文章编号:1001-9081(2013)08-2208-04 doi:10.11772/j.issn.1001-9081.2013.08.2208

海量数据的相似重复记录检测算法

周典瑞*,周莲英
(江苏大学 计算机科学与通信工程学院,江苏 镇江 212013)
(* 通信作者电子邮箱 zdianrui@126.com)

摘 要: 针对海量数据下相似重复记录检测算法的低查准率和低效率问题,采用综合加权法和基于字符串长度过滤法对数据集进行相似重复检测。综合加权法通过结合用户经验和数理统计法计算各属性的权重。基于字符串长度过滤法在相似检测过程中利用字符串间的长度差异提前结束编辑距离算法的计算,减少待匹配的记录数。实验结果表明,通过综合加权法计算的权重向量更加全面、准确反映出各属性的重要性,基于字符串的长度过滤法减少了记录间的比对时间,能够有效地解决海量数据的相似重复记录检测问题。

关键词: 海量数据;相似重复记录;综合加权法;编辑距离
中图分类号: TP311 **文献标志码:** A

Algorithm for detecting approximate duplicate records in massive data

ZHOU Dianrui*, ZHOU Lianying
(*School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang Jiangsu* 212013, *China*)

Abstract: For the problem of low precision and low time efficiency of approximate duplicate records detection algorithm in massive data, integrated weighted method and filtration method based on the length of strings were adopted to do the approximate duplicate records detection in dataset. Integrated weighted method integrated user experience and mathematical statistics to calculate the weight of each attribute to make weight calculation more scientific. The filtration method based on the length of strings made use of the length difference between strings to terminate the edit distance algorithm earlier which reduced the number of the records to be matched during the detection process. The experimental results show that the weight vector calculated by the integrated weighted method makes the importance of each field more comprehensive and accurate. The filtration method based on the length of strings reduces the comparison time among records and effectively solves the problem of the detection of approximate duplicate records under massive data.

Key words: massive data; approximate duplicate record; integrated weighted method; edit distance

0 引言

高质量的数据是保证企业发展的重要前提,因此为了满足业务的需求,需要整合不同的数据源,由于整合的过程会产生一些语法上相同或相似并且代表同一现实实体的相似重复记录,这样会直接影响数据的质量,因此相似重复记录的清除成为提高数据质量的重要步骤。

记录间的相似性检测实质上是表征记录的各属性的相似性检测,由于各个属性对于记录之间的差异性贡献不同,应根据属性的重要程度为各个属性赋予相应的权重,以提高检测的精度。海量数据下的数据检测需要使用大量的时间和资源。

为了检测相似重复记录,目前采用的方法主要有:基本的字段匹配算法[1]、递归的字段匹配算法[2]、基于“排序 & 合并”方法[3]、采用距离函数模型的方法[4]、基于 q-gram 算法[5]、基于聚类的算法[6]和基于人工智能的算法[7]等。传统的方法在进行相似重复记录检测时,需进行大量的磁盘 I/O 操作,这将导致时间和空间复杂度很高,基于聚类的算法计算量较大,准确率较低;而基于人工智能的算法推理过程复杂。李星毅等[8]为了提高查全率和检测的精度,采取多趟检测的技术和主观的等级赋权法,增加了大量的检测时间。传统方法多采用滑动窗口保存重复记录集,窗口的大小指定不合理,导致有些相似重复记录无法正确检测,降低了查全率。

为了克服以上缺陷,本文对传统的检测方法进行改进,首先采用既考虑主观方面的用户经验又结合客观方面的数理统计方法的综合加权法计算各属性的权重,然后把各属性值作为关键字,利用多线程对数据集并行排序,最后在各线程中采用文献[9]中的优先队列技术依次检测各记录,检测过程中采用基于字符串长度过滤法(加速法)减少不必要记录对之间的比对时间,最终合并检测结果集。

1 相关定义

设数据集合 $R=\{R_1,R_2,\cdots,R_n\}$,属性向量 $A=(A_1,A_2,\cdots,A_k)$,A_k 表示数据表第 k 个属性,对于任意记录 $R_i=(R_{i1},R_{i2},\cdots,R_{ik})$ $(1\leqslant i\leqslant n)$,$X_{ij}$ 表示记录 R_i 第 $j(1\leqslant j\leqslant k)$ 个属性的值,用 w_k 来代表属性 A_k 的权值,表示属性对记录的重要程度,称为属性的权重,权重向量 $W=\{W_1,W_2,\cdots,W_k\}$。

定义 1 $G_{st}(1\leqslant s\leqslant m,1\leqslant t\leqslant k)$ 是第 s 个操作用户根据个人的经验为属性 A_t 所指定的等级(从 1 开始,使用连续正整数表示等级,1 表示最高等级,数值越大,等级越低),G_t 表示第 t 个属性的最终统一等级,采用均值法计算出每个属性的最终统一等级。

$$G_i=\frac{1}{m}\cdot\sum_{s=1}^{m}G_{si} \tag{1}$$

收稿日期:2013-02-25;修回日期:2013-04-06。 基金项目:江苏省科技支撑项目(BE2011156)。
作者简介:周典瑞(1987-),男,山东泰安人,硕士研究生,主要研究方向:数据清洗; 周莲英(1964-),女,江苏泰州人,教授,博士,主要研究方向:计算机网络性能分析、信息安全、电子商务、网络信息系统。

其中: $G_i(1 \leqslant i \leqslant k)$ 表示第 i 个属性的最终统一等级，m 表示用户的个数。

定义 2　属性的取值若是各不相同，就会很容易区别记录的相似性。客观上采用随机统计法(多次随机选取一定数目的记录) 计算每一属性取值的变化种数，作为客观描述属性对记录的重要性。$C_{ij}(1 \leqslant j \leqslant k)$ 表示第 i 次第 j 个属性的取值种类数目，C_j 表示第 j 个属性的最终种类数，使用均值法计算出每个属性的最终取值种类数。

$$C_j = \frac{1}{m} \cdot \sum_{i=1}^{m} C_{ij} \tag{2}$$

其中 m 表示选取的次数。

定义 3　根据以上两个定义得到的主观等级向量 $\boldsymbol{G} = (G_1, G_2, \cdots, G_k)$ 和客观属性取值种类数向量 $\boldsymbol{C} = (C_1, C_2, \cdots, C_k)$，计算属性权重向量 $\boldsymbol{W} = (W_1, W_2, \cdots, W_k)$。

$$W_i = \frac{1}{2}\left(\frac{C_i}{\sum_{i=1}^{k} C_i} + \frac{G_i}{\sum_{i=1}^{k} G_i}\right) \tag{3}$$

定义 4　两个字符串 x 和 y 之间的编辑距离 $d(x,y)$ 定义为: 将 x 转换成 y 所需的最少编辑操作次数，其中编辑操作包括字符替换、插入字符和删除字符。由于每个编辑操作有不同的操作代价即消耗不同的操作时间，从一个字符串转换成另一个字符串需要多个编辑操作组成一个编辑操作序列，因此编辑距离是最小的操作序列的代价之和。其形式化定义为: 假设 B 是一个有限的符号字母表，B^* 是 B 上所有的字符串集合，ε 表示空符号，$|x|$ 表示字符串 x 的长度，$|\varepsilon| = 0$。所有的编辑操作: $a \to b$，$a \to \varepsilon$，$\varepsilon \to a$ 分别表示替换、删除和插入操作。如果 a、b 相同的话，则 $a \to b$ 为同义替换，不需要花费时间; 否则为非同义替换，花费替换的时间。每个操作都要消耗一定的操作时间，考虑使用时间表示所需要的操作代价，因此，$T(a \to b)$ 表示替换代价，$T(a \to \varepsilon)$ 表示删除代价，$T(\varepsilon \to a)$ 表示插入代价，如果 $S = E_1, E_2, \cdots, E_n$ 表示字符串转换的一个编辑操作序列，那么将字符串 x 转换成 y 的代价为: $T(S) = \sum_{i=1}^{n} T(E_i)$，因此 x、y 的编辑距离转化为两字符串转换的最小操作代价为 $\min(T(S))$。根据以上描述替换的代价为 rt(若为同义替换 $rt = 0$，否则为 rt)，插入操作代价为 it，删除操作的代价为 dt，其中 t 表示操作时间的基本单位。因此，编辑距离 $d(x,y)$ 的计算公式为:

$$d(x,y) = \min\begin{cases} d(x_{i-1}, y_j) + dt \\ d(x_i, y_{j-1}) + it \\ d(x_{i-1}, y_{j-1}) + rt \end{cases} \tag{4}$$

其中: $d(0,0) = 0$，$d(i,0) = i$，$d(0,j) = j$，$1 \leqslant i \leqslant n$，$1 \leqslant j \leqslant m$。

定义 5　对任意记录 X_i 与 X_j，它们的第 t 维属性分别为 X_{it} 与 X_{jt}，则 X_{it} 与 X_{jt} 的属性相似度[10] 如下:

$$sF(X_{it}, X_{jt}) = \frac{1}{|X_{it}|}\sum_{k=1}^{m} \max(score(a_k, X_{jt})) \tag{5}$$

其中: $score(a_k, X_{jt})$ 表示 X_{it} 中的原子串 a_k 与 X_{jk} 中的每个原子串匹配的分值，$0 \leqslant score(a_k, X_{jt}) \leqslant 1$，$|X_{it}|$ 表示 X_{it} 的长度，m 表示 X_{ik} 中原子串的个数。

定义 6　任意记录 X_i 与 X_j，则 X_i 与 X_j 的相似度为:

$$sR(x_i, x_j) = \sum_{t=1}^{k} sF(x_{it}, x_{jt}) * W_t \tag{6}$$

2　算法设计

2.1　综合加权方法研究与设计

记录之间的相似性检测，实质上是属性的相似性检测。由于各个属性对于记录之间的差异性贡献不同，因此应该根据属性的重要程度，为各个属性赋予相应的权重，提高检测的精度。组合赋权的基本思想为: 结合主观的用户经验和客观的数理统计方法，全面考虑属性的重要程度，生成合理的权重向量。

主观方面: 李星毅等[8] 采用等级法确定等级向量，即首先各用户根据相关经验为各个属性指定一定等级; 然后根据均值法计算属性的最终统一等级，生成如表 1 所示的属性等级表; 最后，根据统一等级生成等级向量。

表 1　属性等级表

用户	A_1	A_2	…	A_k
U_1	G_{11}	G_{12}	…	G_{1k}
U_2	G_{21}	G_{22}	…	G_{2k}
⋮	⋮	⋮		⋮
U_m	G_{m1}	G_{m2}	…	G_{mk}
最终统一等级	G_1	G_2	…	G_k

A_k 表示第 k 个属性，U_m 表示第 m 个用户，G_{ij} 是第 i 个用户为第 j 个属性所指定的等级，G_j 是最终统一等级，等级向量 $\boldsymbol{G} = (G_1, G_2, \cdots, G_k)$。

客观方面: 采用随机统计法(多次随机选取一定数目的记录) 计算每一属性取值的变化种数，作为客观描述属性对记录的重要性。表 2 为属性取值种类数统计表。

表 2　属性取值种类数统计表

操作	A_1	A_2	…	A_k
T_1	C_{11}	C_{12}	…	C_{1k}
T_2	C_{21}	C_{22}	…	C_{2k}
⋮	⋮	⋮		⋮
T_n	C_{n1}	C_{n2}	…	C_{nk}
最终种类数	C_1	C_2	…	C_k

A_k 表示第 k 个属性，T_n 表示第 n 次操作，$C_{ij}(1 \leqslant j \leqslant k)$ 表示第 i 次第 j 个属性的取值种类数目，C_j 表示第 j 个属性的最终种类数，使用均值法计算出每个属性的最终种类数，最终生成属性取值种类数向量 $\boldsymbol{C} = (C_1, C_2, \cdots, C_k)$。

最后，根据定义 3 计算得到属性的权重向量 $\boldsymbol{W}$。

2.2　多线程并发应用

海量数据的检测，必须考虑检测时间，如果顺序地对记录集的每一个属性进行排序和相似重复记录检测，势必会浪费大量的时间。本文考虑到共享加载到内存的数据集，采用多线程并发执行相似重复记录检测算法。这样，不仅充分利用计算机资源，同时又能大幅度提高算法的执行效率和相似重复记录的查全率。当所有的数据加载到内存后，如果按照多轮次检测算法，即第一次按照权重最高的属性排序，然后进行检测; 如果效果不好，再按照权重次之的属性排序，接着再进行检测; 反复进行操作，势必浪费时间。采用多线程的相似检测算法就是按照属性个数开启有限个线程，每个线程中按照属性进行排序，然后执行相同的检测算法，这样既可以减少多轮次检测的时间，又可以保证较高的查全率。

2.3　加速法

从前面的分析可以看出: 在相似重复记录检测过程中，记录间的相似检测是一个重要问题，其关键步骤是记录中各字

段的相似检测，其效率将直接影响整个算法的效率，记录中大多字段采用编辑距离算法来检测，而由于编辑距离算法的复杂度为 $O(m \times n)$，若数据量很大，如不采用一种高效的过滤方法来减少不必要的编辑距离计算，将会导致相似检测时间过长。因此，为了提高相似重复记录的检测效率，本文提出了一种优化相似重复记录检测效率的方法，该方法采用加速法减少不必要的编辑距离计算。根据定义4可以计算出任意两个字符串 x,y（长度为 $|x|,|y|$）的编辑距离一定不比两字符串长度之差小，即 $d(|x|,|y|) \geqslant ||x| - |y||$。如果设定两字段的相似度阈值为 β，则两个字符串的长度之差最多不能超过 $\max(|x|,|y|) * (1-\beta)$。如果能够利用上式对所要比较的记录属性进行基于字符串长度的过滤，则可以大大减少不必要的编辑距离计算时间，从而提高相似重复记录的检测效率。

2.4 优先队列

传统的记录相似检测采用固定窗口大小的滑动窗口顺序扫描记录集，比较当前记录与滑动窗口中的记录之间的相似性，这样大大减少了记录的比对次数，提高了比对的时间效率。但是由于窗口的大小固定，势必引起漏查或者重复比对的问题。同时，窗口的记录只是单一的一条记录，不能代表某类重复记录的所有特征，同样也会导致漏查现象的存在。针对以上缺陷，采用使用重复记录聚类思想的优先队列代替固定窗口大小的滑动窗口。

使用优先队列对排序后的记录集进行相似记录检测的思想：假如当前记录为 R_i，优先队列的第一个聚类项代表为 R_j，通过计算两者的相似性判断当前记录是否属于该聚类项，如果不相似，则直接与优先队列中的下一聚类项代表记录 R_{j+1} 进行比较，直到优先队列的最后一个聚类项代表；如果一直都不相似，则把记录 R_i 作为一个新的聚类项代表添加到优先队列的头部；如果此时队列中的记录总数超过优先队列上限，则根据“最久未使用”原则，删除优先队列中末尾聚类项。如果 R_i 与 R_j 相似，说明该记录具有代表性，应该把该记录添加到当前以 R_j 为代表的聚类项中，并且将 R_i 作为新聚类项的聚类代表记录，同时把 R_i 添加到重复记录集中，然后对 R_i 之后的记录 R_{i+1} 继续进行相似检测。每次记录进行比对时，对比较过的聚类项元素进行标记，如果长时间某些聚类项没有被比较过，说明此聚类项在以后被比较的概率也比较小。因此有必要从优先队列中删除此类聚类项，减少优先队列中的记录，从而减少下一次的比对次数，实现了优先队列的自适应性元素删除功能。

通过使用优先队列对记录集进行相似重复检测，大大减少记录的比对次数；同时采用聚类思想，避免了因为单条记录不能代表多条重复记录而漏查的现象。该方式不受数据规模的限制，特别适合海量数据的相似检测。

2.5 算法流程

首先计算属性的权重，确定每一属性对于记录相似性检测的重要性；然后，多线程并发检测记录集，每个线程针对一个属性对记录集进行排序；最后在每个线程中检测相似重复记录并且合并所有的检测结果。属性相似检测的核心是字符相似度计算，字符相似度的计算通常采用编辑距离算法，由于编辑距离必然大于两字符的长度之差，为了提高检测效率，采用基于字符串长度的过滤策略，即因两字符长度相差较大而相似性较小，省略计算其编辑距离的策略。为了提高检测精度，利用综合加权法计算各属性的权值；并通过多线程并发执行检测算法，有效地提高查全率。

算法描述如下：

步骤1 用户根据属性实际重要程度给各属性指定等级，根据均值法计算各属性的最终统一等级；在各个属性中随机选取一定数目的记录计算其属性取值的种类数，生成属性取值种类数向量，最终把属性等级向量和属性取值种类数向量进行统一化处理，生成属性的权重向量。

步骤2 根据属性的个数创建多个线程。

步骤3 在每个线程中，按照属性值进行排序。采用优先队列顺序扫描记录集，计算当前记录与队列中记录的相似度；然后根据相似阈值判断记录是否相似，并添加到重复记录集中。

步骤4 合并各个线程中的重复记录集。

3 实验分析

3.1 实验设计

实验环境：Intel I3 370 2.40 GHz CPU，物理内存2 GB，硬盘空间320 GB，操作系统Windows 7，数据库软件为Oracle11g，编程语言为Java语言。实验数据来源于镇江市市民信息的采集数据，包括社保的数据、部分试点事业单位的采集数据、财政局的数据等，由于来源广泛、职业的变换导致采集到的数据必然存在大量的重复。度量相似检测算法有效性的三个主要标准包括查全率、查准率和运行时间。为了检验论文中检测算法的有效性，设计以下实验。

文献[8]提出的等级分组方法是一种比较优秀的相似重复记录识别算法，该算法首先根据等级法确定属性的权重，然后选择关键字对数据集进行聚类，最后在各个小的数据集中检测相似重复记录，为了避免漏查，采用多趟查找技术。该算法设计简单，时间复杂度小，检测精度较高。因此，选择等级分组方法作为本文所采用方法的参照。为了便于处理，等级分组方法称为RGM，本文的算法称为IWM。分别从数据源中提取四组数据，对两种算法进行比较，四组数据量分别为53.4万、98.1万、126.2万和153.7万，通过软件和人工等方式对上数据分别处理，使之分别包含0.46万、0.85万、1.31万和1.44万条相似重复记录。

3.2 结果分析

设 I_O 代表原数据集实际的重复记录集合，I_D 代表识别出的重复记录集合。查准率（*precision*）表示为正确识别出来的重复记录占识别出的重复记录的比率，查全率（*recall*）表示为正确识别出的重复记录占数据集中实际的重复记录比率。

$$precision = |I_O \cap I_D| / |I_D|$$

$$recall = |I_O \cap I_D| / |I_O|$$

3.2.1 查准率对比

RGM与IWM的查准率比较结果如表3所示。从表3中可以看出，随着数据量的增加，RGM的查准率下降速度比IWM快，这主要是因为RGM采用比较主观的等级法给属性赋值，没有考虑客观因素，权值的分配不是十分合理；而IWM根据综合加权法综合考虑主、客观方面的因素，使得权重分配更加合理，更能反映现实实体特征，从而提高了查准率。

表3 两种算法的查准率比较

数据量（万）	查准率/%	
	IWM	RGM
53.4	96.3	92.8
98.1	94.6	90.4
126.2	92.8	88.5
153.7	92.1	87.6

3.2.2 查全率、运行时间对比

RGM与IWM的查全率和运行时间的比较结果分别如表

4、5 所示。海量数据下的相似数据检测在查全率和时间效率是相互矛盾的。从表 4 中可以看出,相同的运行时间下,RGM 的查全率略低于 IWM,从表 5 中可以明显地看到,在不同数据量上,IWM 的运行速度比 RGM 快。这都是因为 IWM 采用多线程并行检测技术、加速法和优先队列技术,大大减少记录比对时间和总体测时间,既保证查全率又减少检测时间;而 RGM 为了保证查全率,采用多趟检测技术,增加了检测时间。

表 4　两种算法相同运行时间条件下的查全率比较

数据量(万)	同时间下查全率/%	
	IWM	RGM
53.4	98.2	97.6
98.1	97.3	96.6
126.2	95.9	95.3
153.7	95.0	94.5

表 5　两种算法相同查准率条件下消耗时间对比

数据量(万)	同查全率下消耗时间/min	
	IWM	RGM
53.4	8.6	22.1
98.1	14.8	29.8
126.2	21.7	37.8
153.7	28.3	46.4

综上所述,本文提出的基于海量数据的相似重复记录检测算法的性能要优于基于等级分组的相似检测算法。

4　结语

针对海量数据下相似重复记录检测问题,本文采取了多种有效策略。首先采用主观因素和客观因素综合考虑的综合加权法计算各属性的权重,然后采用多线程依据各属性对数据集并行排序,使用加速法提前结束记录比对算法;最后合并检测结果集。实验结果表明,该方法是一个合理、有效的相似重复数据检测方法。本文方法仍有许多未解决的问题,例如:记录之间的相似度阈值大小是根据经验设定的。由于它对记录的检测精度有一定的影响,所以将在以后的工作中继续研究阈值的设定问题。

参考文献:

[1] MONGE A E, ELKAN C P. The field matching problem: algorithms and applications [C]// Proceedings of the 2nd Conference on Knowledge Discovery and Data Mining. Cambridge: AAAI, 1996: 267 – 270.

[2] MINTON S N, NANJO C, KNOBLOCK C A, *et al*. A heterogeneous field matching method for record linkage [C]// Proceeding of the 5th IEEE International Conference on Data Mining. Piscataway: IEEE, 2005: 314 – 321.

[3] HERNANDEZ M, STOLFO S. The merge/purge problem for large databases [C]// Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data. New York: ACM, 1995: 127 – 138.

[4] BLENK O M, MOONEY R. Adaptive name matching in information integration [J]. IEEE Intelligent Systems, 2003, 18(5): 16 – 23.

[5] 邱越峰,田增平,季文赟,等. 一种高效的检测相似重复记录的方法[J]. 计算机学报, 2001, 24(1): 69 – 77.

[6] 鲁均云,李星毅,施化吉,等. 基于内码序值聚类的相似重复记录检测方法[J]. 计算机应用研究, 2010, 27(3), 874 – 878.

[7] 孟祥逢,鲁汉榕,郭玲,等. 基于遗传神经网络的相似重复记录检测方法研究[J]. 计算机工程与设计, 2010, 31(7): 1550 – 1553.

[8] 李星毅,包从剑,施化吉. 数据仓库中的相似重复记录检测方法[J]. 电子科技大学学报, 2007, 36(6): 1273 – 1277.

[9] MONGE A E, ELKAN C. An efficient domain-independent algorithm for detecting approximately duplicate database records [C]// Proceedings of the SIGMOD 1997 Workshop on Research Issues on Data Mining and Knowledge Discovery. Cambridge: AAAI, 1997: 23 – 29.

[10] 张永,迟忠先. 位置编码在数据仓库 ETL 中的应用[J]. 计算机工程, 2007, 33(1): 50 – 52.

(上接第 2203 页)

[3] NGUYEN T T, C HUI S C, CHANG K Y. A lattice-based approach for mathematical search using formal concept analysis [J]. Expert Systems with Applications, 2012, 39(5): 5820 – 5828.

[4] BAL M, BAL Y, USTUNDAG A. Knowledge representation and discovery using formal concept analysis: an HRM application [C]// WCE 2011: Proceedings of the World Congress on Engineering. London: Newswood, 2011: 1068 – 1073.

[5] CASSIO M, LEGRAND B. Extracting and visualising tree-like structures from concept lattices [C]// IV'11: Proceedings of the 2011 15th International Conference on Information Visualisation. Washington, DC: IEEE Computer Society, 2011: 261 – 266.

[6] JULIEN B, FABRICE G, HENRI B. Interactive visual exploration of association rules with rule-focusing methodology [J]. Knowledge and Information Systems, 2007, 13(1): 43 – 75.

[7] MICHAEL H, CHELLUBOINA S. Visualizing association rules in hierarchical groups [C]// Interface 2011: Statistical, Machine Learning, and Visualization Algorithms. Cary, North Carolina: SAS Institute, 2011: 1 – 11.

[8] DARIO B, CRISTINE D. Visual mining of association rules [C]// Visual Data Mining: Theory, Techniques and Tools for Visual Analytics, LNAI 6208. Berlin: Springer-Verlag, 2008: 103 – 122.

[9] BILAL A, ERHAN A, ALI K. MODENAR: Multi-objective differential evolution algorithm for mining numeric association rules [J]. Applied Soft Computing, 2008, 8(1): 646 – 656.

[10] PACHON A V, VAZQUEZ J. An evolutionary algorithm to discover quantitative association rules from huge databases without the need for an a priori discretization [J]. Export Systems with Applications, 2012, 39(1): 585 – 593.

[11] MARTINEZ B M, RIQUELME J. Analysis of measures of quantitative association rules [C]// HAIS'11: Proceedings of the 6th International Conference on Hybrid Artificial Intelligent Systems. Berlin: Springer-Verlag, 2011: 319 – 326.

[12] SHAHARANEE M, HADZIC F, DILLON S. Interestingness measures for association rules based on statistical validity [J]. Knowledge-Based Systems, 2011, 24(3): 386 – 392.

[13] SAMUEL Y, MEKITIE W, MULUMEBET A, *et al*. Duration and determinants of birth interval among women of child bearing age in Southern Ethiopia [J]. BMC Pregnancy and Childbirth, 2011, 11(38): 1 – 6.

[14] SONG S J, KIM E H, KIM H E, *et al*. Query-based association rule mining supporting user perspective [J]. Computing, 2011, 93(1): 1 – 25.

[15] LIU G M, ANDRE S. AssocExplorer: an association rule visualization system for exploratory data analysis [C]// KDD '12: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2012: 1536 – 1539.

[16] GELYB A, RAOUL M, NOURINE L. Representing lattices using many-valued relations [J]. Information Sciences, 2009, 179(16): 2729 – 2739.

[17] 马瀛通. 人口统计分析学[M]. 北京: 红旗出版社, 1989: 696.