基于频繁序列模式压缩技术的网站结构优化

程舒通[1,2],徐从富[1],但红卫[1]
CHENG Shu-tong[1,2],XU Cong-fu[1],DAN Hong-wei[1]

1.浙江大学 计算机科学与技术学院,杭州 310027
2.杭州科技职业技术学院,杭州 310022
1.College of Computer Science of Technology,Zhejiang University,Hangzhou 310027,China
2.Hangzhou Poly Technique College,Hangzhou 310022,China
E-mail:chengshutong@21cn.com.

CHENG Shu-tong,XU Cong-fu,DAN Hong-wei.Frequent sequential pattern compression-oriented website structure optimization.Computer Engineering and Applications,2007,43(30):133-135.

Abstract: In this article,Website structure optimization based on frequent sequential pattern compression method is to discover the sequence association in course of the frequently visited pages,which can help us to optimize the site topology.After analyzing the shortages of the existing algorithms for frequent pattern clustering,proposes an algorithm for creating compressed partial order based on pattern clustering function.Experiment result shows the compression arithmetic has higher proficiency and qualify,also can obtain much less number and much more information pattern.Thereby can find out more interesting of visited frequent sequential than the normal algorithm.

Key words: data mining;frequent sequential pattern;compression;Web design

摘 要:基于频繁序列模式压缩技术的网站结构优化算法旨在发现用户在浏览过程中频繁访问的序列关联,为优化站点结构提供有力的依据。分析了现有频繁模式聚类算法的不足,提出了在模式聚类函数的基础上生成一个压缩的偏序(Partial Order)的算法,实验结果显示该算法可以对频繁序列模式进行高效、高质量的压缩,可以得到数量更少、信息量更大的模式,从而提高发现的频繁访问序列的兴趣性。

关键词:数据挖掘;频繁序列模式;压缩;Web 设计
文章编号:1002-8331(2007)30-0133-03 **文献标识码:**A **中图分类号:**TP18

1 引言

随着 Web 技术的迅速发展，生活中每一天都似乎和 Internet 网络关联在一起,使其成为获取信息的重要途径。由于全球 Web 站点数目迅速增长,Web 站点的信息量及其复杂度也在迅速上升,因此给用户访问也增加了一定的难度。因此不得不借助关联规则，序列模式和页面聚类等 Web 日志挖掘工具来获得更深层次的用户访问信息。针对网站目前的访问信息,对网站的设计进行优化,建设一个更合理、更易用、注重个性化和相关性的网站成为研究的热点之一[1-4]。

现有的一些 Web 站点设计方法中，提出了利用寻找用户频繁访问的页组来实现调整网站结构的算法,如文[5]是应用基于频繁项集的关联规则来找到频繁访问路径,以及称作“关联规则的超图划分”的聚类方法来进行 URL 集聚类,然后通过推荐引擎进行在线推荐。文[6]提出了结合站点拓扑结构和 Web 页面内容的页面聚类改进算法来提高挖掘结果的兴趣性。

当频繁序列模式的数量非常庞大时,很难从中挖掘有用的知识。所以现阶段频繁序列模式挖掘研究的瓶颈已不在挖掘的效率上,而是在其结果的可用性和可理解性上。为了解决这个问题,一个很自然的想法就是就将挖掘出来的结果再进行精简压缩,从而可以得到数量更少、信息量更大的模式。

本文提出的基于频繁序列模式压缩技术的网站结构优化,挖掘网站中被浏览者频繁访问的网页集合以及序列模式挖掘算法用以挖掘网站中频繁访问的页面序列,以发现隐藏在其中的用户访问模式,实现合理、有效地优化站点的结构,最终为用户提供一个方便快捷信息获取环境。

2 算法概论

提出的算法,它是以浏览者频繁访问网页形成的频繁序列模式为输入。算法充分考虑了时间和效率的关系,提出了一种高效的距离定义方法。并以简便,保持信息不丢失的方法生成偏序。本算法的主要目的就是希望能从序列数据中,挖掘出更精简、更易理解的模式,并且可以排除部分噪音的干扰。利用生成的偏序提高挖掘结果中频繁访问页组的兴趣性。

将频繁序列模式总结成近似偏序,可以先将序列模式进行聚类,然后将每个聚类后的序列模式分别总结成偏序。主要分为如下的两步:

作者简介:程舒通(1976-),男,讲师,研究方向:人工智能、数据挖掘;徐从富(1969-),男,博士,副教授,硕士生导师,研究方向:人工智能、智能 CAD、数据挖掘、知识发现、数据融合;但红卫,硕士,研究方向:人工智能、数据挖掘。

(1)频繁模式进行聚类,它要求所聚类的组之中的模式能有尽量多的共同特性,并且各个组之间的共同特性较少。同时也要除去组内的冗余项。

(2)当将频繁模式进行聚类好后,对每个组中的模式,需要总结出其中的顺序关系(偏序),此偏序所表达的信息要求与聚类中的序列所包含的信息一致,即此过程中,无信息的丢失。

接下来将分别对这两步的主要思想和算法的具体实现进行分析描述。

3 聚类算法

算法的第一步就是将挖掘出来的闭合模式进行聚类,聚类的关键步骤是模式之间的距离的定义和聚类方法的选择。本章首先对距离函数进行讨论,然后讲述聚类的方法。

3.1 距离函数

为了提高聚类效果,考虑将序列模式本身的信息加入进来,因为序列本身的相似程度也隐含的反应了它们一起出现的概率。为了让产生的距离在不同长度的序列中具有可比性,在文[7]中,作者定义了下面的距离函数:

$$D(P_1,P_2)=\frac{\|(P_1-P_2)\cup(P_2-P_1)\|}{\|P_1\|+\|P_2\|}=\frac{\|P_1\|+\|P_2\|-2\|P_1\cap P_2\|}{\|P_1\|+\|P_2\|}$$

此距离将两个模式之间相同项的权重看得比较重,但是不考虑模式支持度信息,相对支持度信息不是重要的模式环境下,这种方法在一定程度上可以简化数据计算复杂度,因此也比较合理了。采用这个距离函数,实验证明可以得到比较好的聚类效果。

3.2 算法分析

在前面小节中,已经定义出任意两个闭合模式之间的距离,有了这个距离函数,就可以对输入的闭合模式进行聚类。需要将距离相近的模式聚类到一起,所以最常用的聚类方法都可以用到这里来。在本算法中,不过多的叙述聚类方法的细节,所以实现了距离函数的 k-中心方法下的聚类。

算法 1 将闭合序列模式聚类

输入:闭合模式集 $P=\{P_1,P_2,\cdots,P_m\}$

聚类簇数 K 输出:需要连接的序列位置列表。

输出:k 个闭合模式集,$C_1,C_2,\cdots,C_k$

1. load pattern $P_1,P_2,\cdots,P_m$
2. initialize $C_1,C_2,\cdots,C_k$

/* 计算两个模式之间的距离 */

for each P_i

3. for each $P_j,j<i$

$DIST_{ij}=D(P_i,P_j)$

/* 对模式进行聚类 */

4. repeat
5. DistributeSamples();/* 指派每个剩余的对象给离它最近的中心点所代表的簇 */
6. CalcNewClustCenters();/* 重新计算该簇的中心点 */
7. until 每个簇的中心都不发生变化;

/* 除掉每个聚类中的冗余项。*/

for each Cluster $C_i(i=1,\cdots,k)$

for each Pattern P_n *in* Cluster C_i

8. if(IsCovered(P_n)==True)

remove P_n from C_i

9. return $C_i(i=1,\cdots,K)$

4 从聚类中生成近似偏序

本算法的第二步就是从每一个聚类中生成一个压缩的偏序(Partial Order),在前面已经叙述了这样做的意义,此步只需要保证从聚类和从它生成的偏序所包含的信息量不变,即偏序所表示的信息刚好是这个聚类中的所有序列的共同信息。

4.1 算法思想

聚类 $\boldsymbol{C}$ 和 $\boldsymbol{p}$ 偏序都是三维向量,其中聚类 $\boldsymbol{C}=\{S,c,sup\}$,偏序 $\boldsymbol{p}=(V,E,l)$。聚类 $\boldsymbol{C}$ 中,S 表示的是这个聚类中的所有序列,c 表示此聚类的聚类中心序列,sup 表示此聚类的支持度(本文中,将聚类中心的支持度看成是聚类的支持度)。偏序 $\boldsymbol{p}$ 中,V 是所有顶点的集合;$E\subseteq V\times V$ 是所有的边的集合,通过边可以表示顶点之间的关系,它们是自反,反对称和传递的;l 是从顶点到属性项的映射函数,在前面的例子中,用的都是单属性项,但也可以将一个顶点和另一个项目集合相映射。

那么算法第二步的目的就是已知一个聚类 $\boldsymbol{C}=\{S,c,sup\}$,求一个偏序 $\boldsymbol{p}=(V,E,l)$,让 $\boldsymbol{C}$ 和 $\boldsymbol{p}$ 相互一致。首先需要保证 $\boldsymbol{C}$ 中的信息都包含在 $\boldsymbol{p}$ 中,其次 $\boldsymbol{p}$ 中也不能包含 $\boldsymbol{C}$ 中没有的信息。在此步中,因为对于每个聚类,只记录了聚类中心序列的支持度大小的信息,而生成的偏序 $\boldsymbol{p}$ 的支持度的信息与此相同,所以在接下来的讨论中,只需要关注序列模式本身的信息。综上所述,当 $\boldsymbol{C}$ 和 $\boldsymbol{p}$ 满足下面两个条件时,就可以说 $\boldsymbol{C}$ 和 $\boldsymbol{p}$ 是相一致的,即 $\boldsymbol{p}$ 所表示的信息和 $\boldsymbol{C}$ 所表示的信息是一致的。

4.2 算法实现

此算法的关键地方就是怎么找到聚类 $\boldsymbol{C}$ 中序列 S 之间的匹配点,从而让它们成为产生的偏序的最大路径恰好是 S 中的所有序列。

为了构造出这种最特殊(specific)的偏序 $\boldsymbol{p}$,要将 S 中的序列能尽量多的匹配。所以一个仍然需要解决的问题就是识别出 S 中的序列哪些点必须匹配,并不是把所有的有相同属性的点叠加起来就是好的。当两个点重叠后,仍然是路径保留,则这两个点可以重叠。有如下定义:

定义 (路径保留点):给定一个序列集合 $S,ss'\in S$ 是条序列,设 $s=<(I_1)\cdots(I_i)\cdots(I_n)>,s'=<(I_1')\cdots(I_j')\cdots(I_m')>$,当满足下列条件时,则序列 s 中的第 i 项和序列 s' 中的第 j 项是路径保留的:

(1)$I_i=I_j'$;

(2)存在 $s''\in S,head(s,i)\diamond tail(s',j+1)\subseteq s''$;

(3)存在 $s''\in S,head(s',j)\diamond tail(s,i+1)\subseteq s''$;

此时,可以说 s 中的第 i 项和序列 s' 中的第 j 项是可以匹配的。

这个定义确保了在对聚类中的序列进行匹配的时候,任何新加的路径都在聚类 $\boldsymbol{C}$ 的序列集 S 中。下面的算法就是解决从一个聚类 $\boldsymbol{C}$ 的序列 S 中,找出匹配点的问题。

算法 2 找出一个聚类 $\boldsymbol{C}$ 中的序列集 S 中的所有可以匹配的点

输入:一个聚类 $\boldsymbol{C}$ 中的序列集 S

输出:可以匹配的位置列表

for all $s,s'\in S$ s.t. $s\neq s'$ *do*

for all positions i of s and j of s' do

if (pathreserve(s_i,s_j'))

output(s_i,s_j',i,j);

```
    end for
end for
```

5 实验结果

在本章中，在真实的数据集上测试了算法 ApproxPO 的性能和效果。此算法是用 C++实现,在 VS.net 2003 的环境下编译运行的。所有的实验都在操作系统为 Windows XP,CPU 为 Celeron(R)2.02 G,内存为 512 M 的单机上运行。

实验部分主要分为聚类效果的实现和偏序的可视化。在实验中主要用到了 Gazelle 数据文件，这个数据曾用在 KDD-Cup2000 竞赛中，现在可以从 http://www.ecn.purdue.edu/KDD-CUP/得到。此数据中一共提供了 70 163 个会话,每个会话中包含了一序列的页面浏览页面,这个数据库文件也是一个稀疏的数据库文件,它包含了 1 037 个不同的属性项,每个会话的平均长度只有 2.50,实验中就称之为 Gazelle_Data。

展示从实际的数据文件 Gazelle_Data 中总结出的几个偏序,并解释其具体意义,并对网站结构提出改进意见。因为在改进后的 clospan 算法计算闭合模式时,输入文件为二进制文件,所以在对数据进行处理时,将具体的属性值都转化成为了数字。

图 1 和图 2 是 Gazzele_Data 中挖掘出来的两个偏序样例，Gazzele_Data 是网站用户的点击流数据,所以每个节点都代表网站的一个子域名(或页面),在图 1 中,1 代表的是 main/home\.jhtml,为此网站的主页;5 代表的是 main/departments\.jhtml,为部门介绍页面,3 代表的是 main/search_results\.jhtml,是一个搜索页面;6 代表的是 products/productDetailLegwear\.jhtml，是一个产品细节描述页面;9 代表的是 main/assortment\.jhtml，是产品分类页面;22 代表的是 main/shopping_cart\.jhtml，是购物车页面；在图 2 中,1、3、6、9、22 页面所代表的意思同上,10 代表的是 main/boutique\.jhtml，是流行服饰小商店的页面;24 代表的是 main/login2\.jhtml,是登录页面;15 代表的是 main/registration\.jhtml,为注册页面;从以上两张图发现,用户通常通过搜索页面来了解商品分类,商品详细信息以及达到购物车页面。因此搜索页面可以成为高度感兴趣的推荐页面放在首页上比较重要的位置。同时可以细分几种功能分别达到商品分类,商品详细信息以及购物车页面。图 2 中从页面 1-6-24-15 这条访问路径可以发现用户往往是先观看了产品的详细信息,感兴趣后才注册登录进行购买的,因此产品详细信息页面成为用户兴趣页面,可以在首页上设置相关链接,让用户能够容易访问到该页面,以增加用户信任度。

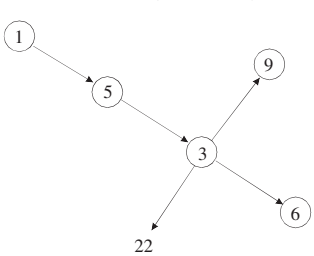

图 1 从 Gazzele_Data 中挖掘出的偏序样例 1

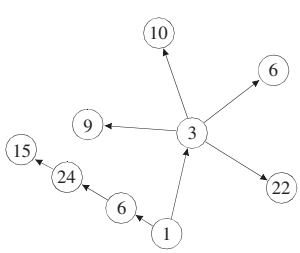

图 2 从 Gazzele_Data 中挖掘出的偏序样例 2

图 3 也是 Gazzele_Data 中挖掘出来的偏序样例,这说明当用户登陆主页之后，有两种常用的方式来查看具体的商品信息,一个是通过搜索页面来搜索商品,另外一个就是通过查看小商品的信息列表来查看商品的信息,这也是一般的购物网站的常见方式。

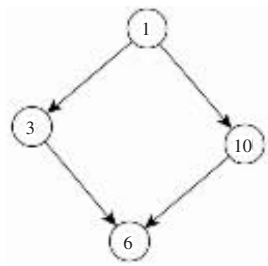

图 3 从 Gazzele_Data 中挖掘出的偏序样例 3

6 结论和展望

对 Web 网站结构优化的方法是在频繁序列模式的基础上试图将闭合序列模式进一步压缩,使其变成更精简也更易理解的偏序的形式。首先讨论了模式之间的距离的计算方法,并且提出了将支持模式间相同项权重值的距离函数。利用此函数,可以在高效的情况下对闭合序列模式进行聚类,且取得很好的聚类效果。其次,描述和证明了怎样在保证信息不丢失的情况下,从一个聚类中总结出一个最精简的偏序出来。实验结果显示本算法能够在高效的情况下,挖掘出一些用其它的方法无法挖掘出来的知识。这对于开发一些目的性强的网站具有较大帮助(如电子商务网站),可以提高 Web 用户的服务质量,使用户享用到满意的个性化服务。(收稿日期:2007 年 3 月)

参考文献:

[1] Nakayama T,Kato H,Yamane Y.Discovering the gap between Web site designers' expectations and users' behavior [J].Computer Networks,2000,33:823-835.

[2] Garofalakis J,Kappos P,Mourloukos M.Web site optimization using page popularity[J].IEEE Internet Computing,1999(7/8):22-29.

[3] Wang Y W,Wang D W.Design strategy of Web page for e-supermarket[C]//Jiang Ping-yu.International Conference on eCommerce Engineering 2001.Xi'an:Machine Press,2001.

[4] Kim J,Yoo B.Toward the optimal link structure of the cyber shopping mall[J].Int J Human-Computer Studies,2000,52:531-551.

[5] Mobasher B,Cooley R,Srivastava J.Creating adaptive sites through usage-based clustering of URLs[C]//Proc Knowledge and Data Engineering Exchange Workshop,IEEE,1999:19-25.

[6] 杨怡玲,管旭东,尤晋元.基于页面内容和站点结构的页面聚类挖掘算法[J].软件学报,2002,13(3):467-469.

[7] Kum H,Pei J,Wang W,et al.ApproxMAP:approximate mining of consensus sequential patterns[C]//UNC-CH 2002,2002.