一种高效的新闻网页噪声过滤方法

邹永强，钟志农
（国防科技大学 电子科学与工程学院，湖南 长沙 410073）

摘 要：网页噪声过滤是网页预处理中关键的一步，其处理结果对后续处理的效率和准确性都有很大的影响。本文基于文本块字符数的统计规律，在总结了新闻网页特点的基础上设计了一种高效的新闻网页噪声过滤算法。该算法不仅完成了新闻正文的提取，也实现了新闻标题和报道时间的提取。试验证明，该算法有很高的处理速度，同时其提取的准确率也有了进一步的提高。

关键词：统计规律；网页噪声过滤；正文提取
中图分类号：TP391 **文献标识码**：A **文章编号**：1674-7720(2011)16-0064-04

An efficient approach to reduce noise in news webpages

Zou Yongqiang, Zhong Zhinong
(College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China)

Abstract: Noise reduction is an essential part of webpage pretreatment. It's processing result has a great effect on efficiency and accuracy of the later process. Depending on the news webpage features and the statistical regularity of the text blocks, an efficient approach to reduce noise in news was designed. This approach can not only extract the main text, but also the news title and the report time. The experimental results show that this approach obtains very high processing speed. And at the same time the accuracy is improved.

Key Words: statistical regularity; reduce noise in webpage; main text extraction

目前，互联网的网页除了表达主题的文本内容之外，还常常包括与主题无关的导航区、超链接、广告信息、版权信息等噪声信息。这些噪声对后续处理是十分不利的，一方面它增加了处理的工作量，耗费了不必要的资源；另一方面它使得处理的效果大打折扣，使结果出错的概率大大增加。因此网页噪声过滤是每个面向网络文本处理的应用技术都要考虑的，尤其是在网络文本挖掘和网络人物追踪等对精度和速度要求都比较高的应用中，其重要性更是不言而喻。

网页噪声过滤的目的是快速准确地识别并清除网页内的噪声，它是提高各种网页分析系统性能的一项关键技术。许多学者为了提高网页滤噪的准确度和效率进行了卓有成效的研究，纷纷提出各自的方法并且不断加以改进[1-6]。而参考文献[7]提出了一种快速且简单的正文提取方法，它不需要构造 DOM 树而是直接把 HTML 源文件看作是文本块的集合，仅通过分析每个文本块的字符数就可以提取出正文。与基于 DOM 的方法相比，这种方法在处理速度上有很大的优势。

本文在分析总结新闻网页特征的基础上利用基本文本块的字符数统计规律，提出了一种高效的过滤方法，它有很高的提取准确率和过滤速度，并且此方法在提取出新闻正文文本的同时还提取出了新闻网页的标题和报道时间。

1 网页噪声过滤算法

尽管网页结构、网页布局千差万别，但还是有一定的规律可循。在参考文献[7]中，网页源文件被分割成多个文本块，然后根据文本块字符数的统计规律，在通过一定的处理后得到 n_{short} 和 n_{long} 两个阈值，最后根据这两个阈值得到要提取的文本块集合。该方法处理速度较快，但是精度上却有所欠缺，而且会发生大段文本块遗漏的现象。问题主要出在阈值的选取上，本文希望通过对参考文献[7]的方法进行改进从而提高提取精度、减少文本块的遗漏，同时实现新闻标题和报道时间的提取。

1.1 新闻网页的特征

新闻网页一般包括新闻标题、新闻报道时间、作者、新闻正文等新闻有效信息，也常常包括导航区、超链接、

版权信息以及图片控件广告等噪声信息。通过大量观察发现新闻有效信息绝大多数处于网页源文件的中间位置，而且由相对较长且位置紧凑的多个段落组成。这些紧挨着的段落字数多少不同，中间还可能插有少量的链接。而噪声信息一般来说字数比较少，而且大多一般处在边缘位置。

再来看新闻网页 HTML 源文件的特征。HTML 源文件由各种标签和标签所修饰的内容组成。这些标签根据作用的不同可以分为网页布局元素（如 ）和网页描述元素（如<font>）。通过对标签的分析可以发现有些标签所修饰的内容全是噪声（如<script>），完全可以滤除它。之后根据网页布局元素将 HTML 源文件划分为多个文本块，这类似于多个段落。对每个文本块的字数进行统计，将其结果用直方图表示出来（如图 1 所示）。在这个网页中新闻有效信息部分从第 67 块到第 104 块，其他部分全是噪声块。从这个统计结果来看，HTML 源文件的特征和网页表现的特征是一致的。

1.2 基于文本块统计的新闻有效信息提取算法

通过以上对新闻网页特点的总结可见，其文本块是有一定的统计规律的。本文的算法就是利用这些统计规律来实现新闻有效信息的提取。

1.2.1 基本文本块的提取

从新闻网页 HTML 源文件中提取出基本文本块主要分以下四步：

（1）检查 HTML 标签，将不完整的补全。

（2）滤除表 1 中类型Ⅰ栏的标签。这些标签所修饰的内容全部为噪声，这些噪声在 HTML 源文件中一般会占很大的比例。

（3）将表 1 中类型Ⅱ栏的标签全部替换为<textblock>，同时把空白都删掉。

表 1 标签分类

类型Ⅰ	<a>,<script>,<noscript>,<style>,<meta>,<! -- -->,<param> <button>,<select>,<optgroup>,<option>,<label>,<textarea> <fieldset>,<legend> <input>,<image>,<map>,<area>,<form>,<iframe>,<embed>,<object>
类型Ⅱ	, ,<dl>,<p>,<Hr>, , , ,<table>,<tr>,<td>,<dt>,<dd> <head>, <body>,<tbody>

（4）根据自定义标签<textblock>从经过上述三步处理的网页源文件中按顺序抽取出每个文本块，得到文本块集合 blockset={b1,b2,b3…}。

HTML 的编写自由度比较大，再加上浏览器的容错能力很强，导致 HTML 源文件里的标签使用很不规范，标签缺失的现象很严重。这会使滤除的出错概率增大，所以第（1）步的处理是很必要的。经过第（2）、（3）两步后，按顺序提取出各个文本块就得到了原始的文本块集合 blockset。下面的处理就是针对这个集合的。

1.2.2 文本块统计与阈值确定

这一步的目的是确定哪些文本块是应该提取的，为此需要建立一个 blockset 的一一映射，这个映射就是数值数组 blocknum，然后对 blocknum 进行处理。这个过程包括以下几步：

（1）对文本块集合 blockset 中每个块的字符数进行统计，之后将统计结果存入数值数组 blocknum 对应的位置处。为便于分析将 blocknum 的结果绘制成直方图。

上面已经说过主体文本是字符数比较多且紧凑的文本块，从这幅统计图中确实可以看出一些符合这个特征的文本块，但是这些紧凑的文本块之间字符数差别比

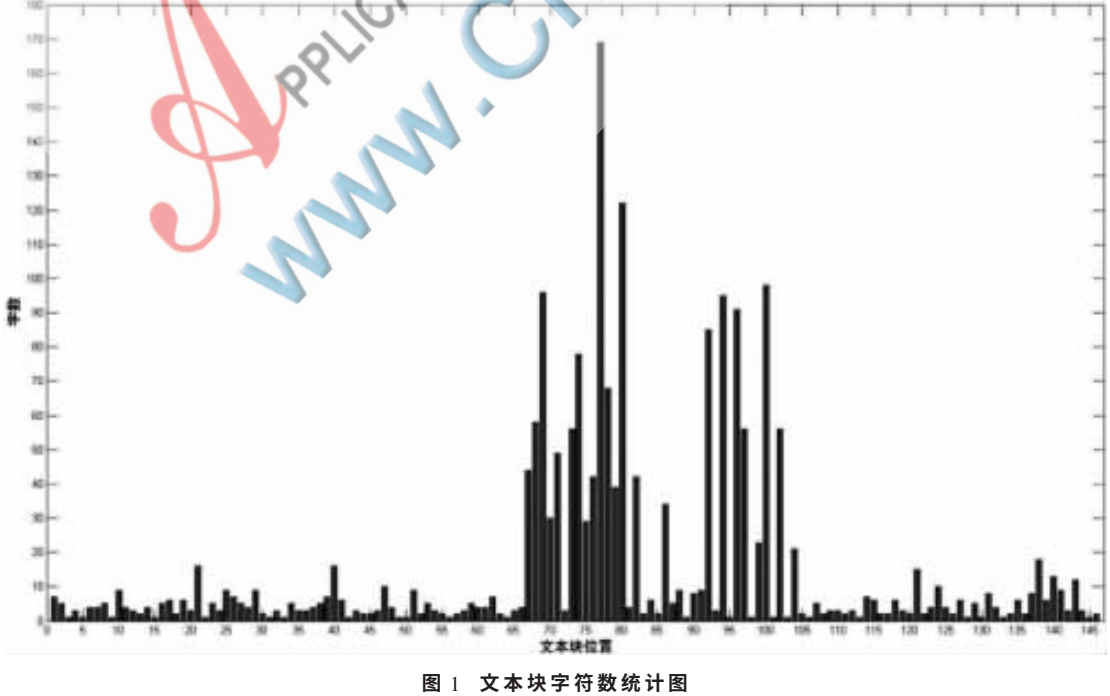

图 1 文本块字符数统计图

较大，其边界也不好判断，为此采用平滑技术对其进行处理。平滑技术的关键是平滑算子的选取，而这个算子选取的依据是以平滑为目的的。由于希望在平滑的同时使边界保持清晰，所以采用加权邻域平均法来处理统计结果，为此设计一个一维中心加权算子：$\frac{1}{4}[1\ \ 2\ \ 1]$。这个算子在平滑过程中考虑到处理点前后各一个临近点，同时给予处理点较大的权值（这里处理点的权值为2，而其临近点权值为1）。处理后得到数值数组sblocknum，绘制成直方图如图2所示。对比图1、图2，可以发现平滑处理之后新闻主体文本块更为突出，其边界也更为明显。这对最终提取出准确的主体文本是非常有利的。

（2）确定文本块最低阈值 $N'_{\min}$ 和标定值 $N_{标定}$，并据此提取出新闻的主体文本。经过多次试验，在总结了规律的基础上本文规定这两个阈值如下：

$$N'_{\min}=\frac{1}{3}(2N_{\min}+N_{avg})$$

$$N_{标定}=\frac{1}{10}\left(\sum_{i=1}^{9}N_{\max_i}+N_{avg}\right)$$

其中 $N_{\min}$ 是除0以外的最小值，N_{avg} 是所有块的平均值，$N_{\max_i}$ 是第 i 个最大值，$N_{\min}$、N_{avg}、$N_{标定}$ 和 $N_{\max_i}$ 都取自平滑后的结果sblocknum。$N'_{\min}$ 由 $N_{\min}$、N_{avg} 两个参数决定，实际上是这两个参数的加权平均。给 $N_{\min}$ 较大的权值，就使得 $N'_{\min}$ 的值落在 $N_{\min}$、N_{avg} 之间且更靠近 $N_{\min}$。经过实验验证，这样取值是合理的，并且达到了较好的效果。标定值 $N_{标定}$ 的取值最终关系到一个文本块集合的取舍，因此必须慎重考虑。因为各个文本块的字符数相差很大，即使平滑后也存在这种情况，因此不能仅根据sblocknum中的最大值来确定 $N_{标定}$ 的值，否则会丢掉很多原本是主体文本的文本块。同时 $N_{标定}$ 也不能太小，否则起不到筛选的作用。本文采用的是数组sblocknum前9个最大值和 N_{avg} 的平均值，这个 $N_{标定}$ 的选取很好地考虑到了这两点，其中前9个最大值也是根据大量实验最终确定的。

在确定了阈值 $N'_{\min}$ 和标定值 $N_{标定}$ 的基础上，本文提出的新闻主体文本块集合mainblockset的提取规则如下：

①设文本块子集 subblockset $=\{b_i \mid b_i \in \text{blockset}, i < \text{blockset.length}\}$，并且subblockset中的文本块是从blockset中按顺序提取的。只有当subblockset满足下列两个条件时：subblockset中所有 b_i 的字符数都不小于 $N'_{\min}$；subblockset中至少有一个 b_i 的字符数大于等于 $N_{标定}$，subblockset才是mainblockset中的一个子集，即subblocksetmainblockset。

②设subblockset1和subblockset2是相继提取出的文本块子集（如图2所示），则它们之间夹的不符合①中第二个条件的文本块集合（如图2中的BSet）也认为是mainblockset的子集。

在图2中，很明显文本块子集subblockset1和subblockset2是符合条件①的集合，所以subblockset1mainblockset，subblockset2mainblockset成立。subblockset1和subblockset2之间夹的文本块集合BSet是不符合①的第二个条件的，但是很显然BSet字符数并不少，仅仅是没有一个文本块的字符数达到 $N_{标定}$ 而已。它极有可能是新闻正文中的一个小段落，把它舍弃将破坏正文的完整性。所以把它留下并且认为它也是正文的一部分，即BSet $\subset$ mainblockset。实验证明，这样做是非常合理的。

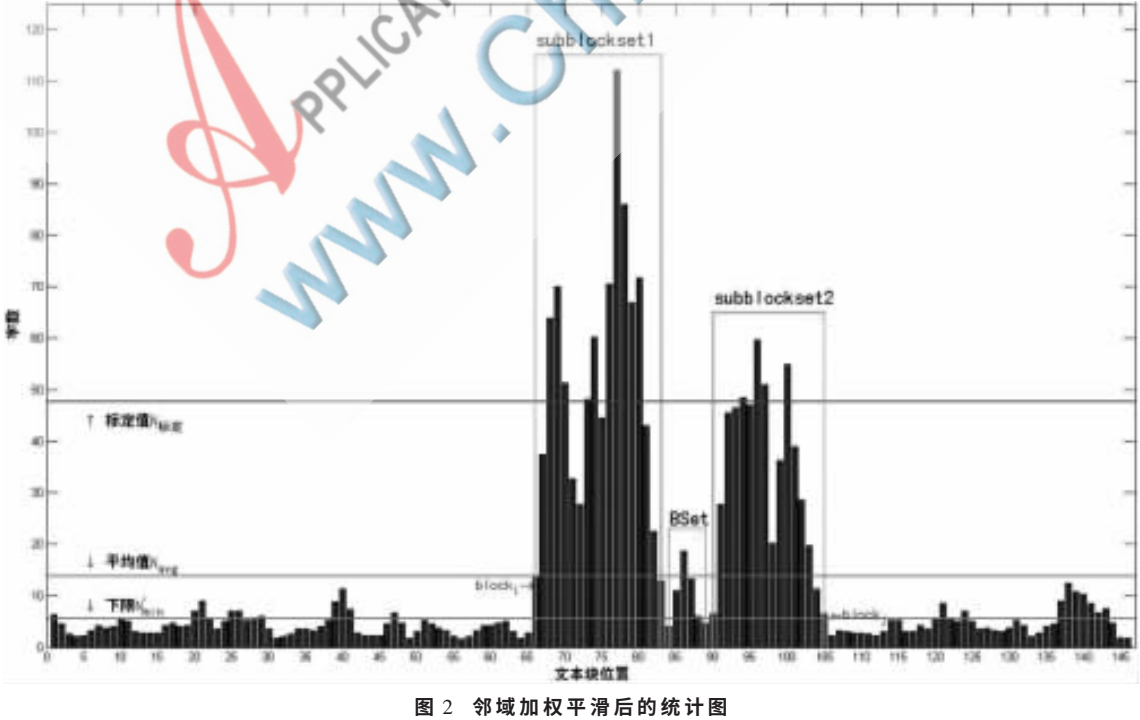

图2 邻域加权平滑后的统计图

1.2.3 新闻标题和报道时间的提取

新闻标题和报道时间是新闻不可或缺的新闻要素，因此本文将其列为提取内容之一。不论是新闻标题还是报道时间，其在新闻网页中出现的位置一般都是固定的，比如标题可能出现在<title>处，也可能出现在紧贴正文的前面，而报道时间大多是在标题和正文之间，也有一些是在正文的结尾处。

正是因为新闻标题和报道时间的位置特殊，本文没有一开始就提取它们，否则提取的标题和时间很可能是正文内部的一些小标题和非报道时间。上一小节抽取出了新闻的主体，实际上也就知道了新闻主体文本的开始块和结尾块的位置。设开始块为 $block_i$，结束块为$block_j$(如图 2 所示，其中 i、j 分别为开始和结束块在 blockset 中的位置)。从块 $block_{i-2}$~$block_{i+2}$ 中抽取出标签<h1>~<h6>所夹的部分，取等级最高的为标题。如果这几个块中没有这个标签，就抽取<titile>所夹的部分，这时抽取出的标题需要用停用词表做进一步的过滤。

一般来讲报道时间格式是比较固定的，常见的格式如：2010-12-15，2010 年 12 月 15 日，2010/12/15，二零一零年十二月十五日等。因此在找到报道时间可能存在的文本块集合后，利用正则表达式匹配的方式找到它。如在本文中把块 $block_{i-2}$~$block_{i+2}$ 和 $block_{j-2}$~$block_{j+2}$ 作为候选文本块集，用一个正则表达式“\w{4}[-,.,/,年]\w{1,2}[-,.,/,月]\w{1,2}日?”(不包括引号)来匹配，就可以找到报道时间。

2 实验研究

2.1 实验准备

检测算法性能的机器环境：CPU 是 Intel(R) Celeron(R) D CPU 3.2 GHz；内存为 DDR2 667 2.00 G；操作系统是 Windows XP Professional SP3。开发环境选用 C#2010 集成开发环境。为了使得到的结果具有普遍性，本文设计了一个网页采集器 NewsSpider，让它从知名的新闻网站(如网易新闻、新浪新闻、腾讯新闻、新加坡联合早报等)中下载了 2 298 个网页，以此作为实验语料集。实验主要测试新闻网页有效信息提取的准确率和处理速度。新闻标题和报道时间的检验比较简单，而对于新闻正文，只有提取出的正文完全覆盖真正的正文，而且未滤除的噪声占正文的比例不大于 5%的时候才算合格，如果这个比例小于 2%就为优秀。这里采用准确率来表示算法的准确性：

$$准确率=\frac{正确过滤的网页数}{总的网页个数}\times 100\%$$

2.2 实验结果

将本文方法与参考文献[7]所提出的算法进行了比较，其结果如表 2 所示(方法一是参考文献[7]提出的方法，方法二是本文的方法)：

表 2 新闻网页过滤实验结果

评价指标	优秀页面	优秀率/%	合格页面	合格率/%	不合格页面	不合格率/%
方法一	934	40.64	2 223	96.74	75	3.26
方法二	1 065	46.34	2 245	97.69	53	2.31

可见，不论是本文方法还是参考文献[7]的方法都取得了不错的效果。本文方法在优秀率和准确率上更胜一筹，主要有两个原因：一是本文在 $N_{标定}$ 的选取上采用数组 sblocknum 前 9 个最大值和 N_{avg} 的平均值，这使 $N_{标定}$ 的取值考虑到更多的其他文本块子集的最大值，防止一些字数不是很多的文本块子集的遗漏；另外本文考虑到了不符合规则①的第二个条件，但是又极有可能是正文的文本块集合(如图 2 中的 BSet)，这样就使正文文本块之间字数比较少的文本块子集也被提取出来。

方法一处理网页的平均速度达到了 51 个/s，本文方法的平均处理速度是 47 个/s。这主要是因为本文方法在计算量上比方法一要稍大一些，但是由于本文提取准确率要高一些，以较小的速度损失换来较高的准确率还是值得的。另外本文方法的速度比基于 DOM 的方法还是快很多的，在准确率和速度上都达到了比较好的效果。

结果中不合格的网页大多是总的正文文本很少的网页，这些网页的正文往往由几句话构成，与其周围的噪声相比特征不十分明显，这极大地影响了提取的准确度。这样的网页占不合格网页的比例达到 70%左右。另外 30%的不合格网页是一些主体文本边界不明显的新闻网页，由于正文开头文本块和结尾文本块的提取比较困难，有时遗漏这些文本块，有时又增加一些噪声块。对于以上两种处理效果不好的网页，还需要研究更为有效的技术来处理。

本文在借鉴现有研究的基础上针对新闻网页提出了一种集网页过滤和基本信息提取于一体的过滤方法。该方法利用了新闻网页基本文本块的统计规律，提高了新闻正文的提取精度，获得了比较高的处理速度，同时提取出了对后续处理很有价值的新闻标题和报道时间。但是在处理一些字符数比较少的网页时，效果较差，这说明算法还有待改进。同时 Internet 上的网页种类繁多，如论坛网页、博客网页等，它们都蕴涵了大量有价值的信息。本文的算法还不适于处理这种多主题的网页，所以研究如何快速准确地过滤这些网页并提取出有价值的信息将是下一步研究的重点。

参考文献

[1] Lin Shianhua, Ho Janming. Discovering informative content blocks from Web documents[C]. KDD 2002: 588-593.

[2] 欧健文,董守斌,蔡斌. 模板化网页主题信息的提取方法[J]. 清华大学学报:自然科学版,2004,32(S1):84-87.

[3] Yu Shipeng, Cai Deng, Wen Jirong, et al. Improving pseudo—relevance feedback in web information retrieval using web page segmentation[C]. Proceedings of WWW2003, Budapest, Hungary, 2003: 11-18.

[4] GUPTA S, KAISER G E, NEISTADT D, et al. DOMbased content extractlon of HTML documents[A]. Proceeding of the 12th International World Wide Web Conference[C]. Budapest: ACM Press, 2003:207-214.

[5] 王琦, 唐世渭, 杨冬青,等. 基于 DOM 的网页主题信息自动抽取[J].计算机研究与发展, 2004, 41(10): 1786-1791.

[6] 徐超.基于 DOM 的网页净化方法研究[D]. 青岛:中国石油大学(华东), 2009.

[7] Zhou Baoyao, Xiong Yuhong, Liu Wei. Efficient web page main text extraction towards online news analysis[C]. 2009 IEEE International Conference on e-Business Engineering, 2009:37-41.

(收稿日期:2011-05-31)

作者简介:

邹永强,男,1986年生,硕士研究生,主要研究方向:文本数据挖掘技术。

钟志农,男,1975 年生,副教授,硕士生导师,主要研究方向:信息处理和数据挖掘技术。