

基于 SQL 语言的粗糙数据查询^{*}

曹付元¹ 梁吉业^{1,2}(山西大学计算机科学系 太原030006)¹(中国科学院计算技术研究所智能信息处理重点实验室 北京100080)²

摘要 查询在数据库中十分重要,而 SQL 是检索数据的一种强有力的工具。本文提出一种 RRDB 的粗糙查询方法,它扩展了标准 SQL 语言。实例表明了这种方法具有较好的应用前景。

关键词 粗糙集理论,粗糙查询,SQL,RRDB

The Rough Data Query Based on SQL Language

CAO Fu-Yuan¹ LIANG Ji-Ye^{1,2}(Department of Computer Science, Shanxi University, Taiyuan 030006)¹(Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)²

Abstract Query is one of the important topics in database, SQL is emerging as powerful tool for retrieving about data. A method of rough query of RRDB is proposed in this paper. The new method extends standard SQL language. An illustrative example is provided that shows the application potential of the method.

Keywords Rough set theory, Rough query, SQL, RRDB

1 引言

Z. Pawlak 提出的粗糙集理论^[1-3],是一种处理不精确、不确定和模糊知识的软计算工具。它已被成功地应用在决策分析、过程控制、数据挖掘、模式识别和智能信息处理等领域^[4,5],但是,由于经典粗糙集理论采用等价关系作为基础,这不能完全满足实际应用的需要,因此提出了各种新的模型和扩展。例如可变精度粗糙集模型(VPRS)^[6],相似关系模型^[7]等。

在实际应用中,信息系统中有不少属性取多值的情况。例如,数据收集的时候并不知道该属性的具体值,而只知道该属性可能是取几个值中的一个;再如在文本分类中,一个文档往往可以是多个类别。为此提出了多值粗糙集模型来满足实际的需要,如多值粗糙集模型^[8]等。查询在信息系统中占有十分重要的地位,在多值信息系统中也有一些查询方法被提出,如基于投影原理的粗糙关系数据库(RRDB)的 Rough 数据查询^[9]等,它们使用的粗糙查询方法是根据要查询处理的语义将 RRDB 分解成标准的数据库关系,然后用 SQL 语言和 Rough 操作对它进行处理,该方法在时间和空间上都存在一定程度的浪费。

本文提出一种新的查询方法,它既有标准 SQL 语言的特点,同时在检索时不是以值的运算,而是以类的运算为条件,扩展了标准 SQL 语言。实例表明该方法可以提高查询效率,减少存储空间,具有较好的应用价值。

2 基本概念

2.1 粗糙集(Rough Set)

Rough Set 是由 Z. Pawlak^[1]于1982年提出的。下面介绍

粗糙集的一些基本概念。

定义1^[1] 设 $A=(U, R)$ 是一个近似空间,其中 U 是一个非空有限集合称为论域, R 是 U 上的一个划分(或等价关系), $[x]_R$ 为 x 在 R 关系下的等价类。设 $X \subseteq U$, 定义 X 在 U 上的下近似和上近似分别如下:

$$\underline{R}X = \{x \in U \mid [x]_R \subseteq X\}$$

$$\overline{R}X = \{x \in U \mid [x]_R \cap X \neq \emptyset\}$$

我们称 $BN_R(X) = \overline{R}X - \underline{R}X$ 为 X 的边界或边界区域, $POS_R(X) = \underline{R}X$ 为集合 X 的 R -正区域, $NEG_R(X) = U - \overline{R}X$ 为集合 X 的 R -负区域。

2.2 粗糙关系数据库(RRDB)

粗糙关系数据库是 Codd 标准关系数据库模型的扩展,它拥有粗糙集理论所有本质特征,如通过使用等价类对对象进行不可区分关系的表示以及通过上、下近似表示一个不确定集。在标准关系数据库中我们有精确和模糊查询,同样在粗糙关系数据库中我们可以利用上、下近似来表示可能和确定的匹配。

定义2^[8](多值信息系统) 多值信息系统是有序对 $M=(U, A=C \cup \{d\}, D)$, 其中 U 为非空有限集合,称为全域。全域的无素被称为对象或实例。 A 是属性集合,有两部分组成。基中 C 是非空有限的条件属性集合; d 是一个决策属性; $C \cap \{d\} = \emptyset$; D 是 A 中属性的值域; D_a 是属性 $a \in A$ 的值域。对于每个 $t \in U, a \in A$, 有 $a(t) \subseteq D_a$, 其中 $a(t)$ 表示 t 在 a 上的取值。

多值信息系统是对信息系统的自然扩展。如果对所有的 $t \in U, a \in A$, 都有 $|a(t)|=1$, 则退化为信息系统。

表1是一个多值信息系统的例子。这是一个学校部分学生身体状况的数据集。

^{*} 本文得到了国家自然科学基金(No. 60275019)和山西省自然科学基金项目的资助。曹付元 硕士研究生,主要研究领域为粗糙集理论、数据挖掘。梁吉业 教授,博士,博士生导师,主要研究领域为粗糙集理论、数据挖掘、人工智能。

表1 多值信息系统

Obj	Age	Height
U1	{child, teen}	{high, skyhigh}
U2	{teen, young-adult}	{high, medium}
U3	{child, pre-teen}	{tall, medium}
U4	{adult, sr}	{medium, low}
U5	{child, youth, teen}	{tall, skyhigh}
U6	{adult, sr-citizen}	{medium, short}
U7	{sr-citizen, elderly}	{skyhigh, tiptop}
U8	{young-adult, youth}	{medium}
U9	{pre-teen, teenager}	{high, tall, tiptop}
U10	{teenager, adult}	{medium, low, little}

定义3 设 $M=(U, A, D)$ 是多值信息系统, 对于 $a \in A$ 有 D_a , R_a 是在 D_a 上的一个划分(或等价关系), 令 $X_a = \{c_1, c_2, \dots, c_n\}$, 若 $X_a \subseteq D_a$, 则定义 $[X_a] = [c_1]_{R_a} \cup [c_2]_{R_a} \cup \dots \cup [c_n]_{R_a}$, 若 $X_a \not\subseteq D_a$, 则定义 $[X_a] = [X_a \cap D_a] \cup (X_a - (X_a \cap D_a))$.

性质1 设 $X_a \subseteq D_a, Y_a \subseteq D_a$, 如果 $X_a \subseteq Y_a$, R_a 为 D_a 上的等价关系, 则有 $[X_a] \subseteq [Y_a]$.

例1 下面以表1中的属性 Age 为例, 我们有:

$$D_{age} = \{child, pre-teen, teen, youth, teenager, youth-adult, adult, sr, sr-citizen, elderly\}$$

在 D_{age} 上的等价关系 R_{age} 定义如下

$$R_{age} = \{[child, pre-teen], [teen, youth, teenager], [young-adult], [adult], [sr, sr-citizen, elderly]\},$$

设 $X_{age} = \{child, youth\}$, 又 $X_{age} \subseteq D_{age}$, 则

$$[X_{age}] = \{child, pre-teen, teen, youth, teenager\}$$

设 $X_{age} = \{child, youth, foetus\}$, 由于 $X_{age} \not\subseteq D_{age}$, 则

$$[X_{age}] = \{child, pre-teen, teen, youth, teenager, foetus\}$$

定义4^[10] 关系模式是关系的描述。一个关系模式是一个五元组。它可以形式化地表示为 $R(A, D, DOM, F)$, 其中 R 为关系名, A 为组成该关系的属性名集合, D 为属性组 A 中属性所来自的域, DOM 为属性向域的映象集合, F 为属性间数据的依赖关系集合。关系模式通常可以简记为: $R(A_1, A_2, \dots, A_n)$, 其中 R 为关系名, $A_1, A_2, \dots, A_n$ 为属性名。

定义5 设 R 是一个关系模式, $M=(U, A, D)$ 为该模式上的一个多值系统, 对于 $a \in A$, 设 $X_a = \{c_1, c_2, \dots, c_n\}$, M' 是选择后的多值系统且和 M 有相同的模式, 定义粗糙选择 $\sigma_{\rightarrow X_a}(M)$, 则选择结果用粗糙关系 M' 的下、上近似分别表示如下:

$$\underline{RM'} = \{t \in U \mid [a(t)] = [X_a]\}$$

$$\overline{RM'} = \{t \in U \mid [a(t)] \supseteq [X_a]\}$$

下近似表示的是查询值的等价类和对应属性值上产生的等价类完全相等, 而上近似则表示查询值的等价类是对应属性值上等价类的子集。

3 基于 SQL 语句的粗糙数据查询

定义6 设 $M=(U, A, D)$ 是多值信息系统, 对于 $a \in A$, 设 $X_a = \{c_1, c_2, \dots, c_n\}$, 则定义查询函数

$$Equ(a, X_a) = \begin{cases} 0, & [a(t)] = [X_a] \\ 1, & [a(t)] \supseteq [X_a] \end{cases}$$

其中 $t \in U$, 我们规定函数结果为0则为精确查询, 为1则为模糊查询。

算法描述

1) 对于 $a \in A$ 有 D_a , 根据某种等价关系 R_a 对 D_a 进行等价划分;

2) 根据要查询的值 X_a , 求其在 D_a 的 $[X_a]$;

3) 求 $a(t)$ 在 D_a 上的 $[a(t)]$;

4) 如果要精确查询当且仅当 $[a(t)] = [X_a]$;

5) 如果要模糊查询当且仅当 $[a(t)] \supseteq [X_a]$ 。

由表1知, 属性 Age 和 Height 的值域为 D_{age} 和 D_{height} , 同时定义在 D_{age} 和 D_{height} 的等价划分 R_{age} 和 R_{height} , 它们分别如下:

$$D_{age} = \{child, pre-teen, teen, youth, teenager, youth-adult, adult, sr, sr-citizen, elderly\}$$

$$D_{height} = \{high, tall, skyhigh, tiptop, medium, low, short, little\}$$

$$R_{age} = \{[child, pre-teen], [teen, youth, teenager], [young-adult], [adult], [sr, sr-citizen, elderly]\}$$

$$R_{height} = \{[high, tall], [skyhigh, tiptop], [medium], [low, short, little]\}$$

为使查询显示更方便、直观, 对表1进行变换, 即对所有对象分别求 $[age(t)]$, $[height(t)]$, 显示的结果如表2。

表2

Obj	Age	Height
U1	{child, pre-teen, teen, youth, teenager}	{high, tall, skyhigh, tiptop}
U2	{teen, youth, teenager, young-adult}	{high, tall, medium}
U3	{child, pre-teen}	{high, tall, medium}
U4	{adult, sr, sr-citizen, elderly}	{medium, low, short, little}
U5	{child, pre-teen, teen, youth, teenager}	{high, tall, skyhigh, tiptop}
U6	{adult, sr, sr-citizen, elderly}	{medium, low, short, little}
U7	{sr, sr-citizen, elderly}	{skyhigh, tiptop}
U8	{teen, youth, teenager, young-adult}	{medium}
U9	{child, pre-teen, teen, youth, teenager}	{high, tall, skyhigh, tiptop}
U10	{teen, youth, teenager, adult}	{medium, low, short, little}

3.1 精确查询

例2 针对 D_{age} 和 R_{age} , 查询在属性 Age 上条件为 $X_{age} = \{child, teen\}$ 的精确元组。则

$$[X_{age}] = \{child, pre-teen, teen, youth, teenager\}$$

$[age(t)]$ 如表2所示, 由于是精确查询, 则 $[age(t)] = [X_{age}]$ 。

实现精确查询的 SQL 语句为

Select * From 表1 Where Equ(Age, X_{age}) = 0;

产生的结果如表3。

表3

Obj	Age	Height
U1	{child, teen}	{high, skyhigh}
U5	{child, youth, teen}	{tall, skyhigh}
U9	{pre-teen, teenager}	{high, tall, tiptop}

3.2 模糊查询

例3 针对 D_{age} 和 R_{age} , 查询在属性 Age 上条件为 $X_{age} = \{child, pre-teen\}$ 的模糊元组, 则 $[X_{age}] = \{child, pre-teen\}$, $[age(t)]$ 如表2所示, 由于是模糊查询, 则 $[age(t)] \supseteq [X_{age}]$ 。

实现模糊查询的 SQL 语句为:

Select * From 表1 Where Equ(Age, X_{age}) = 1;

产生的结果如表4。

表4

Obj	Age	Height
U1	{child, teen}	{high, skyhigh}
U3	{child, pre-teen}	{tall, medium}
U5	{child, youth, teen}	{tall, skyhigh}
U9	{pre-teen, teenager}	{high, tall, tiptop}

对于精确查询和模糊查询有一种特殊情况,即查询的值不在它所对应属性的值域内,则查询的结果为空数据集。

例4 针对 D_{age} 和 R_{age} , 查询在属性 Age 上条件为 $X_{age} = \{foetus, child\}$ 的元组, 则

$[X_{age}] = \{child, pre-teen, foetus\}$

$[age(t)]$ 如表2所示, 由于是模糊查询, 则 $[age(t)] \supseteq [X_{age}]$ 。

实现模糊查询的 SQL 语句为

Select * From 表1 Where Equ(Age, X_{age}) = 1;

产生的结果如下:

Obj	Age	Height
-----	-----	--------

即产生了一个空数据集, 对精确查询同样是一个空数据集。

3.3 组合查询

3.3.1 精确查询

例5 针对 D_{age} 和 R_{age} 以及 D_{height} 和 R_{height} , 查询在属性 Age 上条件为 $X_{age} = \{child, teen\}$ 且在属性 Height 上条件为 $Y_{height} = \{high, tiptop\}$ 的精确元组, 则

$[X_{age}] = \{child, pre-teen, teen, youth, teenager\}$

$[Y_{height}] = \{high, tall, skyhigh, tiptop\}$

$[age(t)]$ 和 $[height(t)]$ 如表2所示, 由于是精确查询则 $[age(t)] = [X_{age}]$ 且 $[height(t)] = [Y_{height}]$

实现组合查询中精确查询的 SQL 语句为

Select * From 表1 Where Equ(Age, X_{age}) = 0 and Equ(Height, Y_{height}) = 0

产生的结果如表5。

表5

Obj	Age	Height
U1	{child, teen}	{high, skyhigh}
U5	{child, youth, teen}	{tall, skyhigh}
U9	{pre-teen, teenager}	{high, tall, tiptop}

3.3.2 模糊查询

例6 针对 D_{age} 和 R_{age} 以及 D_{height} 和 R_{height} , 查询在属性 Age 上条件为 $X_{age} = \{adult\}$ 且在属性 Height 上条件为 $Y_{height} = \{medium\}$ 的模糊元组, 则 $[X_{age}] = \{adult\}$, $[Y_{height}] =$

$\{medium\}$, $[age(t)]$ 和 $[height(t)]$ 如表2所示, 由于是模糊查询则 $[age(t)] \supseteq [X_{age}]$ 且 $[height(t)] \supseteq [Y_{height}]$ 。

实现组合查询中模糊查询的 SQL 语句为:

Select * From 表1 Where Equ(Age, X_{age}) = 1 and Equ(Height, Y_{height}) = 1;

产生的结果如表6。

表6

Obj	Age	Height
U4	{adult, sr}	{medium, low}
U6	{adult, sr-citizen}	{medium, short}
U10	{teenager, adult}	{medium, low, little}

结论 本文把标准关系数据库(RDB)中的标准 SQL 进行扩展, 应用在 RRDB 中, 获得了良好的效果, 实例表明该方法较好地提高了查询效率; 且具有较好的应用前景, 为平滑地集成到各种应用系统中提供了方便。

参考文献

- 1 Pawlak Z. Rough sets. International J of Computer and Information Science, 1982, 11: 341~356
- 2 Pawlak Z. Rought Set—Theoretical Aspect of Reasoning about Data. Kluwer Academic Publishers, Dordrecht, Boston, London, 1991
- 3 Pawlak Z, Grzymala-Busse J W, Slowinski R, Ziarko W. Rough sets. Comm. ACM, 1995, 38: 89~95
- 4 Liang Ji Ye, Chin K S, Dang C Y, Yam R C M. A new method for measuring uncertainty and fuzziness in rough set theory. International Journal of General System, 2002, 31(4): 331~342
- 5 Liang Ji Ye, Xu ZongBen. The algorithm on knowledge reduction in incomplete information systems. International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, 2002, 10(1): 95~103
- 6 Ziarko W. Variable precision rough set modal. Journal of Computer and System Science, 1993, 46: 39~59
- 7 Slowinski R, Vanderpooten D. A generalized definition of rough approximations based on similarity. IEEE Trans on Date and Knowledge Engineering, 2000
- 8 胡可云, 眭跃飞, 陆玉昌, 王驹, 石统一. 多值粗糙集模型. 计算机科学, 2001, 28(专辑5): 1~4
- 9 安秋生, 徐久成, 王国胤, 沈钧毅. 基于投影原理的 RRDB 的 Rough 数据查询. 西安交通大学学报, 2002, 36(8): 859~862
- 10 萨师煊, 王珊. 数据库系统概论. 高等教育出版社, 1995
- 11 Beaubouef T, Petry F E. Fuzzy rough set techniques for uncertainty processing in a relational database. International Journal of Intelligence System, 2000, 15: 389~424
- 12 Beaubouef T, Petry F E. Information-theoretic measures of uncertainty for rough sets and rough relational databases. Information Science, 1998, 109: 185~195