概念格在 Web 日志挖掘中的应用

The Application of Concept Lattice in Web_log Mining

习慧丹　严 晖　（中南大学信息科学与工程学院 湖南长沙 410083）

摘要：概念格作为一种用于数据组织和数据分析的形式化工具，有着广阔的应用领域，如信息检索、数字图书馆、软件工程、数据挖掘等方面。先对概念格进行了简单的介绍，然后讨论了 Web 日志挖掘过程的两个重要阶段：数据预处理和模式发现，并将概念格应用于模式发现以进行频繁页面之间的关联规则挖掘和频繁访问路径挖掘，减少了候选项集的产生，可有效提高挖掘效率。

关键词：概念格 Web 日志挖掘 关联规则 频繁访问路径

1 引言

Web 数据挖掘是一个结合了 WWW 和数据挖掘技术的热门研究课题。按照挖掘对象的不同，可以将 Web 挖掘分为三大类：Web 内容挖掘、Web 结构挖掘和 Web 日志挖掘。用户在访问 Web 站点的时候其浏览访问信息都记录在日志文件中，Web 日志挖掘的目的就是在海量的 Web 日志数据中自动、快速地发现用户的访问模式，如频繁访问路径、频繁访问页面、用户聚类等。这些用户访问模式有着广阔的应用领域，包括为用户提供个性化服务、完善网站结构、完善系统性能以及构建智能化 Web 站点等，在提高网站的声誉和效益等方面都起到重要的作用。

概念格是形式概念分析理论的核心数据结构，是一种非常有效的数据挖掘和分析工具，在理论研究和实际应用上都具有重要意义，将它引入到 Web 日志挖掘中的模式发现阶段，作为一种存储频繁项目集的数据结构，能从海量信息里面提取模式，可有效地提高挖掘效率，为日后行为做指导。

2 概念格

2.1 概念格简介

概念格（concept lattice），也称为 Galois 格，由 R·Wille 首先提出。这里介绍所需概念格的基本概念[1]。假设给定形式背景为三元组（O,A,R），其中 O 是对象集合，A 是属性集合，R 是 O 和 A 之间的一个二元关系，则存在唯一的一个偏序集合与 R 对应，并且这个偏序集合产生一种格结构，这种由背景（O,A,R）所导出的格就称为概念格。格中的每个节点是一个序偶（称为概念），记为（X,Y），其中 $X \in$ 幂集 P(O)，称为概念的外延，即概念所覆盖的对象集合；$Y \in P(A)$，称为概念的内涵，即该概念所覆盖的对象的共同属性。每一个概念关于关系 R 是完备的，即有性质：(1) $X = \{x \in O \mid \forall y \in Y, xRy\}$；(2) $Y = \{y \in A \mid \forall x \in X, xRy\}$。在概念格节点间能够建立起一种偏序关系：给定节点 C1 = (X1,Y1) 和 C2 = (X2,Y2)，则 $C1 < C2 \Leftrightarrow Y1 \subset Y2$，领先次序意味着 C1 是 C2 的父节点。根据这种偏序关系可生成格的 Hasse 图：如果 $C1 < C2$ 并且不存在另一个节点 C3 使得 $C1 < C3 < C2$，则从 C1 到 C2 就存在一条边。

总的来说，概念格是根据形式背景所产生的概念之间的偏序关系建立起来的，并能通过 Hasse 图以图形化形式描述出来，体现概念之间的泛化和特化关系一种特殊的数据结构。

2.2 概念格的构造

概念格的构造是概念格研究中的重点，构造算法分为两大类[1]：批处理算法和渐进式算法。

批处理算法的思想是首先生成所有概念，然后根据它们之间的直接前驱－后继关系，生成边，完成概念格的构造。根据构造格的顺序的不同，可将批处理算法分为：自顶向下算法，自底而上算法和枚举算法。典型的批处理算法有 Bordat 算法，Ganter 算法，Nourine 算法等。

渐进式算法的思想是将当前要插入的对象和现

有格中所有的节点作交运算，根据交的结果不同采取不同的行动：当节点的内涵和新对象的内涵没有交集时，它保持不变；当节点的内涵包含在新对象的内涵中时，将其外延更新为包括新对象即可；当交集在格中没出现过时，则产生新的节点，同时对相应的边进行修改。

由于 Web 日志记录是随时间不断增加的，这就出现了挖掘过程的增量问题，而通过渐进式算法建立起来的概念格能很好的解决这个问题，所以采用它来进行增量挖掘，实现动态更新。

3 Web 日志挖掘的主要过程及概念格的应用

Web 日志挖掘是通过挖掘相应站点的日志文件中的相关数据来发现该站点上的浏览者的行为模式。其基本过程主要分为三个步骤：数据预处理、模式发现、模式分析，如图 1 所示。数据预处理环节是整个过程的基础和实施有效挖掘算法的前提，在 Web 日志挖掘中起着非常重要的作用，模式发现是挖掘的主要过程，所以重点讨论这两个过程。

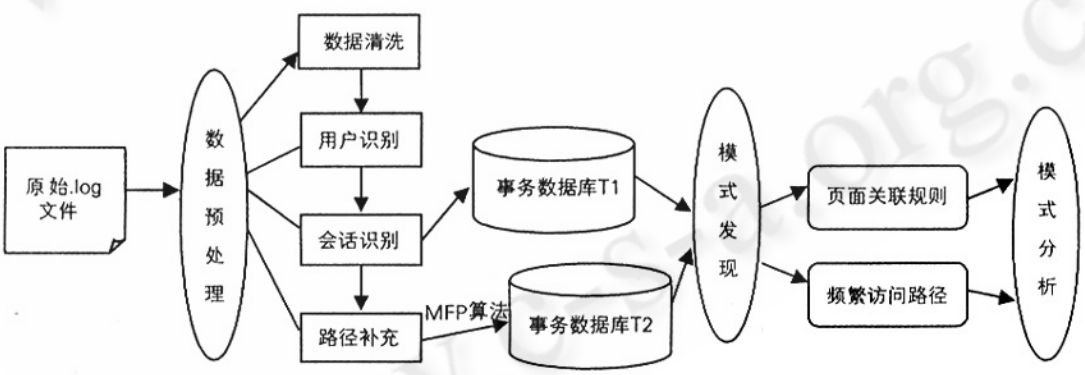

图 1　Web 日志挖掘模型

3.1 数据预处理

根据挖掘的目的，对原始 Web 日志文件中的数据进行提取、分解、合并，最后转化为适合进行数据挖掘的数据格式，并保存到关系型数据库表或数据仓库中，等待进一步处理。它主要包括四个阶段[2]：数据清洗、识别用户、识别会话和路径补充。

(1) 数据清洗。数据清洗指根据需求，对日志文件进行处理，删除日志记录中与挖掘无关的数据。基于 Web 服务器(如 IIS 等)的日志记录功能，当用户在访问相应站点的时候，都会产生大量的日志文件(.log)，因此，这些日志文件就成了 Web 日志挖掘的重要数据来源，它详细地记录了用户访问本站点的信息。根据 W3C 组织规定，服务器日志具有两种格式[2]：通用日志格式(Common Log Format，CLF)和扩展型日志格式(Extended Common Log Format，ECLF)。CLF 只包含固定的几个属性，而 ECLF 包含了可以配置对安全分析有帮助的很多扩展属性，可以使日志记录更加详细，安全环境中的首选日志类型为 ECLF。但是这样也可能增加大量的与挖掘无关的数据而使数据清洗过程较长。

W3C 扩展日志文件格式的日志中可收集以下信息：date(日期)、time(时间)、c－ip(客户 IP 地址)、cs－username(用户名)、s－ip(服务器 IP 地址)、s－port(服务器端口)、cs－method(方法，如 GET/POST/HEAD)、cs－uri－stem(URI 资源)、sc－status(协议状态)、sc－bytes(发送的字节数)、cs(User－Agent)(用户代理)、cs(Referer)(引用页面)等。如图 2 为收集到的一个日志文件：

#software：Microsoft Internet Information Services 5.0

#Version：1.0

#Date：2005－10－21 10：45：39

#Fields：date time c－ip cs－username s－ip s－port cs－method cs－uri－stem cs－uri－query sc－status cs(User－Agent)

2005－10－21 10：45：39 202.197.78.159－202.197.78.167 80 GET

/assignmentms/index.aspx – 200 Mozilla/4.0 + (compatible; + MSIE + 6.0; + Windows + NT + 5.1; + SV1; + .NET + CLR + 1.1.4322)

图 2　日志文件

清洗方法：将原始的日志文件导入到 SQL SERVER 2000 中，然后进行如下操作：①删除 s – port、sc – bytes、s – ip 等不必要的属性；②删除无用的日志记录条目，一般包括后缀名为 gif、jpg、jpeg 和含 swf、js、cgi 等文件的日志记录。由此只保留含 html 文件或 asp 文件的日志记录条目。经过处理，得到的日志记录所包含的属性如表 1 所示：

表 1　部分日志记录

number	IP	Time	URL	Referer	Agent
1	202.197.78.159	21/Oct/2005:10:45:39	A.aspx	–	Mozilla/4.0(Windows + NT)
2	202.197.78.159	21/Oct/2005:10:46:07	B.aspx	A	Mozilla/4.0(Windows + 98)
3	202.197.78.159	21/Oct/2005:10:46:09	D.aspx	B	Mozilla/4.0(Windows + 98)
4	202.197.78.159	21/Oct/2005:10:46:35	C.aspx	A	Mozilla/4.0(Windows + NT)

(2) 用户识别。用户识别对用户访问模式挖掘起着重要的作用，它是指根据表 1 所得到的日志记录信息，将用户与他所访问的页面关联起来，即哪个用户访问了哪些页面。可分两个步骤进行：首先，根据客户 IP 地址和 Agent 初步划分用户，采用如下的启发式规则：如果 IP 地址相同，而代理不同，则认为每个不同的代理就代表不同的用户；然后，根据所请求的 URL 页面和 Referer 页面，结合网站的拓扑结构，进一步划分，构造出用户浏览路径。

(3) 会话识别。会话就是用户在一次访问中所请求的页面集合。一个浏览路径可能包含了该用户多次访问站点的页面，它并没有将用户的每次访问情况区分开来。会话识别的任务就是将用户识别所得到的每一个浏览路径划分为多个路径，其中每个路径代表了一次访问情况。识别方法：根据 Time 属性和设定的时间阈值(实验证明，比较合理的值为 25 分钟)进行划分，即如果两个页面之间的请求时间差值超过了时间阈值就认为用户开始了一个新的会话，这样就得到了会话集合。同时，把每个会话当作一个事务，给定唯一的事务 ID，将会话集合转换成事务数据库，命名为 T1。

(4) 路径补充。主要是针对频繁访问路径挖掘对会话集合进行的进一步处理。如果当前请求的页面与用户上一次请求的页面之间没有超文本链接，那么用户很可能使用了浏览器上的"后退"按钮调用缓存在本机中的页面，这个页面在日志文件中是没有记录的，这样就使得会话集合中的浏览路径不完整。根据引用日志和网站拓扑结构进行路径补充：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。

3.2 模式发现

模式发现指运用各种算法对处理后的数据进行挖掘，生成模式。将概念格应用到这主要是用来进行频繁访问页面之间的关联规则挖掘和频繁访问路径的挖掘。

(1) 频繁访问页面之间的关联规则挖掘。不需要考虑页面之间的顺序关系，针对事务数据库 T1 进行挖掘。由于概念格是由形式背景所导出，所以我们将 T1 理解形式背景(O,A,R)，其中会话集合表示 O，页面集合表示 A，R 就表示一个会话包含哪些页面。

挖掘过程包括以下几部分：①初始化概念格，用 Ctop = (O, A′)(A′ 是对象集合 O 的公共属性集合)和 Cbottom = (O′, A)(O′ 是属性集合 A 的公共对象集合)作为仅有的两个概念构造出初始概念格，然后将 T1 中的每个事务作为待插入的新对象，利用算法 1 构造 T1 所对应的概念格 CL；②当有新的日志增加

时，利用算法 1 对 CL 进行更新；③提取关联规则。算法中的 C. extension 表示节点 C 的外延，C. intension 表示节点 C 的内涵，|C. extension| 表示外延基数。

算法 1 描述：（代码略）。

（2）频繁访问路径挖掘。路径分析可用于发现 Web 站点中最经常被访问的路径，从而调整站点的结构。访问路径考虑页面之间的连续关系，所以频繁访问路径挖掘是针对路径补充得到的会话集合进行挖掘。首先对会话集合进行处理将一个会话划分成多个最大向前路径 MFP（Maximum Forward Path）[4]，获取 MFP 集合，例如：一个用户会话中请求的页面顺序是 A－B－A－C－E－C，则对应的 MFP 为 A－B 和 A－C－E。然后将每个 MFP 作为一个事务，形成事务数据库 T2。频繁访问路径挖掘就是在 T2 的 MFP 集合中找满足最小支持度的连续子序列。主要过程：首先，求出所有的 MFP；然后针对 T2 构造概念格 CL，同时输出频繁访问路径。（限于篇幅）概念格的构造和找频繁访问路径算法可参考算法 1 得到，需注意的是概念格节点的内涵是连续的 MFP 子序列。MFP 算法：$\{s_1,\cdots,s_m\}$ 表示一个会话，$\{y_1,\cdots,y_{l-1}\}$ 表示一个 MFP 字符串，flag 标志标明当前的遍历方向是前进还是后退。

```
for every session{
y1 = s1; l = 2; i = 2; flag = 1; \\前进
while(i≤m){ if(si = = yk) for some 1≤k<l{
    if(flag = =1) 将{y1,…,yl-1}作为 MFP 输出; l = k+1; i = i+1; flag = 0; } \\后退
    else{ yl = si; l = l+1; i = i+1; flag = 1; }}
  if(flag = =1) 将{y1,…,yl-1}作为 MFP 输出;}
```

4　总结

介绍了概念格和 Web 日志挖掘的思想，通过将概念格应用到 Web 日志进行挖掘，可直接得到频繁项目集从而减少了候选项集的产生，并且可实现 Web 日志挖掘的动态更新，具有很强的扩充与实用性。从用户访问日志中发现可信度较高的关联规则和支持度较高的访问路径，显然用户的感兴趣程度是非常高的，它们为完善网站结构、完善系统性能以及构建智能化 Web 站点等提供了有利的依据。根据这些依据，网站设计者可以考虑是否在两个资源之间增添链接或者进行网站拓扑重构，以完善网站结构，减少用户的点击次数和等待时间，从而提高用户的满意度，进而增大客户量。

参考文献

1 Canler B, Wille R. Formal Concept Analysis: Mathematical Foundations[M]. Berlin: Springer, 1999.

2 钟珞等，Web 使用挖掘研究及实现[J]，微机发展，2005，15(1)：33－35。

3 王旭阳、李明，基于概念格的数据挖掘方法研究[J]，计算机应用，2005，25(4)：827－829。

4 李林、崔志明，用户 Web 日志序列模式挖掘研究[J]，微机发展，2005，15(5)：119－121。

5 Dimitrios Pierrakos. Web Usage Mining as a Tool for Personalization: A Survey [J]. User Modeling and User－Adapted Interaction 13：311－372，2003.

6 赞爱国、王新辉，一种基于 Web 日志文件的信息挖掘方法[J]，计算机应用，2004，24(6)：57－59。