# SWLMS: 一个 Web 日志挖掘系统

杨怡玲[1]，管旭东[1]，陆丽娜[2]，尤晋元[1]

（上海交通大学 计算机系，上海 200030）[1]

（西安交通大学 计算机系，西安 710049）[2]

**摘 要**：Internet 的发展带动了 WWW 的发展，继数据挖掘技术成功地应用于传统数据库领域之后，人们对基于 Web 的数据挖掘技术（简称 Web 挖掘）也开始进行研究。Web 日志挖掘是将数据挖掘的技术应用于 Web 服务器上的日志文件，以发现用户的浏览模式，分析站点的使用情况。它可用于协助管理者优化站点结构，提高站点效率。在分析 Web 日志挖掘的困难及对策的基础上，给出了 Web 日志挖掘系统 SWLMS 的体系结构。具体介绍了 SWLMS 中日志的预处理过程，包括数据净化、用户识别、会话识别、路径补充的主要任务及其实现，并着重介绍了预处理之后的序列模式识别过程和算法，包括最大向前路径的识别和频繁遍历路径的发现，并给出了实验结果。

**关键词**：数据挖掘；Web 日志挖掘；序列模式识别；最大向前路径

**中图分类号**：TP311.13

# SWLMS: A Web Log Mining System

*YANG Yi-ling*[1]， *GUAN Xu-dong*[1]， *LU Li-na*[2]， *YOU Jin-yuan*[1]

Dept. of Computer，Shanghai Jiaotong Univ.，Shanghai 200030，China[1]

Dept. of Computer，Xi'an Jiaotong Univ.，Xi'an 710049，China[2]

**Abstract**：Internet brings the wide spread of WWW. After the successful application of data mining (DM) technology to the traditional database domain, web mining, the application of DM to web data, begin to arise. In this paper, we mainly discuss web log mining, the application of DM to log data generated by web servers, which could assist the webmaster to optimize site architecture and increase visiting efficiency. Based on the analysis of difficulties and the corresponding solutions of web log mining, the architecture of SWLMS, our sample web log mining system is addressed. The data-preprocessing phase in SWMLS, including data cleaning, user recognition, session identification and path filling is discussed in detail. Then the sequential pattern recognition phase and its algorithms are presented, including the recognition of maximum forward paths and frequent traversal paths, with some experimental results presented.

**Key Words**：data mining；web log mining；sequential pattern recognition；maximum forward path

Internet 的发展带动了 WWW 的发展，继数据挖掘技术成功地应用于传统数据库领域之后，人们对基于 Web 的数据挖掘技术（简称 Web 挖掘）也开始进行研究。Web 日志挖掘是将数据挖掘的技术应用于 Web 服务器上的日志文件，以发现用户的浏览模式，分析站点的使用情况。它可用于协助管理者优化站点结构，提高站点效率。

收稿日期：1999-08-30

作者简介：杨怡玲（1973～），女，博士生

## 1 Web 日志挖掘的困难和解决方法

Web 服务器日志记录了用户访问本站点的信息。典型的 Web 服务器日志包括以下信息：IP 地址、请求时间、方法（如 GET）、被请求文件的 URL、HTTP 版本号、返回码、传输字节数、引用页的 URL（指向被请求文件的页面）和代理。但是，由于本地缓存、代理服务器和防火墙的存在，使得 Web 日志中的数据并不精确，直接在其上进行挖掘非常困难，而且有可能导致结果的错误。

在 Web 日志挖掘中，主要是提供面向用户的信息分析，所以首先要从 Web 日志中对用户会话进行识别，以次作为信息分析的基础。用户会话是一个用户在规定的时间内请求的所有 Web 页面。日志的不精确性往往增加了识别用户会话的难度。本文介绍的 Web 日志挖掘系统 SWLMS（A Simple Web Log Mining System）在利用主机地址的同时分析日志中每条记录的请求页和引用页的 URL，然后根据 Web 站点的拓扑结构（超链接）和其它启发式规则区分用户会话。

## 2 SWLMS 的体系结构

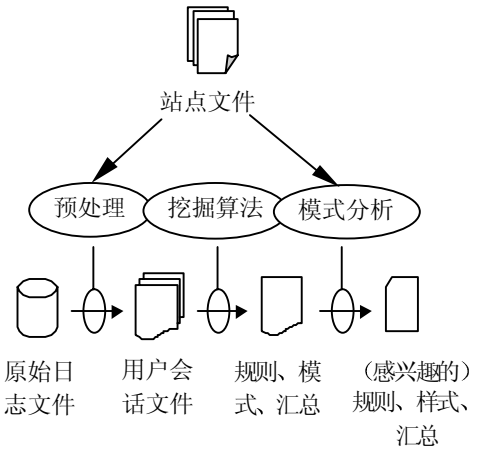


图 1. SWLMS 中的 Web 日志挖掘过程

Fig. 1. Procedure of web log mining in SWLMS

SWLMS 中的 Web 日志挖掘过程一般分为三个阶段，即：预处理阶段、挖掘算法实施阶段、模式分析阶段。如图 1 所示。

下面我们就按图 1 的挖掘过程对 SWLMS 的各处理流程进行具体的介绍。

首先根据访问日志、引用日志和代理日志间的内在关系，将它们拼接成完整的日志并保存在数据库中。图 2 列出了这三种日志的内容。

| | |
|---|---|
| 访问日志 access.log | 202.117.1.2 - [07/Dec/1998:10:25:00–0600] "GET /resource.html HTTP/1.0" 200 782 |
| 引用日志 referer.log | http://www.xjtu.edu.cn/ → /resource.html |
| 代理日志 agent.log | mozilla/2.0 (compatible; MSIE 3.01; Windows 95) |

图 2. 访问日志、引用日志和代理日志的内容

Fig. 2. Sample data of access log, referer log, and agent log

考虑图 2 的日志，引用日志中箭头指向/resource.html，它是访问日志中 GET 命令请求的文件。而引用日志中箭头前的 URL（http://www.xjtu.edu.cn/）说明用户从主页上请求 resource.html 文件。根据这种引用和被引用的关系将图 2 的三条日志组成一条完整的日志。在数据库中建立一个日志表 T_LOG，然后将拼接好的日志保存在 T_LOG 中，结果如图 3 所

示。

| 字段名 | 含　义 | 内　容 |
|---|---|---|
| ip | 用户的IP地址或URL | 202.117.1.2 |
| r_date | 文件访问日期 | 07/Dec/1998 |
| r_time | 文件访问时间 | 10:25:00 |
| method | 方法 | GET |
| request | 被请求文件的 URL | /resource.html |
| status | 服务器状态 | 200 |
| size | 传输字节 | 782 |
| agent | 代理 | Mozilla/2.0(compatible; MSIE 3.01; Windows 95) |
| referer | 引用页的 URL | http://www.xjtu.edu.cn/ |

图 3.　T_LOG 中的一条记录

Fig. 3.　A record in the T_LOG table

我们以“www.xjtu.edu.cn”服务器上 98-12-710:24:53 到 98-12-7 14:58:45 4 个多小时内的日志做实验，日志文件长度约有 2.27 兆字节，保存到数据库中共有 14300 多条记录，这些记录称为原始数据。

# 3．SWLMS 中的预处理过程

Web 日志挖掘首先要对日志中的原始数据进行预处理，包括依赖于域的数据净化、用户识别、会话识别和路径补充等。对日志进行预处理的结果直接影响到挖掘算法产生的规则与模式。可以说预处理过程是 Web 日志挖掘质量保证的关键。

## 3.1 数据净化

数据净化指删除 Web 服务器日志中与挖掘算法无关的数据。大多数情况，只有日志中 HTML 文件与用户会话相关，所以通过检查 URL 的后缀删除认为不相关的数据。在 SWLMS 中使用一个缺省的后缀名列表帮助删除文件，包括的后缀名有 GIF、JPEG、JPG、gif、jpeg、jpg、map 和 cgi。列表可以根据正在分析的站点类型进行修改，例如，对一个主要包含图形文档的站点，此时就不能将图形文件删除。经过数据净化，数据库中的记录变为 5000 多条。

## 3.2 用户识别

由于本地缓存、代理服务器和防火墙的存在，使得识别每一个用户的任务变得很复杂。一般最常被 Web 日志挖掘工具使用的技术就是基于日志/站点的方法，在 SWLMS 中使用了一些启发式规则帮助识别用户。

1） 如果 IP 地址相同，但是代理（Agent）日志中表明用户的浏览器或操作系统改变了，则认为不同的代理表示不同的用户。

2） 将访问日志、引用日志和站点拓扑结构结合，构造用户的浏览路径。如果当前请求的页面同用户已浏览的页面之间没有链接关系，则认为存在 IP 地址相同的多个用户。要说明的是，以上仅是帮助识别用户的启发规则，并非使用这些规则就能准确地识别出用户。

## 3.3 会话识别

在跨越时间区段较大的 Web 服务器日志中，用户有可能多次访问了该站点。会话识别的目的就是将用户的访问记录分为单个的会话（Session）。最简单的方法是利用超时，如果两页间请求时间的差值超过一定的界限就认为用户开始了一个新的会话。为简单起见，在 SWLMS 中使用 30 分钟作为超时界限。

### 3.4 路径补充

在识别用户会话过程中的另一个问题是确定访问日志中是否有重要的请求没有被记录。这就是路径补充所做的工作，解决的方法类似于用户识别中的方法。如果当前请求的页与用户上一次请求的页之间没有超文本链接，那么用户很可能使用了浏览器上的“BACK”按钮调用缓存在本机中的页面。检查引用日志确定当前请求来自哪一页，如果在用户的历史访问记录上有多个页面都包含与当前请求页的链接，则将请求时间最接近当前请求页的页面作为当前请求的来源。若引用日志不完整，可以使用站点的拓扑结构代替。通过这种方法将遗漏的页面请求添加到用户的会话文件中。

## 4. 序列模式识别

经过上面的处理，Web 日志挖掘中数据预处理的工作就基本完成，下面一步是实施挖掘算法。在 SWLMS 中识别用户浏览行为的序列模式，主要是集中在挖掘频繁遍历路径。

### 4.1 最大向前路径

遍历路径就是在用户会话中请求页面所组成的序列。由于用户会话中既包含请求页面又包含路径补充时添加的页面，所以挖掘频繁遍历路径时，首先在每个用户会话中找出所有的最大向前路径（Maximum Forward Path，以下简称 MFP），然后确定其中的公共子路径。最大向前路径的方法是基于 Chen 等[3]中最大向前引用的工作。MFP 是在用户会话中的第一页到回退的前一页组成的路径。例如：一个用户会话中请求的页面顺序是 *A-B-A-C-D-C*，则对应的 MFP 为 *A-B* 和 *A-C-D*。

```
for 每个用户会话 {
  y1=x1; j=2; i=2;
  flag=YES;
  while（i≤m）{
    if（xj==yk）for some 1≤k<j {
      if （flag==YES）
        将{y1,…,yj-1} 作为MFP 输出;
      j=k+1; i=i+1;
      flag=NO;
    } else {
      yj=xi; j=j+1; i=i+1;
      flag=YES;
    }
  }
  if （flag==YES）
    将{y1,…,yj-1} 作为MFP 输出;
}
```

图 4. 发现 MFP 的算法

Fig. 4. Algorithm of finding MFP

图 4 是在一个用户会话中寻找 MFP 的算法。假设$\{x_1, \ldots, x_m\}$代表一个用户会话，$\{y_1, \ldots, y_{j-1}\}$代表一个含有潜在 MFP 的字符串，初值为空，flag 标志标明当前的遍历方向是前进还是后退。算法的主要思想是：每次检查用户会话中的页 $x_i$，试图将该页扩充到潜在 MFP 中。

（1） 若 $x_i \notin \{y_1, \ldots, y_{j-1}\}$，则将 $x_i$ 作为 $y_j$ 加入潜在 MFP 中，并且将 flag 标记为前进。

（2） 否则有 $x_i = y_k$，$1 \le k < j$，

a) 若在此之前，flag 标明的移动方向是前进，则将$\{y_1, \ldots, y_{j-1}\}$作为一个 MFP 加入到结果集合，然后从潜在 MFP 中删除页面$\{y_{k+1}, \ldots, y_{j-1}\}$，并设标志 flag 为向后移动，进入下一次循环。

b) 若 flag 为向后，则此时的$\{y_1, \ldots, y_{j-1}\}$并不是 MFP，直接删除$\{y_{k+1}, \ldots, y_{j-1}\}$，进入下一次循环。

（3） 如果循环到用户会话中的最后一页，flag 标志仍标明向前，则此时的$\{y_1, \ldots, y_{j-1}\}$是一个 MFP。

SWLMS 利用图 4 的算法寻找 MFP，共识别出 1100 个 MFP。这样，挖掘频繁遍历路径问题转化为在所有用户会话的 MFP 中发现频繁出现的连续子序列问题。

## 4.2 挖掘频繁遍历路径算法

挖掘频繁遍历路径在某种程度上与数据挖掘中时序模式相似。频繁遍历路径是 MFP 中满足一定支持度的连续页面序列。包含频繁遍历路径的用户会话的数目叫支持度。定义频繁遍历路径的长度为其包含的页面数。记长度=$k$ 的频繁遍历路径的集合为 $FP_k$，其中最频繁的 $M$ 个遍历路径的集合为 $FP_{k.M}=\{P_{k.1}, \ldots, P_{k.M}\} \subseteq FP_k$，$FP_{k.M}$中的元素按支持度由大到小的顺序排列。

挖掘关联规则时发现频繁项集与挖掘用户浏览模式时发现频繁遍历路径之间有显著区别：在频繁遍历路径中，页面必须形成一个连续的序列，而频繁项集仅是一个事务中项的集合，没有顺序关系。

```
for 每个 F_i {
  for  F_i 中的每个 {x_1 ,x_2 ,..., x_m} {
    if （k≤m）{
      for （j=1；j<m-k+1；j++）{
        if {x_j ,..., x_j+k-1}已经在 FP_k 中
          {x_j ,..., x_j+k-1}的支持度加 1
        else if {x_j ,..., x_j+k-2}的支持度≥s_k-1
                AND {x_j+1 ,..., x_j+k-1}≥s_k-1
          将{x_j ,..., x_j+k-1}插入 FP_k;
      }
    }
  }
}
```

图 5. 发现频繁遍历路径的算法

Fig. 5. Algorithm of finding frequent traversal path

假设 $P_{k-1.M}$是 $FP_{k-1.M}$中的第 $M$ 个频繁遍历路径（即 $FP_{k-1.M}$中的最后一个元素，它的支持度最小），$P_{k-1.M}$的支持度是 $s_{k-1}$，$F_i$是用户会话 $S_i$中 MFP 的集合，$\{x_1, \ldots, x_m\} \in F_i$，构造 $FP_k$（$k$>1）的算法见图 5。

在解释构造 $FP_k$算法之前，我们先讲一下 $FP_k$的候选路径。若两个连续的 $k$-1 长的子路径$\{x_j, \ldots, x_{j+k-2}\}$和$\{x_{j+1}, \ldots, x_{j+k-1}\}$均是 $FP_{k-1}$的元素，即它们的支持度都不小于 $P_{k-1.M}$的支持度 $s_{k-1}$，那么就称$\{x_j, \ldots, x_{j+k-2}, x_{j+k-1}\}$为 $FP_k$的候选路径。例如，假设会话 $s_1$包含两个 MFP：$\{A, B, C, D, E\}$和$\{G, H\}$，寻找 $FP_3$的候选路径，需要考虑 3 个子路径$\{A, B, C\}$、$\{B, C, D\}$和$\{C, D, E\}$。如果$\{A, B\}$和$\{B, C\}$是 $FP_2$中的频繁遍历路径，那么$\{A, B, C\}$就是一个 $FP_3$的候选路径，同样，如果$\{B, C\}$和$\{C, D\}$是 $FP_2$中的频繁遍历路径，那么{B，

$C$，$D\}$也是 $FP_3$ 的一个候选路径。因为$\{G$，$H\}$的长度小于 3 所以不予考虑。

构造 $FP_k$ 的算法主要是基于上面定义的候选路径的概念，从 MFP 中找出长度为 $k$ 的候选路径$\{x_j$，…，$x_{j+k-1}\}$，然后计算它在所有用户会话中的支持度。支持度最大的 $M$ 个路径组成的集合就是 $FP_{k.M}$。

在调用图 5 的算法之前，先要计算每一页在用户会话中的支持度，这其实是处理路径长度 $p\_len$=1 的情况。然后从 $p\_len$ =2 直到 $p\_len$ =$k$ 循环调用图 5 的算法，每一次循环都可以利用上一次循环结果中的支持度。注意，对每个 $p\_len$，所有用户会话的 MFP 只能扫描一次。

SWLMS 寻找长度=6 的频繁遍历路径，并将最小支持度确定为 30，结果找到 11 条频繁遍历路径。最后要将挖掘的结果以直观明了的方式展现给用户，以便更好地被用户所理解。图 6 是将寻找到的频繁遍历路径经过简单的可视化处理后用直方图表示的界面。

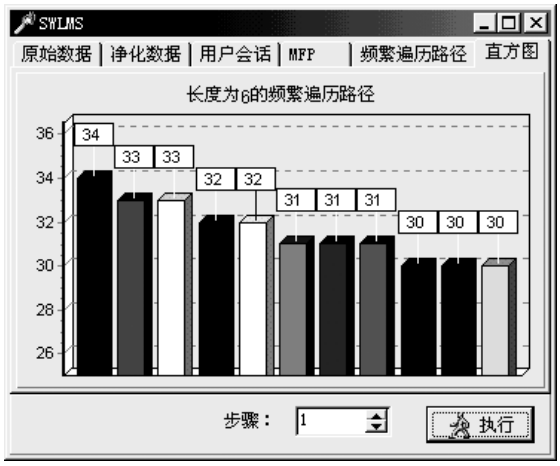


图 6. SWLMS 中长度为 6 的频繁遍历路径结果

Fig. 6. Frequent traversal paths in SWLMS (length=6)

## 5. 结束语

Web 日志挖掘是一个比较新的研究方向，在我国从事这方面的研究人员很少。本文中所做的工作对于学习和研究基于 Internet 的挖掘技术以及建造一个实用的 Web 日志挖掘系统具有很好的参考价值。然而 SWLMS 离一个真正实用的 Web 日志挖掘系统还有相当的距离，本人将在这方面继续作进一步的研究。

**参考文献:**

[1] Wu K L, Yu P S, and Ballman A. SpeedTracer: A Web usage mining and analysis tool. IBM System Journal, 1998, 37(1):89~105

[2] Cooley R, and Srivastava J. Data Preparation for Mining World Wide Web Browsing Patterns. Journal of Knowledge and Information Systems, 1999, 1(1):32-57, http://maya.cs.depaul.edu/~mobasher/papers/webminer-kais.ps

[3] Chen M S, Park J S, Yu P S. Data Mining for Path Traversal Patterns in a Web Environment. In: Proc. of the 16th International Conference on Distributed Computing Systems, 1996, 385~392

[4] 杨怡玲，Web 日志挖掘中的数据准备与用户浏览模式识别:[硕士学位论文]，西安：西安交通大学，1999