

一种多时间间隔序列模式挖掘算法

李广原^{1,2}, 马楠^{1,3}, 曹丹阳¹

(1 北京科技大学 知识工程研究所, 北京 100083; 2 广西师范学院 计算机与信息工程学院, 广西 南宁 530001;
3 北京联合大学 信息学院, 北京 11010)

摘 要: 提出一种多时间间隔的序列模式挖掘算法, 依据挖掘的实际情况设置可变的时间区间, 采用有效的剪枝策略, 分区间精确显示多时间间隔序列模式挖掘结果. 实验证明, 算法具有较高的挖掘性能.

关键词: 数据挖掘; 序列模式; 多时间间隔

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2012)04-0010-04

An Efficient Algorithm for Mining of Multi-time-interval Sequential Patterns

LI Guang-yuan^{1,2}, MA Nan^{1,3}, CAO Dan-yang¹

(1 Research Institute of Knowledge Engineering, University of Science & Technology Beijing, Beijing 100083, China;
2 School of Computer and Information Engineering, Guangxi Teachers Education University, Nanning 530023, China;
3 College of Information, Beijing Union University, Beijing 110101, China)

Abstract: In this paper, we present an efficient algorithm for mining of multi-time-interval sequential patterns. The proposal algorithm set a dynamic time-interval based on the mining process, along with the use of efficient prune techniques, the mining results are shown in a precise way. Experimental results also show that the algorithm has high performance.

Key words: data mining; sequential patterns; multi-time-interval

1 引言

序列模式挖掘, 又称频繁模式挖掘是数据挖掘的一个重要任务, 在计算生物学、用户购物行为模式分析、web 日志挖掘等多个领域有着重要的应用. 目前, 根据挖掘的任务, 已提出了不少序列模式挖掘算法, 比如 GSP 算法^[1], SPADE 算法^[2], 多维序列模式挖掘算法^[3]等. 传统的序列模式挖掘或者没有考虑序列中项与项之间出现的时间间隔, 或者设定一个固定的时间区间, 因而挖掘结果可能会遗漏一些有趣的模式. 文献[4]中提出一种时间间隔序列模式挖掘算法, 但该算法把项与项之间的时间范围限制在一个固定的范围, 文献[5]中对此作了改进, 把时

间间隔设定在若干个固定的范围, 使得挖掘更具有灵活性, 但是, 固定的设置方法挖掘结果不能精确显示.

为此, 我们提出一种挖掘多时间间隔的序列模式算法 MTTAL.

2 问题描述

设 DB 所示为一多时间间隔序列数据库, I 是项集, $I = \{i_1, i_2, \dots, i_n\}$. 数据库中一个事务序列为 $s = \{\langle a_1, t_1 \rangle, \langle a_2, t_2 \rangle, \dots, \langle a_n, t_n \rangle\}$, 其中, $a_i \in I$, t_i 是 a_i 出现的时间, $t_{i-1} \leq t_i, i = 1, 2, \dots, n$. 表 1 所示为一个示例的多时间间隔序列数据库, 下面给出若干定义.

表1 多时间间隔序列数据库

标号	事务序列
1	{<a,1>,<b,4>,<f,5>,<g,12>}
2	{<d,3>,<e,6>,<f,10>}
3	{<b,5>,<c,12>,<e,13>,<f,15>}
4	{<c,2>,<d,3>,<a,8>,<b,10>,<d,13>}
5	{<a,6>,<b,8>,<d,12>,<b,13>,<f,15>}
6	{<c,4>,<e,3>,<c,8>,<a,10>,<d,13>}

定义1 多时间间隔序列 s 可表示为

$s = \{i_1, \delta_{12}, i_2, \delta_{23}, i_3, \dots, i_{n-1}, \delta_{(n-1)n}, i_n\}$, 其中, $i_j \in I, \delta_{(j-1)j} = t_j - t_{j-1}, t_j$ 是 i_j 出现的时间, $2 \leq j \leq n$.

一个多时间间隔序列的长度为该序列所包含的项的个数,多时间间隔序列模式应至少包含两个以上的项,含有 k 个项的多时间间隔序列称为多时间间隔 k 序列. 在表1中,首条序列 $\{<a,1>,<b,4>,<f,5>,<g,12>\}$ 共有9个多时间序列模式,它们分别为 $\{a,3,b\}, \{a,4,f\}, \{a,11,g\}, \{b,1,f\}, \{b,8,g\}, \{f,7,g\}, \{a,3,b,1,f\}, \{a,3,b,8,g\}, \{b,1,f,7,g\}$.

定义2 设 $A = \{<a_1, t_1>, <a_2, t_2>, \dots, <a_n, t_n>\}$ 为一序列, $B = \{b_1, \delta_{12}, b_2, \dots, \delta_{(m-1)m}, b_m\}$ 是一多时间间隔序列, ϵ 是时间间隔阈值,如果满足以下条件,则称 B 包含 A , 记作 $A \subseteq B$.

- (1) $a_1 = b_{j_1}, a_2 = b_{j_1+1}, \dots, a_n = b_{j_1+n};$
 - (2) $t_i - t_{i-1} \in [|\delta_{(j_{i-1})j_i} - \epsilon|, |\delta_{(j_{i-1})j_i} + \epsilon|].$
- 其中, $(1 \leq j_1, j_1 + n \leq m)$.

例如,假设时间间隔值 $\epsilon = 2$, 一个多时间间隔序列 $Y = \{i_1, 2, i_2, 5, i_3, 4, i_4, 8, i_5\}$, 另有一充列:

$X = \{<i_1, 3>, <i_2, 4>, <i_3, 8>\}$, 则 $X \subseteq Y$.

定义3 设两个多时间间隔序列 $A = \{a_1, \delta_{12}, a_2, \dots, \delta_{(m-1)m}, a_m\}$, $B = \{b_1, \gamma_{12}, b_2, \dots, \gamma_{(m-1)m}, b_m\}$.

如果存在 $1 \leq j_1 < j_2 < \dots < j_k \leq m$, 且满足以下条件, 则称 A 是 B 的多时间间隔子序列.

- (1) $a_1 = b_{j_1}, a_2 = b_{j_2}, \dots, a_n = b_{j_m};$
- (2) $\delta_{(i-1)i} \in [|\gamma_{(j_{i-1}-1)j_{i-1}} + \dots + \gamma_{(j_i-1)j_i} - \epsilon|, |\gamma_{(j_{i-1}-1)j_{i-1}} + \dots + \gamma_{(j_i-1)j_i} + \epsilon|].$

例如,假设时间间隔值 $\epsilon = 2$, 一个多时间间隔序列 $Y = \{i_1, 2, i_4, 1, i_2, 3, i_5, 2, i_3, 4, i_5\}$, 另有一多时间间隔充列 $X = \{i_1, 2, i_2, 4, i_3\}$, 则 X 是 Y 的多时间间隔子序列.

定义4 给定多时间间隔序列数据库 DB , 事务

支持度阈值 σ , 一个多时间间隔充列 S 的支持度定义如下:

$$Support(X) = \frac{|Y|}{|DB|} \quad \left| \begin{array}{l} Y \in DB \wedge Y \text{ 是 } X \text{ 的子序列} \end{array} \right.$$

3 MTTAL 算法

为讨论方便,下面再给出若干个定义.

定义5 设 $X = \{<x_1, t_1>, <x_2, t_2>, \dots, <x_n, t_n>\}$ 为一序列, 一个多时间间隔序列 $Y = \{y_1, \delta_{12}, y_2, \dots, y_{m-1}, \delta_{(m-1)m}, y_m\}$ 称为 X 的前缀, 当且仅当满足以下两条件:

- (1) $y_1 = x_1, y_2 = x_2, \dots, y_m = x_m (m < n);$
 - (2) $\delta_{12} \in [|\delta_{12} - t_1 - \epsilon|, |\delta_{12} - t_1 + \epsilon|], \dots$
 $\delta_{(m-1)m} \in [|\delta_{(m-1)m} - t_m - \epsilon|, |\delta_{(m-1)m} - t_m + \epsilon|].$
- ϵ 是时间间隔阈值.

例如: $A = \{<a_1, 2>, <a_3, 4>, <a_5, 5>, <a_3, 7>\}$

$B = \{a_1, 2, a_3, 2, a_5\}, \epsilon = 2$, 则 B 是 A 的前缀.

定义6 设 $X = \{<x_1, t_1>, <x_2, t_2>, \dots, <x_n, t_n>\}$ 为一序列, 一个多时间间隔序列 $Y = \{y_1, \delta_{12}, y_2, \dots, y_{m-1}, \delta_{(m-1)m}, y_m\}$, 称

$X' = \{x'_1, \gamma_{12}, x'_2, \dots, x'_{l-1}, \gamma_{(l-1)l}, x'_l\}$ 为 X 关于 Y 的投影多时间间隔序列, 当且仅当满足以下条件.

- (1) Y 是 X 的前缀
- (2) $x'_1 = x_{m+1}, x'_2 = x_{m+2}, \dots, x'_l = x_n$
- (3) $\gamma_{12} \in [|\delta_{m+2} - t_{m+1} - \epsilon|, |\delta_{m+2} - t_{m+1} + \epsilon|], \dots,$

$$\gamma_{(l-1)l} \in [|\delta_{t_n - t_{n-1}} - \epsilon|, |\delta_{t_n - t_{n-1}} + \epsilon|]$$

定义7 设多时间间隔序列数据库为 DB , DB 中关于 X 的所有多时间间隔投影事务记作 $DB|_X$, X 是一多时间间隔序列.

如果一个多时间间隔序列 S 是频繁的, 则它的所有子序列也是频繁的, 反之, 如果 S 不是频繁的, 则它的所有超集也不是频繁的.

假设 DB 的事务数为 N , 支持度阈值为 σ , 某条事务的开始时间和结束时间分别为 t_b 和 t_e , $(i = 1, 2, \dots, N)$, 称 $t_e - t_b$ 为该事务的时间大小, 则对于任一多时间间隔充列, 如果存在有项与项的时间间隔大于第 $N \times \sigma$ 个事务时间的大小, 则该时间间隔序列为非频繁序列.

下面给出算法描述.

输入: 事务数据库 DB , 多时间间隔阈值 ϵ , 事务

的最小支持度 σ

输出: 分区间多时间间隔序列模式集 L

1. 寻找数据库 DB 中的频繁 1 项集 L_{DB}^1

2. $\forall i \in L_{DB}^1$

3. 构造 $DB|_i$

4. 寻找投影数据库 $DB|_i$ 中的频繁 1 项集

$L_{DB}^1|_i$ if $L_{DB}^1|_i = \emptyset$ then continue // 转下一个 i

5. $\forall j \in L_{DB}^1|_i$

6. if $\delta_{ij} >$ 第 $N \times \sigma$ 个事务的时间大小 // 剪枝

7. continue // goto 5, 转到下一个 j

8. if $\exists (i * j) \in C_{i*j} // i * j$ 表示序列 ij , C_{i*j} 用于存放候选多时间间隔序列模式 ij , 与此类推, $C_{i*j*\dots n}$ 用于存放候选多时间间隔序列模式 $ij \dots n$, 得出 L_{DB}^1 , 候选模式集皆初始化为空集.

9. if $\delta_{ij} \in [|\delta_k - \epsilon|, |\delta_k + \epsilon|] // \delta_k$ 表示在 C_{i*j} 中的某个时间间隔, 用 T_{ij} 来表示 C_{i*j} 中序列的时间间隔的集合.

10. then $freq_{ij} = freq_{ij} + 1 // freq_{ij}$ 为 ij 的计数.

11. else $T_{ij} = \delta_{ij} \cup T_{ij}$

12. $C_{ij} = C_{ij} \cup i * j$

13. if $freq_{ij} \geq \sigma$

14. $L = L \cup i * j //$ 同时记录相应的 δ_{ij} , 以便按

时间区间输出 L

15. $i = i * j //$ 即把 i 变成 $i * j$

16. goto 3

17. 按时间区间输出 L

4 实验分析

为了检验算法 MTTAL 的性能, 我们选择 MI-PrefixSpan^[5] 作为比较对象, 以运行时间和生成模式的数目作为测试的指标, 测试程序采用 C++ 语言, 运行在 CPU 为 PIV3.2 GHz, 内存为 2 GB, 操作系统为 WINDOWS 2003 的微机, 测试数据集采用 IBM 数据生成器来生成^[6], 共生成两组测试数据集, 数据集的相关参数见表 2, 另外, 每个事务中的项的时间采用随机数生成的方式赋值, 并使每条事务中后面出现的项的时间大于前面出现项的时间, 时间范围在 $[1, 100]$, 时间阈值 ϵ 设为 10. MI-PrefixSpan 采用了文献^[7]的时间设置方法. 实验结果见图 1~图 4 所示.

表 2 测试数据集及其指标

数据集	事务中的项 的平均个数	最大潜在频 繁序列的平 均长度	最大潜在频 繁序列中项 的平均个数
T20-S12-I4	20	12	4
T25-S15-I6	25	15	6

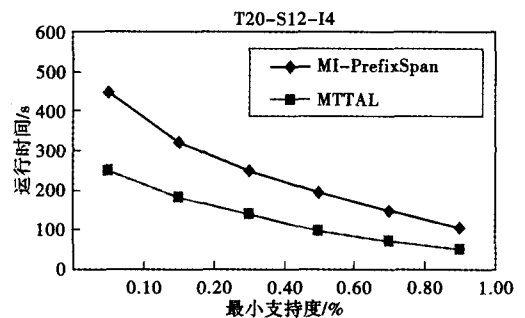


图 1 最小支持度与运行时间对应图

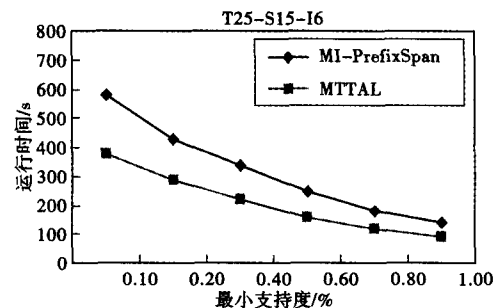


图 2 最小支持度与运行时间对应图

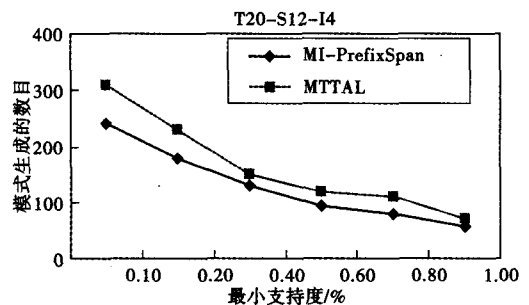


图 3 最小支持度与生成模式数目对应图

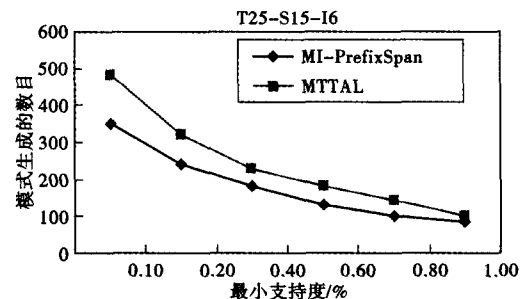


图 4 最小支持度与生成模式数目对应图

从实验结果可以看出,由于采用了新的剪枝技术,能较早地发现多时间间隔候选频繁序列,在两个测试数据集上测试的结果显示,MTTAL 算法在运行时间上均比 MI-PrefixSpan 算法要快.在模式生成的数目上,由于 MTTAL 算法采用时间阈值,根据挖掘的实际情况,每条序列与现有的同序列(这里指项相同)相比来决定实际的时间区间,而算法 MI-PrefixSpan 简单地设置若干个时间区间,这样会造成同属一序列(这里指项相同)的某些序列没有落在这些区间,因而,算法 MTTAL 要比 MI-PrefixSpan 生成较多的模式,另外,MTTAL 分区间显示结果,使得可以清楚地知道同一模式在时间间隔上的分布情况.

5 结束语

本文给出一种新的多时间间隔频繁模式挖掘算法,通过设置时间间隔阈值,分区段显示频繁模式挖掘结果,采用有效的剪枝策略,通过实验对比,算法具较好的挖掘性能.多时间间隔序列模式挖掘可以与传统的序列模式挖掘相结合,成为新的研究方向.比如,多维多层序列模式挖掘与多时间间隔序列模式挖掘相结合,又如,模糊序列模式挖掘与多时间间隔序列模式挖掘相结合等等,这是今后研究的主要方向.

参考文献:

- [1] Srikant R, Agrawal R. Mining sequential patterns: generalizations and performance improvements[C]// Proceedings of the 5th International Conference on Extending Database Technology. Ausrila: Badvoslan, 1996: 3-17.
- [2] Pei J, Han J. PrefixSpan: mining sequential patterns efficiently by prefix2p rejected pattern growth [C]// Proc of the 7th International Conference on Data Engineering. Washington DC: IEEE Computer Society, 2001: 215-224.
- [3] Zakim J. Spade: an efficient algorithm for mining frequent sequences [J]. Machine Learning, 2001, 41 (1): 31-60.
- [4] 吴聪聪, 李忠晔, 何丕廉. 基于矩阵的最大频繁模式挖掘及其更新算法[J]. 微电子学与计算机, 2007, 24 (12): 29-31.
- [5] 缪裕青, 尹东. 分布式存储结构的频繁闭合模式挖掘并行算法[J]. 微电子学与计算机, 2007, 24(10): 161-163.
- [6] Chen Yenliang, Chiang Meiching, Ko Mingtat. Discovering time-interval sequential patterns in sequence databases[J]. Expert Systems with Applications, 2003(25): 343-354.
- [7] Hu Yahan, Tony Cheng-kui Huang, Yang HuiRu, et al. On mining multi-time-interval sequential patterns [J]. Data & Knowledge Engineering, 2009 (68): 1112-1127.
- [8] Agrawal R, Srikant R. Mining sequential patterns[C]// Proceedings of the 11th International Conference on Data Engineering. Taiwan Taipei, 1995: 3-14.

作者简介:

李广原 男,(1969—),博士研究生,副教授.研究方向为数据挖掘、智能信息处理.

马楠 女,(1978—),博士研究生.研究方向为数据挖掘.

曹丹阳 男,(1978—),博士研究生.研究方向为数据挖掘.

- [1] Srikant R, Agrawal R. Mining sequential patterns: generalizations and performance improvements[C]// Proceedings of the 5th International Conference on Extending Database Technology. Ausrila: Badvoslan,