一种具有最大推荐非空率的关联规则挖掘方法*

王大玲+, 于 戈, 鲍玉斌

(东北大学 信息科学与工程学院,辽宁 沈阳 110004)

An Approach of Association Rules Mining with Maximal Nonblank for Recommendation

WANG Da-Ling+, YU Ge, BAO Yu-Bin

(School of Information Science and Engineering, Northeastern University, Shenyang 110004, China)

+ Corresponding author: Phn: +86-24-83687776, Fax: 86-24-23895654, E-mail: dlwang@mail.neu.edu.cn, http://www.neu.edu.cn

Received 2003-04-11; Accepted 2004-01-06

Wang DL, Yu G, Bao YB. An approach of association rules mining with maximal nonblank for recommendation. *Journal of Software*, 2004,15(8):1182~1188.

http://www.jos.org.cn/1000-9825/15/1182.htm

Abstract: To improve quality of personalized recommendation and simplify the preference setup in generating recommendation rules, the characteristics of the association rule for personalized recommendation are discussed, the concepts of recommendation nonblank metric, a new recommendation metric, 1-support frequent itemset and k-maximal association rule are defined, and the idea of getting k-maximal association rule from 1-support frequent itemset is proposed. Moreover, an association rule mining algorithm based on the idea is designed, which is suitable for different sliding window depths. The theoretic analysis and experiment results on the algorithm show that the method has maximal nonblank, higher precision and *F*-measure of recommendation, and simplifies the preference setup of thresholds in mining rules effectively.

Key words: 1-support; association rule; Web usage mining; personalization; nonblank

摘 要: 为了提高个性化推荐的质量,简化推荐规则生成过程中相关参数的设置,讨论了应用于个性化推荐中的关联规则的性质,定义了"推荐非空率"这一新的推荐测度以及"1-支持频繁项集"和"k 最大关联规则"的概念,提出了"在1-支持频繁项集中生成k最大关联规则"的思想,设计了满足该思想且适合于不同滑动窗口深度下推荐的关联规则挖掘算法.理论分析及实验结果表明,该算法具有最大的推荐非空率、较高的推荐准确率和 *F*-测度,并有效地简化了规则挖掘过程中阈值的设置.

关键词: 1-支持;关联规则;Web 使用挖掘;个性化;非空率

中图法分类号: TP311 **文献标识码**: A

通过 Web 挖掘得到用户的兴趣和爱好,以便提供个性化服务,是目前网站建设所采用的一种重要手段.Web

* Supported by the National Natural Science Foundation of China under Grant No.60173051 (国家自然科学基金)

作者简介: **王大玲**(1962－),女,辽宁沈阳人,博士,教授,主要研究领域为数据挖掘,Web 挖掘;**于戈**(1962－),男,博士,教授,博士生导师,主要研究领域为数据库理论与技术;**鲍玉斌**(1968－),男,博士,副教授,主要研究领域为数据仓库,OLAP.

使用挖掘(Web usage mining)是 Web 挖掘的重要技术之一,以此可以得到用户的访问模式,从而根据这种模式为用户定制合适的推荐页面[1].关联规则挖掘方法于 1993 年首次由 Agrawal 提出[2],近年来已应用于 Web 挖掘中.在个性化推荐中应用的关联规则具有如下特点[3,4]:(1) 规则后项代表用户访问过的一个页面的 URL,其长度为 1,称为 1-size;(2) 规则前项代表用户在访问后项之前所浏览过的页面的 URL,其长度是该用户浏览过的页面数目,称为活动会话窗口数目或滑动窗口深度.

根据关联规则挖掘和个性化推荐的特点,所涉及的问题主要包括支持度阈值的设置和滑动窗口深度的选择.围绕这些问题,有许多相关的研究工作.

Liu 等人曾就支持度阈值的设置问题提出了多支持度阈值的关联规则挖掘算法,使不同的项集满足不同的支持度阈值[5].他们提出了对不同用户给定不同的规则数的支持度阈值"自适应"方法,将支持度阈值的设置变成规则数目的设置,使规则数目可控[4].Mobasher 采用扩展 all-*k*th-order 的思想,生成不同滑动窗口深度下的推荐页面,以解决滑动窗口的选择问题[3].他还提出了评价推荐算法的 3 个测度,即推荐覆盖率、准确率和 *F*-测度[6].此外,Han 等人提出了频繁模式增长(FP-Growth)的思想,设计了基于该思想的项存储结构以及在此结构上的频繁模式挖掘算法,包括针对一般频繁模式的 FP-tree[7]和针对 Web Log 数据频繁模式的 WAP-tree[8]. Wang 等人基于 FP-Growth 思想给出了 TD-FP-Growth 算法,以应用于频繁模式的挖掘[9].

为了更高效地实现推荐服务,本文提出了一个新的推荐测度:推荐非空率,并吸收了 Mobasher 的不同滑动窗口深度[3]、Liu 的多最小支持度阈值[4]以及 Lin 的可控推荐参数的思想,设计了一种"1-支持频繁项集"和"*k* 最大关联规则"的挖掘算法,应用 FP-tree[7]生成频繁模式的思想,在初始 FP-tree 的基础上生成频繁项集,采用 1-FIS-Tree 结构存储频繁项集,并在该结构上生成支持个性化推荐的关联规则.

1 推荐测度

目前,衡量一个推荐系统,采用较多的是 Mobasher 给出的评价测度[6],包括覆盖率(coverage)、准确率(precision)以及 *F*-测度(*F*-measure).覆盖率是在推荐的内容中用户喜欢的项占用户喜欢的所有项的百分比,准确率是在推荐的内容中用户喜欢的项占推荐的所有项的百分比.设用户喜欢的页面集合为 *US*,而系统推荐的页面的集合为 *RS*,根据定义,coverag,precision 和 *F*-measure 分别由公式(1)~(3)给出.

$$coverage=\frac{|US\cap RS|}{|US|} \tag{1}$$

$$precision=\frac{|US\cap RS|}{|RS|} \tag{2}$$

$$F\text{-}measure=\frac{2\times coverage\times precision}{coverage+precision} \tag{3}$$

Coverage 和 precision 分别从推荐的广泛性和精确性方面对推荐方法进行衡量,而 *F*-测度则是两者的结合.由公式(3)可见,无论忽视 coverage 和 precision 中的哪一个,都将造成 *F*-测度的降低.

我们认为,coverage,precision 和 *F*-measure 还不能够全面地衡量一个推荐系统的质量.为此,本文提出另一个推荐测度:推荐非空率,简称非空率.

定义 1(非空率(nonblank)). 在用户所访问的页面中,能够给出推荐内容的页面所占的比例.设 *UP* 为用户访问过的页面的集合,*RP* 为具有推荐内容的页面的集合,则 nonblank 可应用公式(4)计算.

$$nonblank=\frac{|UP\cap RP|}{|UP|} \tag{4}$$

Nonblank 与 coverage 具有相似之处,但 nonblank 强调的是在一个页面中是否有推荐内容.一个推荐系统,如果只在用户访问的几个页面中给出了许多用户喜欢的推荐,而在其余的许多页面中却没有给出任何推荐,它的 coverage 也可以达到较大的值,但 nonblank 却是很小的,而 nonblank 过小的推荐系统同样不是一个高质量的系统.显然,一个推荐系统是否能够在用户访问的页面中给出推荐内容,取决于在规则集中是否存在与当前用户

访问模式相匹配的规则.就使用关联规则而言,coverage 涉及的是规则的数量,而 nonblank 涉及的是规则的分布.

我们认为,系统应该保证足够的 nonblank,尽量高的 coverage 和 precision.一般地,nonblank 高,一方面可以带给用户一种信任感,同时也能够起到一定程度的导航作用,这对于初次上网的用户尤为重要.在此基础上,尽可能地提高 coverage 和 precision.

2 算法描述

2.1 相关概念和定理

定义 2(1-支持频繁项集). 即设定的支持数阈值为 1(1-support)时得到的频繁项集,记作 1-FIS.

这里的支持数并非大多数文献中定义的支持度.支持度是一个相对值,与数据集中的记录总数相关,而支持数是一个绝对值,与数据集的记录总数无关.1-support 的设定意味着,一个项集只要在任意记录中出现过一次,即可成为频繁项集,即 1-FIS.

定理 1. 如果采用 1-support 寻找频繁项集,得到的 1-FIS 能够包括数据集中全部项集(证明略).

定义 3(*k* 最大关联规则). 对于所有前项相同的规则,其中 *k* 个支持数最大者称为 *k* 最大关联规则,记作 *k*-MAR.

定理 2. 对于 *m* 个 *i*-size(2≤*i*≤*n*+1)(*n* 为规则前项的最大长度,亦即滑动窗口深度)的 1-FIS $item_{i1},item_{i2},...,item_{im}$,若它们均包含某个(*i*−1)-size 频繁项集 $item_{i-1}$ 中的各项,则仅选择其中支持数最大的 *k* 个项集生成关于 $item_{i-1}$ 的关联规则,即可得到关于 $item_{i-1}$ 的 *k*-MAR(证明略).

定理 3. 若在 1-FIS 中生成不同前项的 *k*-MAR,则应用 *k*-MAR 推荐时,只要 *k*≥1,即可获得最大的推荐非空率(证明略).

在 1-support 下,将产生大量的 1-FIS,但并非这些 1-FIS 都要生成有效的推荐规则.按照定义 3,在所有具有相同前项的关联规则中,只有支持数最大的 *k* 条规则才可以成为推荐规则.*k* 值的设置问题将在第 3.2 节中加以讨论.

同时,也并非所有的 1-FIS 都需要在生成规则之后再取 *k*-MAR.根据定理 2,对于 *m* 个 *i*-size(2≤*i*≤*n*+1)的 1-FIS,$item_{i1},item_{i2},...,item_{im}$,若它们均包含某个(*i*−1)-size 的 1-FIS 中的各项,则仅选择其中支持数最大的 *k* 个 1-FIS 生成前项长为(*i*−1)-size 的 *k*-MAR 用于推荐.

2.2 1-FIS存储结构

虽然无须生成大量的规则,但在此前得到的大量 1-FIS 将导致“项集爆炸”,使算法的时间和空间开销过大.为了解决大量 1-FIS 的存储及 *k*-MAR 生成问题,我们在初始 FP-tree 结构的基础上,设计了一种 1-FIS 的存储结构 1-FIS-Tree 存储 1-FIS,并在该结构上生成 *k*-MAR.

定义 4(1-FIS-Tree). 一种二叉树存储结构,用于存储 *i*-size 和(*i*+1)-size(*i*=1,2,…,*n*)的 1-FIS,树中每个节点结构为⟨*iname*,*support*,*left*,*right*⟩.其中,“*iname*”为存储的 1-FIS 的名字,“*support*”为该 1-FIS 的实际支持数,其左子树指针“*left*”指向比该节点的 *iname* 长度增加 1 的一个超集对应的节点,右子树指针“*right*”指向与本节点的 *iname* 具有相同长度的另一个 1-FIS 对应的节点.

设“*ABEF*”为一个会话,即 URL 序列,且 *n*=3.可生成如下的 1-FIS:

Size=1: *A*, *B*, *E*, *F*;

Size=2: *AB*, *AE*, *AF*, *BE*, *BF*, *EF*;

Size=3: *ABE*, *ABF*, *AEF*, *BEF*;

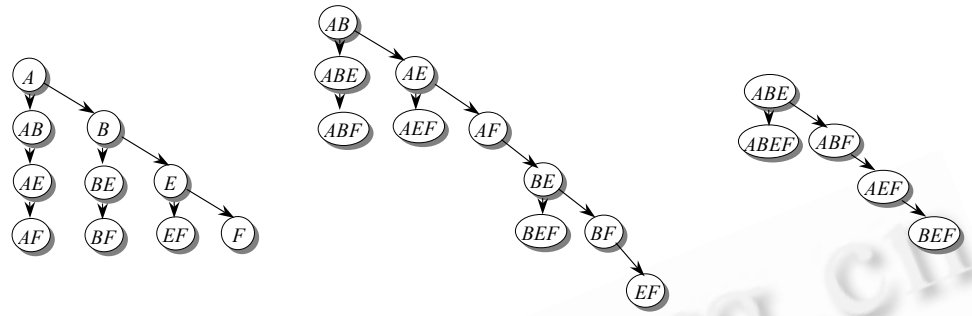
Size=4: *ABEF*.

其存储结构如图 1 所示,这就是 1-FIS-Tree.

2.3 *k*-MAR挖掘算法

根据个性化推荐中的关联规则所具有的 1-size 后项的性质,在挖掘 *i*-size 前项的关联规则时,仅考虑 *i*-size

和(*i*+1)-size 的 1-FIS 即可.基于这一思想,本文所述的挖掘过程包括“建立初始 FP-tree”、“由初始 FP-tree 生成 *i*-size 和(*i*+1)-size (*i*=1,2,…,*n*)的 1-FIS-Tree”以及“由 1-FIS-Tree 生成 *k*-MAR”3 个组成部分.



(a) 1-FIS of 1-size, 2-size (*i*=1) (b) 1-FIS of 2-size, 3-size (*i*=2) (c) 1-FIS of 3-size, 4-size (*i*=3)

Fig.1 1-FIS-Tree storage structure of a session in log file

图 1 Log 文件中一个会话记录的 1-FIS-Tree 存储结构

FP-tree 的结构为一个头表(head table)和一棵树,关于其具体结构,Han 等人曾给出了详述[7].

设 *SF* 为经过预处理 Web Log 后生成的一个会话文件(预处理过程采用 Cooley 等人的方法[10]),*RS* 为存储挖掘结果的规则集,*n* 为给定的规则前项的最大长度,则挖掘算法 Mining-*k*-MAR 如下:

Algorithm Mining-*k*-MAR(*SF*,*RS*,*n*);
Begin
  根据 1-support 建立初始 FP-tree;
  将 1-size 项集按字典序建立 1-FIS-Tree 中的 1-size 节点;
  For *i*=2 to *n*+1
  { For each item in 1-size
    {由表头各项出发,在初始 FP-tree 上遍历,生成以表头各项为前缀的 *i*-size 的 1-FIS;
     按定义 4 中的结构将这些 *i*-size 的 1-FIS 按字典序插入 1-FIS-Tree;
    };
    根据定理 2 生成(*i*–1)-size 前项的 *k*-MAR 并存储于 *RS*;
    从 1-FIS-Tree 上释放所有(*i*–1)-size 项集对应的节点;
  };
  从 1-FIS -Tree 上释放所有(*n*+1)-size 的 1-FIS 对应的节点;
End.

在算法中,建立初始 FP-tree 与 FP-Growth 方法的不同之处在于,将数据集中各事务的项按支持数降序排列后,再将其映射到英文字母空间,以便后面的操作在有序的条件下进行.

3 算法分析与评价

3.1 挖掘算法分析

本文提出的挖掘算法是在初始 FP-tree 基础上建立 1-FIS-Tree.具体来讲,在第 *i* 次遍历 FP-tree 时生成 1-FIS-Tree 上(*i*+1)-size 的 1-FIS 所对应的节点.建立 *i*-size 和(*i*+1)-size 节点以后,生成前项为 *i*-size 的 *k*-MAR,之后释放 *i*-size 节点,再建立(*i*+2)-size 节点,依此类推.由于 1-size 节点是从 FP-tree 的“header table”中得到的,所以,若生成前项为 1-size~*n*-size 的 *k*-MAR,需遍历 *n* 次 FP-tree.

FP-tree 是一种高度压缩的数据结构,通过对数据集的两遍扫描即可建立起 FP-tree,其余操作均在 FP-tree 上进行,特别适合于小支持度(数)频繁项集的生成[7].与 FP-tree 相同,本文的算法在建立初始 FP-tree 时,也是仅需扫描数据集两遍,第 1 遍计算各会话序列中每个 URL 的支持数,并将全体 URL 按支持数的排序结果映射到英文字母空间,第 2 遍将用字母代替的各 URL 在每个会话中按序排列.由于 1-support 的设置,本文的算法中不存在

剔除非频繁项集(URL)的问题,后面的操作是在初始 FP-tree 上进行的,无须再对数据集做任何操作.

与 FP-Growth 算法相比,两者后面的操作都是在 FP-tree 上进行的,同时在建立 FP-tree 上所花费的时间及空间代价是相同的.而 FP-Growth 算法是在 FP-tree 基础上不断地生成条件 FP-tree,从而在其上得到不同长度的频繁项集.由于 FP-Growth 算法中在建立 FP-tree 时是按照各项的支持度(数)的排序结果,而非字典序的排序结果生成节点的排序结果,这样虽然可以有效地减少节点的数量,但由此 FP-tree 生成的频繁项集在项集之间、项集内部各项之间均是无序的,给进一步生成关联规则带来了许多麻烦.FP-Growth 的贡献在于,它无须生成候选频繁项集,且仅遍历数据集两遍,但是其结果仅仅是生成不同长度的频繁模式,而非关联规则,同时,FP-Growth 算法无论初始 FP-tree 还是生成频繁模式时产生的条件 FP-tree 都需要在内存中进行,因此要占用大量的存储空间.当然,在给定合适的支持度(数)阈值时,其代价主要在发现频繁项集所需的时间上,空间代价大多忽略不计,生成关联规则过程的时空代价一般更不被考虑.但是,在 1-support 下,所生成的 1-FIS 数量将大大高于给定合适支持度(数)阈值的情况,这样,就不能不考虑这些 1-FIS 的存储以及进一步生成规则的问题.本文提出的算法在建立初始 FP-tree 时也采用对各会话序列中的 URL 按支持数排序,但同时将排序结果映射到字母集合中去,这就使支持数的排序与字典序融于一体,既有效地减少了 FP-tree 中的节点数量,同时也给后面生成有序 1-FIS 进而生成 *k*-MAR 创造了极大的方便.

定理 4. 设 m 为一个网站的页面数目,则 i-size 的 1-FIS 的数目不会大于 C_m^i(证明略).

定理 5. i-size 和(i+1)-size 的 1-FIS 构成的 1-FIS-Tree 的高度等于 i-size 的 1-FIS 的数目(证明略).

定理 6. i-size 和(i+1)-size 的 1-FIS 构成的 1-FIS-Tree 的节点数目小于 $2^{C_m^i}-1$(m 为网站页面数目)(证明略).

根据算法 Mining-*k*-MAR 建立 1-FIS-Tree 过程为节点的顺序插入,而生成 *k*-MAR 时,最多将遍历所有(i+1)-size 节点.定理 4~定理 6 决定了 1-FIS-Tree 的存储空间和规则生成时间.对于一个网站来说,由于页面数目有限,且用户的访问趋于集中于某些网页,所以导致 1-FIS-Tree 的高度及节点数目有限.在后面生成 1-FIS 及 *k*-MAR 的过程中,由于在生成前项为 i-size 的 *k*-MAR 时,在内存中仅仅存储 i-size 和(i+1)-size 的 1-FIS 节点的 1-FIS-Tree,且立即在此 1-FIS-Tree 上生成 *k*-MAR,同时释放 i-size 的 1-FIS 节点,因而有效地节省了内存空间.

3.2 推荐算法评价

本文提出的算法主要包括两方面的特点:在发现频繁项集时设置 1-support,在生成推荐规则时采用前项长度 n 变长,即在 1-FIS 中生成 *k*-MAR,旨在保证推荐具有最大 nonblank,足够的 coverage 和尽量高的 precision.下面采用美国 Depaul 大学的"在线资源"网站的处理数据(http://maya.cs.depaul.edu/~classes/etc584/resource.html)分别进行规则前项为 1~n 变长与规则前项定长方法,并进行"设定支持度阈值"与 1-support 在 coverage 和 precision 方面的比较.

该数据集具有 683 个 URL,13 745 个会话记录.本文选择其中对前 400 个 URL 的访问记录,剔除访问频度小于 0.1%或大于 85%的 URL,或长度小于 4 的会话记录.处理后,将数据集的 2/3 作为训练集,进行 Web 挖掘以生成推荐的关联规则,将其余的 1/3 作为测试集进行测试.测试推荐测度时,将测试集中每个会话记录的前 n(前项变长时为 i=1~n)个 URL 作为推荐输入,即关联规则的前项,从对应规则中得到的推荐 URL 作为推荐项集合 *RS*,而该会话记录中前 n(前项变长时为 i=1~n)个 URL 后的若干个 URL 为用户喜欢的项集合 *US*.实验结果采用的是 k(k=3)层交叉结果的平均值.

图 2 和图 3 分别给出了在设定支持度阈值(*minsup*)时,推荐规则前项定长 1,2,3,4 和变长 1~4 的 coverage 和 precision 随 *minsup* 变化的情况.由图 2 可见,推荐规则前项为变长时具有较高的 coverage,而由图 3 又可见推荐规则前项定长为 3 时具有较高的 precision,但图 2 显示在 n=3 时的 coverage 则很低,而图 4 显示在 n=1~4 时 *F*-measure 具有较大的优势.Mobasher 等人就是采用该思想生成不同滑动窗口深度下的推荐页面的.因此,下面我们与 Mobasher 的推荐方法进行推荐测度的比较.

在图 2~图 4 中,前项为 1~4 时(即 Mobasher 推荐思想),coverage,precision 和 *F*-measure 分别在 *minsup* 为 0.1%,0.5%和 0.4%时处于最大值.所以,下面将"在 1-FIS 中生成 *k*-MAR"与设定 *minsup* 为 0.1%,0.5%,0.4%时的 coverage,precision,*F*-measure 进行对比.图 5~图 7 分别给出了 n=1~4 及 k=1,2,3,4,5,6,7,8,9 时的 coverage,

precision,*F*-measure 与 *n*=1~4 并设定支持度阈值时(即 Mobasher 方法)的对比情况.由图中可见,虽然“在 1-FIS 中生成 *k*-MAR”的方法在 coverage 上不占优势,但在 precision 上具有较大的优势.除了 *k*=1 以外,其余的 *k* 值下的 *F*-测度 *F*-measure 值均大于设定支持度下的最大 *F*-measure 值.

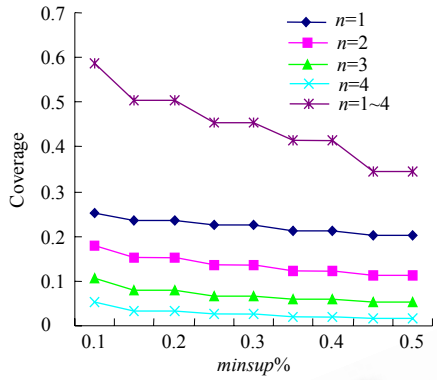


Fig.2 Coverage of different sliding window depth with the variety of *minsup*

图 2 不同滑动窗口深度下覆盖率随支持度阈值变化情况

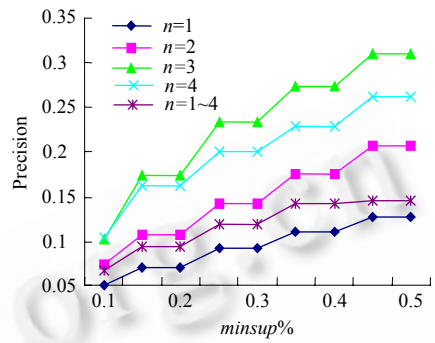


Fig.3 Precision of different sliding window depth with the variety of *minsup*

图 3 不同滑动窗口深度下准确度随支持度阈值变化情况

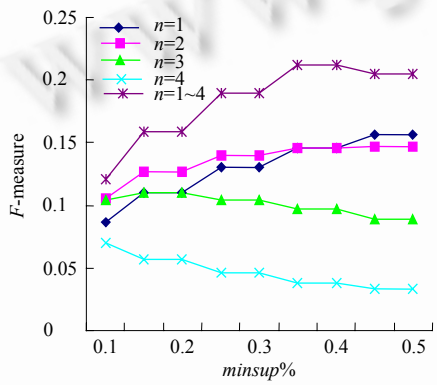


Fig.4 *F*-Measure of different sliding window depth with the variety of *minsup*

图 4 不同滑动窗口深度下 *F*-测度随支持度阈值变化情况

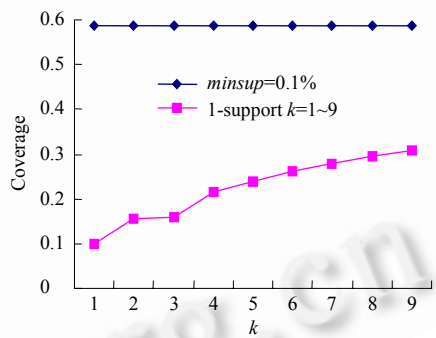


Fig.5 Comparison of coverage at *minsup*=0.1% with *k* in 1-support

图 5 1-支持数下选择不同 *k* 值与 *minsup*=0.1%时覆盖率的比较

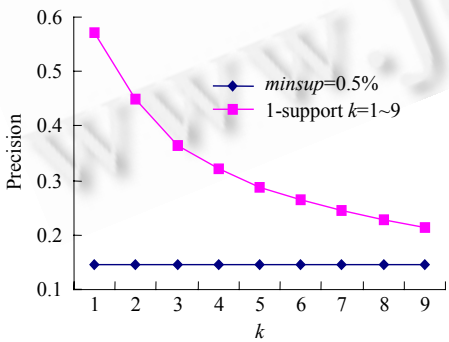


Fig.6 Comparison of precision at *minsup*=0.5% with *k* in 1-support

图 6 1-支持数下选择不同 *k* 值与 *minsup*=0.5%时准确率的比较

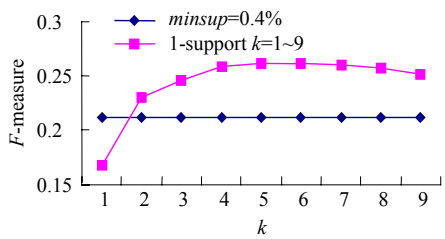


Fig.7 Comparison of *F*-measure at *minsup*=0.4% with *k* in 1-support

图 7 1-支持数下选择不同 *k* 值与 *minsup*=0.4%时 *F*-测度的比较

定理3保证了“在1-FIS中生成k-MAR”的方法具有最大的nonblank,图6和图7又表明了该方法在precision和F-measure上比设置最小支持度阈值具有更明显的优势.虽然k值也需要设定,但这种设定要比设置*minsup*和置信度阈值(*minconf*)简单且有利得多.这是因为:(1) 只要设置$k\geq 2$就可以得到较满意的precision和F-measure;(2) k值的设定对于生成的规则分布是可控的,即可以规定对应于每个相同前项的规则的数目,继而保证最大的nonblank;(3) 根据“在1-FIS中生成k-MAR”算法,生成的k-MAR具有最大的置信度,因而不必专门设置*minconf*.

可见,在1-FIS中生成k-MAR的方法无论在简化参数设置方面,还是在改善推荐测度方面,都是行之有效的.

4 结论与展望

个性化推荐是Web个性化技术的一个重要组成部分.应用Web使用挖掘方法获得用户的访问模式,从而为其制作推荐页面,是一种切实可行的方法,对此,本文做了以下几方面的工作:(1) 提出了推荐非空率nonblank的概念和计算方法,指出nonblank也是衡量一个推荐系统质量的测度之一;(2) 定义了1-支持频繁项集1-FIS和k最大关联规则k-MAR的概念,用以生成支持推荐的关联规则;(3) 设计了满足上述要求的k-MAR挖掘算法及1-FIS-Tree存储结构.算法应用FP-Growth的思想,在初始FP-tree的基础上生成1-FIS并存储于1-FIS-Tree中,并在此结构上生成k-MAR;(4) 对上述算法和结构进行了时间、空间代价分析及推荐覆盖率和准确率的评价.

对算法的分析和评价结果表明,“在1-支持频繁项集中生成k最大关联规则”的方法不仅具有更好的推荐测度,而且参数设置简单、方便.

为了向用户进行更为客观而准确的推荐,还需要进一步做以下工作:(1) 将关联规则挖掘方法与其他规则的挖掘结合起来,如结合用户聚类、页面分类、超链分类和推荐算法,以便得到更精确的用户访问模式,给出更加满足个性化要求的推荐页面;(2) 集成Web使用挖掘和Web内容挖掘技术于一体[11],分析页面内容与该页面访问时间的关系,以便更加准确地给出推荐页面.

References:

[1] Cooley R, Tan PN, Strivastava J. Discovery of interesting usage patterns from Web data. In: Masand BM, Spiliopoulou M, eds. Int'l WEBKDD'99 Workshop. Heidelberg: Springer-Verlag, 2000. 163~182.

[2] Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. SIGMOD Record, 1993,22(2):207~216.

[3] Mobasher B, Dai HH, Luo T, Nakagawa M. Effective personalization based on association rule discovery form Web usage data. In: Chiang HL, Lim EP, eds. The 3rd Int'l Workshop on Web Information and Data Management. New York: ACM Press, 2001. 9~15.

[4] Lin WY, Alvarez SA, Ruiz C. Efficient adaptive-support association rule mining for recommender system. Data Mining and Knowledge Discovery, 2002,6(1):83~105.

[5] Liu B, Hsu W, Ma YM. Mining association rules with multiple minimum supports. In: Zaki MJ, Ho CT, eds. Proc. of the 5th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. New York: ACM Press, 1999. 337~341.

[6] Mobasher B, Dai HH, Luo T, Nakagawa M. Improving the effectiveness of collaborative filtering on anonymous Web usage data. Technical Report, 01-005, 2001. http://facWeb.cs.depaul.edu/research/TechReports/

[7] Han JW, Pei J, Yin YW. Mining frequent patterns without candidate generation. In: Chen WD, Naughton JF, Bernstein PA, eds. Proc. of the 2000 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Press, 2000. 1~12.

[8] Pei J, Han JW, Mortazavi-asl B, Zhu H. Mining access patterns efficiently from Web logs. In: Terano T, Liu H, Chen LP, eds. Knowledge Discovery and Data Mining, Current Issues and New Applications, the 4th Pacific-Asia Conf. Heidelberg: Springer-Verlag, 2000. 396~407.

[9] Wang K, Tang L, Han JW, Liu JQ. Top down fp-growth for association rule mining. In: Cheng MS, Yu PS, Liu B. eds. Advanced in Knowledge Discovery and Data Mining, the 6th Pacific-Asia Conf. Heidelberg: Springer-Verlag, 2002. 334~340.

[10] Cooley R, Mobasher B, Srivastava J. Data preparation for mining World Wide Web browsing patterns. Knowledge and Information System, 1999,1(1):5~32.

[11] Mobasher B, Dai HH, Luo T, Sun YQ, Zhu J. Integrating Web usage and content mining for more effective personalization. In: Bauknecht K, Madria SK, Pernul G. eds. Electronic Commerce and Web Techniques, the 1st Int'l Conf., EC-Web. Heidelberg: Springer-Verlag, 2000. 165~176.