基于结构相似网页聚类的正文提取算法研究

王海涌,冯兆旭,杨海波,张津栋
WANG Haiyong, FENG Zhaoxu, YANG Haibo, ZHANG Jindong

兰州交通大学 电子与信息工程学院,兰州 730070
School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China

WANG Haiyong, FENG Zhaoxu, YANG Haibo, et al. Research on text extraction algorithm based on structure similarity page clustering. Computer Engineering and Applications, 2018, 54(11):122-127.

Abstract: The current Web pages are getting more and more diverse, complex which makes the information extraction more difficult. In this paper, a text extraction algorithm based on structure similarity page clustering is proposed. Firstly, the contribution of each "block" to the template is assigned to different weights according to the composition of the front page of the Web page. Secondly, the similarity of the corresponding blocks in the two Web pages is calculated. The similarity and the weight of each block product as the sum of the two pages' similarity. This algorithm takes into account the influence of Web page structure difference on Web page text extraction. Web page is clustered based on computing the similarity between Web pages. The results are more accurate for the Web page text in the same cluster. The experimental results show that the method has higher accuracy and the evaluation indexes are improved.

Key words: information extraction; similarity; Document Object Model(DOM) tree; hierarchical clustering

摘 要:针对当前互联网网页越来越多样化、复杂化的特点,提出一种基于结构相似网页聚类的网页正文提取算法,首先,根据组成网页前端模板各"块"对模板的贡献赋以不同的权重,其次计算两个网页中对应块的相似度,将各块的相似度与权重乘积的总和作为两个网页的相似度。该算法充分考虑结构差别较大的网页对网页正文提取的影响,通过计算网页间相似度将网页聚类,使得同一簇中的网页正文提取结果更加准确。实验结果表明,该方法具有更高的准确率,各项评价指标均有所提高。

关键词:正文提取;相似性;文档对象模型(DOM)树;层次聚类
文献标志码:A **中图分类号:**TP391 doi:10.3778/j.issn.1002-8331.1701-0161

1 引言

随着信息技术的进步,各种网页制作工具和新的WEB标准,使得产生各种各样网页内容的速度越来越快,与此同时网页上也出现了越来越多的额外信息,包括广告、站内推广信息、其他相关内容的链接,这些广告链接等数据的加入使得开发人员更容易也更多样化的方式来表达信息,不同类型的工具和元素使得网站的内部结构和页面外观更加复杂多样。目前网络数据主要是以HTML页面的形式,由于HTML语言只是告诉浏览器如何显示它定义的信息并执行,经过浏览器分析后的网页适合于人们浏览,但不适合作为由计算机来处理的数据。网页正文提取旨在从半结构化的网页文档中抽取出有价值的信息,它是数据挖掘、话题检测、文本分类、网页聚类等领域的关键环节,面对如此巨大互联网信息库,如何快速有效地提取网页正文信息已成为当前信息处理的迫切需求。

20世纪90年代初,人们开始注重研究信息提取。近年来随着网页的应用越来越广泛,人们逐渐将研究重点转移到网页信息提取上,并取得了不少成果。Arasu等人采用了词频统计与DOM路径相结合的方法。然而对于网页内容很多的网页这种方法的效果不好[1]。Kim等人改进了网页模板的生成方式,利用网页不同区域内

基金项目:甘肃省自然科学基金(No.145RJZA086);兰州交通大学科技支撑基金(No.ZC2014003);兰州市科技计划项目(No.2013-3-79)。

作者简介:王海涌(1974—),男,博士,教授,研究方向为智能信息处理;冯兆旭(1991—),男,硕士研究生,研究方向为智能信息处理;杨海波(1992—),男,硕士研究生,研究方向为数据挖掘;张津栋(1991—),男,硕士研究生,研究方向为智能信息处理。

收稿日期:2017-01-12 **修回日期:**2017-03-01 **文章编号:**1002-8331(2018)11-0122-06

容所占面积大小来设定权重[2]。

当前，常见的网页正文提取算法可分为四大类：基于启发式规则的正文提取算法，基于网页模板的正文提取算法，基于视觉分块的正文提取算法以及基于统计、机器学习的正文提取算法。网络上也有一些网页正文提取产品和项目，如cx-extractor、Readability、Diffbot等，它们在正文提取的效果和性能上各有优劣。其中arc90实验室的Readability算法的基本思想是：先将网页解析DOM树，所有标签小写。然后去除所有“script”标签内容，再通过一对正则表达式的配合提取[3]。2010年1月西南科技大学的熊子奇、张晖、林茂松等人提出利用网页标签和文本内容相似度相结合来提取正文信息[4]。2015年9月中科院计算机网络信息中心杨柳青等人提出并实现了一种基于布局相似性的网页正文提取算法，该算法通过对比同一站点同一专题的网页DOM树中节点数据信息的相似性来实现正文提取[3]，但是该算法未能考虑互联网网页来源繁杂，正文提取结果受网页结构相似度的影响较大。而且论文中提出的算法在树路径和查询参数的相似性计算上采用“相同目录层数/最大目录层数”这样一种简单的度量方法，不能完全反映网页的结构特征，要实现更精准的结果还需作进一步的研究。

2　网页结构特征分析

2.1　基本原理

HTML是一种用于描述网页文档的文本标记语言，可以用DOM树表示，DOM是文档对象模型（Document Object Model）的缩写[5]。DOM是一个浏览器、平台独立于语言的接口，允许用户访问其页面的其他标准组件，HTML文档被解析成DOM树，每个HTML中的元素、属性、文本代表一个树节点[6]。

定义1　DOM树中节点拥有子树数称为节点的度。DOM树中度为0的节点为叶子节点。

定义2　节点的子树称为该节点的子节点；该节点称为子节点的父节点。

定义3　树路径是指从根节点到一个叶节点所经历的所有节点组成的序列，表示如下：

$$P=(m,t_1,t_2,\cdots,t_n,s_1,s_2,\cdots,s_m)$$

其中，m表示该路径在DOM树中出现的次数；$(t_1,t_2,\cdots,t_n)$表示该路径所经历的节点标签组成的序列；$(s_1,s_2,\cdots,s_m)$表示该路径出现的位置[7]。DOM树所有路径的叶节点按遍历排序，用顺序号表示树路径的位置。一个网页的DOM树可用一个树路径集合表示。

2.2　网页结构相似性

网页相似度研究主要用于网页信息抽取。网页相似度的研究可分为基于网页内容的文本相似度研究和基于网页机构的结构相似度研究；前者是对网页中的文本进行中文分词后提取特征词，构造特征向量并计算其相似性[8]。根据文献[6]所述，网页可分为：主题型网页、目录型网页和多媒体网页。

本文主要任务是从大量的网络新闻网页中提取正文内容，这些网页来自许多不同的新闻网站，主要的研究对象是中文主题型网页。网页的结构特征是以块的形式呈现的，而且通常在同一站点下的统一频道中所有主体性网页间的布局结构极为相似，所有的目录型网页的布局结构也极为相似，并且相同或相似的内容往往出现在固定的位置上[9]。澎湃新闻的网页布局实例如图1所示。

图1　新闻网页布局实例

在图中，可以看到同一网站，同一专题下的新闻网页具有相同的布局结构，网站的导航栏位于网页的正上方；在网页的右侧会有一些网站的广告或推荐信息等；再往下是网页的版权信息，中间部分就是所需要的正文信息。而网页中的导航栏、推荐信息、广告信息、版权信息等布局相同，内容也相同；图中的黑框部分就是网页的正文信息，布局相同，内容不同；这样的现象与现代网站开发模式有很大关系，目前大部分网站都采用动态页面以达到节省开发时间和费用的目的，通常采用CSS、JS、HTML来制作通用的网页前端模板，然后从数据库中读取相关数据，并与前台模板相结合展示给用户。

文献[3]基于布局相似性的网页正文内容提取研究，提出了利用相似网页内容布局和样式结构相似的特点提取正文信息的方法[4]，此方法的正确率受相互参照的两个页面在布局结构上的相似程度的影响较大，未考虑所提取目标网页来源庞杂，没有固定统一的格式，难以直接对比两个网页的DOM树去除噪音，因此从网页相似性入手，提出基于结构相似网页聚类的正文提取算法，先将提取到的网页进行相似度的计算，将相似度较高的网页聚为一类，再利用正文提取算法提取正文，并针对文献[3]中简单的树路径相似度计算方法进行改进，提出改进的基于简单树匹配的网页结构相似性度量算法，该方法提高了识别网页结构相似性的能力，对结构差别较大的网页进行良好的区分，进而能够适应更复杂多样的网页结构，在网页正文提取中具有更好的效果。

3　基于网页结构相似的网页正文提取

本文研究对象为中文主题型网页,主要是各大新闻网站在某一段时间内的新闻报道。计算网页相似度,将网页相似度较高的网页进行聚类,利用正文提取算法去除噪声提取正文。

3.1　网页相似度计算

由于基于结构相似网页聚类的正文提取算法中引入了聚类算法,所以就必须使用到网页相似性度量,原算法中采用的度量方法是基于树路径的网页结构相似度匹配算法,由于这种度量方法不能很好地体现网页的层次结构。所以,本文采用改进的基于简单树匹配的网页结构相似性度量方法替换原有的基于树路径的网页结构相似性度量方法。下面将对改进的基于简单树匹配的网页结构相似性度量方法进行详细的介绍。

定义4　根据现有网站开发模式,通常网页布局以各个区域形式呈现,将网页中的各个区域称为“块”。

定义5　将每个“块”视为一个整体,在各个块中又包含各个不同的内容区域,即将这个块划分成多个子块,将这个块称为子块的父块。如此不断迭代对网页结构进行划分,直到该块不能再被划分为子块时,称该块为叶子块。

定义6　将第一次对网页模板进行分块得到的子块称为一级子块。

如图2所示为常见网页的结构示意图,该网页在结构上可以划分为三大“块”,分别为上、中、下三块,最上一块一般包括标题、导航等信息;中间一块可分为左右两块,左边一块通常用于展示噪音链接、索引等信息,右边面积最大的一块通常是网页正文所在的位置;最下边一块一般用于展示网站的版权信息等。通常“块”之间的切割是由HTML中的“块级”标签完成的。

图2　网页结构示意图

图2所示网页的结构代码如下:

```
<div align="center">
<table width="1000" border="0" align="center" cellpadding="0" cellspacing="0">
最上边一块:标题、导航
</table>
<table width="1000" border="0" align="center" cellpadding="0" cellspacing="0">
中间块
<tr>
<td>
<table width="100%" border="0" align="center" cellpadding="0" cellspacing="0">
左边一块:快捷链接、索引等信息
</table>
</td>
<td>
<table width="100%" border="0" align="center" cellpadding="0" cellspacing="0">
右边一块:网页正文
</table>
</td>
</tr>
</table>
<table width="1000" border="0" align="center" cellpadding="0" cellspacing="0">
版权信息
</table>
</div>
```

将上述网页解析为DOM树,可以表示为如图3所示的树,其中,A为树的根节点,a、b、c为根节点的三个直接孩子节点,它们各自为一棵子树,d、e是节点b的两个孩子节点。

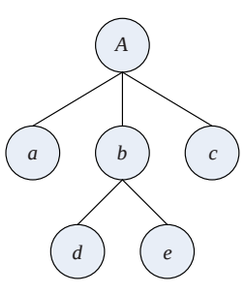

图3　网页DOM树

在实际情况中,网页模板的最上边一块和最下边一块中的内容是静态的、固定不变的,中间块的内容是动态生成的,其中左边块的内容在多数情况下也是静态的,右边块内的内容是由网页脚本在后台从数据库中取得的[10]。在DOM树中表现为,根节点的三棵子树中,第一棵子树与最后一棵子树在结构和内容都是静态的、保持不变的,第二棵子树中大部分节点为动态生成的,其余部分为静态的、保持不变的。

根据上述特征,本文首先提出一种通过对比网页“块”相似度计算网页结构相似度的方法,其基本思想如下:将网页的相似度总和“1”平均分配到网页的各大“块”中,然后对比两个网页的“块”的相似度。对比块的相似度可以将块视为整体1,并平均分配到这个块的所有子块。根据这种规则不断迭代,直到每个“子块”都是

“叶子块”。然后对比两个网页对应位置的“块”的相似度,再根据某种规则综合所有“块”的相似度即可得到两个网页的相似度。将网页的“块”对应于DOM树的节点,则“父块”等同于“父节点”,“叶子块”等同于“叶子节点”。那么就可以把计算两个网页“块”的相似度转化为计算两个网页DOM树的相似度。

一棵树可以表示为 $P=(X_{11},X_{21},X_{22},\cdots,X_{nm})$,其中 n 表示节点的层次,根节点的层次为1,X_{11} 表示树 P 的根节点,m 表示在当前层中节点的序号。X_{nm} 表示第 n 层中从左向右第 m 个节点[11]。

定义7 两棵树 P_1,P_2 的相似度表示为:

$$sim(P_1,P_2)=sim(P_1X_{11},P_2X_{11}) \tag{1}$$

$$sim(P_1X_{nm},P_2X_{nm})=\sigma\times\left[\sum_{i=1}^{Num(n+1)}sim(P_1X_{n+1,i},P_2X_{n+1,i})\right]\times Lp(n) \tag{2}$$

$$Lp(n)=\frac{1}{Num(n)} \tag{3}$$

$$\sigma=\begin{cases}1, & P_1X_{nm}=P_2X_{nm}\\ 0, & P_1X_{nm}\neq P_2X_{nm}\end{cases} \tag{4}$$

其中,P_iX_{nm} 表示树 P_i 的 X_{nm} 节点,$Num(n)$ 表示树中第 n 层节点的总数,并且:

定义8 两个节点相同指两个节点的标签名、属性集以及子节点的个数都相同,以及两个节点在DOM树中的位置也相同。

定义9 两个节点不同是指两个节点的标签名、属性集、子节点的个数以及在DOM树中的位置这些属性中,只要有一个不同,则两个节点不同。

所以,当判断出 $P_1X_{nm}\neq P_2X_{nm}$,$\sigma=0$,此时就没有必要继续计算 $\left[\sum_{i=1}^{Num(n+1)}sim(P_1X_{n+1,i},P_2X_{n+1,i})\right]$ 的值,从而大大减少了计算量。当 $P_1X_{nm}=P_2X_{nm}$ 时,那么这两个节点的子节点个数也是相同的,即它们的 $Num(n)$ 值是相同的。

对比两棵DOM树相似性算法Improved Simple Tree Matching (P_1,P_2),简称ISTM (P_1,P_2),描述如下。

算法1 ISTM (P_1,P_2)

```
Begin
If  P1X11≠P2X11
  return 0;
else
令 Lp(2)=1/Num(2)
If  Num(2)=0
Return 1;
For  i=1,Num(2)
  令 sim=ISTM(P1X2i,P2X2i)
  令 sum+=Lp(2)×sim
Endfor
Return sum;
End
```

算法给出了对比两棵DOM树相似性的过程,其过程如下:

步骤1 首先判断两棵树的根节点是否相同,如果根节点不同,则认为两棵树相似度为0,如果根节点相同,则进行第二步。

步骤2 判断两棵树根节点的子节点个数是否相同,若不相同,则两棵树相似度为0,如果相同,则进行第三步。

步骤3 假设根节点的子节点个数为 N,则每个子节点分配到的权重为 $\frac{1}{N}$。

步骤4 两棵树的相似度等于根节点下每个对应子节点的相似度与节点权重的乘积之和。

步骤5 递归计算子节点的相似度,然后从叶子节点逐层向上传递子节点的相似度。

然而,根节点下所有子节点得到相同的权重并不能很好地体现网页中各个“块”对网页模板的贡献。标题、导航、快捷链接、版权信息在整个“网页模板”中所占的比例远高于模板中“正文”的部分。采用平均分配策略可能导致如下情况:当两个网页正文部分的结构和内容相同,但导航和版权信息部分部分相似时,两个网页的相似度也非常高,这显然与算法的初衷是相背离的。

为了避免上述问题,本文又在上述算法的基础上,引入按“贡献”分配权重。即在相似度计算过程中,标题导航块、版权信息块以及靠近网页两端的在结构和内容相对固定的模块得到较大权重,而靠近网页中心位置,结构和内容变化可能性较大的模块得到较小的权重。而且,按“贡献”分配权重只针对网页的一级“子块”,因为将一级子块作为父块进行再分块后,得到的各个子块对父块的“贡献”是相同的。

修改权重因子 $Lp(n)$ 为 $Lp_n(m)$:

$$Lp_n(m)=\begin{cases}\dfrac{\left|\frac{Num(n)-1}{2}-m\right|+1}{\sum_{m=1}^{Num(n)}\left(\left|\frac{Num(n)-1}{2}-m\right|+1\right)}, & n=2\\ \dfrac{1}{Num(n)}, & n>2\end{cases} \tag{5}$$

综上所述,两棵树的相似度可表示为:

$$sim(P_1,P_2)=sim(P_1X_{11},P_2X_{11}) \tag{6}$$

$$sim(P_1X_{nm},P_2X_{nm})=\sigma\times\left[\sum_{i=1}^{Num(n+1)}sim(P_1X_{n+1,i},P_2X_{n+1,i})\right]\times Lp_n(m) \tag{7}$$

$$Lp_n(m)=\begin{cases}1, n=1\\ \dfrac{\left|\dfrac{Num(n)-1}{2}-m\right|+1}{\sum\limits_{m=1}^{Num(n)}\left(\left|\dfrac{Num(n)-1}{2}-m\right|+1\right)}, n=2\\ \dfrac{1}{Num(n)}, n>2\end{cases} \tag{8}$$

$$\sigma=\begin{cases}1, P_1X_{nm}=P_2X_{nm}\\ 0, P_1X_{nm}\neq P_2X_{nm}\end{cases} \tag{9}$$

算法2 ISTM(P_1,P_2,n)

Begin

If $P_1X_{11}\neq P_2X_{11}$

return 0;

If $Num(n+1)=0$

return 1;

If $n=1$

令 $Lp_1(1)=1$

If $n=2$

令 $Lp_2(m)=\dfrac{\left|\dfrac{Num(n)-1}{2}-m\right|+1}{\sum\limits_{m=1}^{Num(n)}\left(\left|\dfrac{Num(n)-1}{2}-m\right|+1\right)}$

If $n>2$

令 $Lp_n(m)=\dfrac{1}{Num(n)}$

For $i=1,Num(n)$

令 $sim=ISTM(P_1X_{2i},P_2X_{2i},n+1)$

令 $sum+=Lp(2)\times sim$

Endfor

Return sum;

End

算法给出了引入按贡献分配权重对比两棵DOM树相似性算法ISTM(P_1,P_2,n)的基本流程：

步骤1 首先判断两棵树根节点是否相同，如果根节点不同，则认为两个数的相似度为0，如果根节点相同，则进行步骤2。

步骤2 判断两棵树根节点的子节点个数是否相同，若不相同，则两棵树相似度为0；如果相同，则进行步骤3。

步骤3 假设根节点的子节点个数为N，则每个子节点分配到的权重为$\dfrac{\left|\dfrac{Num(n)-1}{2}-m\right|+1}{\sum\limits_{m=1}^{Num(n)}\left(\left|\dfrac{Num(n)-1}{2}-m\right|+1\right)}$。

步骤4 两棵树的相似度等于根节点下每个对应子节点的相似度与节点权重的乘积之和。

步骤5 递归计算子节点的相似度，从第三层子节点开始，每个子节点分配到的权重为$\dfrac{1}{N}$，N为当前层节点的总数。

步骤6 从叶子节点逐层向上传递子节点的相似度。

3.2 网页正文提取

通过上文中网页相似度的计算，运用层次聚类算法对网页进行聚类，得到一组布局相似的网页集；本节将详细介绍针对聚类结果中的簇，如何完成网页正文提取。

利用改进的网页相似度算法对数据集中的网页进行聚类后，结果集中每个簇包含的网页具有如下共同特征：

(1)都是基于一个或几个相似度极高的网页前端模板实现的。

(2)网页之间相同的部分为前端模板包含的内容。

(3)网页之间不同的部分为网页的正文[12]。

根据上述特征，提出提取网页正文的方法，基本思想为：利用DOM树解析工具将网页簇中的两个网页解析成DOM树，比较两棵DOM树并将其中相同的节点和子树删除，然后将剩余部分中的文字提取出来，即为网页正文内容。具体流程如算法所示。算法对比DOM并删除相同节点。

输入：目标网页DOM树D_1，参考网页DOM树D_2。

```
DeleteSame(D1,D2)
Begin
令E1=P1->Children;
令E2=P2->Children;
If E1==null||E2==null
  return D1;
Endif
For i=0:min(Length(E1),Length(E2))
  if(E1[i]==E2[i])
    remove(E1[i]);
  else
    DeleteSame(E1[i],E2[i])
Endfor
return D1
```

在完成了完全相同节点删除后，基于结构相似网页聚类的正文提取算法即已执行完毕。

3.3 算法分析

本文考虑到所提取目标网页来源庞杂，没有固定统一的格式，难以直接对比两个网页的DOM树去除噪音，因此从网页相似性入手，提出一种基于结构相似网页聚类的正文提取算法，先将提取到的网页进行相似度的计算，将相似度较高的网页聚为一类，簇中的网页具有较高的相似度，对比它们的DOM树并将相同的节点删除，这些相同部分就是这些网页中所共有的内容即与网页中的无关信息和冗余数据；剩余的部分就是要提取的正文内容。该方法提高了识别网页结构相似性的能力，对结构差别较大的网页进行良好的区分，进而能够适应更复杂多样的网页结构，在网页正文提取中具有更好的效果。

4　实验结果及分析

4.1　实验环境和数据

为了验证本文所提正文提取方法的有效性，本文对Readability正文抽取算法和基于网页相似度的正文提取算法进行了对比实验。实验平台为PC配置为CPU3.4 GHz，内存2 GB，500 GB硬盘，开发工具为Visual Studio 2010，算法采用C#语言实现，采用HtmlAgilityPack作为DOM树解析工具。

为了对算法正文提取的效果进行更加客观的评估，本文提取用于测试的网页样本以国内主要门户网站的新闻频道网页为主，网页来源网易、新浪、搜狐、腾讯、人民网、新华网、凤凰网等共10个网站，从上述10个网站中随机抽取100个网页，总数量为1 000；再从每个网站样本集中随机抽取一定数量的网页进行测试，抽样测试集容量为100。本文实验中所选取的新闻网页均为中文主题型网页，主题型网页的结构特征是以块的形式呈现的，含有一篇正文报道，而且同一站点下的同一频道中所有主题型网页间的布局结构极为相似，正文以及页面其他信息位置相对固定[13]；不同站点网页结构差别较大的网页能够更好地进行聚类，提高正文提取实验效果。

4.2　实验评价标准

网页正文提取的评价标准是查准率 P（Precision）和查全率 R（Recall），最后统计各项内容的平均值。查准率表示在抽取结果中，标记正文内容所占的比重；查全率表示正确提取正文信息与人工标注正文信息的比例[14]。

计算公式如下：

$$\text{查准率：}P=\frac{C}{A}\times 100\% \tag{10}$$

$$\text{查全率：}R=\frac{C}{B}\times 100\% \tag{11}$$

$$\text{综合评判指标：}F=\frac{2PR}{P+R}\times 100\% \tag{12}$$

其中 A 表示用实验方法得到的网页长度，B 表示人工标注正文部分内容的长度；C 表示 A 和 B 的公共部分长度[15]。

4.3　实验结果

为了验证本文算法的有效性，提取用于测试的网页样本以国内主要门户网站的新闻频道网页为主，将本文算法与文献[3]、Readability正文抽取算法进行了实验测试对比，其中测试结果查准率 P 、查全率 R 以及综合平均对比结果分别如表1至表3所示。

从实验结果来看，三种算法都能较好地提取新闻网页中的正文信息，对于网页中的导航栏等无关信息有效地去除。本文算法在查准率和查全率以及综合评定指标均优于文献[3]算法和Readability正文抽取算法。本文算法在进行正文提取之前先将采集到的网页进行聚类，提高了算法的准确度，降低了来自不同网站，结构复杂对正文提取的影响。从整体上看改进的方法能够实现较高精度的网页正文提取。

表1　正文提取结果查准率P对比　%

网站	本文算法	文献[3]算法	Readability算法
网易	92.24	84.24	85.29
新浪	98.26	91.73	93.88
搜狐	96.42	93.01	92.74
腾讯	95.72	94.00	88.62
人民网	92.93	91.75	87.13
新华网	97.46	92.37	94.24
凤凰网	98.43	95.61	93.69
联合早报	96.82	92.23	90.48
博客园	89.29	87.16	82.47
澎湃新闻	97.05	92.10	91.64

表2　正文提取结果查全率R对比　%

网站	本文算法	文献[3]算法	Readability算法
网易	98.80	96.73	94.15
新浪	95.84	96.15	93.11
搜狐	97.05	93.08	92.79
腾讯	98.74	97.03	95.41
人民网	97.25	94.68	91.85
新华网	94.10	93.98	90.47
凤凰网	97.87	96.24	95.31
联合早报	93.96	92.04	90.03
博客园	90.39	87.20	83.98
澎湃新闻	95.27	92.38	91.61

表3　正文提取结果综合评价　%

算法	P	R	F
本文算法	95.46	95.92	95.67
文献[3]算法	91.42	93.95	92.63
Readability算法	90.01	91.87	90.89

5　结论

本文描述了一种先通过网页聚类再进行正文提取的方法，该方法在正文提取充分考虑网页采集来源的不确定性，以及网页结构的复杂性对正文提取准确度的干扰，引入网页结构权重的概念，并将网页块相似度计算转化为网页DOM树相似度计算，对聚类之后结果簇中的所有网页内相似部分去除，剩余部分则是网页正文信息。实验结果显示本文提出的算法具有非常好的准确度，适合于大规模来源繁杂的网页正文提取。另一方面本文提出的基于结构相似网页聚类的正文提取算法运行效率较低。因此在未来的工作中，将继续研究并解决这个问题，提高算法运行效率。

参考文献：

[1] Arasu A，Garcia-Molina H.Extracting structured data from Web pages（poster）[C]//International Conference on Data Engineering，2003.

（下转139页）