维普资讯 http://www.cqvip.com

基于重复模式的 Web 信息抽取

高 强 张敬之 耿 桦 潘金贵
(南京大学计算机软件新技术国家重点实验室 南京 210093)

摘 要 网页中的大量数据记录往往以重复的 HTML 结构进行有规律的组织,从而形成一致的表现形式。根据这一特征,本文给出一种基于重复模式的 Web 内容抽取方法。通过使用一种叫做后缀树的数据结构,分析页面结构中所包含的重复模式,进而从模式的实例中抽取出对应的数据记录。

关键词 Web 信息抽取,重复模式,后缀树

Web Information Extraction Based on Repeated Pattern

GAO Qiang ZHANG Jing-Zhi GENG Hua PAN Jin-Gui
(State Key Lab. for Novel Software Technology, Nanjing University, Nanjing 210093)

Abstract In a data-rich, multiple-record Web page, the "useful and relevant" information items are usually arranged regularly and compactly, with similar pattern of HTML tags and consistent style of presentation. In other words, the semi-structured Web document often has its own structured features. Based on this observation, this paper presents an automatic approach to extract such kind of information. We use Suffix Tree to obtain the repeated patterns in HTML tags of the target page, and then a series of heuristic rules are applied to choose the appropriate patterns. The information items will be extracted from the instances of these patterns. Experimental results indicate that our approach works effectively in most cases.

Keywords Web information extraction, Repeated pattern, Suffix tree

1 引言

World Wide Web 业已成为当今最庞大而重要的信息来源。然而,WWW 上的文件大多以 HTML 页面的方式存在。一方面,HTML 的 tag 主要用于表达视觉浏览的效果,其本身语法并不严格,也不包含关于内容的语义信息,半结构化特征明显。另一方面,Web 页面中往往存在很多混杂信息,如广告、导航条、版权声明等等,页面的主要内容淹没于这些"噪声"之中。此外,不同的网站设计风格迥异,而且可能经常变更。这些都给 Web 信息的有效利用带来了困难。

Web 信息抽取(Web IE)是一种从 Web 文档中抽取出有用数据的技术。传统的 Web 信息抽取通常采用机器学习的方式,以人工标注的方式训练一系列样本,抽取系统对样本进行学习,产生相应规则,将这些规则应用于目标网页,抽取出信息。这一过程也叫做包装器生成(wrapper generation)。典型的 wrapper 系统有 WIEN[8],STALKER[7],WHISK[9]等。文[1]中对 Web 数据抽取工具分类,并介绍了各自的原理和特点。

然而,手工标注样本毕竟是一项耗时而繁琐的任务,因此一些自动化的 Web 信息抽取方法越来越得到重视。在文[4]中,作者使用多种启发式规则对候选的 HTML tags 各自独立进行评价,最终确定其中一种,作为不同数据区域之间的边界。文[2]中强调,大量的网页是根据后台的数据库经由一定的显示模板所生成的。根据这一特点,利用一组类似网页作为输入,由特定算法推导出生成这些页面所采用的模板,最后抽取出网页中的数据记录值作为输出。文[3]首先对网页的 HTML token 做二元编码,进而利用 Patricia tree 发现编码序列中的候选模式,采取 string alignment 方法产生一系列抽取规则,供用户选择。

在许多网页中,构成同类别数据记录的 HTML tag 序列往往形成一定的重复模式,使得多个数据记录呈现出一致而紧凑的显示效果。而页面中的"噪声"内容,其 HTML tag 序列则松散、混杂而无规律。由于动态网页技术的普及,以及大量网页设计工具的使用,这一特征也将愈加普遍和明显。据此,本文提出一种基于重复模式自动抽取网页信息的方法,以目标网页的 HTML tag 序列构造对应的后缀树,借助后缀树挖掘序列中的重复模式,进而从模式所对应的实例(instance)中抽取出数据记录的信息。

2 页面 tag 序列的后缀树构造

2.1 HTML 页面预处理

将 HTML 源文件看作字符串来获取其中的 tag 序列,是一种直观的做法。但是,由于 HTML 不要求严格的语法,往往可能存在 tag 的缺失和混杂;同时网页中还可能包含 javascript、注释等无关内容。这些都会对之后的处理造成影响。我们选择先对目标网页进行解析,利用 HTML 解析器对源文件进行规范化整理。另一方面,HTML 的结构特征在解析的结果——DOM 树中有着清晰的反映,而 HTML 源文件难以直观体现这种特征。

将解析所得的 DOM 树依前序遍历的顺序展开其节点,得到所需的 HTML tag 序列。在展开过程中,将文本节点均表示为〈text〉。而诸如 script,comment 等节点由于对结构并不存在影响,均可忽略。

例如以下 HTML 片断:

高 强,张敬芝,耿 桦 硕士研究生,主要研究方向为网络多媒体技术、Web 挖掘;**潘金贵** 教授,博士生导师,主要研究方向为中间件、Agent 技术、多媒体远程教育。

维普资讯 http://www.cqvip.com

〈ul〉
〈li〉〈a〉DOW〈/a〉〈span〉-18.73〈/span〉〈/li〉
〈li〉〈a〉S&P〈/a〉〈span〉-4.96〈/span〉〈/li〉
…
〈/ul〉

解析后对应的 DOM 树部分如图 1。

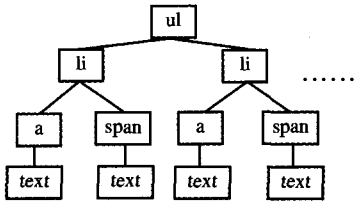

图 1　HTML 片断的 DOM 树

展开后所得的 tag 序列为

〈ul〉〈li〉〈a〉〈text〉〈span〉〈text〉〈li〉〈a〉〈text〉〈span〉〈text〉…

2.2　构造后缀树

后缀树[5,6]是一种重要的数据结构，在文本检索、生物计算等领域有着广泛的应用。构造后缀树是进行后期应用的前提和关键。文[6]介绍了根据输入字符串快速构造后缀树的算法，其空间和时间复杂度都是 $O(n)$，其中 n 是输入串的长度。我们将该算法作用于输入的 HTML tag 序列，从而构造出所需的后缀树。

假设 tag 序列 $S=t_1t_2\cdots t_i\cdots t_n\$$，则 $S_i=t_it_{i+1}\cdots t_n\$$ $(1\leqslant i\leqslant n)$，称为 S 在位置 i 的后缀。$\$$ 为序列结束符，规定 $\$$ 不同于 S 中其他任何的 tag。又称子串 $S'=t_it_{i+1}\cdots t_j$ $(i\leqslant j\leqslant n)$ 在 S 中的位置是 i。S 中的位置约定从 1 开始。

定义 S 对应的后缀树

$T_S(S,\Theta,E,N,L)$

其中：S 为输入 $HTML\ tag$ 序列；Θ 为 T 的根节点；E 为 T 中所有边的集合；N 为 T 中所有内部节点的集合；L 为 T 中所有叶节点的集合。

例如序列 abcabc$，其中 a,b,c 分别代表了一种 HTML tag。所构造的后缀树如图 2 所示。

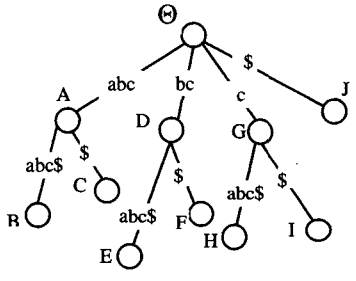

图 2　由序列 ABCABC$ 构造的后缀树

T 中的任一边 $\varepsilon\in E$ 可标记为 S 的一个子串，记作 $label(\varepsilon)$。如图 2 中 $label(\Theta D)=bc$，$label(DE)=abc\$$。

设从根节点 Θ 到任一节点 Φ 的路径经过边 $\varepsilon_1\varepsilon_2\cdots\varepsilon_n$，则该路径的标记为 $label(\varepsilon_1)label(\varepsilon_2)\cdots label(\varepsilon_n)$，记作 $path(\Phi)$。如图 2 中 $path(E)=bcabc\$$。

后缀树 $T_S(S,\theta,E,N,L)$ 具有以下一些重要的性质：

性质 1　T_S 包含有 S 的所有后缀，且每个叶节点唯一对应了其中的一个后缀。若叶节点 $X\in L$，X 所对应的后缀为 $path(X)$。节点 X 中记录该后缀在 S 中的位置。

性质 2　T_S 的每个内部节点都对应了 S 中的一个重复模式。对内部节点 $\Psi\in N$，Ψ 所对应的重复模式记为 PTN_Ψ，有

$PTN_\Psi=path(\Psi)$。

性质 3　内部节点 Ψ 之下的所有叶节点与 PTN_Ψ 的所有实例一一对应，且叶节点所记录的位置正是其对应实例在 S 中的位置。

严格地说，在性质 2 中，内部节点 ψ 所能代表的模式包括 $path(\psi)$ 的所有前缀。如图 2 中 $path(A)=abc$，其中还包括模式 ab 和 a。但由于我们的最终目的是从重复模式的实例中抽取数据记录，而模式 ab 和 a 的实例都包含在模式 abc 所对应的实例之中，不会再提供更多的信息，因此仅需要关注内部节点的路径本身所代表的模式就已经足够。

3　选取有效的重复模式

作为一个符合要求的重复模式，其实例能够分别对应到网页中的数据记录，以达到信息抽取的目标。我们通过遍历后缀树的内部节点，可以得到 HTML tag 序列中所有的重复模式，但其中的大多数都是不符合要求的。因此，需要利用一系列启发式规则，选取出那些合理有效的模式。

3.1　根据模式特征进行过滤

结合 Web 信息抽取的语境，通过考察模式的自身特征，可以初步判断其是否符合要求。在访问后缀树内部节点的过程中，考察其对应的模式，应用以下一组规则将不符合要求的过滤掉：

规则 1(实例数量)　如果一个重复模式的实例数目太少，即该模式出现的次数太少，其包含有用信息的可能性就比较低。在访问后缀树内部节点时，计算该节点下所有叶节点的数量，也就是该模式的实例数量(根据性质 1)。若少于设定的最小实例数量，则淘汰该模式。

最少实例数量作为阈值可以由用户根据目标网页情况指定。

规则 2(模式长度)　如果模式的长度过短，即模式包含的 tag 数过少，尚不足以表达数据记录的结构特征，这类模式将被淘汰。

最短模式长度也可作为一个阈值。

规则 3(关键 tag)　我们的最终目标是抽取网页中的信息。如果模式中没有包含文本 tag，即〈text〉，则信息也无从谈起。有的抽取系统可能关注网页中的超链接，则需要模式中含有〈a〉。然而，如果模式中仅仅有链接和文本，也不足以反映出 tag 序列结构上的特征，也不具备重复模式的意义。

根据以上规则，忽略掉那些不符合这些基本要求的模式。但仅凭模式自身的特征进行取舍是不够的，还需要利用模式的实例集做进一步精化。

3.2　处理候选模式的实例集

在过滤处理之后，仍然会遗留大量不符合要求的候选模式。通过对这些模式的实例集进行分析，进一步选取合理有效的模式，同时对实例集作出适当的处理。

根据 2.2 节中后缀树性质 3，通过访问内部节点 Ψ 之下所有的叶节点，可以获得模式 PTN_Ψ 的全部实例。将 PTN_Ψ 的全部实例按照在源序列中的位置进行排序，以二元组 $\langle p_i, p_i'\rangle$ 来表示模式 PTN 的第 i 个实例，p_i 和 p_i' 分别表示该实例的首尾 tag 在源序列中位置。显然，有 $p_i'=p_i+|PTN|-1$，其中 $|PTN|$ 为该模式的长度。

维普资讯 http://www.cqvip.com

应用以下规则对模式的实例集进行精化：

规则 4（消除重叠） 如果两个实例 $\langle p_i, p_i' \rangle$ 和 $\langle p_j, p_j' \rangle$ 满足 $p_i \leqslant p_j'$ 且 $p_i' \geqslant p_j$，则称这两个实例之间存在重叠。

我们期望有效模式的各个实例分别代表着一条数据记录。如果同一模式的不同实例之间存在重叠，就意味着页面中的某一部分同时属于不同的数据记录。例如序列 ababab，对候选模式 *aba* 而言，有实例〈1，3〉、〈3，5〉存在着重叠。位置 3 上的 *a* 同时被两个实例所占据。很明显，这是不符合要求的。因此，对每个候选模式，我们依次检查并删除产生重叠的相邻实例。

另一种稍复杂的情况为，不同模式的实例之间存在重叠。在这种情况下去除重叠的意义在于，网页内容可以用某种模式来划分成数据记录，也可以用另一种模式划分。不同的划分方法之间自然存在重叠，但必定有一种模式更好，由该模式的实例所代表的数据记录也更合理。

因此，候选模式的实例集之间两两进行比较，检测是否存在重叠。如果存在，则对两个候选模式进行比较。比较结果较优的模式，其实例将被保留；而较差的模式则被选中，去除其包含重叠部分的实例。

首先比较两个模式涉及重叠的实例的数量。当模式 PTN_1 的实例恰好"跨过"模式 PTN_2 的两个实例，从而造成 PTN_1 涉及重叠的实例数少于 PTN_2。这种情形下，为了在消除重叠后可以保留更多的实例，PTN_1 将被选中，删除其包含重叠部分的实例。例如，序列 abcabcabc 的两个候选模式 *abc* 和 *ca* 进行比较，模式 *abc* 涉及重叠的实例为〈1，3〉，〈4，6〉，〈7，9〉；模式 *ca* 涉及重叠的实例为〈3，4〉，〈6，7〉，则模式 *ca* 被选中。

如果两个模式有相同数量的实例形成重叠，则进一步根据模式的特征进行比较。参考 3.1 节的规则 1 和规则 2，实例数更多以及长度更长的模式有着更大的优势被保留。

进一步，倘若实例的数量和模式的长度均相等，我们将涉及重叠的实例映射回序列的 DOM 树，借助 HTML 的层次结构特征对实例进行比较（如 2.1 节中提到）。

仍以 2.1 节中的序列为例：

〈ul〉〈li〉〈a〉〈text〉〈span〉〈text〉〈li〉〈a〉〈text〉〈span〉〈text〉

…

有如下候选模式：

PTN_1 =〈li〉〈a〉〈text〉〈span〉〈text〉

PTN_2 =〈a〉〈text〉〈span〉〈text〉〈li〉

两者模式长度相等，且假设实例数量相同。PTN_1 形成重叠的实例为〈2，6〉，PTN_2 的为〈3，7〉，考虑其映射到 DOM 树中的部分，在图 3 中分别用实线区域和虚线区域表示。

将实例中的所有 tag 映射为 DOM 树中的节点，进而考察这些节点的父节点。如果此父节点不在实例的范围内，称之为该实例的区域外父节点（Parent Outside）。我们通过计算实例的区域外父节点数（记为 NPO）来度量实例的结构优劣。该数越小，表明实例具有较好的结构特征。如图 3 中，实例〈2，6〉的 NPO 为 1，即节点 1〈ul〉。而实例〈3，7〉的 NPO 为 2，即节点 1〈ul〉和节点 2〈li〉。因此，结构特征较差的实例〈3，7〉被选中淘汰。

通过以上一系列的比较和处理，实例集中的重叠现象将被消除。由于这一过程中对模式的实例有大量的删除操作，有必要再次应用规则 1，对实例数量不再符合要求的模式进行过滤。

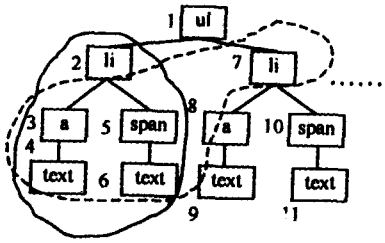

图 3 DOM 树中的实例区域

规则 5（覆盖率） 经过规则 1、2、3、4 的应用，所余下的模式及其实例已经基本符合抽取信息的要求。提出覆盖率的规则是为了满足对网页信息关注程度不同的需要。同时，有些页面中所包含的侧边导航栏具备很好的重复模式，很难根据以上规则进行排除。

定义模式的覆盖率为模式的当前实例中包含的实际文本的字节数与整个网页包含实际文本的字节数的比值。这里的实际文本，是指〈text〉节点中的文本，即页面可见的文本，而非网页的 HTML 源文件的内容。通过设置覆盖率大小的阈值，可以对最终信息的抽取程度作出直观的限定，也可以较容易地解决侧边导航栏的干扰。

至此，HTML tag 序列中的有效重复模式已经获得，进而可以从其实例中直接抽取对应数据记录的信息。

4 实验

在我们的 HTML2RSS 系统中实现了本文提出的方法。该系统主要用于将 HTML 页面中的数据信息转化成 RSS[10] 格式输出。因为所抽取的数据记录来自重复模式的实例，而实例在抽取过程中是以其在网页的 HTML tag 序列中的位置表示的，所以抽取所得的数据记录信息可以很方便地与网页内容进行对应比较。

实验结果主要根据查准率和查全率进行评价：

$$查准率=\frac{提取的有效数据记录数}{提取的数据记录总数}$$

$$查全率=\frac{提取的有效数据记录数}{网页数据记录总数}$$

我们从 http://dir.yahoo.com/ 和 http://www.nifty.com/category/ 提供的网址列表中选取了 70 个网站、共 130 个页面进行实验。

抽取过程需要预先设定一些阈值（3.1 节中提到）。鉴于最终生成 RSS 的需要，将关键 tag 设定为〈a〉和〈text〉。将最少实例数量（MinIns）和最短模式长度（MinLen）分别设置不同的值进行多次测试，以考察这些阈值对抽取效果的影响。实验结果如表 1，其中查准率和查全率为所有被测页面结果的平均值。

表 1 不同参数值的抽取效果

参数	MinLen = 5		MinLen = 7	
	查准率	查全率	查准率	查全率
MinIns=3	76%	90%	85%	84%
MinIns=5	89%	85%	91%	80%

从实验结果看出，基于重复模式的方法能够较为有效地提取出网页的数据记录信息。

进一步分析预设参数对抽取效果的影响。MinIns 值增

（下转第 221 页）

维普资讯 http://www.cqvip.com

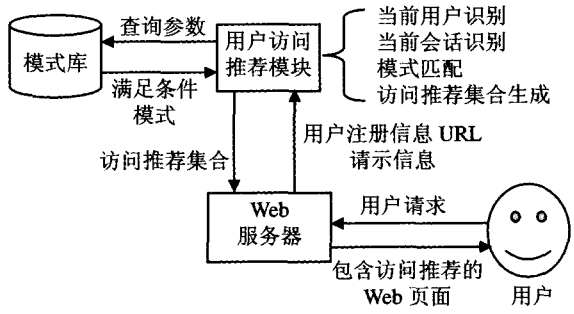

图 7　用户访问推荐示意图

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, I4:249～60

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