近似重复记录的增量式识别算法

许向阳　佘春红
(华中科技大学计算机学院数据库与多媒体技术研究所,武汉 430074)
E-mail:chunhong@public.wh.hb.cn

摘　要　数据清理是数据仓库中的一个重要研究内容,近似重复记录的识别是其中的一个技术难点。文章介绍了近邻排序方法,并以此为基础,研究了在数据模式与匹配规则不变的前提下,数据源动态增加时近似重复记录识别问题,提出了一种增量式算法 IMPN(Incremental Multi-Pass sorted-Neighborhood)。文章最后给出了实验结果。

关键词　数据清理　近似重复记录　增量式识别　特征记录

文章编号 1002-8331-(2003)12-0191-03　**文献标识码** A　**中图分类号** TP311.13

Incremental Algorithm for Detecting Approximately Duplicate Database Records

Xu Xiangyang　She Chunhong
(Dept.of Computer Science and Technology,Huazhong University of Science and Technology,Wuhan 430074)

Abstract: Data cleaning is an important area of data warehouse.Detecting approximately duplicate database records is one of technology difficulties.This paper introduces sorted-neighborhood method.Based on this idea,it studies the problem for detecting approximately duplicate records while receiving increments of data with no changes in data schema and matching rule-set,and presents an incremental algorithm for detecting the records.Finally,it gives out the experimental results.

Keywords: Data cleaning,Approximately duplicate records,Incremental detection,Representative record

1　引言

在建造数据仓库过程中，需要从各种数据源导入大量数据。这些数据中存在数据录入错误、同一对象在不同数据源中的表示各异等质量问题,使得应用于数据仓库前端的决策支持系统产生错误的分析结果而误导决策，影响信息服务的质量。因此数据仓库构建中的一个重要的任务是通过数据清理,将数据转换为一致的形式,保证数据的正确性。

数据清理主要涉及到数据的映射(mapping)、匹配(matching)和合并(merging/purging)。通过映射,将数据格式标准化;通过匹配,发现重复的对象;通过合并,保留或生成一个完整的对象。数据清理活动的核心是近似重复对象的识别。所谓近似重复对象是指表现形式不同但语义上相同的对象[1],从狭义的角度来看，如果两条记录在某些字段上的值相等或足够相似,则认为这两条记录互为近似重复,该文仅考虑狭义的近似重复记录,并简称为重复记录。

识别重复记录的直观方法是采取排序-合并方式，将每一条记录与数据库中其他记录进行比较,该方法的识别精度相当高,但在大数据量的情况下,其处理时间难以忍受。Hernandez等人提出的多趟近邻排序方法[2~4](Multi-Pass sorted-Neighborhood,MPN)和 Monge 提出的优先队列策略(Priority Queue Strategy,PQS)在不影响识别精度的情况下,对记录比较和排序进行改进,提高了响应时间,更适于识别重复记录。

上述重复记录识别算法均以一个输入数据集为前提。如果系统接受到一个新的数据集,必须将其与早期已处理的数据集进行拼接,再对拼接后的整个数据集进行处理。该文主要考虑在数据模式与匹配规则不变的前提下,输入数据集动态增加时重复记录识别问题，提出了一种增量式重复记录识别算法,在识别精度基本保持不变的情况下,该算法大大节省了执行时间与空间的开销。

文章其余部分的结构如下:在下一节将描述基本的邻近排序和多趟邻近排序方法;第 3 节提出增量式算法,并介绍两种预处理策略;第 4 节给出对比实验及结果;第 5 节对全文进行总结。

2　近邻排序方法

近邻排序算法首先对数据集进行排序,然后根据排序的顺序,对范围相对较小的邻近记录应用一些具体的匹配规则进行比较,找出重复记录。基于这样一个观察:一个简单的关键字在

基金项目:国家科技攻关计划项目"科技部科技电子政务系统关键技术及应用系统的研究"(编号:2001BA110B01)
作者简介:许向阳,副教授,主要研究领域为数据库,数据仓库技术。佘春红,硕士研究生,主要研究领域为数据仓库技术。
?1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

排序后不能完全将重复记录聚集在一起,因此一趟近邻排序算法可能遗漏一些重复记录。为避免这种情况,实行多趟邻近排序算法,每次采用一个不同的关键字进行排序。下面加以具体介绍。

2.1 基本近邻排序算法(Sorted-Neighborhood Method,SNM)

给定一个或多个关系表,首先将它们拼接成一个含 N 条记录的数据集,然后采用 SNM 方法。SNM 方法可总结为以下三步:

(1)创建关键字:抽取相关的字段,构造关键字。关键字的选择需要考虑应用的背景,SNM 执行的精度与关键字的抽取密切相关。

(2)排序:用第一步产生的关键字对数据集进行排序。

(3)合并:在排序的数据集上滑动固定大小的窗口,数据集中每条记录仅与窗口内的记录进行比较。如果窗口的大小是 w 条记录,则每条新进入窗口的记录与窗口内先前 $w-1$ 条记录进行匹配比较,最先进入窗口内的记录滑出窗外,如图 1 所示。

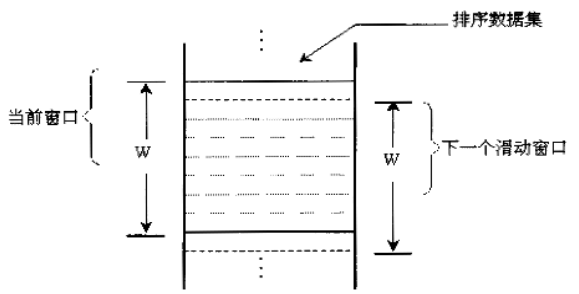

图 1 滑动窗口扫描排序数据集示意图

2.2 多趟近邻排序算法(Multi-Pass Sorted-Neighborhood,MPN)

SNM 方法识别重复记录的精度很大程度上依赖于排序所选择的关键字。在数据清理中,一个关键字不足以将所有重复记录聚集在一起。如果记录中充当或部分充当关键字的字段出现错误,那么该记录很少有机会获得成功的重复记录匹配。例如模式为(IdCard,Name,Age,Sex,Address)的两条记录,一条记录的 IdCard 值是 422400720213002,另一条记录的 IdCard 值是 242400720213002(最左边的两位数位置颠倒),若选择 IdCard 作为关键字,排序后这两条记录物理位置相距较远,不会同时位于较小的滑动窗口内,因此不能被识别成重复记录。为解决这个问题,可以独立地执行多趟 SNM 算法,每趟使用不同的关键字和相对较小的窗口。最后合并每趟扫描产生的重复记录。在合并时假定记录的重复具有传递性,即若记录 $R1$ 与 $R2$ 互为重复记录,$R2$ 与 $R3$ 互为重复记录,则 $R1$ 与 $R3$ 互为重复记录。图 2 描述了一个两趟式的近邻排序扫描方法。

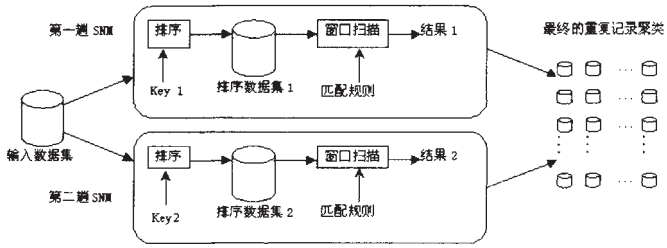

图 2 多趟邻近扫描算法(以两趟为例)

3 增量式重复记录识别算法

2.2 节描述的 MPN 方法是以一个输入数据集为前提的。一旦部分数据集通过重复记录识别处理,分割成一些相似记录聚类后,若再获得模式相同的新的数据集,则必须在执行 MPN 算法之前重新将所有的数据集拼接。MPN 方法的关键是数据排序,并在整个排序的数据集上进行窗口扫描。由于完成近邻排序扫描的时间正比于输入数据集的大小,将新的数据与已处理的数据进行拼接,将导致数据集的急剧增长,大大增加总的执行时间,这在时间和空间上都是不可取的;而且当数据集增加时,以前的一些重复记录聚类可能消失,这是因为排序后新增加的记录可能插在那些以前位于同一窗口内的相似记录之间,使得原来两条相似记录物理位置相距很远,而不能被识别。

在对拼接的数据重新进行处理时,大部分时间花在对已经产生的聚类的重新计算上。如果用于匹配记录的规则不变,观察到仅最近到达的增量数据可以改变当前的聚类,这个结论是增量式重复记录识别处理的核心。对于每条新增加的记录有以下两种处理:第一,加入一个已存在的聚类;第二,创建一个新的聚类。由此,笔者提出增量式 MPN 方法(Incremental MPN,IMPN)。

3.1 算法描述

IMPN 采用 MPN 方法聚类输入数据。IMPN 和 MPN 的最大区别是前者在多趟近邻扫描前需要进行预处理。在第一次进行重复记录识别处理时,利用 MPN 方法聚类数据,然后,每当数据增量到达时,就从上一次产生的每个聚类中选出一些记录,这些记录表示它们所在聚类的特征信息,称为特征记录或"聚类重心"。这些特征记录与增量记录进行拼接,再利用 MPN 方法进行处理。最终结果是为每条增量记录指定一个聚类,这些新的聚类和老的聚类合并在一起,构成最终识别结果,图 3 描述了这一过程。

图 4 给出了 IMPN 的算法描述。该算法由一个循环组成,每次循环系统接受一个增量数据集,增量数据集与特征记录拼

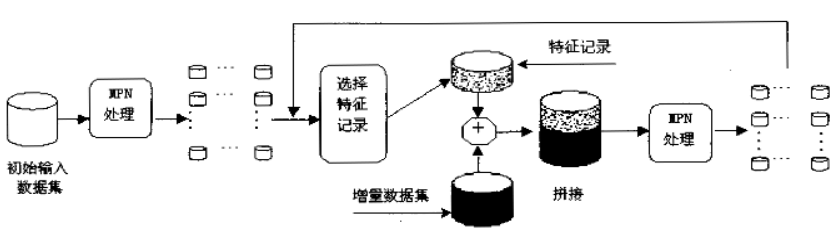

图 3 增量式 MPN 示意图

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

接,这些特征记录从前一次增量式重复记录识别处理的结果获得,然后采用 MPN 方法进行处理。

```
定义:
    I_i: 第 i 个增量数据集
    C_i: 拼接后所产生的数据集
    P_i: 各个聚类的特征记录所构成的集合
初始化:
    i←0
    I_0←第 1 个增量数据集
    C_0←NULL
    P_0←NULL
FOR 每个增量 I_i DO
    1. C_i←Concatenate(P_i, I_i)
    2. 对 C_i采用重复记录识别方法(如 MPN 算法)进行处理, C_i中的
       每条记录均划分到一个聚类中;
    3. P_i+1←NULL;
    4. 对 C_i中的每个聚类 k,按一定的策略选择若干条记录 r_k作为该聚类
       的特征记录, P_i+1←P_i+1∪r_k
ENDFOR
```

图 4　增量式重复记录识别算法

初始条件下,特征记录集为空,第一个数据集增量就是第一个输入数据集,拼接所产生的数据集等同于第一个输入数据集增量,当 MPN 算法执行后,增量数据集中的每条记录划分到重复记录聚类中,算法第一次使用时,所有的记录加入新的聚类,第二次执行算法时,记录要么添加到已存在的聚类中,要么作为一个新的聚类。对于那些输入数据集非常巨大以至内存容纳不下的情况,这里将大数据集分割成许多小的数据片段,每个数据片段作为一个输入数据集,然后利用上述增量式方法进行重复记录识别处理。

在增量式算法执行过程中,假设匹配规则集不发生变化。如果数据过于互相依赖以及规则集更改幅度过大,产生了过多的错误聚类,则要将所得到的全部数据重新执行非增量式重复记录识别算法。根据不同的应用背景,规则集的微小变化和精度上的轻微差异也许可以接受,为避免整个记录识别过程的重新运行,在所实现的系统中,决定由系统与用户的交互解决这一问题。

3.2　特征记录的选取

IMPN 算法中一个关键点就是正确选取每个已生成聚类的特征记录。一般而言,每个聚类需要一个或多个特征记录表示它的特征。特征记录的确定需要以专业知识为指导,不同的应用领域有不同的标准。基本策略包括以下几种:

(1)随机抽样策略:从每个聚类中随机选择记录作为特征记录。

(2)最近 N 元素策略:记录经常按进入关系的时间顺序物理上进行排序,在这种情况下,最近存入关系的记录被认为最能够代表聚类特征,基于这种策略,最近的 N 条记录当作特征记录。

(3)归纳策略:从聚类所表示的记录中归纳数据,产生特征记录,归纳概念的技术称为机器学习。

(4)语法策略:选择最长的较完整的记录作为特征记录。

(5)'使用'策略:选择与聚类中其它记录匹配最频繁的记录作为特征记录。

(2)、(4)是两种较直观的方法,(3)则与人工智能有关,下面介绍(1)、(5)两种方法。

3.2.1　随机抽样策略

给定一个采样率常数 α 以及由通过上述算法产生的聚类集合,得到 $\alpha|c_i|$ 条记录的一个抽样,每个聚类必须至少选择一条记录),$|c_i|$是第 i 个聚类的大小。这里还假设每条记录有这样两个属性:RecordID-唯一地标识每条记录,ClusterID-表示每条记录所属的聚类。

最简单的抽样策略首先根据 ClusterID 排序,将数据集分组,然后从各个分组中,选择一条特征记录。这种方法的缺点是在获得特征记录之前需要对记录分组。为了避免分组,设记录的 ClusterID 来源于所在聚类中记录的 RecordID(如一个聚类包含记录{5,9,11,21},这些记录的 ClusterID 将是 5,9,11,21 四个数中的某一个数),以下是这种选择特征记录的无排序随机抽样方法:

```
FOR 聚类中的每条记录 r DO
    IF r.RecordID==r.ClusterID THEN
        选择 r 作为聚类 ClusterID 特征记录
    ELSE
        IF random()<α THEN
            选择 r 作为聚类 ClusterID 特征记录
        ENDIF
    ENDIF
ENDFOR
```

注:random()为随机抽样函数

3.2.2　'使用'策略

将记录的'使用'定义为在同一个聚类中,该记录与其他记录匹配的次数。这种选择特征记录的基本思想是选择那些在聚类形成过程中'使用'次数最多的记录。

要想对每条记录的'使用'计数,需要对 MPN 算法处理进行一些修改。首先对每条记录增加一个 usage 字段,对'使用'加以记数,在窗口扫描阶段,若两条记录匹配成功,增加它们 usage 字段的值。将每趟的 usage 相加,得到记录 r 最终的 usage 值。设 M 为趟数,r.usagek 为记录 r 在第 k 趟($<=k<=M$)的'使用'计数。这里用以下伪代码描述每条记录最终的 usage值:

```
FOR 所有的记录 r DO
    r.usage=0;
    FOR j=1 TO M DO
        r.usage=r.usage+r.usagej;
    ENDFOR
ENDFOR
```

每条记录通过上述增量式算法的处理,都有一个 ClusterID 和一个 usage。为选择特征记录,还要做以下工作:

(1)用组合关键字 ClusterID+usage 对数据集进行递减排序,排序的结果将数据集分成若干个聚类,每个聚类中 usage 值最大的即'使用'最频繁的记录会最先出现。

(2)根据排序的结果,对每个 ClusterID,选择第一条记录作为特征记录(也可选最近的 N 条记录或将 usage 作为权重,根据权重来抽取记录)。

4　性能分析

对于文章所考虑的增量式重复记录识别问题,一种直接的办法就是将所有增量数据集全部拼接重新运行一遍 MPN 算法。现在就 IMPN 算法与重新运行一遍 MPN 算法进行比较。

在 Pentium Ⅲ微机上,利用 VC6.0 作为开发工具,以文件的形式提供输入数据集,采用'使用'策略选取特征记录,进行了

(下转 220 页)

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

表 2

字段名	数据类型	长度	可空	说明
DEVICE_ID	int	4		设备 ID,该字段唯一确定一个 GSM 车载台设备
DEVICE_TYPE	tinyint	1		设备的类型,对应 DEVICE_TYPE_INFO 表中的 DEVICETYPE_ID 字段
DEVICE _SIMNO	nvarchar	50		设备的 SIM 卡号码,130 开头的为联通,139 开头的为移动
DEVICE _CARID	int	4	√	设备所在汽车的 ID,即 CAR_INFO 表中相应的 CAR_ID 字段
DEVICE _MODE	tinyint	1		设备通信方式
DEVICE _STATE	tinyint	1		设备状态,值为 1 时表示该设备可以正常使用
DEVICE _VERSION	nvarchar	50		设备版本号
DEVICE _DRYPT	tinyint	1		该设备是否要求信息加密,1 为是,0 为否
DEVICE _USER	nvarchar	50	√	设备用户名称
DEVICE_PASSWD	nvarchar	50	√	设备用户密码

表 3

字段名	数据类型	长度	可空	说明
ALARM_TYPE	int	4		报警求助消息的类型,即消息协议中的 MsgID 字段
ALARM	nvarchar	50	√	报警求助消息的内容,格式为 X,Y 报警点的坐标)
ALARM_DEVICE	int	4		发出报警求助消息的设备的 ID 号,即消息协议中的 SourceID 字段
ALARM_RESULT	tinyint	1	√	报警求助消息的处理结果,1 为成功,0 为失败
ALARM_MEMO	nvarchar	50	√	报警求助消息附带的注释,在实际应用中未实现
ALARM_DATETIME	Datetime[1]	8		报警求助消息的时间
ALARM_DES	int	4	√	接受报警求助消息的监控台的 ID 号,即消息协议中的 DestID 字段
ALARM_COUNTER	int	4	√	消息计数器,即消息协议中的 MsgCounter 字段

前景会越来越广,该系统设计是 GPS 系统二次开发的通用监控平台,具有较大的实用意义。 收稿日期:2002 年 7 月)

参考文献

1.柳开洋,韩道范,马爱民.Web Browser/sever 方式的 GPS 车辆管理系统[J].计算机工程与应用,2001;37(4):127~128

2.单广玉.WAP 安全性分析[J].计算机安全,2001;(1)

3.官章全等.Visual C++ 6.0 编程实例详解[M].电子工业出版社,2001

(上接 193 页)

模拟实验。首先,将 1 万条输入记录集平均划分为 1,2,3,4 个数据增量,比较增量式与基本重复记录识别处理所需要的时间开销,结果见图 5。虽然增量式处理在选取特征记录上需要额外的时间,但从图中看到,与非增量式算法重新计算重复记录聚类相比,总的时间开销要小得多。

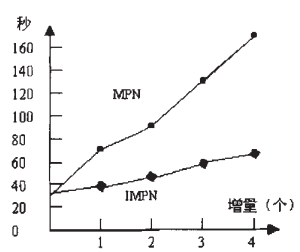

图 5 时间对比实验

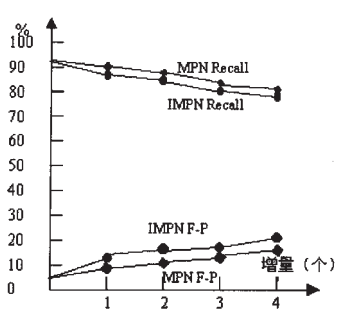

图 6 精度对比实验

对两种算法的识别精度也进行了比较实验。主要从两方面着手,一是召回率 Recall=系统正确识别的重复记录数/实际包含的重复记录数,二是误识别率 F-P(False-Positive)=系统错误识别的重复记录数/系统总共识别的重复记录数。仍以上述增量数据集为例,实验结果如图 6 所示。从图中可以看到增量式算法的精度比非增量式算法的精度稍低,这是因为在增量式算法使用特征记录计算新的聚类,而非增量式算法使用整个记录集计算聚类。

5 结束语

数据清理转换是数据仓库中的一个重要研究领域,重复记录的识别是数据清理中的一个技术难点。文章在介绍基本的多趟近邻排序扫描算法的基础上,提出了增量式重复记录识别算法 IMPN,这种增量式的算法在基本不损失精度的情况下,能够对新增加的数据集进行快速的重复记录识别。笔者还对算法中特征记录的选取策略进行了研究。当然,将有关的数据清理方法集成为一个完整的数据清理子系统,还有待进一步研究。

收稿日期:2002 年 6 月)

参考文献

1.Erhard R,H Hai Do.Data Cleaning:Problem and Current Approaches.IEEE Techn Bulletin Data Engineering,2000

2.M Hern′andez,S Stolfo.The merge/purge problem for large databases[C].In:Proceedings of the ACM SIGMOD,International Conference on Management of Data,1995:127~138

3.M Herntindez.A Generalization of Band-Joins and the Merge/Purge Problem[R].Technical Report CUCS-005-1995,Department of Computer Science,Columbia University,1995

4.M Hern′andez,S Stolfo.Real-world data is dirty:Data Cleansing and the Merge/Purge problem[J].Journal of Data Mining and Knowledge Discovery,1998;2(1):9~37

.1994-2017 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net