因特网语料自动下载分析软件的设计

朱凯　周杰　何婷婷

（华中师范大学计算机科学系，　武汉 430079）

E-mail: hett@163.net

摘要： 随着计算机应用的发展和普及，特别是因特网的普及，获取语料、建立大规模语料库变得越来越容易。本文讨论了如何从因特网上下载语料，如何解析 HTML 页面并提取出其中对语料库有用的语料数据的方法。

关键词： 语料库；网页；下载；HTML；

引言

随着因特网的普及，基于因特网建设超大规模的平衡语料库已经成为可能。由于超大规模通用平衡语料库需要大量的实际语言文字材料作为基础，如果我们在建设基于因特网的语料库的时候由手工下载分析成千上万的网页，显然不可能，因此好的语料自动下载分析软件就显得尤为重要，我们设计的语料自动下载分析软件较成功地应用于基于网络的大规模平衡语料库的建设，本文介绍它的基本设计思想。

1 下载分析软件设计

基于因特网的大规模语料库需要下载大量的 HTML 页面，而因特网上的许多页面相比语料采集工作的需要具有以下不足：① 结构复杂，含有大量的 HTML 标签。②有用的语料信息不是很多，或者根本就没有。③可能含有 HTML 语法错误，这为语料分析程序识别有用的信息带来不便。于是，就要想办法下载大批量的结构简单规范、并且含有语料库建设所需要的信息的网页。

虽然现在已经有很多的下载工具，但是并不符合语料采集工作的要求。原因在于：

(1) 下载软件基本上都只能够下载当前某个网站上可以被用户浏览到的网页，如果语料采集需要某个站点以前的资料，这些软件就无能为力了，比如，如果需要某个站点从 1995 年到 1997 年的文章，现有的下载软件就无法满足要求。

(2)下载的页面除了有用的中文语料外，还包含大量的 HTML 标记及其他无用的信息，结构复杂而且不统一，网页分析程序很难进行精确提取，不能满足语料库建设的需求。

我们注意到有些站点，特别是报纸、期刊、杂志类的网站都提供了检索功能，以方便读者检索阅读。通过检索产生出来的 HTML 页面，大都是由服务器通过用户在客户端输入的查询条件，在数据库中提取满足条件的数据，然后由程序自动组合成 HTML 页面返回客户端。正因为通过检索产生的页面是程序自动生成的，所以页面结构规范一致，信息完整，比如作者、写作时间、分类等等信息都包含在里面，几乎没有冗余信息，这就为页面分析处理程序带来了方便。

下面讨论网页自动下载原理。

1.1 浏览器与服务器交互原理

HTML 页面中有一种被称为表单的元素，它的开始标志和结束标志分别为<FORM>和</FROM>，FORM 的 action 属性就指明了服务器端响应的对象，这个对象可以是一个 ASP 页面或 PHP 页面，还可以是一个组件。下面是一个例子片段：

<form method="post" action="/Search.wct?ChannelID=2771" name="FieldSearchForm" LANGUAGE=javascript onsubmit="return FieldSearchForm_onsubmit();">……<input type="text" name="Content" size="40"><input type="hidden" name="SearchWord" >……<input type="submit" name="Submit" value="检索"><input type="reset" name="Submit2" value="重填"></form>

在例子中 action="/Search.wct?ChannelID=2771"表示服务器方响应的是一个名叫 Search.wct 的对象，而且带了一个参数是 ChannelID=2771，<input type="text"……指明该处是一个供用户输入的文本框，而<input type="submit"……就是提交按钮了。当用户在检索栏中输入检索条件，并点击提交按钮时，浏览器将向服务器提交请求，在请求信息中就包含了用户在表单中输入的信息以及服务器方响应的对象名。

要想利用网站提供的检索功能将网站后台数据库的信息下载下来，关键问题在于如何让程序装配出合法的请求字符串，模拟成浏览器向服务器发送请求信息，然后保存服务器返回的页面。如果对方是一 ASP 或 PHP 页面，一般可以通过对该页面的源程序进行分析，并装配出合法的请求字符串。

还是以前面的例子做说明，如果组成请求串：

http://www.people.com.cn/ Search.wct?name="军事"

当服务器收到该请求后就将参数 name 的值“军事”传给组件，然后组件 Search.wct 将结果以 HTML 页面的形式返回给客户端。

1.2 当返回页面数目较大时的下载方案

一个 ASP 程序产生的结果会有许多的 HTML 页面，例如搜索《人民日报》某年的所有文章可能会有上万张的 HTML 页面，服务器不会将这上万张的 HTML 页面链接在一张页面上返回客户端，而是每二十或三十个链接组成一张页面返回给客户端，然后在每个页面中包含页码号的链接，如果用户想看的结果在服务器返回的头一张页面里找不到时，那么他就可以选择第几页或“下一页”来找。很显然，根据“下一页”这种链接方式来下载网页比根据页码号下载要灵活得多。我们只需要根据“下一页”的链接再找到“下一页”的链接，直到没有“下一页”为止，这样就可以将满足条件的页面全部下载下来了。

至于怎么样找到下一页的链接，要根据不同的站点具体分析，有的站点就是以字符串“下一页”来标志下一页的链接。例如，以下的 HTML 代码：

下一页

如果我们找到了字符串“下一页”就不难找到它的链接为

content.asp?page=2&classid=5&Nclassid=33&order=&updown=

还有些站点的标志可能不一样，比如是一张小图片或一个按钮。如果是一张图片，则可以以图片的名字作为标志。

1.3 隐含参数的问题

另外，有些站点可能并不能通过分析页面将请求链接的字符串装配出来。比如某个站点的搜索页面片段为：

<form method="post" action="search.php">……<input class="text" type="text" name="title" value="" size=40>……<input class="text" type="text" name="key" value="" size=40>……<input class="text" type="text" name="word" value="" size=40>……<input class="text" type="text" name="author" value="" size=40>……<input class="text" type="text" name="start_date" value=""

size=10>……<input class="text" type="text" name="end_date" value="" size=10>……<input class="button" type="submit" name="submit" value="搜索">……<input class="button" type="reset" name="reset" value="重填">……</form>

不难看出服务器端的响应页面是 search.php，该表单里有 title、key、word、author、start date、enddate 共六个参数，如果我们想下载 1999 年全年的文章，可能就要按如下装配请求字符串：

http://chat.ycwb.com/search/search.php?title=&key=&old_key=&enpboards=&author=&start_date=1999-1-1&end_date=1999-12-31&title=&category=所有分类

但事实上是行不通的，原因在于有些隐含的参数没有给全。那我们可以通过 Sniffer 软件达到这一目的。Sniffer 软件可以监控本地机器上 TCP 或 UDP 数据包的内容，通过分析数据包的内容就可以发现其他的参数是什么，以及该如何给参数赋值。

仍然以前面的站点为例，通过查看 TCP 数据包的内容我们可以发现，另外的几个参数为 flg,startnum 和 pagesize，分析发现这三个参数的作用分别是：flg 起控制作用；startnum 指的是请求的开始的页面的编号，如果服务器在数据库中找到了 2000 条纪录，且每 20 条记录的链接组合成为一个页面返回给客户端，当 startnum=201 时，返回的就是第 11 页，记录从第 200 条记录开始；pagesize 也是一个较为重要的参数，代表的是每个页面包含的记录链接条数。实际上，通过这两个参数可以找到一个更好的下载方法，即通过设定 startnum 的值来设定包含下一组记录链接的页面，而没有必要再到页面中去搜索“下一页”的链接。

1.4 对于没有提供检索功能站点的网页的下载

一般的站点，比如新浪，搜狐，由于一般不提供检索功能，就只能按链接下载了。有两种遍历方式可以采用：深度遍历和广度遍历，我们认为采用广度遍历的方式较好，因为对于语料库有用的页面一般集中在上面几层的页面中，这样就可以优先下载这些质量较好的页面。

利用遍历下载有两个重要的问题：

（1）避免重复下载。多个网页中包含一些相同的链接是很普遍的事，要避免重复下载可以设一个链表，在遍历链接的过程中遇到一个新的链接，就检查该链接是否已经在链表中了，如果已经在链表中，则忽略该链接，如果不在链表中，则下载该链接并且在链表中加入该链接。

如果程序在运行过程中中断，下次用户想接着上次下载，或是网站有所更新，想要下载新的页面，这也需要避免下载以前已经下载过的重复链接，解决这个问题的方法是将以前访问过的链接保存在文件中，下次运行程序的时候再将该文件取出比较，但是有一些链接（爬行的起点）是不应该保存的，比如说需要下载搜狐 http://www.sohu.com，那么类似以下的一些链接是不应该保存的：http://www.sohu.com（搜狐主页），http://news.sohu.com（搜狐新闻主页），http://sports.sohu.com（搜狐体育主页），http://business.sohu.com（搜狐财经主页），http://it.sohu.com（搜狐 IT 主页）等等，因为如果保存象这样的链接，那么下次将这些链接再读入之后，程序就不会访问这些页面，也就无法下载更新的页面了，更新的链接一般在这些链接的下层链接中。

（2）遍历必须是收敛的。因为因特网上的网页数目是非常惊人的，如果不做任何处理来下载是肯定不行的，所以必须将下载的链接限定在一定的范围之内。我们知道每一个站点都有一个域名，那么以域名作为该站点的标志是一个可行的办法，可以在遍历网页的链接的时候，对链接进行检查，看是否包含该域名字符串，若包含该域名字符串则可认为该链接是合法的，可以进入遍历过程，否则则认为该链接不应该在遍历的范围之内，应予以忽略。

2 Web 页分析程序的设计

Web 页分析程序的功能是提取网页中对语料库有用的数据，包括语料的元数据和语料本身。

2.1 通用语料解析器原理分析

HTML 文档包括文本和标记。一条基本的标记语句形式为：

<标记名称 属性列表（参数列表）> [</标记名称>]

可以简单地把标记分为两类：包容标记和空标记。包容标记由一个开始标记和一个结束标记构成，中间是数据对象。空标记只有起始标记而没有结束标记。在分析HTML4.0标准的基础上，可以设计一个HTML语法分析器，相当于一个 HTML 语言的解释器，可以将任一个 HTML 文档进行逐字的扫描，通过判断标记的意义来解释数据，生成该文档的树型结构的内部表示，然后提取其中的有用信息。下面举一个简单的例子来说明如何建立文件的逻辑结构。

```
<HTML>
    <HEAD>
        <TITLE>光荣属于中国共产党和中国人民</TITLE>
        <IMG src="prevrec.gif">
    </HEAD>
    <BODY>
        <H1>庆祝中国共产党成立 80 周年</H1>
        <H2>中国共产党走过了 80 年的光辉历程。</H2>
        <P>光荣永远属于中国共产党和中国人民！</P>
    </BODY>
</HTML>
```

这个例子的逻辑结构如图 1 所示。图 1 的树型结构可以转化成图 2 所示的二叉树。

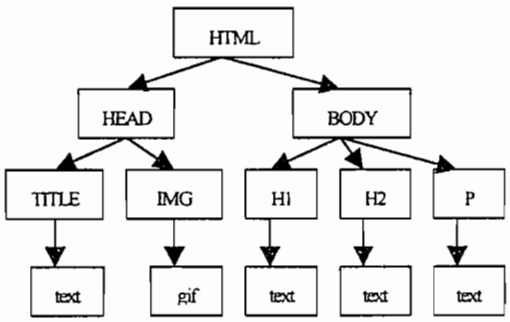

图 1

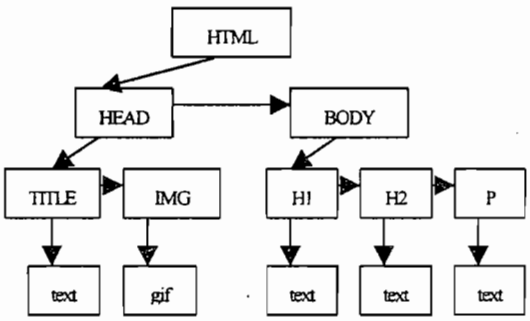

图 2

扫描过程的核心问题是如何建立结点之间的关系。一般来说，一个标记将形成树结构中的一个结点，但有些标记是一个元素的属性，如 B、I、U 等是文本的属性，它们就不单独形成结点，而是作为文本结点的属性。这里采用栈来解决这个问题。在详细介绍之前，先对数据结构定义作一些说明。树的存储结构为 node 类型的数组，其中 node 类型定义如下：

```
class node{
    public:
    CString* pstrData;   // 用来存储结点的私有数据的指针
    Cstring* pstrProperty; // 用来存储标记属性的数据指针
    int nType;   // 标记结点的类型
    UINT NodeID; // 标记结点的位置
    int nChildID; // 标记其孩子结点的位置
    int nBrotherID; // 标记其兄弟结点的位置
```

};

此外，还需要设置前一结点指针 pPriorNode、当前结点指针 pCurrentNode 和结点间关系 bIsBrother 标记变量（True 表示兄弟，False 表示父子，在当前的结点生成后设置）。

在对 HTML 文件进行扫描的过程中，pCurrentNode 为当前结点的 ID 值，pPriorNode 一般为前一结点的 ID 值，因为在一般情况下，结点间的关系在前一结点和当前结点之间产生。但是在某些情况下，pPriorNode 则可能不是前一个结点，而与当前结点之间间隔着若干个结点。这就需要用堆栈来保留前一相关结点的信息。

在对标记进行处理时，如果遇到空标记，只需要将 bIsBrother 设置为 Ture，因为空标记和下一标记之间肯定为兄弟关系。对于包容标记，情况就要复杂一些，栈在这里得到运用。如果遇到包容标记的起始标记，就把标记的相关信息（主要是 ID 值）存入堆栈，并将 bIsBrother 设置为 False，这是因为包容标记如果嵌套下一标记，则其与下一标记应为父子关系；如果遇到该标记的结束标记，就弹出堆栈，并将弹出的 ID 值赋给 pCurrentNode（在下一循环中，再将 pCurrentNode 赋给 pPriorNode），把 bIsBrother 设置为 True，这是因为该包容标记已经结束，其与下一标记之间应为兄弟关系。

2.2 语料自动解析器的实现

在设计语料自动解析器时，用到的 HTML 语料解析器与通用解析器相比，是一个简化了的算法，因为我们比较感兴趣的是 Web 页面中的文字信息，以便为后续的语料库建设打下基础，从网页中能够得到的关于语料的数据有：标题、作者、发表时间、分类、出处、正文。也就是说对网页中的其他数据我们完全忽略，扫描过程中如果不是我们期望的标记，则直接向下扫描。

Web 页分析程序可能遇到的困难有：①空白字符（制表符、空格、回车符以及换行符）以不定数量可能出现在元素之前或之后，也可能出现在元素的中间。②标记和文本中大小写字符可能混用。③标记之后可能有零个、一个或者多个参数；参数值可能用也可能不用引号（“”）作为分界符；参数标记的次序无法设想。

为解决这些问题，我们设计了两种识别有用语料数据的简单方法。

（1）模型匹配

模型匹配方法提供了直接搜寻网页中有用信息的一种有效手段。它包括一个单一的可复用的匹配函数，以参数形式接收一个匹配目标。在匹配函数中，可以放入所需要的任何特征，用星号“*”来替代我们要匹配的结果，程序把它解释为“跳过任意数量的字符”，于是，像<TD>Product</TD>或<TD>105.00</TD>的例子,都可以使用<TD>*</TD>的匹配参数来进行匹配。

（2）有效语料数据识别

下载到的 Web 页中，有些有完整的关于语料的元数据，有些则只有很少的元数据信息；同一个网站下载到的网页，元数据出现的先后顺序也有一定的规则，我们规定：

①元数据的识别按一定的顺序进行。在识别出一个特定的元数据后，排在其前面的元数据不能再被识别，这就保证了同一个元数据项的唯一性。②识别元数据的个数限制。除非在一张网页上有特定数量的元数据已被确定，否则识别算法拒绝这张网页，这就避免了存储不完整的语料记录。

语料自动解析器的设计应该对应每一类网页，抽取一些页面，通过人工干预的手段进行训练，提取其中的元数据的特征标记符号，如果我们只是单纯地让程序去匹配查找，象通用解析器那样去识别一张张页面，将很难保证解析得到的语料数据的质量。

3 小结

本文在一定程度上解决了部分站点的语料下载及分析问题，所设计的软件在语料库的建设实践中得到了很好的检验。但是也存在着许多的不足之处，有待于日后进一步完善。比如本文给出的下载方法需要对每个具体站点作出具体分析，如果需要下载的站点特别多的话，工作量也是很大的。一种改进措施

是，编程由计算机自动对每个站点进行分析，得到向服务器发送的请求字符串。另外，在 Web 页的分析方面，尤其是在程序的通用性、对象识别的精确性和特征值提取的程序自动化方面还有待进一步改进。

参考文献

[1]Dave Sussman, Alex Homer[美], 《ASP.NET 高级编程》, 清华大学出版社, 2002 年 1 月
[2]谢希仁 著, 《计算机网络》, 电子工业出版社, 1999 年 4 月
[3]《HTML 4.01 Specification》, http://www.w3.org, 1999 年 12 月

The Design of software for Downloading and Analysing web-pages Automatically

ZHU Kai ZHOU Jie HE Tingting

(Central china normal university, Wuhan 430079)

E-mail: hett@163.net

Abstract: With the development and popularization of computer application, especially with the popularization of Internet, the collection and the construction of large-scale corpus become more and more easy. This article discussed how to download web-pages from Internet automatically, how to analyse HTML page and acquire useful data for corpus.

Key words: corpus; web page; download; HTML