

文章编号:1001-9081(2010)01-0025-04

基于粗糙集的关联规则挖掘方法

贺超波¹, 陈启买²

(1. 仲恺农业工程学院 计算机科学与工程学院, 广州 510225; 2. 华南师范大学 计算机学院, 广州 510631)

(hechaobo@tom.com)

摘要:对粗糙集进行了相关研究,并提出一种以粗糙集理论为基础的关联规则挖掘方法,该方法首先利用粗糙集的特征属性约简算法进行属性约简,然后在构建约简决策表的基础上应用改进的 Apriori 算法进行关联规则挖掘。该方法的优点在于消除了不重要的属性,减少了属性数目和候选项集数量,同时只需一次扫描决策表就可产生决策规则。应用实例及实验结果分析表明该方法是一种有效而且快速的关联规则挖掘方法。

关键词:数据挖掘;关联规则;粗糙集;分辨矩阵;Apriori 算法

中图分类号: TP392 **文献标志码:** A

Approach for mining association rules based on rough set

HE Chao-bo¹, CHEN Qi-mai²

(1. College of Computer Science and Engineering, Zhongkai University of Agriculture and Engineering, Guangzhou Guangdong 510225, China; 2. School of Computer, South China Normal University, Guangzhou Guangdong 510631, China)

Abstract: In this paper a novel approach for mining association rules based on rough set theory was presented. It firstly used the rough set's feature attributes reduction algorithm to reduce attributes, and then applied the improved Apriori algorithm to mine association rules based on the reduction decision table. The advantages of this method were that it reduced the number of the attributes by reducing the unimportant attributes, also reduced the number of candidate item sets and could produce the association rules by scanning decision table only once. The results of application and experiments show that this method is fast and effective.

Key words: data mining; association rule; rough set; discernable matrix; Apriori algorithm

0 引言

关联规则的挖掘一直是数据挖掘领域研究热点之一,其目的是在大型事务数据库中发现项目之间的关联模式^[1]。自 1994 年 Agrawal 等人提出传统的关联规则挖掘算法 Apriori 以来,针对该算法运行效率的不足,众多研究人员提出了许多新的改进算法。如 Partition^[2]、频繁闭项集法^[3]、FP-Growth^[4]以及 TBAR^[5]等算法。尽管这些算法各具优点且在性能和效率上均明显高于传统的 Apriori 算法,但当数据集属性数目较多时,这些算法的挖掘效率仍然较低。解决该问题的根本原则是进行数据预处理时进行属性的约简,从而减少数据挖掘规模,为此已有不少相关解决方法,大致可以分为三类:1) 基于代数理论的^[6];2) 基于信息熵理论的^[7-8];3) 基于粗糙集分辨矩阵和分辨函数的^[9]。

粗糙集(Rough Set)理论^[10]主要用于处理和提取含糊性和不精确性的知识并在数据挖掘领域得到成功应用。利用粗糙集理论进行数据挖掘,其主要思想是在保持分类能力不变的前提下,利用基于粗糙集的属性约简方法减少属性的数目,进而可以归纳总结出适用于决策支持的规则。

针对以上关联规则挖掘相关问题,本文以粗糙集理论为基础,提出了一种新的关联规则挖掘方法。并从减少属性数目、减少候选项集数量以及决策表扫描次数等方面对 Apriori 算法进行了改进。文中给出了相关算法,并通过应用实例及实验结果分析验证了该方法的有效性。

1 粗糙集基本理论

1.1 粗糙集理论的基本定义以及性质

定义 1 一个知识表达系统 S 定义为一个五元组, $S = \langle U, C, D, V, f \rangle$, 其中 U 表示对象的集合, 记为 $U = \{x_1, x_2, \dots, x_n\}$; $R = C \cup D$, 是属性的集合, 其中 C 表示条件属性集, 而 D 表示决策属性集; $V = \bigcup_{r \in R} V_r$ 是属性值的集合, 即属性的值域集, 其中 V_r 是属性 $r \in R$ 的值域; f 是信息函数, $f: U \times R \rightarrow V$, 即 $f(x, R) \in V_r$, 它指定了 U 中每一对象 x 的属性值。

定义 2 一个决策表定义为 $DT = (U, C \cup D, V, f)$, 其中 U, C, D, V, f 等符号的意义均如定义 1 所述。

定义 3 对于任意 $x_i, x_j \in U$, 如果 $C(x_i) = C(x_j)$, 且 $D(x_i) = D(x_j)$, 则 DT 被称为一致性决策表, 否则被称为不一致决策表。

定义 4 对于 $P \subseteq R$, 我们可以定义一个 P 上的不可分辨关系:

$$\text{ind}(P) = \{(x, y) \in U \times U \mid \forall a \in P, f(x, a) = f(y, a)\}$$

定义 5 对于 $x \in U$, 可用集合 $[x]_P = \{y \in U \mid (x, y) \in \text{ind}(P)\}$ 来表示包含元素 x 的等价类。

定义 6 令 X 是 U 中根据条件属性集 C 定义的分类, Y 是 U 中根据决策属性 D 定义的分类, 对于 $X_i \in X, Y_i \in Y$, 定义一个函数 $Dx: Des(X_i) \rightarrow Des(Y_i)$, 函数 Dx 称为决策表 DT 中的决策规则。

定义 7 对于每个规则 $Des(X_i) \rightarrow Des(Y_i)$, 度量规则的

收稿日期:2009-07-18;修回日期:2009-09-03。

基金项目:广东省科技计划项目(2009B010800036);仲恺农业工程学院自然科学基金资助项目(G3091813)。

作者简介:贺超波(1981-),男,广东河源人,讲师,硕士,主要研究方向:数据库技术、数据仓库、数据挖掘;陈启买(1965-),男,湖南衡阳人,教授,主要研究方向:数据仓库、数据挖掘。

参数有支持度(Support)和置信度(Confidence),分别定义为:

$$Support = |X_i \wedge Y_i| / |U|$$

$$Confidence = |X_i \wedge Y_i| / |X_i|$$

性质1 由定义4和定义5推导可得出如下结论:假设 $X \subset R, Y \subset R, a \in V_x, a \in V_y$, 则有 $[XY]_a = [X]_a \cap [Y]_a$ 。

1.2 属性约简概念及其求解方法

属性约简是粗糙集用于数据分析的重要概念,可以减少属性数目,提高分析效率。属性约简定义为不含多余属性并保证分类正确的最小条件属性集。假设条件属性集 C 的约简是 C 的一个非空子集 C' , C 和 C' 必须满足以下两个性质:

1) $ind(C, D) = ind(C', D)$, 不存在 $C'' \subset C'$ 有 $ind(C'', D) = ind(C', D)$ 。

2) 一个决策表可能同时存在几个约简, C 的约简的集合记作 $Red(C)$ 。这些约简的交集定义为决策表的核(Core), $Core(C) = \cap Red(C)$, 核中的属性是影响分类的重要属性。

计算最小约简的复杂性随着决策表的增大呈指数增长,是一个典型的 NP 完全问题。文献[9]中提出一种基于分辨矩阵的条件属性约简有效方法,可以获的最优解或近似最优解。其主要思想是通过构造分辨矩阵,化简由分辨矩阵导出的分辨函数成标准式,该标准式所有的蕴含式包含的属性就是决策表所有的约简集合,最后选取包含核属性的最小约简集作为最终约简集。该方法中分辨矩阵 M 是一个 $|U| \times |U|$ 矩阵,其中的每一个元素 M_{ij} 都是 C 的一个子集,即 $M_{ij} \in C$ 。 M_{ij} 的具体定义如下:

$$M_{ij} = \{m_{i1}, m_{i2}, \dots, m_{in}\}; i, j = 1, 2, \dots, |U|$$

$$m_{ik} = \begin{cases} \Phi, & c_{ik} = c_{jk} \\ c_k, & c_{ik} \neq c_{jk} \end{cases}; k = 1, 2, \dots, |C|$$

可知 M 为主对角线为 Φ 的对称矩阵,在实际应用中只计算它的上三角阵。构造分辨函数可把 M_{ij} 的每个属性值进行“或”运算,然后再“与”其所有的 $M_{ij} (M_{ij} \neq \Phi)$, 其中 $i, j = 1, \dots, |U|$, 结果用 F 表示。应用分辨矩阵进行属性约简的优势在于可以利用该矩阵的对称性将计算量减半,并且可以对约简后的决策表直接提取规则。

2 基于粗糙集理论的关联规则挖掘方法

2.1 基本思想

该方法的基本思想是首先利用基于分辨矩阵的条件属性约简算法确定决策表 K 的一个约简集 K_reduct , 然后利用 K_reduct 将 K 转化为适合 Apriori 算法分析的具有布尔属性的新决策表 K' 。在新的决策表上对条件属性集运用 Apriori 算法首先生成频繁 1 项集,并删除其中支持度小于阈值的项,记为 L_{c1} ;在 L_{c1} 基础上继续生成频繁 i 项集,其中 $L_{ci} = L_{ci-1} \times L_{c1}$, i 大于 1, 小于或等于约简集的长度,并删除 L_{ci} 中支持度小于阈值的项;同理对于决策属性集 D ,可在 L_{d1} 基础上计算 $L_{dj} = L_{dj-1} \times L_{d1}$, j 大于 1, 小于或等于决策属性集的长度;接着 L_{ci} 与 L_{dj} 运用性质 1 进行两两连接运算生成长度为 $i+j$ 的频繁项集,记为 L_{i+j} , L_{i+j} 中支持度与置信度大于或等于阈值的频繁项集可放入规则集 RS 中。每次计算 L_{ci}, L_{dj}, L_{i+j} 都可以根据性质 1 利用之前的计算结果,避免了再次扫描决策表。最后对规则集 RS 中的规则进行合并优化后即所需挖掘的关联规则。

该方法的实现步骤描述如下:

步骤 1 数据概化形成决策表、确定条件属性以及决策属性。

步骤 2 运用基于分辨矩阵的属性约简方法对条件属性

进行约简和求核,当存在多个核值时,选择包含核值的最小约简集。

步骤 3 将约简后的决策表转换为基于布尔属性的新决策表。

步骤 4 根据支持度和置信度阈值,运用改进后的 Apriori 算法发现频繁 K 项集 L_k ,并生成规则集 RS 。

步骤 5 优化合并 RS 中的关联规则,由规则集 RS 生成形式为: $Des(c_1) \wedge Desc(c_2) \wedge \dots \wedge Desc(c_n) \rightarrow Des(d_1) \wedge Desc(d_2) \wedge \dots \wedge Desc(d_n)$ 的关联规则。

2.2 关键算法描述

该方法主要涉及到属性约简和求核、转换生成新决策表以及关联规则集 RS 的生成 3 个关键算法。其中属性约简和求核算法在文献[9]已提到,下面给出本文设计的另外两个算法的描述。

算法 1 Decisiontable _transform,使用约简集转换生成新决策表。

输入: K 为具有完全属性的决策表; C 为条件属性集合; D 为决策属性集合; K_reduct 为条件属性约简集。

输出: 具有布尔属性的新决策表 K' 。

- 1) For K_reduct 中每一个条件属性 C_i {
- 2) 计算 C_i 的值域集 V_{Ci} ;
- 3) 将 V_{Ci} 中每一个元素写入新决策表 K' 的条件属性集 C' 中;}
- 4) For D 中每一个决策属性 D_i {
- 5) 计算 D_i 的值域集 V_{Di} ;
- 6) 将 V_{Di} 中每一个元素写入新决策表 K' 的决策属性集 D' 中;}
- 7) For K' 中每一个元组 t {
- 8) For $K_reduct \cup D$ 中每一个属性 a {
- 9) If $f(t, a) = a(m), a(m) \in C' \cup D'$ Then
- 10) 在 K' 中, $f(t, a(m)) = 1$;
- 11) Else
- 12) 在 K' 中, $f(t, a(m)) = 0$; }

算法 2 RS_generate, 关联规则集 RS 生成算法。

输入: K' 为新决策表; C' 为 K' 中的条件属性集; D' 为 K' 中的决策属性集; $minsup$ 为最小支持度; $minconf$ 为最小置信度。

输出: 关联规则集 RS 。

- 1) For C' 中每一元素 c_i {
- 2) 计算生成 c_i 的频繁 1 项集 L_{ci}'
- 3) If $|L_{ci}'| / |K'| \geq minsup$ Then
- 4) $L_{c1} = L_{c1} \cup c_i // L_{c1}$ 为条件属性符合 $minsup$ 的频繁 1 项集
- 5) For D' 中每一元素 d_i {
- 6) 计算生成 d_i 的频繁 1 项集 L_{di}'
- 7) If $|L_{di}'| / |K'| \geq minsup$ Then
- 8) $L_{d1} = L_{d1} \cup d_i // L_{d1}$ 为决策属性符合 $minsup$ 的频繁 1 项集
- 9) For ($k = 2; k \leq |C'|; k++$) {
- 10) $L_{Ck} = L_{C(k-1)} \times L_{c1}$;
- 11) 结合性质 1 计算 L_{Ck} 中各元素的支持度并删除小于 $minsup$ 的元素;
- 12) For ($k = 2; k \leq |D'|; k++$) {
- 13) $L_{Dk} = L_{D(k-1)} \times L_{d1}$;
- 14) 结合性质 1 计算 L_{Dk} 中各元素的支持度并删除支持度小于 $minsup$ 的元素;
- 15) For ($i = 1; i \leq |C'|; i++$) {
- 16) For ($j = 1; j \leq |D'|; j++$) {
- 17) $L_{i+j} = L_{i+j} \cup (L_{Ci} \times L_{Dj})$
// L_{i+j} 为同时具有条件属性、决策属性的频繁集
- 18) 结合性质 1 计算 L_{i+j} 中各元素的置信度 $conf'$, 如果 $conf' \geq minconf$, 则加入 RS 中;
- 19) }}

2.3 性能分析

该方法首先通过属性约简算法减少了条件属性数目,然后在算法 RS_generate 中只在生成频繁 1 项集时才需扫描决策表,往后寻找更大的频繁项集都可以根据粗糙集理论性质 1 利用前一次计算结果,避免了再次扫描决策表;同时通过支持度阈值过滤不符合要求的频繁集,减少了候选项集的数量,节省了存储空间。可见该方法可降低运算时间和空间复杂度,提高挖掘关联规则的效率。

3 应用实例

以某银行信用卡申请批准数据^[11]为例挖掘其中的关联规则,对数据库的关系表进行数据预处理后得到表 1 所示的决策表。

表 1 某银行信用卡申请批准数据

申请号	账户	余额	职业	月工资	决策
1	银行	中等	有	低	同意
2	银行	低	有	高	拒绝
3	无	低	有	中等	拒绝
4	其他金融机构	高	有	高	同意
5	其他金融机构	中等	有	高	拒绝
6	其他金融机构	高	有	低	同意
7	银行	高	无	中等	同意
8	无	低	无	低	拒绝

为方便使用分辨矩阵对该决策表进行约简和求核,使用字母编号代替各属性值,概化表 1 得到表 2。

表 2 概化后的决策表

U	A	B	C	D	E
1	A ₁	B ₂	C ₁	D ₃	E ₁
2	A ₁	B ₃	C ₁	D ₁	E ₂
3	A ₃	B ₃	C ₁	D ₂	E ₂
4	A ₂	B ₁	C ₁	D ₁	E ₁
5	A ₂	B ₃	C ₁	D ₁	E ₂
6	A ₂	B ₁	C ₁	D ₃	E ₁
7	A ₁	B ₁	C ₂	D ₂	E ₁
8	A ₃	B ₃	C ₂	D ₃	E ₂

表 2 中 U、A、B、C、D、E 分别代表申请、账户、余额、职业、月工资、决策等属性;A_i、B_j、C_k、D_l、E_m (i, j, k, l, m ∈ 1, 2, ..., N) 分别为各属性在表 1 中对应位置的取值。

确定 A、B、C、D 为条件属性, E 为决策属性,接下来根据条件属性构建决策表的分辨矩阵,由分辨矩阵定义可得如下分辨矩阵:

$$\begin{bmatrix} 0 & AC & C & C & BC & CD & A & D \\ & 0 & BC & CD & CD & C & A & B \\ & & 0 & C & C & C & D & AB \\ & & & 0 & ACD & ABC & B & 0 \\ & & & & 0 & AC & 0 & 0 \\ & & & & & 0 & B & D \\ & & & & & & 0 & C \\ & & & & & & & 0 \end{bmatrix}$$

由分辨矩阵可计算分辨函数 $F = (A \wedge B) \vee (B \wedge C)$, 即约简集为 {A, B}, {B, C}, 核为 B, 选取 {A, B} 为最小约简集 K_{reduct}, 删除多余属性, 根据算法 1 将表 2 转化形成具有布

尔属性的决策表(表 3)。

表 3 约简后决策表的布尔表

U	A			B			E	
	A ₁	A ₂	A ₃	B ₁	B ₂	B ₃	E ₁	E ₂
1	1	0	0	0	1	0	1	0
2	1	0	0	0	0	1	0	1
3	0	0	1	0	0	1	0	1
4	0	1	0	1	0	0	1	0
5	0	1	0	0	1	0	0	1
6	0	1	0	1	0	0	1	0
7	1	0	0	1	0	0	1	0
8	0	0	1	0	0	1	0	1

假设 $minsup = 25\%$; $minconf = 75\%$, 执行算法 RS_generate 进行关联规则挖掘。其中产生的满足 $minsup$ 的频繁 K 项集有: $L_{C_1} = \{A_1, A_2, A_3, B_1, B_2, B_3\}$, $L_{D_1} = \{E_1, E_2\}$, $L_{C_2} = L_{C_1} \times L_{C_1} = \{(A_2, B_1), (A_3, B_3)\}$, $L_{D_2} = \emptyset$; $L_2 = L_{C_1} \times L_{D_1} = \{(A_1, E_1), (A_2, E_1), (A_3, E_2), (B_1, E_1), (B_3, E_2)\}$, $L_3 = L_{C_2} \times L_{D_1} = \{(A_2, B_1, E_1), (A_3, B_3, E_2)\}$ 。在 L_2 中, $B_1 \rightarrow E_1, B_3 \rightarrow E_2$, 置信度均为 $100\% > minconf$, 放入规则集 RS 中; L_3 中 $A_2, B_1 \rightarrow E_1, A_3, B_3 \rightarrow E_2$ 置信度均为 $100\% > minconf$, 放入规则集 RS 中, 则 $RS = \{(B_1, E_1), (B_3, E_2), (A_2, B_1, E_1), (A_3, B_3, E_2)\}$, RS 中的关联规则可进一步合并优化得到最小化关联规则 $B_1 \rightarrow E_1$ 与 $B_3 \rightarrow E_2$, 最后由表 1 得到的关联规则为: $Des(B_1) \rightarrow Des(E_1)$ 即余额 = 高 \rightarrow 决策 = 同意 (置信度为 100%), $Des(B_3) \rightarrow Des(E_2)$ 即余额 = 低 \rightarrow 决策 = 不同意 (置信度为 100%)。

4 实验与结果分析

实验目的是对传统的 Apriori 算法与采用本文算法进行关联规则挖掘执行时间效率上的比较, 实验环境部署在一台 CPU 为 Intel Pentium Dual 2.0 GHz, 内存 2 GB, 操作系统为 Windows XP SP2 的 PC 机上, 开发语言采用 VC++ 6.0, 测试数据来自 5 个通用的 UCI 数据集: Zoo、breast-cancer、Tic-Tac-Toe、Solar Flare 和 mushroom。实验中首先采用属性约简算法进行属性约简, 结果如表 4 所示。

表 4 实验数据集及属性约简结果

数据集	原属性数	约简后属性数	总记录数
Zoo	17	10	101
breast-cancer	10	7	286
Tic-Tac-Toe	10	8	958
Solar Flare	10	8	1389
mushroom	22	8	8124

然后在选取不同数据集情况下分别执行 Apriori 算法与本文算法进行挖掘规则数和时间效率上的比较, 同时对 breast-cancer 以及 mushroom 数据集在分别选取不同记录数情况下观察 Apriori 算法与本文算法执行时间的变化, 结果如表 5、图 1、2 所示 (设 $minsup = 10\%$, $minconf = 50\%$)。

从表 5 可以看出, 对于同一数据集, 本文算法挖掘的关联规则数与 Apriori 相差不大, 可以将置信度阈值设置较低便可与其接近。图 1、图 2 表明随着记录数的增多, 传统的 Apriori 算法随着频繁项集数量以及扫描数据集次数增多, 执行时间

有突变性的增加,而本文方法有效地应用了粗糙集的相关性质,减少了频繁项集数量以及数据集扫描次数,执行时间变化相对平稳而且平均约减少3~4倍时间。

表5 Apriori 算法与本文算法挖掘规则数和用时

数据集	Apriori 算法		本文算法	
	规则数	用时/s	规则数	用时/s
Zoo	11	0.33	9	0.12
breast-cancer	5	2.34	3	0.45
Tic-Tac-Toe	32	4.46	18	1.13
Solar Flare	19	5.32	12	1.98
mushroom	37	56.80	28	9.30

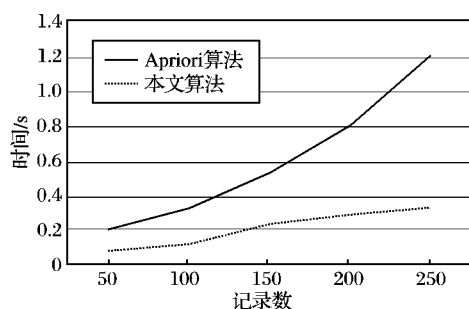


图1 breast-cancer 数据集测试结果

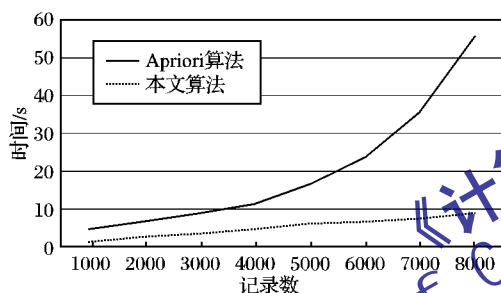


图2 mushroom 数据集测试结果

5 结语

本文给出的关联规则挖掘方法基于粗糙集理论,对传统的Apriori算法进行了改进,解决了该算法需多次扫描数据库

及产生庞大的候选项集的问题。理论及实验证明该方法可有效进行属性约简以及关联规则的挖掘,在运行时间复杂度、空间复杂度上都有一定程度上的改善。但在实际应用中最终参与决策规则挖掘的条件属性及决策属性的选择仍然依靠人为判断,所以研究如何建立相关分析模型自动确定条件属性、决策属性有待进一步的工作。

参考文献:

- [1] HAN JIAWEI, KAMBER M. Data mining concepts and techniques [M]. San Francisco: Morgan Kaufmann Publishers, 2005.
- [2] SAVASERE A, OMIECINSKI E, NAVATHE S. An efficient algorithm for mining association rules in large database[C]// Proceedings of the 21th International Conference on Very Large DataBase. San Francisco: Morgan Kaufmann Publishers, 1995: 432 - 443.
- [3] PASQUIER N, BASTIDE Y. Discovering frequent closed item sets for association rules[C]// Proceedings of the 7th International Conference on Database Theory. London: Springer-Verlag, 1999: 398 - 416.
- [4] HAN JIAWEI, PEI JIAN. Mining frequent patterns without candidate generation[C]// Proceedings of the 20th ACM SIGMOD International Conference on Management of Data. New York: ACM, 2000: 1 - 12.
- [5] BERZAL F, CUBERO J, MARIN N. TBAR: An efficient method for association rule mining in relational databases[J]. Data and Knowledge Engineering, 2001, 37(1): 47 - 64.
- [6] 叶东毅. 基于近似精度递归计算的一个属性约简算法[J]. 小型微型计算机系统, 2003, 24(12): 2272 - 2274.
- [7] 刘振华, 刘三阳, 王珏. 基于信息量的一种属性约简算法[J]. 西安电子科技大学学报, 2003, 30(6): 835 - 838.
- [8] 王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759 - 766.
- [9] 唐建国, 谭明术. 粗糙集理论中的求核和约简[J]. 控制与决策, 2003, 7(4): 449 - 452.
- [10] PAWLAK Z. Rough sets[J]. International Journal of Computer Information Science, 1982, 11(5): 341 - 356.
- [11] PAWLAK Z. Rough sets: Theoretical aspects of reasoning about data [M]. Netherlands: Kluwer Academic Publishers, 1991.

(上接第24页)

更少的存储空间,因为 SAB^x 树节点的数据项为一维数值,并且采用动态的曲线填充的方法使得二进制串长度随秩的变化而变化,可以在一定程度上节省存储空间。

4 结语

针对移动对象当前及未来信息的索引方法,本文在 B^x 树的基础上提出了一种自适应 SAB^x 树。该方法考虑到移动对象在空间中分布密度不同,引入了 Hilbert 曲线动态地对整个索引空间进行填充,根据秩的范围大小划分 Hash 辅助索引表的各个单元,各单元通过 pt 指针域对相应的 B^x 树进行存储管理。实验结果表明 SAB^x 树比 TPR 树节省了很多存储空间,比现有的 B^x 树在范围查询效率方面有所提高。本文将继续研究 SAB^x 索引结构的 K 近邻查询和连续查询算法。

参考文献:

- [1] TRAJCEVSKI G, WOLFSON O, HINRICHS K. Managing uncertainty in moving objects databases[J]. ACM Transactions on Database Systems, 2004, 29(3): 463 - 507.
- [2] BECKMANN N, KRIEGER H, SCHNEIDER R, et al. The R*-

Tree: An efficient and robust access method for points and rectangles[C]// Proceedings of the ACM SIGMOD on Management of Data. New York: ACM, 1990: 322 - 331.

- [3] SALTENIS S, JENSEN C S, LEUTENEGGER S T, et al. Indexing the position of continuously moving objects[C]// Proceedings of the ACM SIGMOD on Management of Data. New York: ACM, 2000: 331 - 342.
- [4] TAO Y, PAPADIAS D, SUN J. The TPR*-tree: An optimized spatio-temporal access method for predictive queries[C]// Proceedings of the VLDB. Berlin: Morgan Kaufmann, 2003: 790 - 801.
- [5] PATEL J M, ARBOR A, CHEN Y, et al. STRIPES: An efficient index for predicted trajectories[C]// Proceedings of the ACM SIGMOD on Management of Data. New York: ACM, 2004: 635 - 646.
- [6] JENSEN C S, LIN D, OOI B C. Query and update efficient B^+ -Tree based indexing of moving objects[C]// Proceedings of the VLDB. Berlin: Morgan Kaufmann, 2004: 768 - 779.
- [7] MOON B, JAGADISH H V, FALOUTSOS C, et al. Analysis of the clustering properties of the Hilbert space-filling curve[J]. IEEE Transactions on Knowledge and Data Engineering, 2001, 13(1): 124 - 141.