改进的中文静态网页新闻正文自动抽取算法

何春辉 王孟然
(湘潭大学 工程训练中心，湖南湘潭 411105)

摘要： 网页新闻正文自动抽取属于信息抽取领域中的重要研究问题，现有基于行块分布进行新闻正文自动抽取的方法对短文本段落的新闻正文抽取效果较差。为了改善这种现状，提出了一种改进的中文静态网页新闻正文自动抽取算法。该方法给出了较好的行块分割策略来构建行块分布函数，并提出使用最长公共子序列作为新闻正文内容起始行块和结束行块的快速定位方法的判别准则。最后在1 000个新闻网页上对算法的性能进行了实验验证，得出新算法的平均抽取准确率为95.0%，平均召回率为96.54%，正文平均遗失率为1.6%，抽取单个网页的平均耗时为0.13 s。实验结果充分说明了新算法能适应大规模的网页新闻正文自动抽取任务。

关键词： 行块分布；自动抽取；快速定位；最长公共子序列
中图分类号： TP391.1 **文献标志码：** A **文章编号：** 1009－0312(2018)05－0046－05
DOI:10.16002/j.cnki.10090312.2018.05.009

对信息检索和文本挖掘及敏感信息监测等领域而言，网页新闻正文的自动抽取是原始数据获取的一个关键环节。利用信息抽取技术得到的新闻正文内容质量高低会直接影响上层应用。因此，如何能准确高效的自动抽取网页新闻正文成为了学术界和工业界关注的热点问题。

目前最流行的新闻正文抽取方式大致归结为以下几类：1) 基于模式匹配[1]来抽取网页新闻正文。该方式的优势是抽取准确性较高，但劣势也很明显，它在复杂网页的抽取上代价太大，需要对特定类型的网页制定大量抽取规则，这些规则需要大量人力来跟踪维护且无法适应大规模网页通用性抽取的要求。2) 基于DOM树[2－4]来抽取网页新闻内容，根据网页的结构信息构建DOM树，然后利用树的节点信息来抽取新闻正文。该方式的优势是准确性可靠，但缺点是DOM树的构建对网页结构的完整性依赖很高且树的构建和遍历过程需要的空间复杂度较高。3) 基于文本密度统计[5－7]和行块[8－10]分布来抽取网页新闻正文。该方法的优势是无需制定规则，特别适合进行大规模网页的抽取，但缺点是算法需要启发式的计算行块的骤升和骤降点。4) 基于机器学习方法[11]来抽取新闻内容。该方法的优势是可以通过提取重要特征来准确识别新闻正文，但缺点是需要事先标注大量的网页新闻样本来训练模型。

通过上述方法对比分析，发现它们都各有优缺点，但皆无法很好的适应大规模网页新闻正文的自动抽取任务。为了较好的改善这种现状，提出了一种改进的中文静态网页新闻正文自动抽取算法，它借助了行块分布的优势来进行网页新闻正文的抽取，无需人工制定相关抽取规则，同时利用改进的行块分割策略和起始行块与结束行块定位方法对新闻正文进行抽取。这种方法既能高效准确的抽取新闻正文内容，又具有很好的通用性，适用于大规模中文网页新闻正文的自动抽取任务。

1 改进的中文静态网页新闻正文自动抽取算法流程

国内最初是由哈尔滨工业大学的陈鑫提出基于行块分布函数进行网页正文抽取，他给出了两个核心指标：1) 正文区域的密度；2) 行块的长度。通过结合这两个指标可以较好的实现新闻正文抽取任务。这种方法的难点是给出行块分布函数后，首先需要遍历所有行块，求出长度最大的行块；然后以长度最大的行块为中心，使用启发式的方法对剩下的行块进行遍历，根据设定的阈值来得到文本内容长度出现骤升和骤降的行块位置；接下来将得到的骤升行块和骤降行块之间的内容按原文出现顺序进行合并；最

收稿日期： 2018－04－25
作者简介： 何春辉(1991—)，男，湖南永州人，工程师，硕士，主要从事文本信息挖掘研究。Email：xtuhch@163.com。

后选出合并后长度最长的行块内容作为最终的正文内容。

这种方法优点是无需制定规则和构建 DOM 树，同时不需要事先标注大量的训练样本，适合进行大规模网页正文的抽取任务。但它存在以下不足：1）它在抽取包含短文本段落的网页新闻效果较差；2）它在定位骤升和骤降行块位置时，只使用行块内容的长度作为判别依据，这会使得一些内容较长的噪声也被当做正文内容给抽取出来。上述问题对基于行块分布函数进行正文抽取的通用性和准确性造成了很大的影响。为了改善这种现状，提出了一种改进的中文静态网页新闻正文自动抽取算法，该算法的整体流程如图 1 所示。

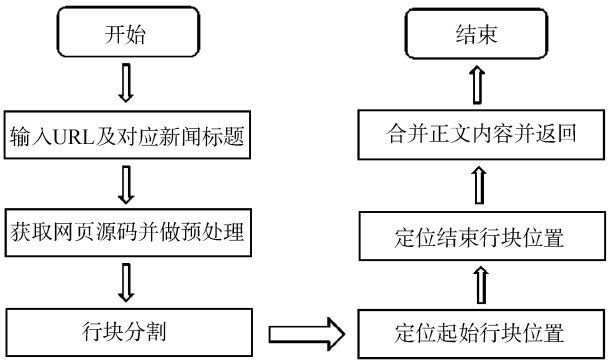

图 1　改进的静态网页新闻正文自动抽取算法整体流程图

图 1 中输入的新闻标题是使用开源工具从对应 URL 的网页源码中抽取得到，抽取准确率很高，针对特殊情况，如果标题为空，认为 URL 属于非正常网页，正文内容自动设置为空。

2　网页预处理和行块分割策略

网页预处理质量好坏对后续的正文抽取有直接的影响，预处理步骤是指对发送 URL 请求之后得到的网页源码进行相关的处理。这个步骤的主要任务是去除网页里面的 HTML 标签、CSS 样式、SCRIPT 标签、空格、空行、注释信息、特殊字符等与正文内容无关的噪声信息。通过以上处理步骤后，一张网页就转变为一个以行作为基本单元的简单文本文件，网页中剩下的所有文本内容就会按原始出现的先后顺序保存到该文本文件内。

经过预处理步骤得到剩下的内容后，就能以行作为基本单元从头到尾逐行遍历取出每一行对应的文本内容来构建行块分布函数。构建行块分布所采用的分割策略如表 1 所示。

表 1　构建行块分布函数的分割方法及步骤

输入：经过预处理后以行作为基本单位的文本文件
输出：行块分布行数
Step1：初始化行块分布函数 Block（key = B_ num，value = content），B_ num = 0，content = null
Step2：从文本文件的第一行开始遍历，取出本行的文本内容计算长度并赋值给 L，转 Step3
Step3：If L < 8：//内容长度小于 8 的段落可认为是非正常段落，阈值可根据情况调整
转 Step4
Else：
统计文本里面指定的标点符号数量并赋值给 C，判断 C 的取值情况
If　C > = 1：//阈值可根据实际情况进行调整
content = 本行内容，Block. Add（B_ num，content），B_ num = B_ num + 1，转 Step4　　Else：
转 Step4
Step4：自动遍历下一行，以此类推，直到遍历完最后一行转 Step5
Step5：返回行块分布函数 Block，转 Step6
Step6：行块分割结束

引入特定标点符号作为过滤特征是考虑到正常的新闻正文内容会包含标点符号，而导航栏、网站信息、广告等一般不含标点符号，该条件对中文新闻具有普适性，并不影响通用性。

3　起始行块和结束行块的快速定位方法

如第 2 节所述，经过行块分割后可得到行块分布函数。这个行块分布函数里面会包含网页中绝大部分有用的文本段落信息，因为行块分割时是按照原文出现的先后顺序进行分割的，所以行块分布函数里

面所有的内容都是按顺序连续排列的。如何能够从该分布函数里面快速准确的定位正文所在区域成了一个难点问题。如果条件太严格就会出现抽取部分正文当做最终的正文，导致部分正文信息“遗失”。如果条件太宽松了又会抽取一些与正文不相关的噪声信息，从而增加后期的处理难度。因此为了能够准确的快速定位正文起始行块和结束行块的位置，文中引入了基于两个字符串之间的最长公共子序列的长度作为行块位置的定位判别准则，具体原理见 3.2 和 3.3 小节。

3.1 字符串最长公共子序列求解方法

两个字符串之间的最长公共子序列可以说明两个字符串之间是否有公共交集，这样可从字符层面说明字符串之间是否具有相关性，它可以有效地过滤噪声找出相关联的信息。因此引入字符串的最长公共子序列作为起始行块和结束行块的定位判别准则。对于两个字符串 $I=(s_1,s_2,\cdots,s_i)$ 和 $J=(t_1,t_2,\cdots,t_j)$ 之间的最长公共子序列长度[12]（LCS）计算公式见式（1）：

$$\mathrm{LCS}[i,j]=\begin{cases}0,\text{if}\ (i=0)\ \text{or}\ (j=0)\\ \mathrm{LCS}[i-1,j-1]+1,\text{if}\ (i,j>0,s_i=t_j)\\ \max\{\mathrm{LCS}[i,j-1],\mathrm{LCS}[i-1,j]\},\text{if}\ (i,j>0,s_i\neq t_j)\end{cases}, \tag{1}$$

式中 LCS[i,j] 表示两个字符串之间的最长公共子序列的长度，是一个非负整数。

3.2 正文起始行块定位方法

经过行块分割后可得到行块分布函数，考虑到行块分布函数里面所有的内容都是按顺序连续排列的。因此需要准确定位正文的起始行块位置只需从前往后遍历，定位出起始边界即可。利用字符串最长公共子序列长度来辅助定位起始行块位置，具体计算步骤如表 2 所示。

表 2 正文起始行块定位方法及步骤

输入： 新闻标题 title 和行块分布函数 Block
输出： 起始行块位置编号

```
Step1: 初始化当前行块编号 i，当前行块内容 content，起始行块位置编号 Start = 0
Step2: For ( i = 0; i < = Block. Size( ) /2; i + + ) { //经大量测试只需遍历一半行块即可找出起始位置
            content = Block( i) . value; //将第 i 块的内容赋值给 content
            L = LCS( title,content) ; //计算标题和 content 之间的最长公共子序列长度并赋值给 L
            If ( L > = 2) { //公共序列长度不小于 2,阈值可根据实际情况进行调整
                Start = i; //得到起始行块位置的值
                Break;
            } Else{ continue; }
        }
Step3: return Start;
Step4: 起始行块位置定位结束
```

通过上面这个方法可以快速准确的定位到正文在分布函数中的起始行块位置。

3.3 正文结束行块定位方法

表 3 正文结束行块定位方法及步骤

输入： 新闻标题 title 和行块分布函数 Block
输出： 结束行块位置编号

```
Step1: 初始化当前行块编号 i，当前行块内容 content，结束行块位置编号 End = Block. Size ( ) - 1
Step2: For ( i = Block. Size( ) - 1; i > = Block. Size( ) /2; i - - ) { //同理,遍历一半行块即可找出结束位置
            content = Block( i) . value; //将第 i 块的内容赋值给 content
            L = LCS( title,content) ; //计算标题和 content 之间的最长公共子序列长度并赋值给 L
            If ( L > = 2) { //公共序列长度不小于 2,阈值可根据实际情况进行调整
                End = i; //得到起始行块位置的值
                Break;
            } Else{ continue; }
        }
Step3: return End;
Step4: 结束行块位置定位结束
```

经过行块分割后可得到行块分布函数，考虑到行块分布函数里面所有的内容都是按顺序连续排列的。因此需要准确定位正文的结束行块位置只需从后往前遍历，定位出结束边界即可。利用字符串最长公共子序列长度来辅助定位结束行块位置，具体计算步骤如表 3 所示。

通过上面方法可快速准确的定位新闻正文在分布函数中的结束行块位置。3.2 和 3.3 节所述定位算法的复杂度呈现线性增长，因此效率比较高；在准确率方面，因为使用了 LCS 的长度作为评估准则，所以准确性也能得到极大的保证。最后根据定位的行块编号，将位于起始行块和结束行块之间的所有行块内容按顺序进行合并从而得到最终的正文内容。

4　实验分析

从新浪、搜狐、腾讯、网易等大型门户新闻网站以及多家不同类型的新闻网站中选取了 1 000 个不同的新闻网页构成新方法的性能评测样本数据集。为了客观的反映方法的性能，使用平均准确率和平均召回率和平均遗失率以及单个网页正文抽取的平均耗时四个指标来进行评估。四个指标的定义分别如式(2)所示。

$$\overline{P}=\frac{\sum_{i=1}^{k}P_i}{N},\overline{R}=\frac{\sum_{i=1}^{k}R_i}{M},\overline{L}=\frac{\sum_{i=1}^{k}L_i}{N},\overline{T}=\frac{\sum_{i=1}^{k}T_i}{N}, \tag{2}$$

其中 $\overline{P}$ 是指平均准确率，$\overline{R}$ 是指平均召回率，$\overline{L}$ 是指平均遗失率，$\overline{T}$ 是指单个网页的平均耗时，k 是新闻网站的类别数量，P_i 是指第 i 类新闻网站中正文抽取正确的网页数量，R_i 是指第 i 类新闻网站中抽取到全部正文内容的网页数量，T_i 是指第 i 类新闻网站中正文抽取的耗时，L_i 是指第 i 类新闻网站中未能抽取到正文的网页数量，N 是指样本数据集的大小，M 是指抽取到全部正文的网页总数量。

使用文中提出的中文静态网页新闻正文自动抽取算法对上述网页进行正文抽取的实验结果如表 4 所示。

表 4　中文静态网页新闻正文自动抽取算法实验结果

指标名称及网站名称	新浪新闻	搜狐新闻	腾讯新闻	网易新闻	其它网站	汇总
网页数量	100	100	100	100	600	1 000
取得全部正文网页的数量	100	97	100	95	592	984
正文抽取正确/错误的数量	96/4	94/3	95/5	91/4	574/18	950/34
抽取耗时/s	11.4	13.6	12.5	14.7	81.1	133.3

由表 4 实验结果求出 $\overline{P}=950/1\ 000=95.0\%$，$\overline{R}=950/984=96.54\%$，$\overline{L}=(1\ 000-984)/1\ 000=1.6\%$，$\overline{T}=133.3/1\ 000=0.13$ s。

使用文献[4]提出的 CEPR 方法对上述网页进行正文抽取的实验结果如表 5 所示。

表 5　CEPR 方法正文自动抽取的实验结果

指标名称及网站名称	新浪新闻	搜狐新闻	腾讯新闻	网易新闻	其它网站	汇总
网页数量	100	100	100	100	600	1 000
取得全部正文网页的数量	99	98	98	94	586	975
正文抽取正确/错误的数量	94/5	94/4	93/5	92/2	570/16	943/32
抽取耗时/s	17.3	18.5	18.1	20.2	117.6	191.7

由表 5 实验结果求出 $\overline{P}=943/1\ 000=94.3\%$，$\overline{R}=943/975=96.72\%$，$\overline{L}=(1\ 000-975)/1\ 000=2.5\%$，$\overline{T}=191.7/1\ 000=0.19$ s。实验环境为 64 位 Win7 系统，Intel Core i7 处理器，16G 运行内存。

根据上面两个实验的对比结果可知，本文提出的方法平均准确性为 95.0%、平均遗失率为 1.6%，抽取单个网页平均耗时为 0.13 s，这些结果均比现有的 CEPR 方法要好，只是在平均召回率指标上稍逊于 CEPR 方法但也到达了 96.54%。总体上说明新方法的性能有一定的提升，尤其是在耗时方面表现出色，这对于大规模网页正文抽取具有非常大的优势。

5 结语

根据新闻正文内容在网页中比较集中的特点，提出了一种改进的中文静态网页新闻正文自动抽取算法。首先使用模式匹配技术抽取出新闻的标题并结合常用的预处理技术剔除网页源码中与新闻正文无关的内容，然后结合文中提出的行块分割策略构建行块分布函数，再使用文中提出的正文起始行块和结束行块快速定位方法来定位正文在行块分布函数中所处的区域，最后通过合并分布函数中起始行块和结束行块之间的所有行块内容得到新闻的正文内容。该方法由于不依赖于规则和 DOM 树，它可以快速准确的从不同网站的网页新闻中自动抽取正文内容，通用性很高，且对较少段落的短文本新闻也具有很好的适应性。算法不足之处如下：1）算法依赖于新闻标题，如果新闻标题缺失，算法将无法提取新闻正文；2）算法目前只能提取网页新闻的正文和标题，无法提取新闻中的其他信息。未来会考虑增强算法的抽取功能，让它能抽取新闻的发布时间、新闻来源、作者和编辑等更多有用的信息。

参考文献

[1] SIRSAT S，CHAVAN V. Pattern matching for extraction of core contents from news web pages [C] // Second International Conference on Web Research. IEEE，2016：13 - 18.

[2] 潘心宇，陈长福，刘蓉，等. 基于网页 DOM 树节点路径相似度的正文抽取 [J]. 微型机与应用，2016，35(19)：74 - 77.

[3] 马晓慧，李泓莹. 一种 DOM 树标签路径和行块密度结合的 Web 信息抽取方法 [J]. 智能计算机与应用，2017，7(4)：13 - 16.

[4] WU G，LI L，HU X，et al. Web news extraction via path ratios [M]. ACM，2013.

[5] 朱泽德，李淼，张健，等. 基于文本密度模型的 Web 正文抽取 [J]. 模式识别与人工智能，2013，26(7)：667 - 672.

[6] 钱爱兵. 一种基于统计的中文网页正文抽取方法 [J]. 情报学报，2009，28(2)：187 - 194.

[7] 林子熠，沈备军. 基于统计的自动化 Web 新闻正文抽取 [J]. 计算机应用与软件，2010，27(12)：232 - 235.

[8] 姬鑫，钟诚. 基于分块的新闻网页信息抽取算法 [J]. 计算机应用与软件，2015(4)：317 - 322.

[9] 邱江涛，唐常杰，李川，等. 基于块分布的新闻网页内容提取 [J]. 吉林大学学报(工学版)，2009，39(5)：1326 - 1330.

[10] 李烯，徐朝军. 基于分块和统计相结合的新闻正文抽取 [J]. 情报理论与实践，2010，33(1)：117 - 120.

[11] 罗永莲，赵昌垣，贾玉芳，等. 基于朴素贝叶斯 Web 新闻内容的抽取方法 [J]. 计算机与现代化，2016(1)：59 - 63.

[12] APOSTOLICO A，GUERRA C. The longest common subsequence problem revisited [J]. Algorithmica，1987，2(1)：315 - 336.

Improved Automatic Extraction Algorithm for Chinese Static Web Page News Body

HE Chunhui　WANG Mengran

(Engineering Training Center，Xiangtan University，Xiangtan 411105，China)

Abstract　The automatic extraction of web page news content is an important research issue in the field of information extraction. The current method of automatic extraction of news body based on the blocks distribution is less effective in extracting short text paragraph. In order to improve this situation，an improved automatic text extraction algorithm for Chinese static web pages is proposed. This method gives a better block segmentation strategy to build a block distribution function，and puts forward using the longest common subsequence as a rapid positioning method norm for the start and end blocks of news content. Finally，the performance of the algorithm was tested on 1 000 news web pages. The average extraction accuracy rate of the new algorithm was 95.0%，the average recall rate was 96.54%，the content average loss rate was 1.6%，and the average time consumed to extract single web page was 0.13 seconds. The experimental results fully illustrate that the new algorithm can adapt to the large-scale automatic extraction of web news content.

Key words　block distribution；automatic extraction；rapid positioning；longest common subsequence