基于子树相似度计算的网页评论提取算法研究*

朱毅华　张超群　曾　通　吴龙凤　徐玛丽　王东波　李晓晖
(南京农业大学信息科学技术学院　南京 210095)

【摘要】将网页评论的识别与自动提取转化为 DOM 树结构中的子树循环体识别问题,提出一种基于网页 DOM 子树相似度计算的方法,从网页中 < BODY > 节点向下逐层遍历识别出满足约定条件的评论块节点树。针对目前 DOM 树相似度计算算法在评论提取方面的性能不足,本算法同时考虑树节点的标签与位置信息构建叶节点路径,通过求解两个 DOM 子树的叶节点路径相似度矩阵得到两个子树的相似度。比较其他几种基于 DOM 相似度计算方法和一种基于标签权重的网页评论提取方法在性能和效率上的差异。实验表明,基于本算法的网页评论提取方法具有较高的查准率和查全率,总体优于现有网页评论提取方法。

【关键词】DOM 树　子树相似度　评论提取
【分类号】TP393

The Research of Recognizing the Reviews in Webpages Based on Calculating the Similarity of DOM - SubTrees

Zhu Yihua　Zhang Chaoqun　Zeng Tong　Wu Longfeng　Xu Mali　Wang Dongbo　Li Xiaohui
(College of Information Science and Technology, Nanjing Agricultural University, Nanjing 210095, China)

【Abstract】The processing of recognizing and extracting the reviews from webpages is transformed into recognizing the DOM - SubTrees which is cyclical in the DOM - Tree. Each node is iterated in the DOM, and the similarity between DOM - SubTrees is calculated, then those nodes meeting the requirements are found out. The proposed method can calculate the similarity between DOM - SubTrees in the end. To make it suitable in recognizing the reviews in webpages, the paper transforms the DOM - SubTrees into the paths of leave - nodes which consider the name and the position of tag. The authors compare 4 methods which are used in calculating the similarity between DOM - SubTrees, and also compare the algorithm with other algorithms which recognizes the reviews in webpages by using the weight of tags in the DOM - Tree. The experiments show that the algorithm has higher precision and recall rates, and more effective than other algorithms.

【Keywords】DOM - Tree　Sub - tree similarity　Review extraction

1　引　言

随着 Web2.0 的兴起,通过互动方式发布的网络信息迅速增多,其中包括大量通过评论、回复方式表达的具有

收稿日期:2013-07-22
收修改稿日期:2013-08-28
*本文系教育部人文社会科学研究青年基金项目"基于信息生态学的网络舆情管理机制与平台研究"(项目编号:10YJC870053)和江苏高校哲学社会科学研究重点项目"涉农网络舆情的政府监管研究"(项目编号:2011ZDIXM027)的研究成果之一。

情感倾向性的信息。除了提供网民表达个人诉求和意愿的方式,网络评论的价值越来越体现在其成为网站、政府获得用户态度的重要渠道,为企业把握用户的消费态度、政府把脉公众舆论倾向提供重要的决策依据,因而成为消费者行为分析、网络舆情监控等方面研究的关注焦点。与单一的网页正文信息提取目标不同,评论性网页往往集中了大量不同用户、不同时间发布的信息,加上 Web 文档中常见的脚本、广告、导航等噪音干扰,快速准确地识别和提取出网页中各个独立的评论块成为舆情、口碑分析等网络信息挖掘系统预处理环节中必不可少的工作。本文通过对评论性网页中评论块特征的分析,对网页源码进行基于 DOM 树的分析,提取具有评论块特征的代码段,将网页标签过滤后即可得到网页中的评论块。本文区别于现存方法,以评论块所具有的特征(结构类似,循环出现)为依据,结合 DOM 树分析的方法有效地从网页中提取用户评论块。实验证明该方法对于博客类、新闻类、SNS 类网页具有较高的识别效率。

2　相关研究

目前在 Web 信息抽取领域的研究重点多集中于网页中正文信息的提取,较少涉及评论信息的提取。总体上各类 Web 信息抽取方法可分为 4 种:

(1)基于网页视觉特征的方法。利用网页在视觉上的特点和布局上的特征制定规则抽取网页信息[1-3]。由于网页评论信息在视觉特征上并不总是有别于其他内容,这种方法在对评论信息这样数量较大、视觉特征不够明显的信息抽取中意义不大。

(2)基于语义特征的方法。利用统计语言模型的方法提取文本中代表性词语并结合文档对象模型(Document Object Model, DOM)进行语义特征扩展,从而提取出所需信息[4,5]。此类方法对代表性不够强、语义不够丰富的词语的忽视容易导致网页抽取信息的缺失。此外,该方法具有比较复杂的规则,使用过程中规则的获取和总结较为困难。

(3)基于模板的方法。通过对网页源代码进行人工分析,总结网页信息的提取规则,进而生成网页模板包装器[6,7]。这种方法具有很高的信息抽取准确性,在查全率和查准率上至少能达到 90%,但如果网页结构进行了调整,即便是很微小的变动也可能导致信息提取的失败。为了克服其对人工依赖较大的缺陷,李效东等[8]以 DOM 树路径作为信息抽取的“坐标”,利用归纳学习的算法半自动化生成提取规则,作为网页数据源包装器组成的重要构建。

(4)基于 DOM 树分析的方法。李姜[9]基于 DOM 结构对网页进行分块,结合信息熵的迭代计算技术进行评论块的自动发现与抽取。杨奕锦[10]对去噪后的标签树进行权重赋值,然后利用标签权值判断标签树的相似性,利用标签树的位置连续性来识别数据记录区域,实现了一个电子商务类网站顾客评价的抽取系统。刘伟等[11]提出的评论抽取方法将抽取过程分为两步,首先通过深度加权的方法从网页中抽取评论记录,其次通过比较 DOM 树中节点的一致性将用户评论内容提取出来。

此外还有一些其他的评论提取算法,例如文献[12]针对不同博客发布系统构建评论抽取模式,实现从博客中抽取评论信息。文献[13]结合网页评论在网页中循环出现的特征,利用过滤策略移除非评论块并使用 SVM 算法作为评论块与非评论块分类器,实现从博客中提取评论信息。以上方法中很多仍需要人工干预,一些算法规则复杂难以实现,但基于网页 DOM 树结构分析思想的提出为网页评论信息提取提供了有益的思路,是本研究的起点。

3　网页评论提取算法

3.1　问题描述

评论信息是用户针对网站主题如博客文章、商品介绍、论坛话题等发布的回复或评论,其内容结构对于不同类型的网站略有不同,一般包括评论作者、时间和内容等。网页评论信息提取面临的主要问题为:

(1)网页结构复杂,含有大量的与主题无关的信息,这些“噪音”的存在将大大影响评论提取的质量;

(2)同一网站栏目的网页结构大致相同,而不同网站栏目间的网页结构却可能千差万别,这就要求提出一种通用算法能对不同结构的网页进行统一处理;

(3)用户评论的内容除文字内容外还可能是图片、超链接甚至视频,这些元素的存在使得不同评论块在结构上并不完全相同。

统计分析显示:

(1)用户评论一般以相似结构集中显示于页面的

某个区域；

(2)同一区域中的所有评论块所使用的 HTML 标签及其结构非常相似；

(3)网页中用户评论块一般集中在一个父 HTML 标签下，从而对应于相同的 DOM 父节点。

这些特点源自目前绝大多数网站基于“模板 + 数据库”方式构建，评论信息由数据库中的记录根据统一的模板动态生成，从而呈现出内容与格式上的循环。

因此，设计出一种算法自动识别出评论信息的“循环体”，是解决评论信息抽取问题的重要思路。DOM 树是网页元素的物理结构，评论信息的“循环体”结构必然反映在相似 DOM 子树的重复出现上，基于此，本文将评论信息的识别问题转化为 DOM 子树的相似度比较问题，再结合一定的规则特征，通过对网页 DOM 树节点的遍历以准确定位评论信息的位置。

3.2 网页评论提取步骤

基于上述思路本文提出以下假设：一个页面包含多条评论信息，而重要的评论信息往往出现在热点话题中，页面中评论数量应达到一定的阈值，从而使根据相似度算法区分出评论块成为可能；同时，区别于网页设计中经常用来生成导航栏等内容的相似循环结构，真实的评论块往往具有更大的节点深度。该假设忽略了数量较少评论页面中的重要评论信息，但可以认为重要的信息产生更多回应评论的可能性也较高，通过爬虫的定期重复访问一定程度上可弥补该缺陷。

基于以上假设，本算法自网页的 < BODY > 节点开始进行循环迭代，计算当前节点的所有子树间的相似度矩阵并寻找符合以下条件的节点：

(1)要提取的页面中至少存在 b 条评论信息，那么在当前节点的子树集合中，若某个子树表示一个评论块，那么与该子树相似度大于阈值 a 的个数应至少为 b - 1；

(2)当前节点的深度大于设定值 level。

b 与 level 由预设的经验值给出，算法流程如图 1 所示。

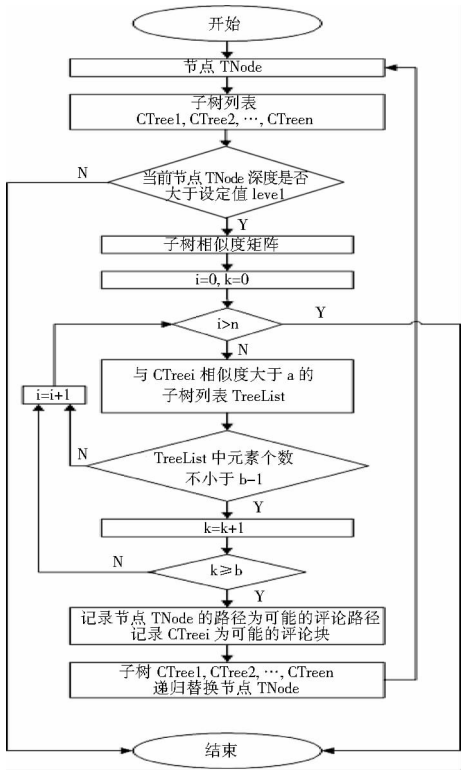

图 1 网页评论提取关键步骤

算法关键步骤描述如下：

(1)获取当前节点以 TNode 表示；

(2)获取 TNode 的所有子树列表 CTree1, CTree2, …, CTreei, …, CTreen；

(3)如果节点 TNode 的 DOM 树的深度大于设定值 level，则执行步骤(4)，否则识别过程结束，跳至步骤(7)；

(4)构建 TNode 所有子树的相似度矩阵；

(5)对相似度矩阵进行统计分析。对于子树 CTreei 而言，判断条件为存在不小于 b - 1 个相似度大于阈值 a 的同级子树，若满足上述条件的子树 CTreei 的个数大于 b，则判断当前记录自 < HTML > 标签至节点 TNode 的路径为一条可能的评论路径，而 CTreei 即为一条评论块；

(6)依次使用子树 CTree1, CTree2, …, CTreei, …, CTreen 递归替换节点 TNode 进行步骤(2)；

(7)结束。

本算法的关键在于步骤(4)选择合理的 DOM 子树相似度计算方法，该方法要求能给出子树结构上相似程度的定量值，同时当子树存在一定差异时能够定量衡量差异的大小。目前用于 DOM 树相似度计算的方法主要有：

(1)基于编辑距离的方法(Edit Distance, ED)。其思想是通过计算将一棵 DOM 树转化为另一棵 DOM 树所需要进行的编辑次数来衡量其相似度[14,15]。

（2）基于最长公共子串的方法（Longest Common Substring，LCS）。通过广度优先遍历将 DOM 树转化为由标签构成的串，以字符串最长公共子串作为相似度的度量标准[16]。

（3）基于简单树匹配的方法（Simple Tree Matching，STM）。利用动态规划计算两棵树的最大匹配节点个数得到两个 DOM 树之间的相似度[17]。

（4）基于最大合成树的方法（Maximal Combined Tree，MCT）。将两棵树合并为一棵新树，新树生成后通过比较原 DOM 树与新生成的 DOM 树的差异从而计算相似度[18,19]。

这些 DOM 树相似度计算的方法主要用于网页间的相似度计算，用于评论提取时都有不同程度的问题。如基于编辑距离的方法与简单树匹配的方法因为过于严苛要求子节点顺序，导致评论节点对应的子树之间相似度偏低。基于最长公共子串的方法在将 DOM 树转化为标签串的过程中丢失了节点的位置信息，导致相似度整体偏高。本文借鉴上述几种方法的优点，提出基于叶节点路径的 DOM 子树相似度的计算方法（Paths of Leaves，POL），在相似度计算结果和性能上都有较为明显的改进。

3.3　基于叶节点路径的 DOM 子树相似度计算

（1）相关定义

定义 1 叶节点：一个节点的子树的个数称为节点的度。度为 0 的节点被称为叶节点。

定义 2 叶节点路径：是指在一棵 DOM 子树中自根节点到叶节点所经过的节点的标签序列表示。

定义 3 叶节点路径长度：是指自 DOM 子树根节点到叶节点所经过的节点的个数。

定义 4 叶节点路径集合：一个 DOM 子树所包含的所有叶节点路径的集合。

定义 5 节点深度：节点的叶节点路径集合中，叶节点路径长度最大值即为该节点深度。

（2）具体方法步骤

首先生成 DOM 树的全部叶节点路径，求解两个 DOM 树的叶节点路径相似度矩阵，进而得到两个子树的相似度。具体算法如下：

①获取两个子树 Tree1，Tree2；

②分别获取两棵树的叶节点路径 P11，P12，P13，…，P1n 和 P21，P22，P23，…，P2m。图 2 展示了一个树结构转化为路径的例子；

③初始化相似度矩阵 Sim[n][m]；

④执行双层循环

```
from i = 1 to n //遍历 D1 叶节点路径
  from j = 1 to m//遍历 D2 叶节点路径
    Sim[i][j] = Sim(P1i,P2j)//相似度矩阵赋值
```

⑤利用相似度矩阵计算 Tree1 与 Tree2 的相似度。

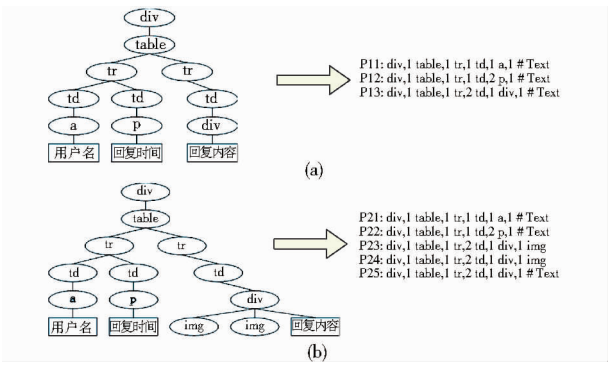

图 2　DOM 树转化为叶节点路径

在步骤④中 Sim(P1i,P2j) 表示计算叶节点路径 P1i 和 P2j 的相似度，计算方法实现如下：

①取两个叶节点路径并将其表示为 <标签名，次序> 的序列，其中标签名为路径中节点名称，次序为该节点相对于父节点而言是第几个子节点；

②计算两个叶节点路径合并后的向量维度；

③将两个叶节点路径表示为该维度向量；

④运用余弦相似度计算两个叶节点路径的相似度。

计算过程如图 3 所示：

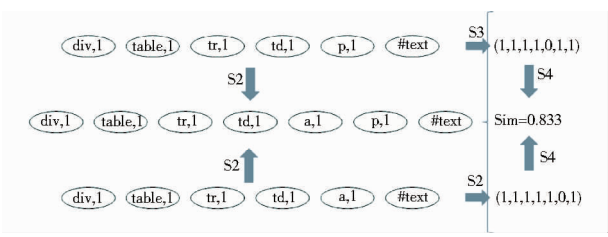

图 3　叶节点路径相似度计算过程

算法中将两个叶节点路径表示为统一维度的向量时有两种方式：一种是单纯考虑某一个维度在路径中是否出现，出现则为 1，不出现则为 0；另一种方式考虑路径中节点的层次对于确定子树的结构具有不同的贡献作用。实验发现，带权重的方法无论倾向于高层节点还是倾向于低层节点，都会导致相似度计算结果集中于偏高或偏低的区域，造成相似度阈值选取困难，因此本文采取不加权的方式。

子树 Tree1 与 Tree2 最终相似度的计算方法为：对

每条叶节点路径在另一棵子树的叶节点路径集合中找一条最相似的路径,将相似度乘以自身长度。综合考虑所有叶节点路径求得子树的相似度。子树相似度计算公式如下:

$$Sim(Tree_1, Tree_2) = \left(\frac{\sum_{i=1}^{n} [Max(P_{1i}) \times Len(P_{1i})]}{\sum_{i=1}^{n} Len(P_{1i})} + \frac{\sum_{i=1}^{m} [Max(P_{2i}) \times Len(P_{2i})]}{\sum_{i=1}^{m} Len(P_{2i})} \right) / 2$$

其中,$Max(P_{1i})$表示 D2 的叶节点路径集合中与P_{1i}最相似的叶节点路径的相似度,$Len(P_{1i})$表示叶节点路径长度。

以图 2 为例,两个子树的叶节点路径相似度矩阵如表 1 所示:

表 1　子树叶节点路径相似度矩阵

Sim	P21	P22	P23	P24	P25
P11	1.0	0.667	0.5	0.5	0.667
P12	0.667	1.0	0.333	0.333	0.5
P13	0.5	0.5	0.833	0.833	1.0

则最终相似度计算如下:

$$Sim(Tree_1, Tree_2) = \left(\frac{1.0\times6 + 1.0\times6 + 1.0\times6}{6+6+6} + \frac{1.0\times6 + 1.0\times6 + 0.833\times6 + 0.833\times6 + 1.0\times6}{6+6+6+6+6} \right) / 2 = 0.967$$

因而可以认为这两棵子树是同一评论区域的两个评论块。

图 2 中两个子树描述了网页评论中较为常见的现象,即网站支持用户在评论中附加图片,虽然两条评论中一条包含图片另一条不包含图片,但就结构而言应该具有较高的相似度。在本例中,以编辑距离(ED)、简单树匹配(STM)、最长公共子串(LCS)、最大合成树(MCT)、本文提出的叶节点路径(POL)算法计算的相似度结果如表 2 所示:

表 2　5 种子树相似度计算方法计算结果

方法	ED	STM	LCS	MCT	POL
相似度	0.937 5	0.928	0.937 5	0.849	0.967

为反映几种相似度算法在评论块识别中的区分度差异,将上述 5 种子树相似度计算方法应用于实际页面中进行评论块识别的效果如表 3 所示。实验随机选择数据集中 10 个网页分别在不同相似度水平的比较算法进行评论识别的效果。

表 3　5 种子树相似度计算方法对评论识别效果比较

方法	相似度水平	识别率/%	错误率/%	正确识别相似度范围	错误识别相似度范围
ED	0.6	45.45	54.55	0.920 – 0.975	0.667 – 0.972
	0.7	48.78	51.22	0.920 – 0.975	0.750 – 1.000
	0.8	62.50	37.50	0.920 – 0.975	1.000
	0.9	62.50	37.50	0.920 – 0.975	1.000
STM	0.6	45.45	54.55	0.903 – 0.952	0.644 – 0.951
	0.7	48.78	51.22	0.903 – 0.952	0.750 – 1.000
	0.8	62.50	37.50	0.903 – 0.952	0.975
	0.9	62.50	37.50	0.903 – 0.952	0.975
LCS	0.6	19.05	80.95	0.900 – 0.973 3	0.694 – 1.000
	0.7	21.28	78.72	0.900 – 0.973 3	1.000
	0.8	21.28	78.72	0.900 – 0.973 3	1.000
	0.9	21.28	78.72	0.900 – 0.973 3	1.000
MCT	0.6	30.77	69.23	0.783 – 0.877	0.662 – 0.923
	0.7	40.00	60.00	0.783 – 0.877	0.743 – 0.944
	0.8	61.29	38.71	0.803 – 0.892	0.887 – 0.994
	0.9	0.00	100.00	–	0.887 – 0.994
POL	0.6	100.00	0.00	0.852 – 0.983	–
	0.7	100.00	0.00	0.852 – 0.983	–
	0.8	100.00	0.00	0.852 – 0.983	–
	0.9	100.00	0.00	0.900 – 0.994	–

实验结果表明 POL 算法具有明显优势。从各算法的相似度计算原理及测试结果初步得出以下结论:

(1)MCT 过分考虑节点的差异,忽视子树在结构上的相似性,导致计算子树相似度时相似度整体偏小。

(2)ED、STM 在思想上类似,都是侧重于从树之间的差异来求得两棵树的相似度,两者的计算结果也较为接近。LCS、MCT 从两棵树相似的角度出发,这两种算法所得相似度的值与子树的大小有很大关系,相似度值不能真实反映两个节点相似程度。表 3 中 LCS、MCT 的错误主要是因为其对 DOM 树大小敏感,相对拥有较小体积文本块的相似度计算偏高,易产生误识别。

(3)POL 在两个集合中找到最相似的路径从而计算两棵子树的相似度,一方面考虑了包含真实评论信息的叶子节点的差异性,同时考虑了子树在结构上的相似性。在计算相似度时将路径的相似度与其长度相乘,求和后取平均值,抵消了子树的大小对相似度值的影响,即使是包含很多子孙节点的两个子树,只要它们在结构上大体相同,也可以有较为合理的相似度值,从而对评论块具有较高的区分度。

4　算法分析与测评

为了测试本文算法的性能与效率,选取了其他 DOM 树相似度计算算法作为对比进行分析和实验测评。此外,由于文献[8]是相关研究中专门用于评论

自动提取的方法,本文也进行了专门的对比。

4.1　测试数据集

目前尚无针对网页评论提取的测试集,本文采集了来自 20 个不同类型知名站点的 20 个包含评论信息的网页构建数据集。各网站的页面都使用了模块技术,因此可以认为同一站点栏目中网页内容组织方式大体相当,如果对其中一个网页能够自动识别评论块,那么对于同一站点中的其他网页也能够正确识别。数据来源如表 4 所示:

表 4　数据集中数据来源

序号	类型	站点名称	URL	评论数量
1	博客	网易博客	http://blog.163.com/	40
		天涯博客	http://blog.tianya.cn/	12
		新浪博客	http://blog.sina.com.cn/	50
		搜狐博客	http://blog.sohu.com/	19
		和讯博客	http://blog.hexun.com/	10
2	新闻网站	新浪新闻	http://news.sina.com.cn/	32
		腾讯新闻	http://news.qq.com/	20
		搜狐新闻	http://news.sohu.com/	34
		凤凰资讯	http://news.ifeng.com/	20
		网易新闻	http://news.163.com/	37
3	电子商务	亚马逊	http://www.amazon.cn/	10
		京东商城	http://www.jd.com/	8
		淘宝特卖	http://www.taobao.com/	20
		苏宁易购	http://www.suning.com/	10
		当当商城	http://www.dangdang.com/	5
4	网络社区	猫扑	http://tt.mop.com/	15
		凯迪社区	http://club.kdnet.net/	15
		豆瓣网	http://book.douban.com/	9
		天涯社区	http://focus.tianya.cn/	94
		西祠胡同	http://www.xici.net/	31

(注:访问时间为 2013 年 6 月 5 日 14 时 20 分。)

4.2　测度指标

选择评论抽取的查全率(Recall)和查准率(Precision)以及 F1 值作为算法评估效率的指标,公式分别如下:

$$R = \frac{\text{判断正确的评论块数量}}{\text{总评论块数量}}$$

$$P = \frac{\text{判断正确的评论块数量}}{\text{判断为评论块的数量}}$$

$$F1 = \frac{2 \times P \times R}{P + R}$$

其中,F1 值是综合考虑了查全率和查准率的复合评估指标。

性能指标主要是从时间上反映算法的优劣情况,是指算法在特定平台上完成一次识别任务所消耗的时间,实验中对同一任务进行多次实验后取平均结果。实验测试平台为:CPU:Intel® Core™ i5 - 2410M;内存:8GB;操作系统:Fedora17;使用 Java 语言实现算法。

4.3　实验及结果分析

(1)实验一:不同子树相似度计算方法运用于评论抽取的效果

该实验以数据集中的所有网页为测试对象,评测不同子树相似度计算方法运用于评论抽取的效果,此处选择 ED、MCT、LCS 与 POL 进行对照实验。实验结果如图 4 - 图 6 所示:

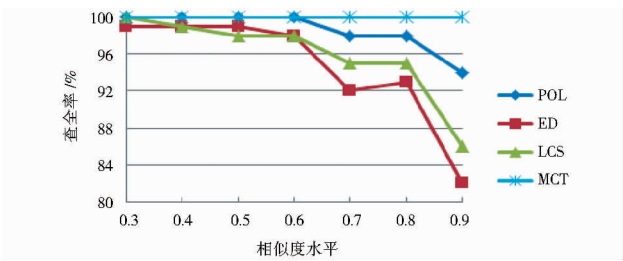

图 4　4 种子树相似度计算方法在不同相似度水平的查全率比较

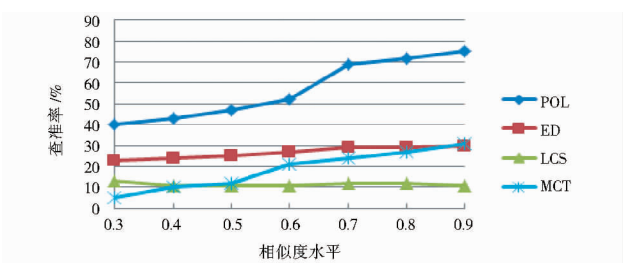

图 5　4 种子树相似度计算方法在不同相似度水平的查准率比较

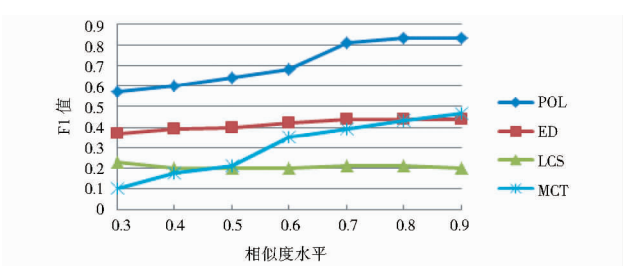

图 6　4 种子树相似度计算方法在不同相似度水平的 F1 值比较

实验数据表明,POL 算法运用于评论提取中效果表现优异,查准率和查全率都优于其他子树相似度的计算方法。此外,ED、LCS 算法随着相似度阈值的增加查准率变化较为缓慢,表明这两种算法将大量非评论内容识别为评论内容,并且认为它们之间的相似度较

高。MCT 虽然在查全率方面几乎接近于 1,但是在查准率方面表现一般,表明该方法虽然能够把评论块都找出来但是却错误地把很多网页中的噪音认为是评论块。

(2)实验二:相似度阈值对效果的影响

统计表明多数网站导航栏、标题列表、广告列表等节点的深度小于 7,将经验值 level 设置为 6 较为合适。但也存在例外,比如豆瓣网的导航栏的深度达到 9,评论提取时会误将导航栏识别为评论块区域。在网页的评论数大于 5 的情况下,表 5 给出了 POL 算法中相似度阈值 a 在 0.7、0.8、0.9 水平上的实验结果。

表 5　不同相似度阈值水平的算法效率比较

类型	评论数量	0.7					0.8					0.9				
		A	T	R(%)	P(%)	F1(%)	A	T	R(%)	P(%)	F1(%)	A	T	R(%)	P(%)	F1(%)
博客网站	131	131	140	100	93.6	96.7	130	137	99.2	94.9	97.0	126	133	96.2	94.7	95.4
新闻网站	143	137	183	95.8	74.9	84.1	137	176	95.8	77.8	85.8	125	143	87.4	87.4	87.4
电子商务	53	48	159	90.6	30.2	45.3	48	135	90.6	35.6	51.1	47	128	88.7	36.7	51.9
网络社区	164	164	216	100	75.9	86.3	164	215	100	76.3	86.6	164	213	100	77.0	87.0
合计	491	480	698	97.8	68.8	80.7	479	663	97.6	72.2	83.0	462	617	94.0	74.9	83.4

实验结果表明,当相似度阈值取 0.8 时 POL 算法在查准率和查全率上都达到较为理想的取值。结果显示本算法对于电子商务类型的站点的识别效率相对较低,而对于博客、新闻网站、网络社区等类型的站点算法的识别效率较高。经分析其原因如下:

①电子商务类网站中经常出现的同类商品推荐功能类似于评论块“循环体”形式,对识别造成了干扰。

②电子商务类网站的导航栏存在对商品进行多级分类的情况,可能会被“误识别”为评论块,造成查准率下降。

③出于安全方面的需求,电子商务网站在页面中大量嵌入脚本代码,一方面加大了算法的计算复杂度,另一方面也造成页面结构的混乱,对评论识别的效率造成直接的影响。

(3)实验三:与文献[8]方法的性能与效率比较

与文献[8]中提出的评论提取方法作对照,将 POL 算法的相似度阈值选择 0.8,level 设定值为 6,b 设为 5,对两种算法的性能进行了测试。两种评论提取算法的性能比较如图 7 所示:

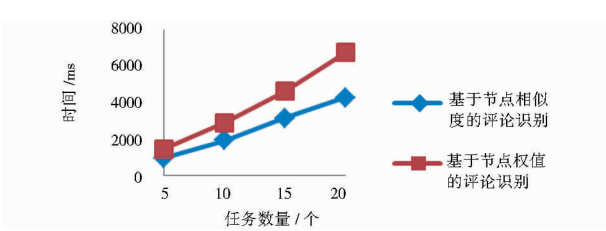

图 7　两种评论提取算法的性能比较

从性能上看,基于节点(即子树)相似度计算的方法耗时小于文献[8]提出的基于节点权值的评论提取算法。同时,随着网页数量的增加,基于节点权值的评论提取算法耗时增幅高于线性,基于节点相似度的评论提取算法呈现为线性。

从算法的效率上看,两种方法的查全率与查准率如图 8、图 9 所示:

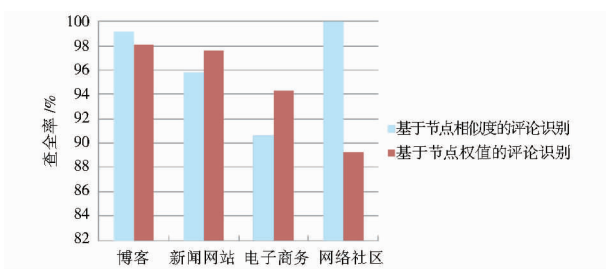

图 8　两种评论提取算法的查全率比较

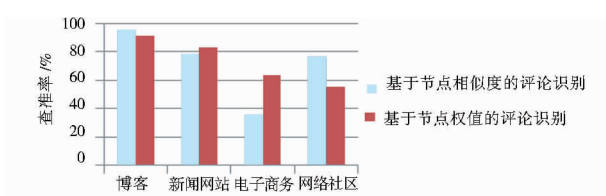

图 9　两种评论提取算法的查准率比较

结果表明基于节点相似度计算的评论提取方法更加适用于对博客、网络社区两种类型的网页评论提取;而在新闻网站、电子商务类型网站中基于节点权值的评论提取算法表现更为优异。可能的原因如下:

①基于前述原因,电子商务、新闻网站网页中的同类商品推荐、新闻列表、分类导航等具有和评论块相似特征的网页噪音,干扰了基于节点相似度计算的网页评论识别效果。而基于节点权值的计算方法利用权值对这些噪音进行了过滤。

②对于博客、网络社区两种网站而言,用户在参与评论的过程中更加注重互动,用户的评论中有对其他用户评论的引用或者是对其他用户评论的“再评论”,导致不同 DOM 子树结构上的不平衡。本文的算法偏重于 DOM 结构本身的相似性,减小了对评论引用及“再评论”对评论块识别的影响,

从而对于博客、网络社区这两类网站具有较高的识别效率。

5　结　语

本文所实现的评论抽取算法在有效性以及适用性上与其他算法相比都具有一定的优势，但在网页噪音的过滤方面还有待改进之处。后续的工作集中在以下方面：进一步吸收其他算法中的一些思想，对算法的查准率进行优化；继续优化算法，减小时间复杂度，提高实用性；由于越来越多采用 Ajax 技术网站的出现，许多网页评论是通过异步请求加载到页面中的，单纯通过 URL 的方式采集到的网页并不包含真实评论的内容，也就无法构建其 DOM 树，考虑使用 WebKit 等浏览器内核解决上述问题，在浏览器内核对页面中的脚本代码、异步请求进行解析后反馈包括全部内容的完整 DOM 树后即可完成用户评论的提取。

参考文献：

[1] 安增文，徐杰锋. 基于视觉特征的网页正文提取方法研究[J]. *微型机与应用*，2010(3)：38－41.（An Zengwen，Xu Jiefeng. The Research on Vision－based Web Page Information Extraction Algorithm [J]. *Microcomputer & Its Applications*，2010(3)：38－41.）

[2] 杜鹏. 基于视觉特征的 Web 页面信息抽取技术的研究[D]. 兰州：西北师范大学，2009.（Du Peng. Research on Vision－based Web Page Information Extraction Technology[D]. Lanzhou：Northwest Normal University，2009.）

[3] Cai D，Yu S P，Wen J R，et al. VIPS：A Vision －based Page Segmentation Algorithm[R]. Microsoft Technical Report，MSR－TR－2003－79. 2003.

[4] Liao X，Cao D，Tan S，et al. Combining Language Model with Sentiment Analysis for Opinion Retrieval of Blog－Post[C]. In：*Proceedings of Text Retrieval Conference* 2006 (*TREC*'06)，Maryland，USA. 2006：211－213.

[5] Hu M，Sun A，Lim E. Comments－oriented Blog Summarization by Sentence Extraction[C]. In：*Proceedings of the* 16*th ACM Conference on Information and Knowledge Management* (*CIKM*'07). New York：ACM，2007：901－904.

[6] 连小刚. 基于 DOM 的 Web 信息抽取系统设计与实现[D]. 武汉：华中科技大学，2009.（Lian Xiaogang. Design and Implementation of Web Information Extraction Based on DOM [D]. Wuhan：Huazhong University of Science and Technology，2009.）

[7] Banko M，Cafarella M J，Soderland S，et al. Open Information Extraction from the Web[C]. In：*Proceedings of the* 20*th International Joint Conference on Artificial Intelligence* (*IJCAI*－07). Hyderabad：AAAI Press，2007：2670－2676.

[8] 李效东，顾毓清. 基于 DOM 的 Web 信息提取[J]. *计算机学报*，2002，25(5)：526－533.（Li Xiaodong，Gu Yuqing. DOM－based Information Extraction for the Web Sources[J]. *Chinese Journal of Computers*，2002，25(5)：526－533.）

[9] 李姜. 基于 DOM 的评论发现及抽取模型研究[J]. *计算机工程与设计*，2007，28(9)：2150－2153.（Li Jiang. Reviews Discovery and Opinions Extraction Model Based on DOM [J]. *Computer Engineering and Design*，2007，28(9)：2150－2153.）

[10] 杨奕锦. Web 页面用户评论信息抽取技术研究[D]. 杭州：浙江大学，2011.（Yang Yijin. Study on Information Extraction Technology in Web Pages of Review [D]. Hangzhou：Zhejiang University，2011.）

[11] 刘伟，严华梁，肖建国，等. 一种 Web 评论自动抽取方法[J]. *软件学报*，2010，21(12)：3220－3236.（Liu Wei，Yan Hualiang，Xiao Jianguo，et al. Solution for Automatic Web Review Extraction [J]. *Journal of Software*，2010，21(12)：3220－3236.）

[12] Parapar J，Lopez－Castro J，Barreiro Á. Blog Posts and Comments Extraction and Impact on Retrieval Effectiveness[C]. In：*Proceedings of the* 1*st Spanish Conference on Information Retrieval* (*CERI*2010)，Madrid，Spain. 2010：5－16.

[13] 高虹安. 部落格贴文评论撷取及其在意见探勘上的应用[D]. 台北：台湾大学，2008.（Kao H. Comment Extraction from Blog Posts and Its Applications to Opinion Mining[D]. Taipei：National Taiwan University，2008.）

[14] 张瑞雪. 基于 DOM 树的网页相似度研究与应用[D]. 大连：大连理工大学，2011.（Zhang Ruixue. Research & Application of Web Similarity Based on DOM Tree [D]. Dalian：Dalian University of Technology，2011.）

[15] 聂卉，黄贵鹏. 树编辑距离在 Web 信息抽取中的应用与实现[J]. *现代图书情报技术*，2010(5)：29－34.（Nie Hui，Huang Guipeng. The Application and Implementation of Tree Edit Distance in Web Information Extraction [J]. *New Technology of Library and Information Service*，2010(5)：29－34.）

[16] 罗刚. 解密搜索引擎技术实战(Lucene & Java 精华版)[M]. 北京：电子工业出版社，2011.（Luo Gang. Actual Battles of Decoding Searching Engine Technology (Lucene & Java Essentials) [M]. Beijing：Publishing House of Electronics Industry，2011.）

[17] 何昕，谢志鹏. 基于简单树匹配算法的 Web 页面结构相似性度量[J]. *计算机研究与发展*，2007，44(S3)：1－6.（He Xin，Xie Zhipeng. Structural Similarity Measurement of Web Pages Based on Simple Tree Matching Algorithm [J]. *Journal of Computer Research and Development*，2007，44(S3)：1－6.）

[18] Manning C D，Schütze H，Raghavan P. 信息检索导论[M]. 北京：人民邮电出版社，2010.（Manning C D，Schütze H，Raghavan P. Introduction to Information Retrieval[M]. Beijing：Posts & Telecom Press，2010.）

[19] 刘兵. Web 数据挖掘[M]. 北京：清华大学出版社，2013.（Liu Bing. Web Data Mining [M]. Beijing：Tsinghua University Press，2013.）

（作者 E－mail：wangdongbo0102@gmail.com）