一种隐式关联页面的挖掘方法①

徐 昊, 谢文阁

(辽宁工业大学 电子与信息工程学院, 锦州 121001)

摘 要: 点击流数据是分析互联网用户心理倾向的关键, 用户感兴趣的页组关联就隐藏于 WEB 日志之中. 网站页面间的隐式关联可以通过分析点击流数据实现. 给出了一种挖掘关联页面的方法. 关联页面发现算法采用了一种类似于 Apriori 的模型. 算法克服了前人关联页面算法的一些缺点, 能够更好地适应复杂的互联网环境.

关键词: Web 日志; 隐式关联页面; 点击流数据

Mining Method of Implied Association Page

XU Hao, XIE Wen-Ge

(School of Electronic and Information Engineering, Liaoning University of Technology, Jinzhou 121001, China)

Abstract: The Clickstream data is the key to the analysis of Internet users psychological tendency, and the association of the user interesting pages group is hidden in the WEB log. Implied association between Web pages can be achieved by analyzing the click stream data. This paper puts forward a method of mining association page. Associated page searching algorithm adopted a model similarly to the Apriori. This algorithm overcomes some shortcomings of predecessors' association page algorithm can better adapt to the complex Internet environment.

Key words: Web log; implied association page; click stream data

1 引言

以辩证的观点来看, 事物间是存在着普遍联系的. 对于 Web 网站来说, 网站管理员通过添加链接使得整个网站的页面联系起来, 这是一种显式的网页间的联系. 然而, 隐藏在 Web 日志中的网页间的隐式的链接关系却更能够真实的反应网页间的用户兴趣相关性. 就如同著名的 "啤酒与尿布"的案例一样, 大量的隐性页面关联就隐藏在 Web 日志之中的点击流序列之中. 杨怡玲[1]等通过对页面的分析, 将支持度计算转化为范化内容链接比的计算, 该算法后期能够快速收敛, 但前期计算量很大; 鲍钰[2]提出了一中隐式页面关联规则发现模型, 能够有效的发现关联页面, 但是其目标页面的确定方法有些片面, 且最终只能够发现到频繁 2 项集, 适应性不好. 本文就是在这种背景之下, 综合了上述论文的优点,给出了一种挖掘关联页面的方法.

2 Web日志的预处理

点击流的数据挖掘通常有两种: 一种是把点击流看作具有进化特性的数据流, 在只进行单次或较少次数扫描情况下, 以较少内存开销, 实时完成数据挖掘. 第二种是把点击流看作静态数据(主要是存储于 Web 服务器之中的日志数据), 以牺牲一定处理速度为代价, 获得更加准确的数据挖掘结果[3]. Web 日志文件一般有三种: 服务器端日志文件、代理服务器端日志文件、客户端日志文件. 一般情况下, 我们通常使用的是第一种文件格式. 原始日志文件数据并不能直接用来进行 Web 挖掘,通常需要将其进行日志预处理操作.

2.1 数据清洗

目前常见的 Web 日志格式有两种: 一种是 apache 的 NSCA,另外一种是微软 IIS 的 W3C. 这两种日志格式都包括几个主要的部分: 访问主机, 访问时间, 请求页面, 状态码等[4]. 因此, 这里我们要删除状态码不

① 收稿时间:2014-01-05;收到修改稿时间:2014-02-28

为 200 且请求类型不为 GET 的点击流纪录. 另外, 我们还需要删除用户在浏览网页时随着网页而下载的文件(如用户访问网站时, 随着页面一起下载网站 LOGO 图片等), 这些日志对于分析并无用处, 属于冗余信息.

2.2 用户识别与会话识别

用户识别就是将日志文件按照实际的用户进行分组. 在实际的应用中, 通常采用 cookie 标记的方式来确定用户, 即使用户改变了 IP 地址, 只要浏览器未变, 其 cookie 标记就仍是唯一的. 但是用户的浏览器可以自行禁用 cookie, 使得利用 cookie 标记用户变得无效. 为了简单起见, 本文在实验中采用最简单的识别方式, 即如果主机 IP 地址相同, 则认为是同一用户. 会话识别就是将用户的访问页面按照每次的会话而进行分组. 会话识别的主要作用就是确定某一用户在一次持续的页面浏览中所浏览的页面序列. 常用的会话识别方法是设置时间阈值, 相邻两次的页面访问超过阈值则认为是一次新的会话. 通常将这个阈值设定为 30 分钟. 通常在实际的应用中, 网站的管理者并不十分感兴趣究竟是哪一个具体的页面与另外的某一具体页面存在关联关系, 而是关心某一类页面与另一类页面的关联关系. 因此, 根据实际情况对页面进行恰当分类也是十分重要的. 例如可以按照页面内容进行分类(时政, 体育, 军事, 娱乐等).

2.3 确定访问的目标页

所谓的目标页就是用户最终想要访问的页面. 对于普通的用户来说, 其访问网站必然是有主页开始逐步地深入到其所感兴趣的页面. 但是在这个过程中, 用户的一系列访问都被记录在了点击流日志中, 如果在这些点击流中识别出用户的兴趣页面, 删除无关页面的干扰, 将大大减少算法的执行时间. 在文献 2 中, 作者提出了一种目标页面确定算法. 它的主要思想是, 在访问过程中发生回溯的页面即为目标页面. 但是该算法依靠用户的点击回溯来确定目标页面, 结果并不令人信服, 而且在复杂的互联网环境下, 其有效性也值得怀疑. 在文献 4 中提到了一般用户可以忍受的页面响应时间为 30 秒. 我们据此可以推断出, 普通用户对于不感兴趣的页面的关注度不会超过 30 秒的时间. 因此, 只要是在某一页面停留时间超过 30 秒, 我们就认为这是一个目标页面. 具体描述如算法 1 所示:

输入: 会话集合 S={P1, P2, ……}
输出: 目标页面集合 A={P1′, P2′, ……}
A=空集;
对于 会话集合 S
If (i=1) add P1 to A;
For(i=2;S≠?;i++)
  If(Pi.time-Pi-1.time>30s)
    Add Pito A;

算法 1

3 关联页面挖掘算法

我们可以通过关联规则的频繁项集的发现来实现关联页面的发现. 但是传统关联规则只能够发现没有顺序的页面集合, 对于有时序的点击流序列来说并不适合. 本文所述的算法能够发掘具有时序的点击流序列. 具体步骤为: 首先计算所有页面的计数, 将所有大于支持度的页面形成 L1. 在 L1 的基础上, 将 L1 的所有页面两两组合, 形成 C2. 由于用户访问页面时存在时序, 因此, 在形成 C2 时, 若 L1 中包含 A, B 两个页面. 那么 AB, BA 都包括在 C2 之中, 它们是不同的. 形成C2之后, 扫描会话, 删除不存在于会话中的元素, 形成 C2′. 计算 C2′个元素的元素计数, 大于支持度的元素形成 L2. 从 L2 开始, 对于所有的在 Li-1 中的元素, 设 P1={x1,x2, …,xi-1}, P2={y1,y2, …,yi-1}. 如果 (x2=y1 and …and xi-1=yi-2), 那么就形成新的页面 {x1,x2, …,xi-1,yi-1}, 并加入到 Ci. 扫描会话, 删除 Ci 中不存在于会话中的元素, 形成 Ci′,计算 Ci′中所有元素的计数, 大于支持度的形成 Li. 如此递归的计算, 直到 Li 为空集为止. 最终结果是所有集合 L 的并集. 具体描述如算法 2 所示, 算法流程如图 1 所示:

L_1={频繁 1 项集}
C_2={从 L_1 中两两结合产生}
For 所有会话
C_2'={删除在 C_2 中不存在于会话中的元素}
For 所有的 c ∈C_2'
c.count++
L_2={ c ∈C_2'|c.count≥minsup}
for (i=3; L_{i-1}≠空集;i++) {
对于所有的在 L_{i-1} 中存在的页面
令 P1={x1,…,xi-1}
令 P2={y1,…,yi-1}
if (x2=y1 and …and xi-1=yi-2) {
产生新页面 N={x1,…,xi-1,yi-1};

Add N to C_i

}

C_i'={删除在 C_i 中不存在于会话中的元素}

For 所有的 c ∈Ci′

c.count++

L_i={ c ∈C_i'|c.count≥minsup}

}

关联页面 L=$\bigcup_1^k Li$

算法 2

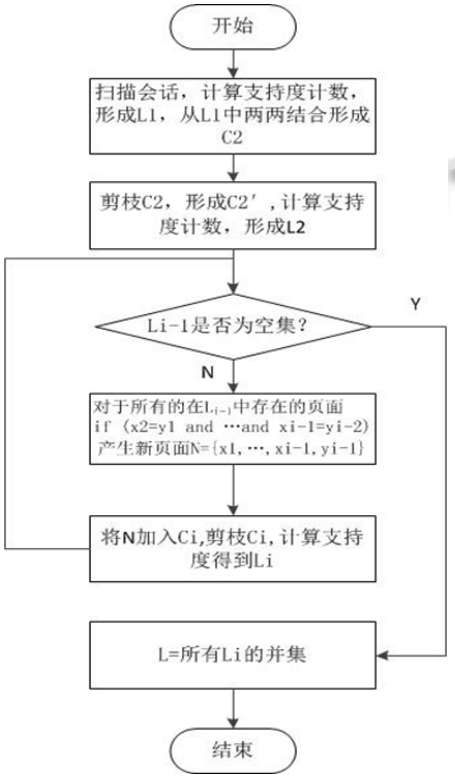

图 1

4 实验结果与分析

实验数据为 NASA-HTTP, 其格式为 (host, timestamp, request, HTTP reply code, bytes in the reply), 格式含义为, host 代表请求主机, timestamp 代表请求时间, request 代表请求页面, HTTP reply code 代表返回代码, bytes in the reply 代表字节返回数. 为了便于统计, 实验中将页面分为了 7 个类别: /的页面访问被划为 A; ksc 页面被划为 B; images 被划为 C; shuttle 被划为 D; history 被划为 E,icons 被划为 F, 其余被划为 G. 为简化起见, 本次实验抽取了数据源中的前 500 条数据进行计算. 设置阈值为 0.05, 通过最终计算, 得到了 L={BC, CD, DC, DF, FD, CDC, DFD}. 挖掘结果如表 1

表 1

集合	结果
频繁 1 项集	B,C,D,E,F
频繁 2 项集	BC, CD, DC, DF, FD
频繁 3 项集	CDC, DFD
最终结果 L	BC, CD, DC, DF, FD, CDC, DFD

通过这个结果, 我们就可以了解到, 用户在访问 B 类页面时往往会跳转到 C 类页面; 同时, 用户也往往会先访问 C, 再访问 D, 再跳转回来访问 C. 其余结果也可按照上述类比分析.

5 结语

在当前互联网被广泛使用的背景下, 如何有效的发掘用户的兴趣成为各类网站共同关注的问题. 本文所采用的方法既不像文献 1 那样需要大量计算页面内容(范化内容链接比), 也不像文献 2 只能够挖掘到频繁 2 项集. 因此, 本文所述算法能够更好地适应复杂的互联网环境.

参考文献

1 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.软件学报,2002,13(3):467–469.

2 鲍钰.WEB 日志挖掘及其应用研究[博士学位论文].上海:华东师范大学,2009.

3 马超,沈微.基于闭合有间隔频繁子序列的点击流聚类.计算机工程,2010,36(23):72–75.

4 Kimball R, Merz R.张丽萍,等译.Web 数据仓库构建指南.北京:清华大学出版社,2005.

5 陈燕.数据挖掘技术与应用.北京:清华大学出版社,2011.

6 姚家奕.数据仓库与数据挖掘技术原理及应用.北京:电子工业出版社,2009.