基于 s-Tree 算法的个性化推荐服务研究*)

宁小红　余森森
(浙江师范大学　杭州 310012)

摘　要　本文提出了基于关联规则的挖掘最大频繁访问的新算法——s-Tree 算法，并以此去分析用户的访问模式，挖掘出特定用户访问模式和浏览偏爱路径信息，进而优化站点结构，为用户提供“一对一”个性化的 Web 页面访问预测及内容推荐。

关键词　Web 使用挖掘，个性化服务，推荐

Study on s-Tree Algorithm for Personalized Recommendation

NING Xiao-Hong　YU Sen-Sen
(Zhejiang Normal University, Hangzhou 310012)

Abstract　This article proposes a new algorithm based on the connection rule excavation most greatly frequent visit——s-Tree algorithm, and analyzes the user by this visit pattern, excavates the specific user visit pattern and the browsing is partial to routing information, then optimizes the stand structure, provides “one to one” the personalized Web page visit forecast and the content recommendation.

Keywords　Web Usage Mining(WUM), Personalization, Recommend

1　引言

随着 Web 技术的迅速发展，Internet 上信息成指数级增长。然而，信息过载和资源迷向已经成为制约人们高效使用 Internet 信息的瓶颈，可见传统的信息服务技术已经远远滞后于方便、快捷的信息发布技术，面对激烈的竞争，网站需要引入一种能够根据用户的特点自动组织和调整信息的服务模式，即信息服务方式从传统的“一对多”传播发展到“一对一”的个性化服务方式[1]。

目前，基于 Web 使用挖掘(Web Usage Mining, WUM)的个性化服务技术发展迅速，它利用 Web 挖掘的方法抽取用户感兴趣的潜在有用模式与信息，当用户在访问网站某些特定信息时，能针对不同用户访问特点提供不同服务策略和服务内容的智能个性化服务。这些智能个性化服务可大大缩短用户在网络上的访问延迟，提高信息服务的质量。

2　基于 Web 挖掘的个性化推荐服务系统分析

2.1　基于 Web 挖掘的个性化服务

Web 个性化服务分为两个阶段：首先是使用 Web 挖掘技术根据 Web 日志文件等数据信息获取的用户兴趣爱好、Web 访问模式等个性化信息，根据用户访问的 Web 页面内容及用户群访问的相似性，利用基于路径分析技术和关联规则的方法进行 Web 页面和用户的分类、发现频繁访问路径并对访问路径进行优化，从而改进站点结构等；然后是通过挖掘到的用户个性化信息将用户访问模式需求同 Web 结合，通过用户群的相似性进行 Web 页面访问预测及内容推荐(Recommending)。

2.2　Web 个性化推荐服务的种类

Web 个性化服务系统是通过提供的多种功能实现系统目标的。从目前实现的角度可以将个性化服务系统分为 4 种：记忆型、向导型、定制服务型和工作任务辅助支持型。

记忆型通过在系统中记录和存储使用者信息，提供诸如问候、书签及分配权限等服务，它是 4 种类型中最简单的一种；

向导型通过引导用户快速、高效地获取用户所寻求的信息，提供个性化超链接推荐、页面向导等服务[2]，比如 YAHOO 分类搜索引擎；

定制服务型可以根据用户的知识、兴趣以及爱好对 Web 页面内容、结构和布局进行个性化设定，比如提供个性化的网页布局、颜色或位置信息的自动适应，内容定制、超链接定制，价格和支付服务等；

工作任务辅助支持型类似于个人助理，它可以帮助用户发送邮件、下载软件等个性化差使、进行个性化查询、谈判等，是最复杂的一种服务类型。

3　基于 Web 使用挖掘的个性化站点的结构

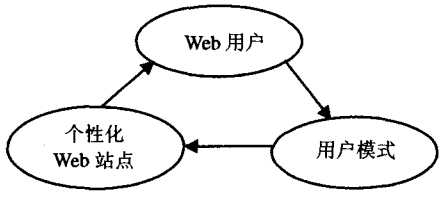

图 1　个性化站点示意图

个性化 Web 站点实质上就是一种以用户需求为中心的 Web 站点。首先，不同的 Web 用户通过各种途径访问 Web

*)2006 年度浙江省教育厅科研项目立项：“基于 Web 使用挖掘的个性化推荐系统的研发”，项目编号 20060440。宁小红　讲师，主要研究方向：Web 挖掘、入侵检测；余森森　副教授，主要研究方向：数据挖掘、现代教育技术。

站点；其次，系统学习用户的特性，创建用户访问模型；最后系统根据所得到的知识调整服务，以适应不同用户的个性化需求。图1是个性化Web站点的示意图。

图2是基于使用挖掘的个性化站点的结构。用户访问Web服务器，留下Web使用信息，个性化Web站点收集站点文件和Web使用信息，通过对数据预处理得到合适的数据，然后通过Web使用挖掘的算法挖掘得到用户模式库，这个过程离线实现。当用户访问改站点时，用户访问推荐模块对用户信息进行识别，并且到模式库中进行匹配比对，实时为用户产生推荐页面集合，和其他响应信息一起返回给用户。

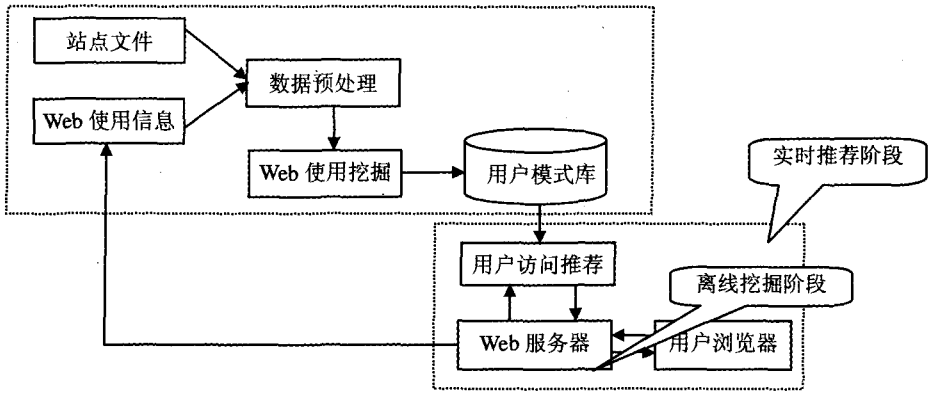

图2 基于Web使用挖掘的个性化站点的结构

4 Web使用挖掘的实现过程及推荐

Web使用挖掘的输入数据(数据源)有：Web服务器日志(访问日志、引用日志和代理日志)、Web站点的拓扑结构和站点文件、用户注册信息、用户调查信息、Cookies以及可选的与网站相关的数据库数据，这些数据一般都存储在DBMS中[3]。

Web使用挖掘通常要经过下面三个阶段：(1)数据预处理阶段；(2)挖掘阶段；(3)模式分析与在线推荐。其中，数据预处理和日志挖掘算法是Web日志挖掘中的关键技术。数据预处理结果的质量关系到使用挖掘过程和模式分析过程的质量，而挖掘算法的选择与改进是保证挖掘成功的重要因素。所以，在Web使用挖掘技术的研究中也侧重于这两方面，如图3所示。

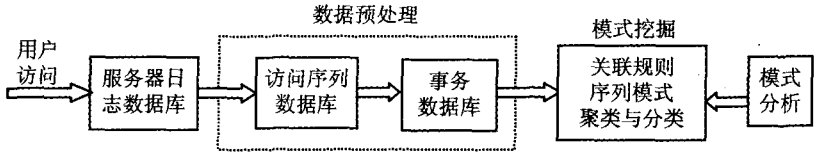

图3 Web使用挖掘的实现过程

4.1 数据预处理

对原始Web日志文件中的数据进行提取、分解、合并，最后转化为适合下一步进行数据挖掘的数据格式，并保存到关系数据库表或数据仓库中。其主要任务是把Web日志转化为适合数据挖掘的可靠的精确的数据，解决由于日志的不精确所带来的问题。主要包括数据清理(Data Cleaning)、会话识别(Session Identification)、浏览页识别(Pageview Identification)以及事务识别(Path Completion)。由于存在客户端缓存和代理服务器端缓存，要求较高的个性化服务需要进行路径完善(Path Completion)。如果网站没有使用Cookies技术或内嵌的会话标示技术，还需要进行用户识别(User Identification)。经过数据预处理过后的结果是用户事务。

4.2 模式发现

利用WUM挖掘算法是对预处理过程得到的用户事务进行挖掘。挖掘出有效的、新颖的、潜在的、有用的及最终可以理解的信息和知识。在基于Web使用挖掘的推荐系统中，常用的技术有路径分析技术、关联规则、序列模式、分类聚类技术等。其中利用路径分析和关联规则挖掘技术来预定用户的浏览模式在网页个性化研究领域引起多方面的关注[5]。

通过对客户访问路径模式的发现，可以得到客户频繁访问的路径，即热门路径。例如，通过对路径模式发现结果分析可知：如果 $tp=\{d\}$ 是长度为0的热门路径，则说明页面 d 的访问频率大，可以在该页面上放置广告、新闻等内容。站点的主页一般都是热门页面。此外，若某页面是多个热门路径的“交点”，则该页面位置亦很重要。利用路径分析技术进行Web挖掘就是要从图中找出最频繁的路径访问模式或大的参引访问序列。

关联规则是指发现用户会话中经常被用户一起访问的页面集合，这些页面之间并没有顺序关系。如果关联规则中的页面之间没有超链接，则这是一个我们感兴趣的关联规则。如55%的用户访问Web页面/Products/DigitalCamera时，也访问了/Products/Computer，则当用户访问DigitalCamera页面时，可以考虑将Computer页面推荐给他。挖掘关联规则通常使用Apriori算法或其变形算法。

4.3 模式分析与在线推荐

模式分析与在线推荐是整个Web使用模式挖掘的最后一个环节。其目的是根据实际应用，通过选择和观察，把发现的规则、模式和统计值转换为知识，再经过模式分析得到有价值的模式，即我们感兴趣的规则和模式。实现这一点需要一些技术，如：可以像SQL的知识查询机制；或把Web使用数据装入数据仓库，以便执行OLAP操作；也可以采用可视化技术，使得数据中的总体模式或趋势变得更加直观[7]。

推荐引擎是基于挖掘的个性化服务体系结构在线部分的关键，其主要任务就是根据当前的用户会话产生实时的推荐

集，推荐集可能包括超链接、广告、文本和商品等，推荐对象以超链接的形式添加到用户当前会话的最后一个浏览页中。一般而言，用户当前会话中的最后几个浏览页最能反映用户当前的行为特征。我们可以在用户当前会话中创建一个长度为 j 的滑动窗口，滑动窗口中动态保存用户当前会话中最后的 j 个浏览页，从而可以用这个滑动窗口代表用户的当前会话。

在体系结构中，总体使用特征表示为浏览页空间上的 n 维向量；同时，用户当前会话也可以转换为浏览页空间上的 n 维向量。即如果 C 表示一个总体使用特征，则 C 可以表示为

$$C=\{w_1^C, w_2^C, \cdots, w_n^C\}$$

其中 $w_k^C=\begin{cases} weight(Page,C), \text{if } Page\in C \\ 0, else \end{cases}$

用户当前会话 S 为：$S=\{s_1, s_2, \cdots, s_n\}$，

其中 $S_t=1$(if Page accessed by user)；$S_t=0$(else)。

从而给定一个特征 C(总体使用特征)和用户当前会话 S，计算浏览页 Page 推荐系数 REC(S,Page)：

$$REC(S,Page)=\sqrt{weitht(Page,C)*match(S,C)}$$

如果浏览页 Page 在用户当前会话 S 中，则其推荐系数 REC(S,Page)赋值为 0。给定用户当前会话 S，总体使用特征集 UP 和最小推荐系数阈值 r，基于总体使用特征的推荐集 UREC(S)的计算方法如下：

$$UREC(S)=\{W_i^C \mid C\in UP, \text{ and } REC(S,W_i^C)\geqslant r\}$$

把 UREC(S)中推荐系数最大的浏览页作为最终的推荐集。

5 Web 使用挖掘的算法

本文提出一种基于用户会话有向树表示和内容页面优先的支持度计算方法的最大频繁访问模式——s-Tree 算法。该算法采用 Apriori 思想，通过连接频繁模式和分层频繁弧生成候选项集，同时采取预剪枝策略降低算法的开销。

5.1 s-Tree 算法

s-Tree 算法采用 Apriori 思想，通过逐层搜索的迭代方法，用 k-阶频繁树(包含 k 条弧的频繁树)加上一条频繁弧生成(k+1)-阶候选集，然后通过剪枝，频繁度计数，得到所有(k+1)-阶频繁图，最后找出用户的最大频繁访问模式树。

算法 s-Tree(D,min-sup)
输入 用户会话树数据库 D;
最小支持度阈值 min-sup
输出 最大频繁访问模式树集合 S
扫描数据库，获得频繁弧集合 $Arc_1, Arc_2, \cdots, Arc_n$;
再次扫描事务数据库 D，对每个会话树进行预剪枝，得到新数据库 D' 的大小;
$i\leftarrow 1$; $F_1\leftarrow Arc_1$;
While($F_i\neq\emptyset$)
Do C_{i+1}=Generation(F_i) //连接生成候选频繁树
若 $C_{i+1}=\emptyset$，算法终止返回 S;
For 每个候选子树 $t_{i+1}\in C_{i+1}$
t_{i+1}.Count$\leftarrow$0
for 数据库 D' 中每个会话树 T
if t_{i+1}包含在 T 中
t_{i+1}.Count++;
$F_{i+1}\leftarrow\{t_{i+1}\in C_{i+1} \mid \sup(t_{i+1})\geqslant$min-sup$\}$
//这里的支持度计算考虑了内容页面的影响因子后的支持度删除 S 中被 F_{i+1} 中项包含的子树
$S\leftarrow S\cup F_{i+1}$
$i\leftarrow i+1$
Return S

考虑所挖掘的会话树都具有层次结构，我们摒弃了 Apriori 算法直接连接两个频繁模式树的做法，而是采取给频繁模式树增加一条频繁弧的方法来生成候选的模式树。判断两个模式是否可连接，是一个非常耗时的操作。这样在初始时，首次扫描数据库，确定频繁弧集合，且将频繁弧集合按照层数分类。分别用记号 Arc_i 表示层数为 i 的频繁弧集合。于是初始的频繁 1-项集就为 Arc_1。然后再次扫描数据库对会话树进行剪枝。对数据库中的每个会话树，若它包含非频繁的弧，就对它进行剪枝，剪掉从非频繁弧到叶节点这段路径。如图 4 所示的会话树，若弧(C,H)是非频繁弧，则将从 C 到 I 的路径给剪掉，经过这样的预剪枝后，数据库明显减小，而不会影响最后的挖掘结果。

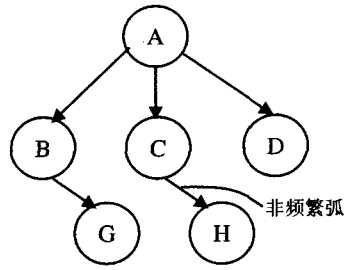

图 4 有向树表示

用 C_i 来表示具有 i 条弧的候选项集，用 F_i 表示具有 i 条弧的频繁项集。在连接步中，通过给 F_i 中频繁树增加相应的频繁弧生成候选项集 C_{i+1}。在这里 s-Tree 算法采取两种策略降低候选项集生成的开销。

第一，若频繁树的层数为 m，则所增加的频繁弧应在频繁弧集合 C_{m-1} 和 C_m 中选取。也就是说算法只会在第 $m-1$ 层和 m 层顶点上增加相应的弧。如图 5，算法只可能在处于第 2 层 B,C,D 和第 3 层的点 G,H 上增加一条弧[8]。

假设有一个 n 阶频繁树 t_n，其层数为 $m(m\geqslant 2)$，若它在 $m-1$ 层和 m 层之外的点上增加一弧后也是频繁树，记为 t_{n+1}，层数也为 m，则根据 Apriori 性质，t_{n+1} 去掉一条连接 $m-1$ 层和 m 层的弧 l 后仍应为频繁树，记为 t_n^*，其层数为 m 或 $m-1$，这样 $n+1$ 阶树 t_{n+1} 就可以通过频繁树 t_n^* 在第 $m-1$ 层某点上增加弧 l 来完成，所以算法并不会丢掉任何可能的候选树。

这样通过在频繁 k-阶树上有选择地增加弧，可以有效地减少生成的(k+1)-阶候选树的数目，大大降低算法的时间开销。

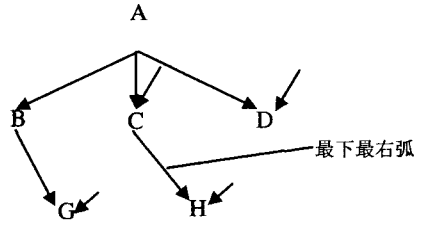

图 5 增加弧位置示意图

第二，会话树同一节点的子节点之间按照字母序排列，所以在连接弧的时候，算法只考虑那些字母序在会话树最下最右的弧后的弧。例如图 2 所代表的会话，弧(C,H)为其最下最右的弧，所以算法只会在图中箭头所标记的 C,D,G 和 H 节点上增加弧，并且只会在 C_i 中选择弧连接到第 $i(i>0)$ 层相应的点上。这不会丢掉任何可能的候选项，而且可减少重复生成候选图的数目，进一步降低算法的时间开销。

生成候选集后还要进一步对候选集做剪枝，删除那些不可能是频繁树的项。在剪枝阶段，算法根据 Apriori 性质，对每个候选树，看其所有子会话树是否是频繁树，若不是，则将其从候选项集中删除。假设 c_{i+1} 是通过频繁图 t_n 弧 l 所得到

的候选树，由于树 t_n 所有子会话树都是频繁树，因此算法只验证 c_{i+1} 的包含弧 l 的所有子会话树是否为频繁图，若有一个不是，则将 c_{i+1} 从候选集中删除。

对于 C_i 中的候选树，只有达到了最小支持度阈值才算是最终的频繁树。在计数过程中，算法首先找出所有的弧数大于 i 的事务，若其数目小于 min-sup · |D|，则 C_i 中所有的候选树都不是频繁树，算法终止。否则逐个扫描看候选项是否包含其中，计算每个候选项的支持度，将候选项集中不满足最小支持度阈值的删除，从而得到 F_{i+1}。

5.2 算法实验比较

s-Tree 算法用 C++ 标准库实现，在 CPU 为 Intel Pentium 4 1.8GHz，256M 的 RAM 上装有 Windows XP Professional 的 PC 机上运行，测试数据集 BMS-WebView-1 和 BMS-WebView-2 从 KDD-CUP 2000 的主页(www. ecn. purdue. edu)上下载。它们包含几个月实际的 Web 点击流数据，其特征见表 1。首先我们将这两个数据集的形式进行处理，将会话转化成为有向树的形式。运行时间是指从数据读入到结果输出的时间。

表 1 测试数据集特征

数据库	大小	会话数目	网站网页数	最大会话访问网页数	会话平均网页数
BMS-WebView-1	2.07M	59602	497	267	2.51
BMS-WebView-2	5.01M	77512	3308	161	4.62

(1)运行时间分析

我们分别对两种数据集在不同的支持度阈值条件下做了实验，运行时间是指从数据的输入到结果的输出。其结果如图 6 所示。可以看到，随着支持度的增大，其运行时间将急剧减小。对于 BMS-WebView-1 这个数据集，在阈值 0.3%左右运行时间下降得最快，从阈值 0.25 时的 83.8s，下降到阈值为 0.33%时的 2.7s。这是因为在这个阈值点，挖掘出的频繁模式数目也是急剧变化的，挖掘出来的频繁模式也是急剧减少的。而对于 BMS-WebView-2 这个数据集，在阈值 0.33%这个点下降较快。

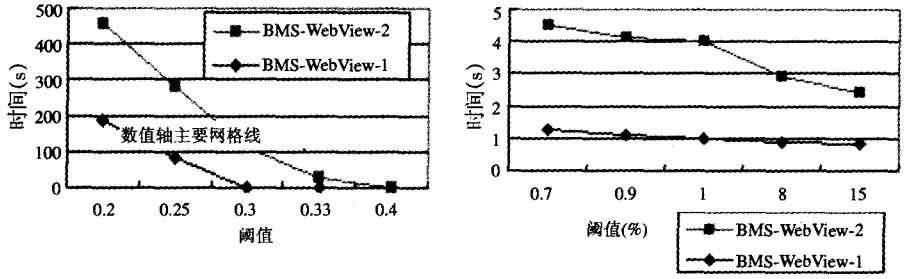

图 6 运行时间和阈值之间的关系

(2)挖掘结果分析

通过对挖掘结果的分析可以看到，在相等阈值情况下，s-Tree 算法挖掘出的频繁模式要比一般算法多。这是由于我们对包含较多内容页面的模式增加了权重，挖掘出了包含较多内容页面而被一般算法所忽视的有意义的模式。从表 2 中可以看出，我们的算法将发现较多的模式。

表 2 对数据集 BMS-WebView-1 挖掘出来的频繁模式数目(阈值 0.3%)

模式大小	一般算法	s-tree 算法	模式大小	一般算法	s-tree 算法
6	8	13	(8,10]	4	7
7	5	7	(10,20]	0	2
8	5	9	(20,)	0	0

6 用户个性化服务推荐模式

6.1 当前用户识别

有几种方式能够辨认出用户：如果用户进行了注册，每次登录到站点就会被识别出来；如果网站的 Web 服务提供 Cookie，则具有相同 Cookie 值的页面请求来自同一个用户；复杂一点的情况也可以利用主机地址和其他收集到的信息，分析日志中每条记录的请求页和引用页的 URL，然后根据挖掘的结果识别用户。

6.2 当前会话识别和模式匹配

s-Tree 算法基于 Apriori 思想，采用有向树识别并表示用户会话，挖掘结果中更多的信息；提出内容页面优先的支持度计算方法发现一般算法不能发掘的感兴趣模式。然后模式匹配为当前的用户会话从用户模式库中搜索最相似的模式。

6.3 访问推荐集合的生成

根据当前用户会话与模式库中的模式匹配结果，从中挑选为用户推荐的页面，同时要保证推荐集合中的所有的页面均没有被当前用户会话访问过。同时综合与协调多种推荐模式生成的不同推荐结果集合，并最终生成返回给用户的推荐结果集合。

6.4 结果修正

根据用户浏览行为的反馈对推荐进行修正，如果用户接受推荐，则进行巩固；否则，根据用户的反馈进行重新推荐。

总结 本文提出的 s-Tree 算法采用有向树表示用户会话，克服了现有会话表示的不足，使得挖掘的结果包含更多的信息。根据 Web 页面中不同页面的性质提出的内容页面优先的支持度计算方法，能够挖掘出一般算法所不能发掘的感兴趣的模式；算法基于 Apriori 思想，同时采用频繁模式连接分层的频繁弧和预剪枝策略，降低了算法的开销。

参 考 文 献

1 Aggarwal C, Yu P. Data Mining Techniques for Personalization [J]. IEEE Data Engineering Bulletin, 2000, 23(1): 4～9

2 Cooley R, Mobasher B, Srivastava J. Data preparation for mining world wide Web browsing patterns. Journal of Knowledge and Information Systems, 1999(1): 230～241

3 Srivastava J, Cooley R, Deshpande M, et al. Web Usage Mining: Discovery and App lications of Usage Patterns from Web Data [J]. SIGKDD Exp lorations, Newsletter of SIGKDD, 2000(1): 12～23

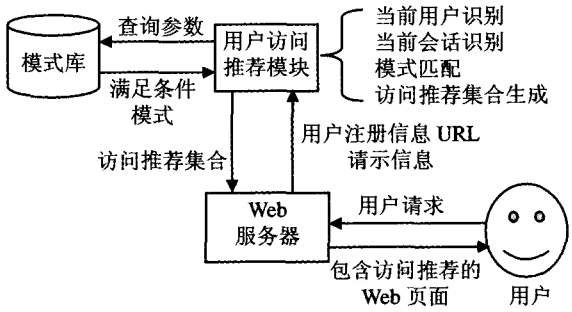

图 7　用户访问推荐示意图

4 Pierrakos D, Paliouras G. Web Usage Mining as a Tool for Personalization: A Survey [J]. Kluwer Academic Publishers, 2003. 311～372

5 张慧颖,焦霖楠. 用户访问模式聚类分析在网页推荐中的应用[J]. 计算机工程,2006,32(15):64～66

6 Han Jiawei, Kamber M. Data Mining: Concepts and Technlques [M]. Morgan Kaufmann Publishers. inc, 2001

7 Wu Song, Jin Hai, Tan Guang. Symmetrical Declustering: A Load Balancing and Fault To lerant Strategy for Clustered Video Servers [C]. In: International Conference on Computational Science and Its Applications, 2003. 199～208

8 詹宇斌,殷建平,张玲,等. 一种基于有向树挖掘 Web 日志中最大频繁访问模式的方法[J]. 计算机应用,2006,26(7):1662～1665

9 Han J, Kamber M. Data mining: concepts and techniques [M]. San Francisco: Morgan Kaufmann Publishers Inc, 2001

10 Scime A. Web mining: application and techniques [M]. Hershey: Idea Group Publishing, 2004

(上接第 203 页)

4 Aggarwal C C, et al. A Framework for Clustering Evolving Data Streams. In: Proc. of the 29th VLDB Conf., 2003. 81～92

5 Aggarwal C C, et al. A Framework for Projected Clustering of High Dimensional Data Streams. In: Proc. of the 30th VLDB Conf., 2004. 852～863

6 Park N H, Lee W S. Statistical Grid-Based Clustering over Data Streams. ACM SIGMOD Record, 2004, 33(1): 32～37

7 Lu Y, et al. A Grid-Based Clustering Algorithm for High- Dimensional Data Streams. In: Proc. of the 1st International Conference on Advanced Data Mining and Applications, 2005. In Lecture Notes in Computer Science, 2005, 3584: 824～831

8 Agrawal R, et al. Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. In: Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD'98), 1998. 94～105

9 Goil S, et al. MAFIA: Efficient and Scalable Subspace Clustering for Very Large Data Sets: [Technical Report, No. CPDC-TR-9906-010]. Center for Parallel and Distributed Computing, Department of Electrical & Computer Engineering, Northwestern University, 1999

10 Hinneburg A, Keim D A. Optimal Grid-Clustring: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering. In: Proc. of the 25th VLDB Conference, 1999. 506～517

11 Baumgartner C, et al. Subspace Selection for Clustering High-Dimensional Data. In: Proc. 4th IEEE Int Conf On Data Mining (ICDM'04), 2004. 11～18

12 Sheikholeslami G, Chatterjee S, Zhang A. WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases. In: Proc. of the 24th VLDB Conference, 1998. 428～439

13 Parsons L, Haque E, Liu H. Subspace Clustering for High Dimensional Data: A Review. ACM SIGKDD Explorations Newsletter, 2004, 6(1): 90～105

14 Liu B, Xia Y, Yu P S. Clustering Through Decision Tree Construction. In: Proc. of the Ninth International Conference on Information and Knowledge Management, 2000. 20～29

15 Aggarwal C C, et al. Fast Algorithms for Projected Clustering. In: Proc. of the 1999 ACM SIGMOD International Conference on Management of Data, 1999. 61～72

(上接第 212 页)

大,即意味提高对模式重复程度的限制,则抽取的准确性即查准率可以得到提高,但有可能丢失掉一些重复程度并不十分频繁的数据记录,使得查全率有所下降。而 MinLen 值增大,意味着具备相当结构特征的数据记录才会被抽取,提高了查准率,但查全率会因此受到一定影响。我们可以根据实际需要,通过适当调整参数来取得较好的抽取效果。

结束语　本文描述了一种基于重复模式的自动抽取网页信息的方法。根据页面的 HTML tag 序列构造后缀树,利用后缀树的优越性质取得序列中所有的重复模式以及对应的实例;应用相关规则,对候选模式进行过滤和精化,最后保留合理的模式,并从其实例中抽取网页数据记录的信息。实验结果表明,该方法是有效可行的。目前,我们正致力于进一步改善抽取的准确性,同时对系统的性能进行优化。

参 考 文 献

1 Laender A, Ribeiro-Neto B, Silva A, et al. A brief survey of Web data extraction tools. SIGMOD Record, 2002, 31(2)

2 Arasu A, Garcia-Molina H. Extracting Structured Data from Web Pages. SIGMOD-03, 2003

3 Chang C H, Lui S L. IEPAD: Information extraction based on pattern discovery. WWW-10, 2001

4 Embley D W, Jiang Y, Ng Y K. Record-Boundary Discovery in Web Documents. In: Proc. SIGMOD'99, 1999

5 McCreight E. A space-economical suffix tree construction algorithm. Journal of the ACM, 1976, 23: 262～272

6 Ukkonen E. On-line construction of suffix trees. Algorithmica, 1995, 14: 249～60

7 Muslea I, Minton S, Knoblock C. A Hierarchical Approach to Wrapper Induction. In: Proceedings of the 3rd International Conference on Autonomous Agents, 1999

8 Kushmerick N, Weld D, Doorenbos B. Wrapper induction for information extraction. In: Proc. Int Joint Conf. Artificial Intelligence, 1997

9 Soderland S. Learning Information Extraction Rules for Semi-structured and Free Text. Machine Learning, 1999

10 http://blogs.law.harvard.edu/tech/rss