计算机科学2000Vol.27№.2

万维网知识挖掘方法的研究

The Research of Web Mining

沈达阳
（汕头大学计算机科学研究所　汕头515063）

孙茂松
（清华大学计算机系　北京100084）

Abstract The application of knowledge mining technology, is the key to improve the information searching ability of World Wide Web. Aiming at different types of information, the paper introduces several Web mining technologies, which are applied in ParaSite, DRS and WebKB respectively, after comparing their characteristics, an idea of building common Web mining systsem is presented.

Keywords World Wide Web, Knowledge mining, Web mining, Information searching

1. 引言

万维网（World Wide Web）的出现使计算机拥有海量的信息资源，然而这些信息却很少以计算机可理解的结构存在，因为，万维网上的页面本来就是以人，而不是计算机为其阅读对象的。因此，复杂的文本结构、图像、声音等多种信息的存在，既把万维网变成一种丰富多采的媒体，又造成了计算机对万维网信息进一步处理的障碍。

目前，处理万维网信息的最有效，也是最广泛的手段，是万维网的搜索索引。通用搜索引擎所用搜索索引建立的方法，主要有以下两种：

1）人工维护的搜索索引：如 Yahoo！和一些虚拟图书馆系统，利用大量的人力浏览 Internet 页面，对其进行分类，因此，建立的搜索索引覆盖面较窄，但比较精确，一般只适用于相对小型、静态的信息（如大学的 Web 信息），因为编辑人员不可能跟上庞杂，快速增长的信息（如：个人主页或美国经济新闻，某个主题的学术文章等）。

2）Internet 搜索蜘蛛（Internet spider）产生的搜索索引：如 AltaVista，Lycos，Excite 等，这些自动生成的索引覆盖面很广，但在精确性方面却很差。用户往往被迫人工从庞杂的反馈中，过滤出所需的信息。

因此，利用知识发现（KDD）和自然语言处理（NLP）方法从 Internet，尤其是超级文本组成的万维网中发现知识，是提供高精度和大覆盖面搜索索引，改善 Internet 信息搜索效果的关键。通常，我们把这些方法称为万维网的知识挖掘（Web Mining）

2. 特定知识的挖掘方法

2.1 结构关系知识的挖掘方法

Spertus[1] 在 ParaSite 系统中提出一种比较简单的万维网结构知识挖掘方法，她把超级链结按方向分成这样几类：①向上链结：所指向的文档目录是同一服务器的上层目录；②向下链结：所指向的文档目录是同一服务器的下层目录；③交叉链结：所指向的文档父目录与本目录没有父子关系；④向外链结：所指向的文档在其他的服务器。

通过考察 Yahoo！的目录结构，可发现以下的启发性规则：

·（分类封闭）在一个层次索引中，从一个页面开始，由向下或交叉链结得到的页面，其主题和原始页面的主题相关。

·（索引连接）从一个索引开始，任何由本页面的向外链结得到的页面，其主题很可能是相同的。

·（重复索引连接）从一个索引 P 开始，沿着向外链结指向索引 P′，通过 P′ 的向外链结得到的页面很可能和 P 的主题相关。

·（作者定位）如果 P 是个人主页，而文件 P′ 在 P 下面的目录中，那么 P′ 的作者很可能就是个人主页 P 所标识的人。

·（目录和链结）如果页面 P 是在 P′ 的上层目录中，而存在从 P′ 到 P，和 P 到 P′ 的超级链结，那么 P 就有可能是个人主页。

·（空间位置）如果 URL（Unified Resource Locator，即 Internet 资源的地址）U1 和 U2 在同一个页面中距离很近，那么它们就有可能具有类似的主题或特征。

利用这些规则，ParaSite 系统中的搜索蜘蛛就可

以得到页面之间的结构关系，从而实现发现个人主页、搜索新页面和自动索引等目的。

2.2 个人页面的挖掘方法

为了精确地定位万维网个人主页，Shakes[4]在个人主页搜索系统 Aboy! 中，使用了 DRS（Dynamics Reference Sift）页面过滤方法，这种基于元搜索的过滤方法，通过利用内在的知识库，来挖掘特定搜索引擎的反馈，从而提供最大限度的包容性和高精确率的实时信息。它包含下面的关键元素。

1）信息引用源：如从通用搜索引擎反馈来的内容广泛地候选 URL；

2）交叉过滤器：用于过滤候选 URL 的模块，它基于一个附加的、与1相独立的、不同类型的信息源（如 Email 地址的数据库）；

3）启发性过滤器：采用领域有关的启发性信息，来分析候选 URL 所指向文本的内容，从而增加精确度；

4）分类桶：用于把候选 URL 分在有等级和标号的桶（匹配和不命中）之中；

5）URL 生成器：当1）—4）不能产生有用的候选时，用于合成候选的 URL；

6）URL 模式抽取器：利用成功的搜索结果，学习机构或国家 URL 的通用模式（即某一机构或国家 URL 的构成模式）。该模式被5）所采用。

通过 Aboy! 的万维网界面，用户提供姓名以及机构、国家名称（可选的）。DRS 把姓名传给信息引用源 MetaCrawler（华盛顿大学的元搜索引擎）之后，接收到许多主页候选；它同时把名字传给 email 目录服务（WhoWhere 和 IAF），得到许多 Email 地址。然后利用内部数据库确定目标单位的 URL，如果用户填完所有的输入区域，所有信息就传给下一步的过滤。

在过滤这一步，DRS 采用两种类型的过滤器来筛选索引：基于交叉的过滤和基于启发的过滤。前者是利用目标人员的机构名和 email 地址的交叉排斥，来排除冗余信息，而基于启发的过滤是使用人名和个人主页特征来确认个人主页的可能候选。

在过滤了明显不正确的 URL 之后，DRS 系统把剩下的 URL 排列在一些称为桶的类别中。对于主页领域，每个桶有三个属性：①目标人的名字和候选 URL 的匹配程度；②候选的 URL 和指定的机构、国家匹配程度；③页面和一般个人主页的相似程度。如果三者都有精确的匹配，应该就是个人主页，部分相匹配的桶用按领域相关的主观标准来排队。

DRS 体系最关键之处是它的模式抽取。它能够从成功的搜索结果中抽取 URL 的通用模式，记下来，用于将来的使用。在以后的搜索中，如果搜索失败，它就能够自动合成可能的个人主页的 URL。这保证 DRS 具有一定的适应学习能力，使其性能在应用的过程中，能逐步提高精确率和覆盖面。

3. 通用知识的挖掘方法。

WebKB[3]是卡内基·梅隆大学（CMU）的万维网信息挖掘研究项目，致力于建造一个大型的知识库，作为万维网的镜像。这样的知识库能够推动对万维网更有效的信息抽取，以及对基于万维网的知识推理和问题求解的支持。为了建造这样的知识库，Carven 等许多人提出了多种形式的知识挖掘方法。

WebKB 中的信息挖掘方法采用了许多比较成熟的 NLP 技术，同时又结合万维网的信息特征，对原来的 NLP 技术做了许多改进。在 WebKB 中，把页面信息分为以下两种内在的表示类型（在大学信息的应用范围内）：①类别，如：Person，Student，Department，Research，Project，Course 等；②关系，如：AdvisorOf，MemberOf 等。另外，人工标注了一些训练用的超级文本，如课程的主页，工作人员的主页等（WebKB 标注了 CMU 计算机系所有主页的类别和页面之间的关系），然后，WebKB 假设：

·每一种内在表示的具体实例都由一个或多个超级文本的连续段来表示。所谓"超级文本的连续段"就是单独的 Web 页面，或在一个 Web 页面中连续的字符串，或通过超级链结联系在一起的 Web 页面的集合。例如，一个 Person 实例的描述，可能是单独一个页面，如个人主页，或者任意一种页面中介绍个人的文本，或者由若干相互连接的页面一起介绍的个人信息。

·关系 R 的每个实例 R(A,B)，在 Web 上用三种方式表示：①R 用一段超级文本表示，该文本是用来连接文段 A 和 B 的。②代表 A 的连续文段包含代表 B 的连续文段。③通过本系统的学习，认为代表 A 的文段和 B 有某种关系。

因此，WebKB 需要解决三种基本的学习任务：1）通过对超级文本主体部分的分类来辨别其类别实例。2）通过对超级文本链的分类来辨别其关系实例。3）通过从 Web 页面中抽取小文本片段来辨别类和关系的实例。

WebKB 通过以下两种方法来学习。

3.1 文本分类的统计方法

文本分类的统计方法包括利用标记好的数据来为每个类建造统计模型，然后选择在统计上和新页面中的词最为类似的类别，来实现对其进行分类。

在可学习的文本分类算法中，有许多很流行的 NLP 方法，如 Bag of Word，Bigram 等。文[4]采用的是 Naive Bayes 方法（基于 Kullback-Leibler 分布做了一些修改）。

$$c' = \arg\max_c \left[\frac{\log Pr(c)}{n} + \sum_{i=1}^{T} Pr(w_i|d) \log\left(\frac{Pr(w_i|c)}{Pr(w_i|d)}\right) \right]$$

其中，n 是 d 中词的个数，T 是系统词典的大小，w_i 是系统词典中第 i 个词，那么 $Pr(w_i|c)$ 就是从 c 类文本中得到词 w_i 的概率，而 $Pr(w_i|d)$ 就代表了文本 d 中词 w_i 出现的概率。根据以上公式，把文本 d 归入类别 c'。

和传统的一般文本分类相比，Web 页面分类的特点是，超级文本的冗余隐含着许多不同的表示方法。除了利用出现在页面中的词来分类之外，WebKB 还可以用这样的分类方法：1)在指向页面的超级连接(即出现在〈a〉…〈/a〉中的文本)中出现的词；2)仅仅出现在 HTML 文本的文章标题(即〈title〉…〈/title〉)或头部(即〈head〉…〈/head〉)中的词。对于某些类别，这些方法分类的精确度更高。

3.2 一阶文本分类方法

由于 Web 超级文本具有复杂的结构，Web 也可以用有向图表示出来，即：页面对应图中的节点，超级链结对应图中的边。3.1节的统计方法能够考虑到每个节点的词，以及连接每个节点的边中的词，然而，对于某一给定页面，如果要把它的连接模式，或相邻页面中出现的词，纳入辨别模型中，3.1节的方法就无能为力了。如给出以下的规则：

```
IF 页面 A 包含词(textbook,TA)AND 页面 A 连接到
   一个包含词(assignment)的页面
THEN 页面 A∈类别 Courese
ENDIF
```

象这样能够体现图的一般特征的规则，就需要一阶的表示方式。

FOIL[5]是一种用于学习 Horn 子句的最大覆盖算法。为了适应超级文本的结构特征，需要关系型的学习器来表示图结构，又需要具有特征选择功能的统计型学习器，来刻画图中的节点和边，因此，一般采用改进的 FOIL 算法：FOIL-PILFS(FOIL with Predicate Invention for Large Feature Spaces)，其算法如下：

```
输入：目标关系的正例集合 T+，反例集合 T-，和背
   景关系集合 B
BEGIN
初始化子句 C：R(X1,…,Xk)：-true
T=T+∪T-
WHILE (T 包含反例)AND (C 不复杂)
    BEGIN
    CALL predicate_invention(得到新的候选文字)
    把候选文字(包括原来的背景谓词和新的谓词)加
       入 C
    用更新的 C 来改变 T 中的二元组
        FOR EACH 新的谓词 Pi(Xi)
    BEGIN
        IF Pi(Xi)被 C 选中 THEN 把 Pi(Xi)加入到
           B
    END
输出：得到的子句
END
```

其中 predicate_invention 函数是 CHAMP[5]算法的扩展，不过 CHAMP 算法仅仅在基本关系算法找不到匹配的子句时，才构造新的谓词，而 FOIL-PILFS 在搜索子句的每一步都寻求构造新的谓词。特别是，在搜索过程的每一步，如果子句包含变量 $X_1,\cdots,X_n$，对于任何一个变量 X_i，如果训练文本集中有与其类型对应的文本，就考虑构造新谓词来刻画它。

```
FUNCTION Predicate_invention
输入：部分字句 C，每个类型的文本集合，参数 e(表示
   选择范围的大小)
BEGIN
    FOR EACH C 中的变量 Xi
        FOR EACH 和 Xi 类型有关的文本 Dj
        BEGIN
            S+ =在 Dj 的正例二元组中和 Xi 结合常
               量(词)
            S- =在 Dj 的反例二元组中和 Xi 结合常
               量(词)
            根据词/类别互信息，对 S+∪S- 中的词
               排序
            n=|S+∪S-|×e
            F=排在前 n 位的词
               用 Naive Bayes 算法学习 Pi(Xi)的
               权重-特征集 F，训练 S+∪S-
    END
输出：所有经过学习的 Pi(Xi)
END
```

在类别学习过程的表示方法中，系统发现的谓词意义如下：

Has_word(Page)：每个这样的布尔谓词表示页面中出现单词 word

Link_to(Page,Page)：每个这样的关系代表数据集合中连接页面的超级链结。第一个参数是链结出现的页面，第二个是被连接的页面。

以下是训练中学得的规则，可用于辨别 student 和 faculty 两种类别的页面：

```
student(A):-not(has_data(A)),not(has_comment
   (A)),link_to(B,A),has_jame(B),has_paul(B),
   not(has_mail(B))
faculty(A):-has_professor(A),has_ph(A),
   link_to(B,A),has_faculty(B)
```

在关系学习过程的表示方法中，系统发现的谓词意义如下：

class(Page)：页面 Page 是属于类别 class 的；

like-to(Hyperlink,Page,Page)：连接页面的超级连结；

has_word(Hyperlink)：表示词被包含在每个超级链结的标记文本中；

has_neighborhood_word(Hyper link)：表示词被包含在每个超级链结的“邻居”中，所谓超级链结的“邻居”，就是包含超级连接的段落、列表、表格、标题等。

以下是训练中学得的规则：

```
member_of_project(A,B):-research_project(A),
    person(B),link_to(C,A,D),link_to(E,D,B),
    neighborhood_word_people(C)
department_of_person(A,B):-person(A),depart-
    ment(B),link_to(C,D,A),link_to(E,F,D),
    link_to(G,B,F),neighborhood_word_graduate
    (E)
```

3.3 文本段的抽取

有时候，要抽取的信息不能用 Web 页面或页面之间的关系来表示，而仅仅是嵌在页面中的文本段。同样是基于 FOIL，WebKB 开发了一种信息抽取的学习算法：SRV[4]，它是采用爬山法的一阶学习器，算法的输入是一些页面，页面中所要抽取信息的实例已经被标记出来。算法的输出是一些用于信息抽取的规则，而信息抽取的过程就是检查任何长度的文本，试图找到能够和规则匹配的文本片段。抽取出来的规则如下：

```
ownername(Fragment):-
    some(Fragment,B,[],in_title,true),
    length(Fragment,<,3),
    some(Fragemnt,B[pre_token],word,"GMT"),
    some(Fragment,A,[],longp,true),
    some(Fragment,B,[],word,unknown),
    some(Fragment,B,[],quadrupletonp,false).
```

其中，Fragment 是要匹配的文本段，length(Fragment，Relop，N)用于限制抽取信息中词的个数，其中 Relop 是比较运算符，N 是数字。

some(Fragment，Var，Path，Attr，Value)用于检验 Fragment 中词的特性，其中，Var 是当前词单元，Path 是被检验单元的路径(和本单元的关系)，Attr 是要检验的属性，Value 是属性的值。

下面是和以上规则匹配的万维网页面，其中 Bruce Randall Donald被抽取出来，被认为是本页面的属主。

```
LAST-MODIFIED: WEDNSDAY, 26-JUN-96 01:37:46
    GMT
<TITLE>BRUCE RANDALL DONALD</TITLE>
<H1>
<IMG
    SRC = "FTP: // FTP. CS. CORNELL. EDU/PUB/
        BRD/IMAGES/BRD. GIF">
<P>
BRUCE RANDALL DONALD<BR>
```

4. 各种挖掘方法的比较

ParaSite 的方法比较简单，精度不高，而且发现的知识十分有限；DRS 方法在专业范围内可用性比较好，也有一定的学习功能，但应用范围比较有限，一般只用于某个特定类别的信息搜索。WebKB 中提供了多种信息挖掘的方法，功能很强，既可以适应多个特定的类别，理论上也可以用于挖掘通用的信息，但它需要有一定规模的训练文本，而且，随着信息类型的通用化，训练文本的规模和质量也要随着提高，因此实现的难度也随着变大。这些方法，各有其特点，在构造万维网挖掘系统时，要综合使用这些方法。一般地说，ParaSite 中的启发性规则对大部分的系统都有所帮助，也容易实现，DRS 则针对比较专门的信息类型，而 WebKB 可以适应规模较大的系统。不同的系统，可以针对各自的特点，对这些方法进行剪裁。例如，在我们开发的 PersonIndex[6,7]中，使用了 ParaSite 中的一些启发性规则，以及 DRS 中的 Email 过滤和 URL 模式。由于 WebKB 中通用的训练文本很难得到，我们就使用专业的语料库来代替，如人名语料库，机构名语料库等来做局部的训练，也取得了很好的效果。

主要参考文献

1 Spertus E. ParaSite: Mining Structural Information on the Web. In: Proc. of the 6th World Wide Web Conf. 1997

2 Shakes J, Langheinrich M, Etzioni O. Dynamic Reference Sifting: A Case Study in The Homepage Domain, Proc. of WWW6

3 Craven M. Learning to extract symbolic knowledge from World Wide Web: [Technical report]. CMU CS Dept, 1998

4 Lewis D. Training algorithms for linear text classifiers. In: Proc. of the 19th Annual Int. ACM SIGIR Conf. 1996

5 Quinlan J R. FOIL: A midterm report. In: Proc. of the 12'h European Conf. On Machine Learning, 1993

6 Shen Dayang, YuBin, Lin zuoquan. WebIndex, the Intranet Information Collecting Agent. In: Proc. of APWeb98, Beijing, 1998

7 沈达阳，孙茂松. Internet 中文个人信息搜索，中文信息学报(已录用)，1999

(上接第90页)

空间模型相比平均精度有明显改善。前者为37%，后者为51%，增加14%。

参 考 文 献

1 姚天顺，等. 自然语言理解. 清华大学出版社，1995

2 吴立德，等. 大规模中文文本处理. 复旦大学出版社，1997

3 刘开瑛，等. 中文文本中抽取特征信息的区域与技术. 中文信息学报，1998，12(2)

4 Yan T W, Garcia-Molina H. SIFT-A Tool for Wide-Area Information Dissemination. In: Proc. of the 1995 USENIX Technical Conf.

5 Yan T W, Garcia-Molina H. Distributed selective dissemination of information. In: Proc. of the Third Intl. Conf. on Parallel and Distributed Information System. 1994. 89～98

6 Callan J P. Passage-Level Evidence in Document Retrieval. In: Proc. SIGIR' 94. 1994

7 Salton G, et al. Automatic Structuring and Retrieval of Large Text Files. Comm. of the ACM, 1994, 37(2): 97～108

8 Buckley C, et al. Automatic Query Expansion using SMART: TREC-3. In: Proc. of the Third Text Retrieval Conference. 1995