第 19 卷 第 3 期 控 制 与 决 策 2004 年 3 月
Vol 19 No. 3 Control and Decision Mar. 2004

文章编号: 1001-0920(2004)03-0307-04

一个可以准确反映Web浏览兴趣的度量值——偏爱度

邢东山, 沈钧毅
(西安交通大学 电子与信息工程学院, 陕西 西安 710049)

摘 要: 在分析如何准确反映Web浏览兴趣的基础上提出偏爱度的概念, 并依据这个概念设计了基于用户浏览偏爱树的偏爱路径挖掘算法. 首先用Web日志构筑用户浏览偏爱树(PNT); 然后利用PNT树进行用户浏览兴趣模式的挖掘, 发现用户浏览偏爱路径. 该算法可广泛应用于电子商务领域.

关键词: 浏览偏爱树; Web使用挖掘; 数据挖掘; Web日志; 电子商务
中图分类号: TP391 **文献标识码**: A

A new measurement for Web navigation interest——preference

XING Dong-shan, SHEN Jun-yi
(School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China. Correspondent: XING Dong-shan, E-mail: dsxing@163.net)

Abstract: The concept of preference is proposed on the analysis of how to represent user navigation interest. Based on this concept, the preferred navigation tree is presented for mining Web access patterns. Then the preferred path is mined from this tree. This method can be widely used in E-business.

Key words: preferred navigation tree; Web usage mining; data mining; Web log; E-business

1 引 言

发现用户的浏览兴趣模式对于Web管理员按照用户的爱好优化网站设计等具有重要的意义. 目前, 人们已提出了一些用户浏览兴趣模式的挖掘算法: 如Chen等[1]将访问日志数据转换成一组连续的访问页面(最大向前引用序列MFP)来找出频繁访问模式; Borges和Levene等[2,3]将Web日志以图的形式表示, 节点表示页面, 边表示超链接, 边的权重表示用户访问的概率; 然后用关联规则[4,5]的方法在这个图上抽取浏览模式; Myra Spiliopoulou[6]设计的WUM, 按照统计和结构化属性通过MINT挖掘语言提取兴趣浏览模式. 然而, 这些研究尚存在一些不足, 其中主要问题是发现的模式和规则不能准确反映用户浏览兴趣. 另外, 这些方法还存在可视化不好、不易理解、可扩展性差等缺点.

本文提出一个可以准确反映Web浏览兴趣的度量值——偏爱度, 并依据这个概念构造浏览偏爱树(PNT); 然后在PNT树上挖掘用户浏览偏爱路径; 最后进行性能分析.

2 一个可以准确反映Web浏览兴趣的阈值——偏爱度

定义1(偏爱度) 设U是网站中所有页面统一资源定位(URL)的集合, W是所有浏览子路径的集合. 如果存在$w \subset W$, 对于$\forall x \in w$ (x是$\forall u \in U$组成的浏览页面序列, 称其中第j个浏览页面为第j位), 它们的前m位都相同, 而$m+1$位有n种不同的浏览页面, 则称在m位上有n种不同的选择, 其中第k($k = 1, 2, \cdots, n$)种选择的偏爱度定义为

收稿日期: 2002-07-01; **修回日期**: 2003-08-27.
基金项目: 国家自然科学基金资助项目(60173058).
作者简介: 邢东山(1972—), 男, 河南郑州人, 博士, 从事Web挖掘技术和电信欺诈挖掘技术的研究; 沈钧毅(1939—), 男, 江苏常熟人, 教授, 博士生导师, 从事数据库理论和数据挖掘的研究.

$$(C_k \cdot T_k) \Big/ \left(\left(\sum_1^n C_i \right) \cdot \left(\sum_1^n T_i \right) \Big/ n^2 \right). \qquad (1)$$

其中: C_i 表示第 i 种选择的支持度, 即用户通过第 i 种选择进入下一个页面的次数; T_i 表示用户通过第 i 种选择进入下一个页面的离散化时间的总和. 离散化时间是指将离散化技术应用于用户浏览时间的表示上, 将时间属性域划分为区间, 用区间的标号代替实际的时间值. 如可按照用户在页面上的停留时间将浏览分成简单经过、大致浏览、正常浏览和兴趣浏览 4 种, 则离散化为

$$T = \begin{cases} 1, & 0 < t \le T_{\max\_passing}; \\ 2, & T_{\max\_passing} < t \le T_{\max\_simple\_viewing}; \\ 3, & T_{\max\_simple\_viewing} < t \le T_{\max\_normal\_viewing}; \\ 4, & T_{\max\_normal\_viewing} < t \end{cases} \qquad (2)$$

这里: $T_{\max\_passing}$, $T_{\max\_simple\_viewing}$ 和 $T_{\max\_normal\_viewing}$ 分别代表设定的最大简单经过时间, 最大大致浏览时间和最大正常浏览时间.

3 用户浏览偏爱树(PNT)

定义 2(用户浏览偏爱树) 用户浏览偏爱树(PNT)是按用户浏览页面序列构筑的多层结构树. 树上的每个节点代表一个浏览页面, 树干表示沿着同一路径到下一个节点的访问序列. 这个结构与文献[7]较为相似, 不同的是节点记录了用户通过同样的前缀路径到达该点的次数, 即支持度, 也记录了从上一个节点到该节点的偏爱度量.

PNT 树结构可描述为:

```
TYPE Nodeptr = ^Nodetype
  Nodetype = RECORD
    URL: String;
    Count: Integer;
    Time: Integer;
    Preference: Real;
    Child, Younger: Nodeptr
  END
```

这里: Child 和 Younger 分别表示指向子节点和兄弟节点的指针, URL 表示该节点页面的地址, Count 表示用户通过同样的路径到达该点的次数, Time 表示浏览时间的离散化表示.

4 PNT 树的构筑

4.1 预处理 Web 日志生成用户事务数据库

首先对 Web 日志进行数据清洗: 删除 Web 日志中与数据挖掘不相关的冗余项, 如 URL 的后缀为 GIF, JPEG, CGI 等的记录; 在此基础上进行用户识别: 由于防火墙和代理服务器等原因不可能准确地识别用户, 只能尽可能地使用比较合理的启发式规则, 如将用户的 IP 地址与 Agent 绑定来鉴别一个用户; 然后生成用户事务: 利用一定规则得到用户连续请求的访问页面序列, 如利用时间最大间隔法进行用户事务鉴别[8], 即如果页面请求之间的时间间隔达到一定的值(在一个页面上停留了一定的时间), 就认为开始了一个新的用户事务; 最后将所有的用户事务存储在用户浏览事务数据库中.

4.2 生成 PNT 树

在对用户浏览事务数据库的扫描过程中, 对数据库中的属性进行数据归类, 以形成概念汇聚点. 对数据库中记录的属性字段按归类方式进行抽象, 建立具有一定层次的树状结构 —— PNT 树.

算法 1 构筑 PNT 树

输入: 用户访问数据库 D_s; 输出: PNT 树

创建一个新节点 R 为根节点, 其 Count 值为 D_s 中用户事务的个数

目前的节点指向 R

For I = 1 to D_s 中用户事务的个数

目前的节点指向树的根节点

n = D_s 中第 i 个用户事务 S_i 的大小

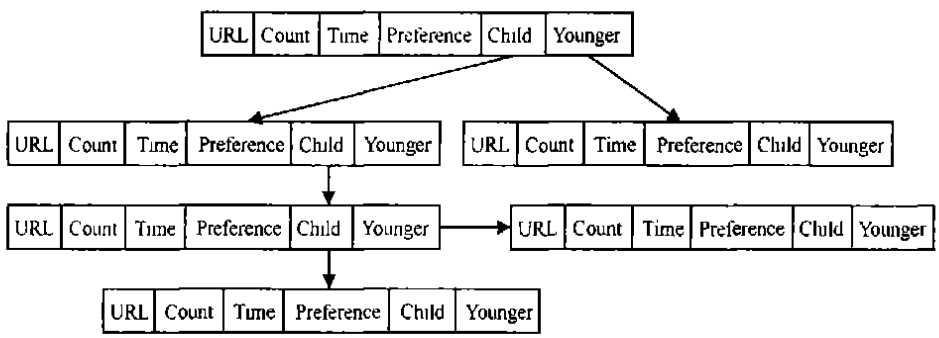

图 1 PNT 树结构

For j = 1 to nD。

If 目前节点有一个子节点的URL 等于 s_{ij} 的URL, Then

该子节点的Count 值增 1

Time 值增加 s_{ij} 的离散化时间

Preference = (该子节点的Count/所有子节点的Count)·(该子节点的Time/所有子节点的Time), 并使目前的节点指向该子节点

Else

创建一个新子节点(s_{ij} 的URL: 1: ((1/所有子节点的Count)·(s_{ij} 的Time/所有子节点的Time)))

使目前的节点指向新节点

EndIf

EndFor

EndFor

5　利用 PNT 树挖掘兴趣路径

遍历整个PNT 树, 在树上的支持度和偏爱度都大于阈值的分支就是一个用户浏览偏爱路径

算法 2　利用 PNT 树的用户浏览偏爱路径挖掘算法

输入: PNT 树以及支持度和偏爱度阈值; 输出: 用户浏览偏爱路径 设 T 是一个指向多叉树根节点的指针, A 是一个辅助堆栈

Step1:

置堆栈A 为空

置链接变量 $P \leftarrow T$

Step2:

If P = NULL Then

Goto Step4

EndIf

Step3:

访问节点 P

If P Preference ≥ 偏爱度阈值 and P Count ≥ 支持度阈值, Then

将 P 点标记为偏爱节点压入堆栈A 中

Else

将 P 点标记为非偏爱节点压入堆栈A 中

$P \leftarrow$Child(P)

Goto Step2

EndIf

Step4:

If A 〈〉NULL Then

将堆栈A 中的节点X 弹出

If X 是偏爱节点, Then

将该节点和堆栈A 后面是偏爱的节点以及这些节点前一位不是偏爱节点的节点按序输出到候选偏爱路径集合中

Else

$P \leftarrow$Younger(P)

Goto Step2

EndIf

Else

算法终止(已遍历完毕)

EndIf

Step5:

对候选偏爱路径集合中的序列进行匹配

If 一个序列S 没有包含在集合里的其他任何序列中, Then

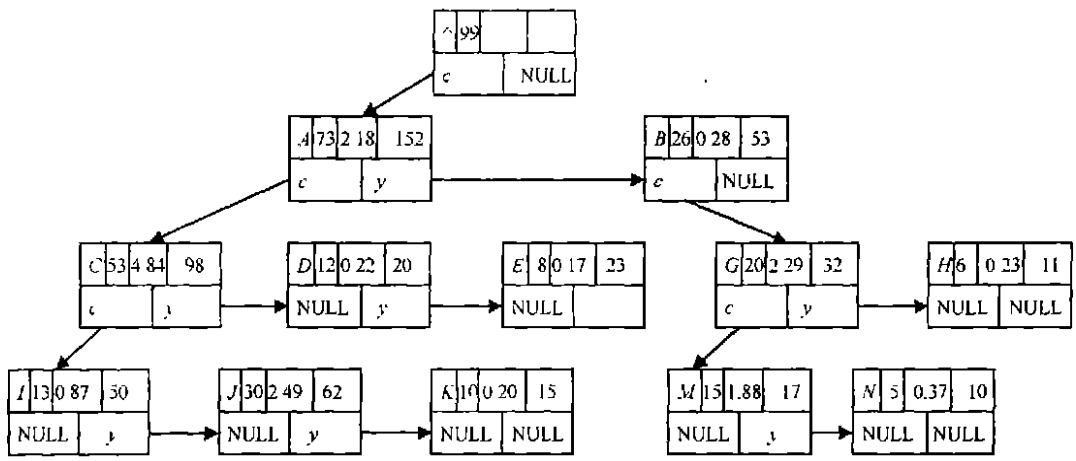

图 2　PNT 树挖掘实例

序列 S 就是用户浏览偏爱路径

End If

如图 2 所示, 将偏爱度阈值设为 1, 支持度阈值设为 10, 进行用户浏览偏爱路径挖掘, 可以找到偏爱路径为^ ACJ; BGM.

6 挖掘结果的表示

为便于用户理解和评估, 必须对挖掘结果进行相应表达方式转换, 以直观明了的方式显示给用户. 对于用户浏览路径这种序列模式的表达, 常用的传统的文本方式和表格方式都不直接, 会造成不直观、用户理解困难等缺憾. 可直接用挖掘时采用的树结构表示, 其中挖掘出来的偏爱路径用粗线表示, 可有效地避免上述缺点, 系统本身也不需额外的开销.

7 性能分析

首先测试算法的准确性. 本文将文献[1]算法和本文算法挖掘出来的兴趣路径应用于所开发的教育网站上, 发现利用文献[1]算法挖掘出来的频繁路径改善网站结构后, 用户的平均停留时间比先前的多 13%, 而利用本文算法挖掘出来的偏爱路径改善网站结构后, 用户的平均停留时间比先前的多 21%. 用户在网站的平均停留时间越长, 说明用户对网站的兴趣越高, 这便从实践中证明了本文算法比文献[1]算法能更准确地挖掘出用户浏览的兴趣路径.

然后测试算法的可扩展性. 作者用 Microsoft Visual C++ 6.0 语言实现了本文算法和文献[1]算法(MF+SS), 在内存为 128M 的赛杨 450 计算机上进行了性能测试. 将下载的日志文件进行分割以形成大小为 500 K, 1M, 1 500 K, 2M 和 2 500 K 的 5 个测试用例, 计算执行 CPU 时间, 得到图 3 所示曲线.

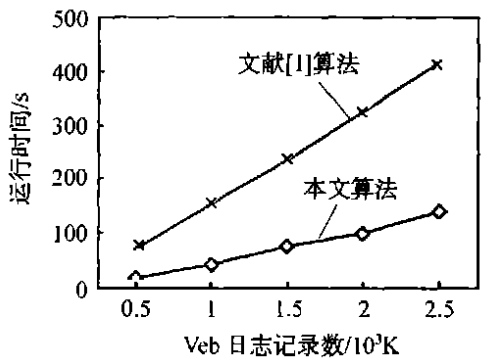

图 3 算法执行时间比较曲线

由图 3 可以看到, 本文算法的执行时间小于文献[1]的执行时间, 而且随着日志数量的上升, 本文算法比文献[1]算法的执行时间的增幅小, 增长趋势较为缓慢. 这是由于文献[1]算法需要多次扫描用户事务数据库, 需要大量的 I/O, 浪费了大量的运行时间, 而本文算法仅需一次. 以上说明本文算法的可扩展性比较好.

8 结 论

本文利用所提出的一个衡量 Web 浏览兴趣的阈值——偏爱度概念, 设计了基于用户浏览偏爱树的偏爱路径挖掘算法. 首先用 Web 日志构筑用户浏览偏爱树(PNT); 然后在此树的基础上挖掘用户浏览偏爱路径. 该算法可以准确地挖掘用户浏览偏爱路径, 而且可扩展性很好.

参考文献(References):

[1] Chen M S, Park J S, Yu P S. Data mining for path traversal patterns in a Web environment[A]. *Proc of the 16th Int Conf on Distributed Computing Systems*[C]. Hong Kong: IEEE cs Press, 1996. 385-392.

[2] Borges J, Levene M. Mining association rules in hypertext databases[A]. *Proc the 4th Int'l Conf on Knowledge Discovery and Data Mining*[C]. Menlo Park: AAAI Press, 1998. 149-153.

[3] Yang D L, Yang S H, Hong M C. An efficient web mining for session path patterns[A]. *Proc of Int Computer Symposium 2000, Workshop on Software Engineering and Database Systems*[C]. Taiwan, 2000. 107-113.

[4] Nanopoulos A, Manolopoulos Y. Finding generalized path patterns for web log data ining[A]. *Proc of East-European Conference on Advances in Databases and Information Systems (ADBIS'00)*[C]. Prague: Springer Verlag, 2000. 215-228.

[5] Fayyad U M, Piatetsky-Shapir G, Smyth. *Advances in Knowledge Discovery and Data Mining*[M]. Menlo Park: AAAI Press, 1996. 1-34.

[6] Myra S, Lukas C F. WUM: A tool for Web utilization analysis[A]. *Proc of EDBT Workshop WebDB'98*[C]. Valencia: Springer-Verlag, 1998. 109-115.

[7] Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation[A]. *Proc 2000 ACM-SIGMOD Int Conf on Management of Data (SIGMOD'00)*[C]. Dallas, 2000. 1-12.

[8] Srivastava J, Cooley R, Deshpande M, et al. Web usage mining: Discovery and applications of usage patterns from Web data[J]. *SIGKDD Explorations*, 2000, 1(2): 12-23.