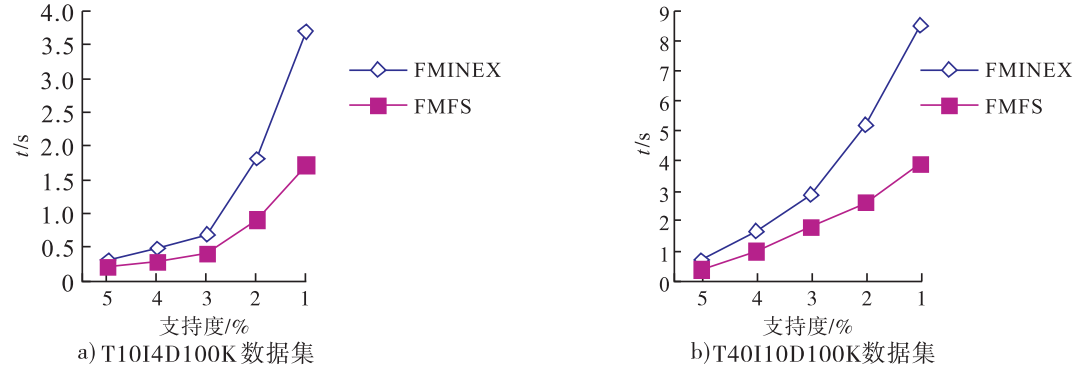

图 2 FMFS 和 FMINEX 算法在数据稀疏型数据集上的运行结果(δ =5)

Fig. 2 Results on sparse data sets of FMFS and FMINEX algorithms(δ =5)

持度比较高的情况下.

4 结语

δ 无关集能够实现压缩频繁项集和查询支持度之间的折中,以牺牲比较小的准确率为代价换取查询效率的快速提高. 本文在分析了传统挖掘无关集算法后,得出挖掘效率低下的原因,并提出 FMFS 算法. 该算法使用 Rymon 枚举树作为搜索空间,结合特定的剪枝策略,以及本文提出的一种快速判断无关集的定律,达到了快速挖掘无关集的目的. 实验结果证明,FMFS 算法不仅能够挖掘出所有的无关集,还比 FMINEX 算法要快速. 随着云计算时代的到来,特别是以亚马逊、阿里云为代表的商业化云计算提供商日益壮大,如何利用云计算实现无关集的快速挖掘,将是今后数据挖掘领域研究的一个重点方向.

参考文献:

[1] HAN J W, KAMBER M, PEI J, et al. 数据挖掘概念与技术[M]. 范明, 孟小峰, 译. 北京: 机械工业出版社, 2004: 1 – 261.

[2] 纪允. 析取闭合项集的快速生成和恢复算法研究[D]. 合肥: 合肥工业大学, 2013.

[3] 王红梅, 党源源, 胡明, 等. 基于排序树的频繁项集挖掘算法[J]. 吉林大学学报(工学版), 2016, 46(4): 1216.

[4] 朱玉全, 孙志挥, 赵传申. 快速更新频繁项集[J]. 计算机研究与发展, 2003, 40(1): 94.

[5] BASTIDE Y, TAOUIL R, PASQUIRE N, et al. Mining frequent patterns with counting inference[J]. SIGKDD Explorations, 2000, 2(2): 66.

[6] 许普乐, 张勤, 纪允. 基于 FP 树的一种快速挖掘生成器算法[J]. 安庆师范学院学报(自然科学版), 2013, 19(1): 48.

[7] PASQUIER N, BASTIDE Y, TAOUIL R, et al. Discovering frequent closed itemsets for association rules[C]// 7th Intl. Conf. on Database Theory. Heidelberg: Springer, 1999: 398.

[8] BAYARDO R J. Efficiently mining long patterns from databases[C]// Proc of the ACM SIGMOD Int Conf on Management of Data. New York: ACM Press, 1998: 85.

[9] MANNILA H, TOIVONEN H. Multiple uses of frequent sets and condensed representations: Extended abstract[C]// Proc of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96). [S. l.]: [s. n.], 1996: 189.

[10] BOULICAUT J F, BYKOWSKI A, RIGOTTI C. Free-sets: A condensed representation of boolean data for the approximation of frequency queries[J]. Data Mining and Knowledge Discovery, 2003, 7(1): 5.

[11] 许普乐, 纪允, 张勤. 应用 FP 树快速生成无关集算法[J]. 安庆师范学院学报(自然科学版), 2016, 22(2): 60.

[12] RYMON R. Search through systematic set enumeration[C]// Proc of Third Int'l Conf. on Principles of Knowledge Representation and Reasoning. [S. l.]: [s. n.], 1992: 539.

[13] 田卫东, 纪允. 一种频繁核心项集的快速挖掘算法[J]. 计算机工程, 2014, 40(6): 120.