一种基于拓扑信息的物流频繁路径挖掘算法

杨俊瑶 蒙祖强 蒋 亮

(广西大学计算机与电子信息学院 南宁 530004)

摘 要 为了高效地从海量物流数据中获取频繁路径,根据物流网络及物流的特征设计了一种物流数据模型以及一种充分考虑了物流网络拓扑信息的频繁路径序列挖掘算法 PMWTI(Path Mining With Topology Information)。在 PMWTI 中设计了一种用于候选路径序列深度剪枝的代价容忍度剪枝方法,该方法在利用 Apriori 性质剪枝的基础上进一步去除了部分不可能是频繁路径序列的候选路径序列,这在一定程度上缩减了候选路径序列规模,从而减少了对数据集的扫描。实验表明,相比没有采用该剪枝方法的同等算法,PMWTI 具有更高的频繁路径挖掘效率。

关键词 物流,频繁路径,序列模式,数据挖掘

中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.4.053

Logistics Frequent Path Sequence Mining Algorithm Based on Topological Information

YANG Jun-yao MENG Zu-qiang JIANG Liang

(College of Computer and Electronic Information, Guangxi University, Nanning 530004, China)

Abstract In order to get frequent paths from massive logistics data, according to the feature of logistics networks and logistics, this paper provided a logistics data model and a frequent path sequence mining algorithm PMWTI(Path Mining With Topology Information) taking the topological information of logistics networks into consideration. In PMWTI, a cost tolerable degree pruning method used for the deep pruning of candidate path sequences was designed. This method discards some candidate path sequences which are gained by Apriori pruning method but can't be frequent path sequences. It can downscale the candidate path sequences, so that the algorithm scans the dataset less. The experimental result shows that, compared with the same algorithm which do not adopt this pruning method, PMWTI has better mining efficiency.

Keywords Logistics, Frequent path, Sequence pattern, Data mining

1 引言

物流中会产生大量的物流数据,其中之一是具有时空特性的 RFID 数据。通过对这种 RFID 数据的加工可以获取物流的路径数据。在这些路径数据中可以通过频繁序列模式挖掘出频繁路径,频繁路径可为优化物流路由、研究物流规律等提供参考。物流中频繁路径序列挖掘的相关研究主要集中在 RFID 数据仓库模型与频繁路径挖掘算法上。在数据仓库模型方面,文献[1,2]给出了 RFID-CUBOID 数据模型,该模型中,物品的属性信息与位置信息相分离,成为相对独立的两部分,并在数据立方体中引入了路径信息。频繁路径挖掘算法的相关研究主要集中在传统频繁序列模式挖掘算法在频繁物流路径挖掘上的应用、分布式的频繁物流路径挖掘、多维度的频繁物流路径挖掘等方面。

文献[3]提出了一种 GSP 算法的改进算法 ImGSP,用于挖掘频繁物流路径序列。该算法主要的改进思想是通过序列的长度限制,删减一些不用搜索的序列,从而在一定程度上缩减了数据集的规模。文献[4]通过改进传统关联规则,提出了频繁装卸点、频繁直达(转运)路径等物流中频繁模式挖掘相关概念。在此基础上结合物流数据的时空属性,挖掘 RFID 物流数据中的非平凡模式,为科学的管理物流与优化物流线路提供可靠及时的决策依据。在 RFID 物流数据挖掘中的数据储存方式及相关频繁物流路径挖掘算法的研究方面,文献[5]给出了一种分析处理操作是基于图的联机框架,其中物品流动的路径信息以图的方式表示;文献中作者给出了频繁子图挖掘算法 RFSM,该算法从用户给定维值的图集上挖掘 RFID 频繁图,而不是在整个图集上进行挖掘。文献[6]主要在多维 RFID 频繁路径挖掘、单机环境下 RFID 频繁路径挖掘、分布式环境下的 RFID 频繁路径挖掘等几方面做了相关研究,其中还利用路径编码的方法压缩 RFID 路径数据。文献[7]针对 EPC 网络中 RFID 数据的特点设计了适用于 EPC 网络且利于从中进行频繁路径挖掘的 RFID 路径数据模型,在该数据模型的基础上给出了一种基于增长模式的 RFID 频繁路径挖掘算法 FPM,并在 FPM 算法的基础之上根据供应

到稿日期:2014-05-03 返修日期:2014-08-05 本文受国家自然科学基金项目(61363027),广西自然科学基金项目(2012GXNSFAA053225)资助。

杨俊瑶(1986—),男,硕士,主要研究方向为数据挖掘,E-mail:faerysword@163.com;蒙祖强(1974—),男,博士,教授,主要研究方向为数据挖掘、粒度计算等;蒋 亮(1988—),男,硕士,主要研究方向为数据挖掘、粒度计算等。

链多层级的结构特点，给出了一种多级支持度的 RFID 频繁路径挖掘算法 MSFPM。

以上关于频繁路径挖掘的研究都是借鉴现有频繁序列挖掘算法的思想，将 RFID 数据中的位置序列转化为传统的事务序列形式再进行路径挖掘，在算法的优化与改进上虽然也会考虑物流数据的特点，但却忽略了物流网络的空间性质。除了利用传统的 Apriori 特性[8]（即频繁模式的子模式必然频繁）对候选序列进行剪枝外，还可以利用物流网络的拓扑结构的特点设计相应的剪枝算法来对候选序列作更进一步地削减，从而提高算法效率。本文在传统的 Apriori 算法的基础上，设计了频繁序列模式挖掘算法 PMWTI 来进行频繁路径挖掘。在 PMWTI 中生成候选序列时充分考虑了被传统的频繁路径挖掘算法所忽略的物流网络的拓扑信息，在利用 Aprori 特性进行剪枝的基础上更进一步地去掉了不可能频繁的序列，在一定程度上缩减了候选序列的规模，与没有采用该剪枝方法的传统频繁路径挖掘算法相比，PMWTI 具有更好的挖掘效率。

2 一种面向高效挖掘的物流路径数据模型

2.1 频繁模式挖掘的相关概念

为方便描述，本文在此先介绍几个频繁模式挖掘的相关概念。

项集：项的集合，包含 k 个项集的集合称为 k 项集，例如，包含 A、B、C 3 个项的集合称为 3 项集，记为 ABC。

序列：项集的项按一定顺序排列形成序列，k 序列即 k 个项的排列。例如包含 A、B、C 3 个项，且其顺序为 B、C、A 的序列称为 3 序列，记为 BCA。

支持度：数据集中某项集（序列）出现的频度，以百分比或小数表示。

最小支持度：项集（序列）支持度的最低要求。

频繁模式：支持度不低于最小支持度的项集（序列），也称作频繁项集（序列）。

路径序列：序列中的项表示地点的序列，例如 A、B、C 表示 3 个地点，从 B 到 C 再到 A 的序列即为一条路径序列，记为 BCA。

频繁路径：满足最小支持度的路径序列。

候选集（序列）：可能成为频繁模式的项集（序列）。

2.2 传统频繁模式挖掘中基于矩阵的数据模型

频繁模式的挖掘算法的时间消耗主要在数据集的扫描上，关于如何提高数据集的扫描效率已有不少相关研究，其普遍采用 0-1 矩阵的表示方式，文献[9-11]等对此有相关介绍。其中矩阵的每一列映射为某一项，而每一行对应着数据集中的一个项集，该行中 0 表示相应的项不在当前项集中，1 则表示在当前项集中。如表 1 所列，各列分别映射项 A、B、C、D、E，则表 1 中第二行与第三行所代表的项集分别是 ACE 与 BCD。这种可以随机查找的数据结构的采用使候选集扫描中的串搜索转化为了简单的逻辑判断。例如在判断表 1 中第二行所示记录 ACE 中是否包含 A、E 项时，不用逐项搜索串 ACE，而只要判断 A、E 所对应的列的值的逻辑与运算的结果是否为 1，为 1 则表示 A、E 项存在；反之则不存在。

表 1 数据集的 0-1 矩阵示例

A	B	C	D	E
1	0	1	0	1
0	1	1	1	0

在频繁序列模式的挖掘中这种数据模型不再适用，序列信息无法简单地通过 0、1 表示出来，所以频繁序列模式的挖掘中使用的通常都是低效的串查找。然而由于物流的路径序列具有独特的时空属性，在序列中不会出现相同项。本文根据这一特点建立了一种物流数据模型，其中对传统的物流数据进行了相关处理，在 0-1 矩阵的基础上做了相应变动，将传统的 RFID 路径数据转化成一种蕴含路径序列信息且利于高效挖掘频繁路径的数据集。接下来首先建立一种物流路径数据模型，然后介绍频繁路径挖掘算法 PMWTI。

2.3 一种适用于物流领域频繁路径挖掘的数据模型

传统的物流数据为 RFID 数据，可以表示为{EPC，Location，Time}，其中 EPC（Electronic Product Code）是物品独一无二的标志，Location 是在时间 Time 时所在地点（Time 在这里并没有采用常规的时间表示方式，而是象征性地以数字表示）。在这些数据中只选取 Location 表示的地点为具有代表性的起始点或中转枢纽的那些数据。这些具有代表性的点即图 1 所示物流网络中的节点 A、B、C、D、E、F，图 1 中边为各节点之间的路径，该图相关的 RFID 数据示例如表 2 所列。此外，还可以根据挖掘需要建立一张包含物品某些属性表，在实现路径序列与属性分离的同时又可以进行多维度的频繁路径挖掘。例如，表 3 是物品种类（KindID）与物流权重（Weight）的属性表，其中物流权重是路径频繁程度的度量，即该权重越大则对应的路径越频繁。如果将这两个属性结合起来进行挖掘，则可得出某一类物品的频繁路径。为了简化问题，后面的实验进行效率比较时只考虑了物流权重，而没有考虑诸如物品种类等其他属性。

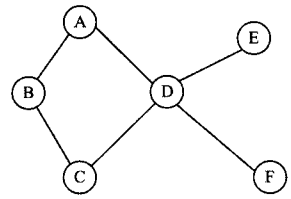

图 1 物流网络拓扑图示例

表 2 RFID 数据示例

EPC	Location	Time
1	A	1
2	E	1
3	F	1
2	D	2
1	D	2
3	D	2
2	A	3
1	F	3
2	B	4

将表 2 的 RFID 数据按 EPC 分类，拥有相同 EPC 的记录中的 Location 列对应的值按 Time 的大小（即时间的先后顺序）整理为{epc_x，location_1，location_2，…，location_i}，其中 location_1 至 location_i 为表 2 中 EPC 的值为 epc_x 的 Location的值，且按时间的先后顺序进行排序，该序列即为以

epc_x 为 EPC 的物品由按时间先后顺序所经过的地点所组成的 RFID 路径序列。表 2 的整理结果如表 4 所列。

表 3　物品的属性示例

EPC	KindID	Weight
1	111	100
2	222	100
3	111	200

表 4　物品的 RFID 路径序列示例

EPC	Location Serial
1	ADF
2	EDAB
3	FD

在挖掘过程中，EPC 仅仅作为区分数据集中每条记录的标记，没有挖掘意义，所以算法中如表 4 所列的数据去掉 EPC 后被转换为如表 5 所列的矩阵，以二维整形数组表示，所有元素都初始化为 0。其中每一行表示以某 EPC 为标记的物品的物流信息与相关属性，相关属性按挖掘需求选取。比如，我们要获取某类物品(KindID)的频繁路径，可以在序列的维度上加上 KindID 的限制。在实验中我们只考虑某一路径的频繁程度，所以用的是一个物流权重 Weight。此处表 5 其实也是逻辑上的一张表，它由表 3 与表 4 在逻辑上按 EPC 联接而成，两表的 EPC 映射到相同行号。将表 3 中的信息转换到表 5 中，经过的点按时间顺序依次记为 $1,2,\cdots,i$，没有经过的点自然就是初始状态 0。这样判断某一行是否包含某一路径可以通过检验该行中该路径对应的值是否是公差为 1 的等差数列来判断。比如，表 5 第三行中 D、A、B 对应的值依次为 2、3、4，是公差为 1 的等差数列，因此该行所表示路径包含了 DAB 这条路径。如此就把物流的频繁路径挖掘转换成了一种类似经典的关联规则挖掘。其中采用二维整形数组的数据结构是为了将串搜索转换为简单的数值比较，且在二维数组中能够随机访问，这样可以大幅提高算法运行的效率。

表 5　最终的 RFID 路径数据集

A	B	C	D	E	F	KindID	Weight
1	0	0	2	0	3	4	10
3	4	0	2	1	0	2	20
0	0	0	2	0	1	5	15

3　基于物流网络拓扑信息的频繁路径挖掘算法 PMWTI

通过对物流网络的分析得知，物流中 $A\rightarrow B$ 与 $B\rightarrow A$ 是不同的，且 $A\rightarrow B\rightarrow C$ 与 $A\rightarrow C$ 又不存在包含关系，$A\rightarrow C$ 并不是 $A\rightarrow B\rightarrow C$ 的子序列，而是另一条不相干的路径，所以物流频繁路径的挖掘又与经典的关联规则挖掘不同，是一种特殊的序列模式挖掘。由于 AB 与 BA 不是同一序列，因此如果不进行任何剪枝，则产生的候选 k 序列的数目为 A_n^k 而不是 C_n^k(n 为网络中节点个数)。幸运的是在这种特殊的网络中，剪枝方法除了传统的 Apriori 特性外，还可以根据物流网络本身的特点设计相应的剪枝算法。

用于挖掘频繁序列的传统的类 Apriori 算法的流程主要分为两个交替进行的步骤：1)获取候选 K 序列，其中会用到相关的剪枝算法来减少候选 K 序列的产生；2)扫描数据集以判断候选序列是否频繁，从而获取频繁 K 序列，然后又转到第 1)步，获取候选 $K+1$ 序列，如此循环往复，直到找不出更长的序列为止。其中从第 2)步转到第 1)步时借助了第 2)步的结果，根据频繁模式的任何子模式都是频繁的这一特点对候选 $K+1$ 序列进行剪枝，去掉不可能是频繁序列的 $K+1$ 序列。显然这两个步骤中最耗时的是第 2)步，而直接影响该步骤耗时程度的却是第 1)步。第 1)步产生的频繁序列越少则第 2)步耗时越少。物流网络有它本身的特点，在频繁路径的挖掘中主要体现在两个方面。

1)序列中任何相邻两个节点在拓扑图中必然是相邻的。比如图 1 中 AC 序列是不可能出现的，只可能出现 ABC 或 ADC。因此，由频繁 K 序列生成频繁 $K+1$ 序列时与传统 Apriori 或其他序列模式挖掘算法不一样，只要符合要求的两个 K 序列存在，则可以连接生成相应的 $K+1$ 序列。这两个序列满足：其中一个序列的前 $K-1$ 个节点组成的 $K-1$ 序列正好是另一个序列去掉第一个节点的子序列，比如，如果 ABC 与 BCD 是频繁的，则连接它们可以得到候选序列 $ABCD$；

2)物流网络的拓扑信息可用来进行候选频繁序列剪枝。由于物流趋向于走代价最小的路径，从理论上来讲，如果两点之间某一条路径是频繁的，那么两点之间代价小于等于该频繁路径代价的所有路径也是频繁的。在物流网络中，两点之间存在一条或多条最小代价路径，无论该路径频繁与否，代价超过最小代价太多，以至于无法接受的路径不会是频繁路径。而具体超过多少才算过多，这需要具体问题具体分析。

本文对物流网络中从节点 i 到节点 j 的路径序列 R_{ij} 定义了一个“代价容忍度”参数 TD，TD 表示 R_{ij} 的代价与从 i 节点到 j 节点的最小路径代价的比值，显然该比值是大于等于 1 的。在生成候选路径序列时，可以设定符合要求的路径的代价容忍度上限，即所有代价容忍度超过上限的路径序列都被去掉。比如上限如果设为 1.2，即所有代价超过最小代价的 20%的路径序列将被剪枝掉。

将使用传统的序列模式挖掘算法(可理解为代价容忍度上限为无穷大)所得挖掘结果，与使用了代价容忍度上限的挖掘结果进行对比，显然如果代价容忍度过低，潜在的频繁的路径序列可能因剪枝而被去掉，从而使两者的挖掘结果不一致，因此必须适当调高代价容忍度；而如果代价容忍度过高，则算法将退化为传统的序列模式挖掘算法，从而使剪枝效果不明显。因此要获得最佳的挖掘效果，具体应用于某一物流领域时可以先用该领域中的物流数据做一些测试以得出最佳代价容忍度。

在介绍 PMWTI 算法之前先介绍几个其中会用到的相关概念与方法。物流网络基本的拓扑信息，包括网络中节点个数与网络的邻接矩阵。设网络中节点个数为 N，则相应的邻接矩阵 neib 为 $N\times N$ 的矩阵，在算法的实现中 neib 为 $N\times N$ 的二维数组，其中 $neib[i][j]$表示节点 i 与节点 j 的邻接距离(代价)，若 $neib[i][j]$等于 0，则表示节点 i 与节点 j 不相邻。利用最短路径算法可以从邻接矩阵中求出任意两点之间的距离，计算结果类似邻接矩阵，同样用矩阵表示，在此称之为最小代价矩阵。算法实现中最小代价矩阵用 $N\times N$ 的二维数组 $least$ 表示，其中 $least[i][j]$表示节点 i 与节点 j 之间最短

路径的代价，若 $least[i][j]$ 等于 0，则表示节点 i 与节点 j 之间无路径。本文在 PMWTI 中引入的候选序列剪枝方法在此记为代价容忍度剪枝法，该剪枝方法为：首先求出该候选序列所表示路径的代价 $cost$；然后算出代价容忍度参数（记为 TD）与该路径所连两点的最小代价的乘积 $ultimate$，设该路径为从节点 i 到节点 j 的路径，则 $ultimate=least[i][j]\times TD$，$ultimate$ 即为从节点 i 到节点 j 所能接受的路径代价的最大值；最后将 $cost$ 与 $ultimate$ 做比较，如果 $cost>ultimate$，则相应的候选序列不符合要求，必须从候选序列里去掉。

PMWTI 算法描述如下：

输入：物流路径数据集 set，最小支持度 SUP，代价容忍度参数 TD，物流网络中的基本拓扑信息。

输出：所有满足支持度 SUP 的频繁路径的集合。

step 1　读取数据集并将之转换为如表 5 所列矩阵并存于主存之中。

step 2　读取物流网络的基本拓扑信息，获得邻接矩阵 neib 与节点个数 N，然后根据 neib 求出最小代价矩阵 least。

step 3　获取候选 K 序列，如果所得候选 K 序列的集合为空，则挖掘结束；否则执行 step 4。首次执行此步骤所求候选序列为物流网络中的所有节点的集合；不是首次执行，则首先通过 step 4 获得的频繁 K－1 序列连接（连接方法前文已有描述）获得初步的候选 K 序列，然后对该序列使用代价容忍度剪枝法。

step 4　扫描 step 1 中的矩阵，从候选序列中求出频繁序列。如果所求频繁序列集合为空则挖掘结束，否则转到 step 3。

PMWTI 中引入了最小代价矩阵，记节点数为 n，则该矩阵的求出与存储所需时间与空间代价分别为 $O(n^3)$ 与 $O(n^2)$，通常情况下与频繁路径挖掘中的代价相比是可忽略不计的，除非最小支持度设定过大，导致算法很快结束，没有挖掘出较长模式，当然这种挖掘也没有意义。

4　实验分析

本文研究建立在文献[12]的基础之上，实验所用物流网络（取 100 个节点）由该文献中的方法随机生成。在之后的进一步研究中对其中提出的 SWPL 算法上做了一定修改，引入了一个类似代价容忍度的参数，用于获取以某个点为起点的所有合乎要求的路径，这个要求即路径的代价不超过最小代价与该参数的乘积，在此姑且称该算法为 M-SWPL（Multiple-SWPL）。实验所用数据集的生成是建立在 M-SWPL 的基础上的，首先设定该容忍度参数（实验中使用的是 1.2），对每一节点（记为 i 节点）都执行 M-SWPL 算法以求出所有符合要求的从 i 节点出发的路径；然后给其中所有以 i 节点为起点的任意 j 节点之间的路径赋予一个随机的物流权重，该权重即为从 i 节点到 j 节点的总的物流权重；最后将该权重按一定比例分配给这些得出的从 i 节点到 j 节点的路径，该分配比例与路径的代价有关，在程序实现中路径所分得权重与路径代价的平方的倒数成正相关，启发来自万有引力定律。简而言之，代价越小所分权重越大，所代表的路径相对越频繁。

实验仅进行一次，用于实验的所获得的数据集中共有 1206480 条记录，其中所有记录的物流权重之和为 480800030，平均每条记录的权重为 2328.6，以取最小支持度为 0.001 为例，即某路径频繁则其物流权重之和必须大于等于 480800030×0.001。首先测试数据集，其中使用了两个算法。第一个是本文提出的算法 PMWTI；第二个是传统的类 Apriori 挖掘算法，即由 PMWTI 算法描述去掉利用拓扑信息进行剪枝的那部分所组成的算法，在此记为 TRADITIONAL。将使用 PMWTI 挖掘的结果与相同最小支持度下 TRADITIONAL 挖掘出的结果进行比较，其中 PMWTI 采用了一系列的代价容忍度，代价容忍度取值分别为 1.0、1.1、1.2、1.3、1.4、1.5、1.6、1.7、1.8、1.9、2.0。最小支持度取值分别为 0.001、0.003、0.005、0.007、0.01、0.03、0.05、0.07。比较结果如图 2 所示，其中横坐标为最小支持度，纵坐标为保证两算法挖掘结果一致的最低代价容忍度。例如最小支持度为 0.001 时，最低代价容忍度为 1.6，即 PMWTI 中代价容忍度取 1.6 及以上的挖掘结果与 TRADITIONAL 的挖掘结果一致，而取 1.5 及以下挖掘结果与 TRADITIONAL 的挖掘结果不一致。分析挖掘结果可知，最小支持度从 0.001 到 0.07 所挖掘出来的最长频繁路径的长度依次为 9、8、7、6、5、3、1、1。之所以最小支持度为 0.001 时最低代价容忍度高达 1.6，是因为相对于实验所用数据集最小支持度过低，以至于几乎所有路径都成为频繁路径，挖掘结果没有什么意义，在这种情况下并不能充分发挥 PMWTI 的优势。当最小支持度从 0.001 提高到 0.003 时，最低代价容忍度从 1.6 降到了 1.3，即当最小支持度稍微调高，所挖掘出来的频繁路径具有一定意义时，PMWTI 效果则得到了良好的体现。随着最小支持度的提升，最低代价容忍度逐渐降低至 1.0，这意味着当最小支持度高到一定程度时要么所挖掘出来的频繁路径为最小代价路径，要么没有任何一条路径满足最小支持度，例如，最小支持度为 0.05 与 0.07 时所挖掘出来的都是单一的节点而非路径序列。

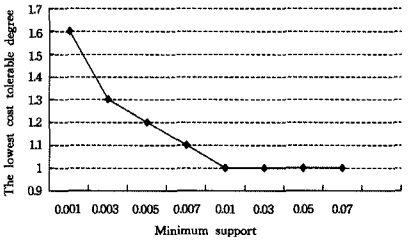

图 2　不同最小支持度下的最低代价容忍度取值

PMWTI 与 TRADITIONAL 的效率比较如图 3 所示，其中横坐标为最小支持度，纵坐标为挖掘所用时间（单位为秒），PMWTI 的代价容忍度取 1.6。显然 PMWTI 在任何支持度下效率都远远优于 TRADITIONAL，完全不考虑任何物流网络拓扑信息的 TRADITIONAL 在从物流数据中挖掘频繁路径时在效率上表现得非常糟糕，两者不具可比性。经过分析发现，TRADITIONAL 的主要问题出在生成候选 2 序列上。由于 TRADITIONAL 中通过频繁 1 序列获取候选 2 序列时是通过排列得到，即如果有 n 个频繁 1 序列则会生成 A_n^2 即 $n(n-1)$ 个候选 2 序列。然而其中有些候选 2 序列在数据集中是不可能出现的，因为物流网络中的路径序列具有路径这一属性，而路径中相邻的任意两点在物流网络拓扑上必是相邻的（在此称这种序列为合法序列），所有不相邻的节点组成的序列是不可能出现在物流数据集中的（在此称序列中相邻

但物流网络中不相邻的序列为非法序列)。对于这个问题,通过引入基本的拓扑信息即可解决,在此提出第三个算法,即在TRADITIONAL中加入邻接矩阵neib这一参数,生成候选2序列ij时根据neib进行判断,如果节点i与节点j不相邻,则将候选2序列ij去掉;否则将之加入候选2序列。之所以只在生成候选2序列时加入这种判断,是因为之后候选K序列($K>2$)的连接生成是建立在合法序列上的,所有通过合法序列的连接得到的序列都是合法序列,候选序列在生成过程中自然而然地获得了这种剪枝效果,在此记该算法为NATURAL。NATURAL较TRADITIONAL在效率上有大幅提升,与PMWTI比较接近,因此下面给出的比较数据中只包含NATURAL与PMWTI的实验数据,TRADITIONAL不再参与比较。

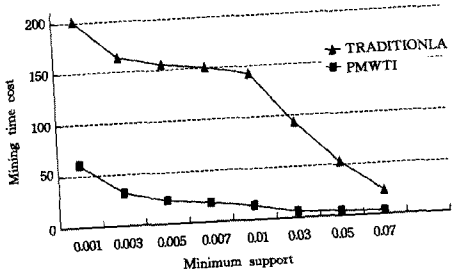

图3 不同最小支持下PMWTI与TRADITIONAL效率的比较

首先在取最小支持度为0.003的情况下对PMWTI与NATUAL进行比较,如图4所示。其中纵坐标表示算法所用时间,单位为秒;横坐标表示代价容忍度,由于代价容忍度仅对PMWTI有效,NATURAL无此参数,因此图4中NATURAL耗时为恒定的37.328秒。由图4可知,随着代价容忍度的不断提高,PMWTI耗时逐渐接近NATURAL;而当代价容忍度过低时,虽然PMWTI相比之下挖掘效率非常高,但挖掘结果可能与常规的挖掘算法的结果不一致,因此选取适当的代价容忍度是很有必要的。

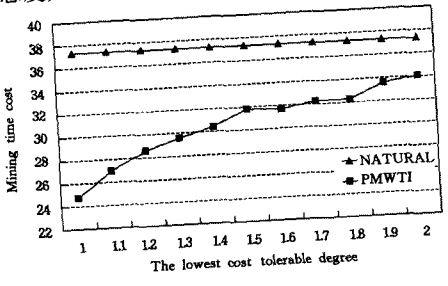

图4 NATURAL与不同代价容忍度下PMWTI挖掘效率的比较

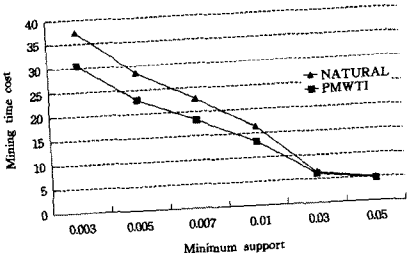

图5 不同支持度下NATURAL与PMWTI的挖掘效率比较

由之前测试结果可知,最小支持度取0.003及以上时代价容忍度取1.3及以上的挖掘...度取值1.4、最小支持度取值范围为0.003到0.05时,对PMWTI与NATUAL再次进行比较,如图5所示,其中横坐标为最小支持度,纵坐标为算法耗时,单位为秒。由图5可见,在一般的最小支持度下(0.003到0.01),PMWTI耗时约为NATURAL的80%。当最小支持度过高(0.03及以上)时两算法效率相当,PMWTI的效率就体现不出来了,分析挖掘结果,根本原因是在这种最小支持度下已经挖掘不出较长的频繁路径,时间基本都消耗在频繁1序列与频繁2序列的挖掘上,PMWTI的剪枝策略基本上没有用武之地。

结束语 本文针对物流数据的特点,为高效地从这些数据中挖掘出频繁路径而设计了相应的物流数据模型与挖掘算法PMWTI,其中数据模型与挖掘算法是相互独立的。该数据模型是建立在以空间换取时间的基础上的。算法PMWTI中设计了一种通用的候选路径序列剪枝策略——代价容忍度剪枝法。该剪枝策略可以用于物流网络中任何生成了候选路径序列的频繁路径挖掘算法,相比没有采用该剪枝方法的同等算法,该策略减少了数据集的扫描,有效地提高了挖掘效率。对于多维的频繁路径挖掘,本文只是稍稍提及,而考虑多维属性的频繁路径挖掘是很具有应用价值的,这可成为今后的研究方向。

参考文献

[1] Han J,Gonzalez H,Li X,et al. Warehousing and mining massive RFID data sets[C]// ADMA' 06. 2006

[2] Gonzalez H,Han J,Li X,et al. Warehousing and analyzing massive RFID data sets [C]// Proc. of the 22nd Int. Conf. on Data Engineering (ICDE'06). 2006:83-92

[3] 胡孔法,张长海,陈峻,等. 一种面向物流数据分析的路径序列挖掘算法ImGSP [J]. 东南大学学报:自然科学版,2008,38(6):970-974

[4] 赵秀丽,徐维祥. 在物流RFID数据库中挖掘时空模式[J]. 物流技术,2011,30(9):101-104

[5] 胡孔法,孙艳,陈峻,等. 现代物流系统中基于频繁子图的RFID路径挖掘算法[J]. 计算机集成制造系统,2010,16(11):2490-2494

[6] 陈竹西. 面向RFID海量数据的若干数据挖掘技术研究[D]. 扬州:扬州大学,2009

[7] 谭晓博. 基于EPC网络的RFID频繁路径挖掘研究与开发[D]. 上海:上海交通大学,2012

[8] Agrawal R,Srikant R. Mining sequential pattern [C]// Proc. of the 11th International Conference on Data Engineering. Taipei, 1995

[9] 张笑达,徐立臻. 一种改进的基于矩阵的频繁项集挖掘算法[J]. 计算机技术与发展,2010,20(4):93-96

[10] 付沙,廖明华,宋丹. 基于压缩矩阵方式的Apriori改进算法[J]. 微电子学与计算机,2012,29(6):28-32

[11] 罗丹,李陶深. 一种基于压缩矩阵的Apriori算法改进研究[J]. 计算机科学,2013,40(12):75-80

[12] 杨俊瑶,蒙祖强. 基于时间依赖的物联网络模型的路径规划[J]. 广西师范大学学报:自然科学版,2013,31(3):152-156