

一种粗关系数据结构数据排序方法

魏玲玲¹, 邱桃荣²

(1 江西科技学院 计算机系, 江西 南昌 330098; 2 南昌大学 计算机系, 江西 南昌 330031)

摘 要: 目前的排序方法处理单个数字或字符串, 数据库数据项中的数据不再是原子值, 所以根据粗关系数据结构特点及等价类, 对传统的插入排序方法进行改进, 使其能运用于粗关系数据结构数据排序, 并将改进的插入排序方法应用到部分国家地域信息实例中, 从而证明了针对粗关系数据结构改进的插入排序方法的有效性和实用性。

关键词: 粗关系数据结构; 等价类; 关键字; 排序

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2013)04-0149-03

Data Ordination Approach of the Rough Relational Database

WEI Ling-ling¹, QIU Tao-rong²

(1 Department of Computer, Jiangxi BlueSky University, Nanchang 330098, China;

2 Department of Computer, Nanchang University, Nanchang 330031, China)

Abstract: The current sort methods are processing single number or string, and the date of data items is no longer an atomic value, then according to characteristic of the rough relational data structure and equivalence class, it is improved the traditional insertion sort method and to make use of the rough relational data structure data sorting, finally applied to a part of the national geographical information example, then proves the improved insertion sort is practical and effective for the rough relational data structure.

Key words: rough relational databases structure; equialvace class; keyword; sort

1 引言

随着计算机技术的发展, 其应用领域不断扩大且存储数据量的急剧膨胀, 出现了诸如不确定性、不完全、模糊等不规范信息。1993 年, T. Beaubouef 和 F. E. Petry 提出粗关系数据库模型(the rough relational database model, RRDm)。粗关系数据库模型是在传统关系数据库模型的基础上发展而来, 打破了传统关系数据库必须满足第一范式理论限制, 元组的属性值是一个由多个原子值组合而成的非原子值, 它以处理不确定信息为主。

排序是计算机内经常进行的一种操作, 其目的是将一组“无序”的记录序列调整为“有序”的记录序列。人们常在信息中查找某条信息, 要使查找操作快速, 就必须以某种合理的次序存储数据, 那就可以很

好的提高查找速度。目前的排序方法处理单个数字或字符串, 但随着数据量的增加, 数据项中的数据发生了变化, 不再是唯一确定值, 所以本文根据数据不确定性特点, 改进已有的插入排序方法, 使其能运用于粗关系数据库数据排序^[1]。

2 基本概念

2.1 粗关系数据结构

关系的描述称为关系模式, 将粗关系数据库模型形式化表示为三元组^[2-6]: $R(U, A, D_j)$, 其中 U 表示所有元组的集合, $U = \{u_1, \dots, u_i, \dots, u_n\}$ 是对象的全体, 为非空有限集; A 表示数据库的属性集, $A = \{A_1, \dots, A_i, \dots, A_m\}$ 是属性的全体, 也是非空有限集; $D_j (1 \leq j \leq |A|)$ 为属性 A_j 的值域, 它由若干个等价类构成的等价关系; 它的元组 t_i 采用 $(d_{ij},$

收稿日期: 2012-07-21; 修回日期: 2012-08-25

基金项目: 国家自然科学基金项目(61070139)

$d_{i2}, \dots, d_{in})$ 的形式, 其中 d_{ij} 是元组的一个属性值. 在传统关系数据库模型中 $d_{ij} \in D_j$, 而在 RRDB 中 $d_{ij} \subseteq D_j (d_{ij} \neq \emptyset)$, 元组 t_i 称粗元组. 在 D_j 上存在一个等价关系 θ_{A_j} , 构成 D_j 上的等价划分 $D_j = \{E_1, E_2, \dots, E_n\}$.

例如: 表 1 是部分国家地域信息, 它有两个主要属性: 国家及地域特征信息, 这两个属性对应的域、等价类及粗关系数据库表如下:

COUNTRY = {[us, usa, united-states], [belize, british-honduras], [cuba], [vieques], [mexico], [int, international]}.

FEATURE = {[swamp, bog, marsh, wetland], [beach, coast, sand], [road, airport, parking lot], [building, urban], [water, river, lake, sea, ocean], [grass, pasture, meadow], [cropland, farmland], [forest, woods, jungle, woodland]}.

表 1 地域信息

ID	COUNTRY	FEATURE
U123	us	marsh, lake
U124	us	marsh
U125	usa	marsh, pasture, river
U126	us	forest, river
U147	us	sand, road, urban
U157	us, mexico	sand, road
M007	mexico	sand, road
M008	mexico	beach
M009	mexico	sand
C039	belize	jungle
C040	belize, int	jungle, coast, sea

2.2 排序

排序整理文件中的记录, 使之按关键字递增或递减次序排列起来. 其确切定义^[7]:

输入: n 个记录 $R_1, R_2, \dots, R_n$, 其相应的关键字分别为 $K_1, K_2, \dots, K_n$.

输出: $R_{i1}, R_{i2}, \dots, R_{in}$, 使得 $K_{i1} \leq K_{i2} \leq \dots \leq K_{in}$ (或 $K_{i1} \geq K_{i2} \geq \dots \geq K_{in}$).

比较两个关键字的大小和改变指向记录的指针或移动记录本身是大多数排序算法的两个基本操作.

其中改变指向记录的指针或移动记录本身的实现依赖于待排序记录的存储方式. 常用存储方式有:

(1) 存储结构为顺序表.

(2) 存储结构为链表.

(3) 顺序的方式存储待排序的记录, 并建立一个辅助表, 也就是关键字和指向记录位置的指针组成

的索引表.

目前排序方法按所用策略不同, 可归纳为五类: 插入排序, 选择排序, 交换排序, 归并排序和分配排序. 本文提出的排序方法主要是对插入排序进一步调整, 使其适合粗关系数据库中数据排序.

3 粗关系数据结构数据排序

3.1 排序

已知一组升序排列数据 $a[1][1][1], \dots, a[1][1][p], a[2][1][1], \dots, a[n][m][p]$, 一组无序数据 $b[1][1][1], \dots, b[1][1][p], b[2][1][1], \dots, b[n][m][p]$, 需将二者合并成一个升序列.

首先比较 $b[1][1][1]$ 与 $a[1][1][1]$ 的值, 有以下情况:

若 $b[1][1][1]$ 等于 $a[1][1][1]$, 则比较 $b[1][1][2]$ 与 $a[1][1][2]$ 的值, 则若 $b[1][1][2]$ 大于 $a[1][1][2]$, 则跳过, 直到 $b[1][1][y]$ 小于 $a[1][1][y]$ 的值停止, 则将 $a[1] \sim a[n]$ 分别向后移动一位, 将 $b[1]$ 插入到原来 $a[1]$ 的位置这就完成了 $b[1]$ 的插入.

若 $b[1][1][1]$ 大于 $a[1][1][1]$, 则跳过, 比较 $b[1][1][1]$ 与 $a[2][1][1]$ 的值, 若 $b[1][1][1]$ 仍然大于 $a[2][1][1]$, 则继续跳过, 直到 $b[1][1][1]$ 小于 a 数组中某一数据 $a[x][1][1]$, 则将 $a[x] \sim a[n]$ 分别向后移动一位, 将 $b[1]$ 插入到原来 $a[x]$ 的位置, 这就完成了 $b[1]$ 的插入;

$b[2] \sim b[n]$ 用相同方法插入. 若无数组 a , 可将 $b[1]$ 当作 $n=1$ 的数组 a .

3.2 算法及分析

算法: 对 R 列表进行排序

输入: 无序列表 R ;

输出: 有序列表 R ;

int i, j ;

SeqList R ;

Sort(SeqList R)

{

int k ;

for($i=2; i \leq n; i++$)

if($R[i][1] = R[i-1][1]$)

do{

$k++$;

}while($R[i][k] < R[i-1][k] \mid k < p$)

if($k < p$)

continue;

else{

```
R[0]=R[i];
j=i-1;
do{
    R[j+1]=R[j];
    j--;
}while(R[0] < R[j]);
R[j+1]=R[0];
}
if(R[i][1] < R[i-1][1])
{
    R[0]=R[i];
    j=i-1;
    do{
        R[j+1]=R[j];
        j--;
    }while(R[0] < R[j]);
    R[j+1]=R[0];
}
```

本算法的优点是稳定、查询速度高;但比较次数不一定,比较次数越多,数据移动越多,特别是当数据总量庞大的时候,但用链表可以解决这个问题。

4 算例

以上列表 1 为例,使用上面的算法运行后,可得出表 2 结果。

5 结束语

本文主要介绍一种粗关系数据结构的排序方法,它根据粗关系数据结构的特点,适当调整了直接插入排序方法,使其能适应粗关系数据结构存储的数据排序,在排序时,消耗了一定的时间和空间,但提高了后期数据查询的速度。

参考文献:

[1] 武森,王静,谭一松. 考虑数据排序的改进 CABOSFV 聚类[J]. 计算机工程与应用,2011,47(34):127-129.

表 2 排序后地域信息

ID	COUNTRY	FEATURE
U124	us	marsh
U123	us	marsh, lake
U125	usa	marsh, pasture, river
M008	mexico	beach
M009	mexico	sand
U157	us, mexico	sand, road
M007	mexico	sand, road
U147	us	sand, road, urban
C040	belize, int	jungle, coast, sea
U126	us	forest, river
C039	belize	jungle

注: 排序时数据先后顺序按等价类的顺序

[2] 安秋生. 粗糙函数依赖的近似度量[J]. 计算机工程与应用, 2009, 45(1): 144-146.

[3] T Beaubouef, F E Petry, B. Buckles. Extension of the relational database and its algebra with rough set techniques[J]. Computational Intelligence, 1995, 11(2): 233-245.

[4] 安秋生,徐久成,沈钧毅,等. Rough 关系数据库模型及其关系操作[J]. 计算机科学, 2002, 29(7): 72-89.

[5] 王丹,吴孟达,刘银山. 粗糙关系数据库空间结构及其粗糙集模型[J]. 计算机工程与应用, 2005, 41(34): 163-167.

[6] 魏玲玲. 粗关系数据库查询模型及其应用[D]. 江西: 南昌大学, 2007.

[7] 黄刘生. 数据结构[M]. 北京: 经济科学出版社, 2000.

作者简介:

魏玲玲 女, (1981—), 硕士, 讲师. 研究方向为数据库技术、粗糙集.

邱桃荣 女, (1964—), 博士, 教授. 研究方向为数据库技术、粗糙集.