

Web 访问序列模式挖掘算法的研究

李陶深 王伟娜 陈庆峰

(广西大学计算机与电子信息学院 南宁 530004)

摘 要 针对现有 Web 访问序列模式挖掘算法和 PrefixSpan 算法存在的问题,提出一种基于投影位置的 Web 访问序列模式挖掘算法(PWSPM)。该算法通过序列模式分析,发现用户的行为模式,预测用户对网页的访问模式,进而改进站点的性能和组织结构,提高用户查找信息的质量和效率,以及对用户开展个性化的信息服务。实验和应用结果表明,提出的算法具有更好的执行效率,适用于 Web 日志挖掘,可用于构建智能化 Web 站点和解决个性化的信息服务问题。

关键词 Web 访问,序列模式,数据挖掘,PrefixSpan 算法,Web 日志挖掘

中图法分类号 TP311

文献标识码 A

On Sequential Pattern Mining Algorithm for Web Access

LI Tao-shen WANG Wei-na CHEN Qing-feng

(School of Computer, Electronics and Information, Guangxi University, Nanning 530004, China)

Abstract In view of the problems existing in present sequential pattern mining algorithm for Web access and PrefixSpan algorithm, a sequential pattern mining algorithm for Web access based on projection position-based (PWSPM) was proposed. This algorithm uses sequence pattern analysis to find the user's behavior model and predict user's access pattern to home pages. And then, according to analytical results, it improves sites performance and organizational structure to increase the quality and efficiency of the users to find information. Experimental and application results show that PSPM-Web algorithm has better runtime performance and extensibility. It can apply in Web log mining and is used to build intelligent Web sites and solve the personalized information services.

Keywords Web access, Sequential pattern, Data mining, PrefixSpan algorithm, Web log mining

1 引言

在互联网快速发展和广泛应用的今天,Web 成为信息发布、交互及获取的主要工具,Web 日志的信息量也以惊人的速度增加。因为 Web 日志提供了有关 Web 动态的丰富信息,通过对 Web 日志记录进行挖掘,能够发现用户访问 Web 页面的模式和用户的期望值,进而可以改进和优化站点结构,识别电子商务的潜在客户,增强对终端用户的因特网信息服务的质量和交付。因此,Web 序列模式挖掘对提高整个 Web 系统的性能及其智能应用都具有十分重要的意义。

序列模式挖掘在 Web 点击流量分析中有着广泛的应用,而基于投影数据库的算法是经典的序列模式挖掘算法之一^[1]。它利用投影数据库来减小搜索空间,较好地提高了算法的性能。基于频繁模式增长的 PrefixSpan 算法^[2]是现有的序列模式挖掘算法中性能最好的算法之一,它不需要产生候选的序列模式,大大缩减了算法运行需要检索的存储空间,从而有效地减少了构造投影数据库需要的开销。但是,在构造和扫描投影数据库时,PrefixSpan 算法还是需要花费大量的时间与空间^[3],特别在挖掘长序列模式和密数据集的过程中,

算法可能会生成大量的投影数据库,其中许多投影数据库是重复相同的,从而导致了算法性能的下降。如果序列模式数量巨大,那么算法在构造投影数据库的代价是不可忽视的。

针对 PrefixSpan 算法在挖掘密数据集和长序列模式过程中存在的问题,人们提出了一些改进的算法。文献[3]提出了一种序列模式更新算法(ISBPB)来解决序列模式挖掘中的增量挖掘问题,该算法利用增量挖掘中形成的频繁项集来减小投影数据库,节省了扫描投影数据库所需消耗的时间,从而提高了算法的效率。文献[4]提出了一种无重复投影数据库扫描的序列模式挖掘算法(SPMDS),通过对投影数据库的伪投影计算其 Hash 值,然后根据该值判断是否存在重复的投影数据库,并使用一些简化手段来减少对投影数据库的搜索,节省了算法的扫描时间。文献[5]提出了一种基于位置信息的序列模式挖掘算法(PVS),算法记录了每一个生成的投影数据库的位置信息,并利用这些位置信息有效地避免重复产生相同的投影数据库,从而提高算法的运行效率。文献[6]提出了一种基于改进 PrefixSpan 的序列模式挖掘算法(IPMSP),算法在构建投影数据库时,通过删除非频繁项存储、检查序列数据库关于前缀的前缀来避免重复产生和重复扫描投影数据

到稿日期:2013-05-21 返修日期:2013-07-22 本文受国家自然科学基金项目(60973074)资助。

李陶深(1957—),男,博士,教授,主要研究方向为分布式数据库、无线 Mesh 网络、云计算、网络与信息安全,E-mail: tshli@gxu.edu.cn;王伟娜(1987—),女,硕士,助理工程师,主要研究方向为数据挖掘。

库,从而减少投影的次数和数量,缩短 PrefixSpan 算法的执行时间,提高算法的执行效率。文献[7]采用投影和分离数据库等技术来提高 PrefixSpan 算法的挖掘效率。文献[8]则在 PrefixSpan 算法中加入了剪枝步策略,通过减少扫描某些特定序列模式的生成过程,减小了投影数据库的大小,从而减少了扫描投影数据库所需花费的时间,最终提高了算法的挖掘效率。

国内外对于 Web 访问序列模式的应用方面的研究工作主要集中于分析 Web 站点性能、改进 Web 站点设计等方面。其中,优化 Web 站点的链接结构主要从两个方面考虑:一是通过挖掘 Web 日志,发现用户访问的期望位置^[9],通过在期望位置 and 实际位置之间建立导航链接或者调整网站的网页链接结构和内容,对网站进行优化,更好地为用户服务;二是通过对 Web 日志进行挖掘,发现用户访问页面的相关性^[10],从而使密切联系的网页之间增加链接,以方便用户使用。目前,将 PrefixSpan 算法应用于 Web 访问序列模式挖掘的研究并不多见。

针对 Web 日志数据的特点,本文将 PrefixSpan 算法扩展应用到 Web 日志挖掘中,提出一个基于投影位置的 Web 序列模式挖掘的改进算法。算法通过序列模式分析,挖掘用户访问网页的行为模式,改进网站的结构和性能,最终用于构建智能化 Web 站点,为用户提供个性化的信息服务。

2 问题定义与算法描述

2.1 问题的定义

定义 1 Web 访问序列定义为: $s = (p_1, p_2, \dots, p_m)$, 其中 m 为访问序列的长度, $p_k (1 \leq k \leq m)$ 是一个页面项, 长度为 m 的访问序列称为 m -序列。

定义 2 设 $s_a = \langle a_1 a_2 \dots a_n \rangle$, $s_b = \langle b_1 b_2 \dots b_m \rangle$, 如果存在整数 $1 \leq j_1 < j_2 < \dots < j_m \leq n$, 使得 $a_{j_1} = b_1, a_{j_2} = b_2, \dots, a_{j_m} = b_m$, 则称序列 s_a 为 s_b 的子序列, 又称序列 s_b 是 s_a 超序列。

定义 3 Web 访问序列数据库 S 是元组 $\langle uid, s \rangle$ 的集合, 其中 uid 是用户 ID, s 是 Web 访问序列。

定义 4 Web 访问序列 s_a 在 Web 访问序列数据库 S 中的支持度是指数据库中包含 s_a 的元组个数, 记为:

$$support_S(s_a) = |\{ \langle SID, s_a \rangle \mid \langle SID, s_a \rangle \in S \wedge (s_a \subseteq S) \}|$$

定义 5 给定阈值 $min_support$, 如果访问序列 s 在 Web 访问序列数据库 S 中的支持度不低于 $min_support$, 则称访问序列 s 为频繁 Web 访问序列模式, 长度为 k 的频繁 Web 访问序列模式记为 k -模式。

定义 6 设 Web 访问序列 $s = \langle p_1, p_2, \dots, p_m \rangle$, α 是页面项, $s \diamond \alpha = \langle p_1, p_2, \dots, p_m, \alpha \rangle$ 表示 s 连接 α , 即序列扩展。如果 $s' = s \diamond \alpha$, 那么 s 是 s' 的前缀, α 是 s' 的后缀。

2.2 算法的设计思路

经过研究分析, Web 日志数据具有以下特殊性^[11,12]:

(1) 在同一个时间里, 同一个用户不可能同时去访问两个不同的页面。

(2) 在单个用户的访问序列中很少会出现重复的访问页面, 因此该数据是非常稀疏的。

(3) 不同的用户对网页内容的兴趣是有差异的, 因此用户访问序列中会呈现出长短不均匀的特点。如果用户对某些页面不感兴趣, 那么他可能访问一个页面就会很快离开; 如果用户对某方面的页面和内容非常感兴趣, 那么它就可能会访问上千个类似的页面。

虽然 PrefixSpan 算法具有良好的可伸缩性, 可适用于不同特点数据集的挖掘, 但是由于 Web 日志数据集之间具有不同特性, 直接将 PrefixSpan 算法应用在 Web 日志数据并不能得到较好的挖掘性能。为此, 本文将 PrefixSpan 算法扩展应用到 Web 日志挖掘中, 提出一种基于投影位置的 Web 序列模式挖掘的改进算法(PWSPM)。算法扩展和改进的思路如下:

(1) 鉴于 Web 日志中同一时间不可能同时访问两个页面, 项集扩展只需要在通常的序列模式挖掘过程中进行, 因此在 Web 日志的序列模式挖掘中, 本文算法用序列扩展去代替项集扩展;

(2) 针对 Web 日志数据的稀疏性可能会产生大量的非频繁项, 导致算法为扫描不可能出现的序列模式而花费大量不必要的挖掘时间, 本文算法根据 Apriori 性质, 删除频繁度小于最小支持度 min_sup 的非频繁项的存储, 从而减少非频繁项不必要的存储空间和扫描不可能出现序列的挖掘时间;

(3) 针对 Web 访问序列存在的不均匀性, 在递归挖掘长序列模式过程中, 算法直接定位投影序列位置, 递归进行局部频繁项挖掘, 以避免构造大量重复的投影数据库, 降低构造投影数据库的时间;

(4) 将改进的 PWSPM 算法用于 Web 日志数据挖掘, 获取用户访问 Web 站点的模式信息, 据此进一步分析和研究日志记录中的规律, 预测用户的访问习惯, 更好地改进和优化 Web 站点的性能和组织结构, 提高用户查找信息的质量和效率, 以便实现对用户开展个性化的信息服务。

2.3 算法的描述

假设 S 为 Web 访问序列数据库, $min_support$ 为最小支持度阈值, L 为 Web 访问序列模式的长度, 算法的输出 Web 访问序列模式 $Web_patterns$, 则 PWSPM 算法的描述如下:

PWSPM($S, min_support$)

```
{
    扫描访问 Web 序列数据库  $S$  一次, 得到全部项集  $n$  及其支持度;
    if(支持度  $\geq min\_support$ ) 删除  $S$  中的非频繁项, 得到  $S'$ ;
    将满足最小支持度阈值  $min\_sup$  且长度为 1 的序列模式  $b$  加到
    Web 访问序列模式  $Web\_patterns$  中;
    for(对于每一个 Web 访问序列模式  $Web\_patterns$ ) {依次记录投影
        位置放入投影位置表  $P\_Table$  中; 调用  $recursion(Web\_patterns, L+1, P\_Table, S')$  直接定位出投影序列位置, 递归挖掘
        局部频繁项。
    }
    输出 Web 访问序列模式  $Web\_patterns$ 
}
```

过程 $recursion(Web_patterns, L, P_Table, S)$ 的功能是根据 Apriori 性质删除对非频繁项的不必要的存储, 并利用投影位置直接定位出投影序列位置进行递归挖掘局部频繁项,

以此来减少扫描不可能出现的序列模式而花费的不必要的挖掘时间,提高算法的执行效率。该过程描述如下:

```
recursion (Web_patterns, L, P_Table, S)
{
    根据 P_Table 表中定位的投影位置进行扫描,找出满足 min_sup 的
    频繁页面项  $\alpha$ ;
    if(找不到满足条件的频繁页面项) 结束递归过程。
    采用序列扩展  $Web\_patterns \diamond \alpha = \langle p_1, p_2, \dots, p_m, \alpha \rangle$ , 即长度为  $L+1$ 
    的 Web 访问序列模式  $b$ ;
    for(对于每一个 Web 访问序列模式  $b$ ) {
        将  $b$  添加到  $Web\_patterns$ , 得到了  $Web\_patterns'$ ;
        输出  $Web\_patterns'$ ;
        for(每个  $Web\_patterns'$ ) {
            依次记录投影位置构造新的投影位置表  $P\_Table$ ;
            调用方法 recursion ( $Web\_patterns', L+1, P\_Table, S$ );
        }
    }
}
```

3 算法的实验结果与性能分析

3.1 实验数据准备

本文的实验选用了两个真实的数据集(DMResearch 和 Kosarak)来进行算法的实验对比。其中,DMResearch 数据集来自于中国数据挖掘研究院的 Web 站点(<http://www.dmr-research.net>),收集的时间范围为:2004 年 10 月 7 日至 2004 年 12 月 12 日。经过数据的预处理后,实验的 DMResearch 数据集包含 12193 个用户序列和 8846 个不同页面。而 Kosarak 数据集来源于一个新闻门户网站(<http://fimi.ua.ac.be/data/>)的日志,是一个包含有 990000 个序列的点击数据的公共数据集。实验用的 Kosarak* 数据集是从 Kosarak 中随机地提取的一个稀疏数据子集,其平均长度为 8.03。DMResearch 数据集和 Kosarak* 数据集的特点如表 1 所列。

表 1 DMResearch 数据集和 Kosarak* 数据集的特点

	DMResearch	Kosarak*
页面项总数	8846	8641
用户序列总数	12193	7000
Max. len	10745	41269
Min. len	1	1
平均序列项数	28	8.03

3.2 实验结果与性能分析

本文的实验是在一台 PC 机上进行,机器的配置为:CPU 2.27GHz,2GB 内存,500M 硬盘;操作系统为 Windows 7 操作系统;编译平台为 Eclipse。算法的代码用 JAVA 语言编写实现。PrefixSpan(with pseudo-proj)算法^[2]是采用伪投影技术的 PrefixSpan 算法,该算法的效率比 PrefixSpan 高。实验的目的是采用 DMResearch 和 Kosarak* 两个数据集,对 PrefixSpan(with pseudo-proj)与本文提出的 PWSPM 算法的执行效率进行比较分析,以验证 PWSPM 算法的可行性。

图 1 为在 DMResearch 数据集上 PrefixSpan(with pseudo-proj)算法和 PWSPM 算法的执行效率对比图。从实验结果来看,与 PrefixSpan(with pseudo-proj)算法相比,PWSPM 算法的执行效率提高了 66.7%~76.9%。这是因为 DMRe-

search 有大量非频繁项,PrefixSpan(with pseudo-proj)算法在进行挖掘处理时,扫描大量不可能产生模式需要花费很多的不必要的时间,而 PWSPM 算法采用序列扩展来生成频繁序列,利用 Apriori 性质删除频繁度小于最小支持度 min_sup 的非频繁项,减小了算法访问序列数据库和投影数据库的大小,不仅减少了不必要的项集判断时间,同时通过减少非频繁项不必要的存储空间来节省计算非频繁项频度的时间开销,因而较好地提高了算法的挖掘效率。

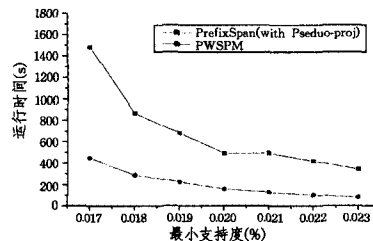


图 1 在 DMResearch 上的执行时间比较

图 2 给出了在 Kosarak* 数据集上 PWSPM 算法和 PrefixSpan(with pseudo-proj)的执行效率比较结果。从实验的结果来看,与 PrefixSpan(with pseudo-proj)算法相比,PWSPM 算法的执行效率提高了 48.2%~67%。效率提高的原因在于:Kosarak-1 是一个有大量非频繁项的稀疏数据集,平均长度为 8.03,PSPM 算法在进行序列挖掘处理时通过舍弃存储非频繁项来减小搜索空间,利用直接定位投影序列位置进行递归挖掘局部频繁项,大大减少了投影数据库的构造数量,从而不用再为扫描不可能出现的序列模式而花费不必要的挖掘时间,因而提高了算法的执行效率。

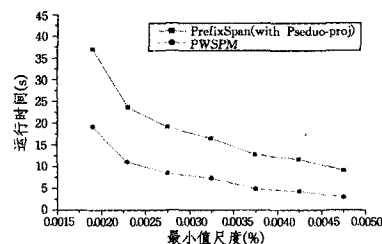


图 2 在 Kosarak* 上的执行时间比较

结束语 根据 Web 日志数据具有的特殊性,本文改进了 PrefixSpan 算法,提出了一种基于投影位置的 Web 访问序列模式挖掘算法(PWSPM)。算法利用序列模式分析去挖掘网站用户访问模式、浏览路径和行为倾向,从中发现某些页面存在的逻辑关系,有效地获取用户访问 Web 页面的行为模式信息,从而进一步分析和研究日志记录中的规律,更好地预测用户对网页的访问模式。可以应用本文算法挖掘和预测得到的信息,有针对性地提出改进和优化网站的性能和组织结构的方案,来高效地提高用户查找信息的质量和效率,从而提高网站可用性。

参考文献

- [1] Han J, Pei J, Mortazvial B, et al. FreeSpan: Frequent pattern projected sequential pattern mining[C]//Proceedings of the 6th ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2000:355-359

- [2] Pei J, Han J, Mortazavi-Asl J, et al. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth[C]// Proceedings of the 7th International Conference on Data Engineering. IEEE Computer Society Washington, DC, USA, 2001: 215-224
 - [3] 陆介平, 刘月波, 倪巍伟, 等. 基于投影数据库的序列模式挖掘增量式更新算法[J]. 东南大学学报, 2007, 36(3): 457-462
 - [4] 张坤, 朱杨勇. 无重复投影数据库扫描的序列模式挖掘算法[J]. 计算机研究与发展, 2007, 44(1): 126-132
 - [5] 张利军, 李战怀. 基于位置信息的序列模式挖掘算法[J]. 计算机应用研究, 2009, 26(2): 529-531
 - [6] 汪林林, 范军. 基于 PrefixSpan 的序列模式挖掘改进算法[J]. 计算机工程, 2009, 35(23): 56-58, 61
 - [7] Saputra D, Rambli D R A, Foong O M. Sequential Pattern Mining using PrefixSpan with Pseudoprojection and Separator Database[C] // 2008 International Symposium on Information Technology. Kuala Lumpur, Malaysia, Aug. 2008: 1-7
 - [8] 公伟, 刘培玉, 贾嫻. 基于改进 PrefixSpan 的序列模式挖掘算法[J]. 计算机应用, 2011, 31(9): 2045-2047
 - [9] 鲍钰, 黄国兴, 张召. 基于 Web 日志挖掘的网站结构优化方法[J]. 计算机工程, 2003, 29(12): 82-84
 - [10] 胡建武, 何贞铭, 张贻权. Web 日志挖掘及其实现[J]. 计算机工程与应用, 2004, 40(14): 156-158
 - [11] Yang Z L, Wang Y T, Kitsuregawa M M. An Effective system for mining web log[C] // Proceedings of the 8th Asia-Pacific Web conference on Frontiers of WWW Research and Development (APWeb'06). Harbin, China, January 2006: 40-52
 - [12] Mabroukeh N R, Ezeife C I. A Taxonomy of Sequential Pattern Mining Algorithms[J]. Journal of ACM Computing Surveys, 2010, 43(1): 1-41
-
- (上接第 14 页)
- [10] Ravishankar M, Eisenlohr J, Pouchet LN, et al. Code generation for parallel execution of a class of irregular loops on distributed memory systems[C]// The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC12), 2012. Los Alamitos, CA, USA; IEEE Computer Society Press, 2012
 - [11] Ravishankar M, Eisenlohr J, Pouchet LN, et al. Code generation for parallel execution of a class of irregular loops on distributed memory systems[R]. OSU-CISRC-5/12-TR10. The Ohio State University, 2012
 - [12] Strout M M, George G, Olschanowsky C. Set and relation manipulation for the sparse polyhedral framework[C]// Proceedings of the 25th International Workshop on Languages and Compilers for Parallel Computing (LCPC2012), 2012. Springerburg LNCS series, 2013
 - [13] LaMille A, Strout M. Enabling code generation with sparse polyhedral framework[R]. CS-10-102 Colorado State University, 2010
 - [14] Guo Min-yi, Pan Yi, Liu Zhen. Symbolic Communication Set Generation for Irregular Parallel Applications[J]. The Journal of Supercomputing, 2003, 25(3): 199-214
 - [15] 胡长军, 李静, 王珏, 等. 一类非规则并行应用问题的通信集生成算法[J]. 计算机学报, 2008, 31(1): 120-126
 - [16] Lee S, Min S J, Eigenmann R. OpenMP to GPGPU: a compiler framework for automatic translation and optimization[C]// Proceedings of the 14th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPOPP'99), 2009. NY, USA; ACM New York, 2009: 101-110
 - [17] Basumallik A, Eigenmann R. Optimizing irregular shared-memory applications for distributed-memory systems[C] // Proceedings of the 11th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPOPP'06), 2006. NY, USA; ACM New York, 2006: 119-128
 - [18] Campanoni S, Jones T, Holloway G, et al. HELIX: Automatic Parallelization of Irregular Programs for Chip Multiprocessing[C]// Proceedings of the 10th International Symposium on Code Generation and Optimization (CGO'12), 2012. NY, USA; ACM New York, 2012: 84-93
 - [19] Kim H, Johnson N P, Lee J W, et al. Automatic Speculative DOLL for Clusters[C] // Proceedings of the 10th International Symposium on Code Generation and Optimization (CGO'12), 2012. NY, USA; ACM New York, 2012: 94-103
 - [20] Zhuang X, Eichenberger A E, Luo Y, et al. Exploiting parallelism with dependence-aware scheduling[C]// Proceedings of the 2009 International Conference on Parallel Architectures and Compilation Techniques (PACT'09), 2009. Washington DC, USA; IEEE Computer Society, 2009: 193-202
 - [21] Lin Y, Padua D. Compiler analysis of irregular memory accesses[C]// Proceedings of The ACM SIGPLAN 2000 Conference on Programming Language Design and Implementation 2000. NY, USA; ACM New York, 2000: 157-168
 - [22] Zhao J, Zhao R C, Han L. A Nonlinear Array Subscripts Dependence Test[C]// Proceedings of the 14th IEEE International Conference on High Performance Computing and Communications. Liverpool, England, 2012: 764-771
 - [23] Aho A V, Lam M S, Sethi R, et al. Compilers principles, techniques, and tools (Second edition) [M]. Boston: Addison-Wesley, 2007
 - [24] Amarasinghe S P, Lam M S. Communication optimization and code generation for distributed memory machine; [C]// Proceedings of The ACM SIGPLAN 1993 Conference on Programming Language Design and Implementation (PLDI'93), 1993. NY, USA; ACM New York, 1993: 126-138
 - [25] Mitchell N, Carter L, Ferrante J. Localizing Non-affine Array References[C] // Proceedings of the 1999 International Conference on Parallel Architectures and Compilation Techniques (PACT'99), 1999. DC, USA; IEEE Computer Society Washington, 1999: 192-202
 - [26] 丁锐, 赵荣彩, 韩林. 基于主导值的计算和数据划分算法[J]. 计算机科学, 2012, 39(3): 290-294