基于文本块密度和标签路径覆盖率的网页正文抽取*

刘鹏程，胡　骏，吴共庆
（合肥工业大学 计算机与信息学院，合肥 230009）

摘　要： 大多数网页除了正文信息外，还包括导航、广告和免责声明等噪声信息。为了提高网页正文抽取的准确性，提出了一种基于文本块密度和标签路径覆盖率的抽取方法（CETD-TPC）。结合网页文本块密度特征和标签路径特征的优点，设计了融合两种特征的新特征，利用新特征抽取网页中的最佳文本块，最后，抽取该文本块中的正文内容。该方法有效地解决了网页正文中噪声块信息过滤和短文本难以抽取的问题，且无须训练和人工处理。在 CleanEval 数据集和从知名网站上随机选取的新闻网页数据集上的实验结果表明，CETD-TPC 方法在不同数据源上均具有很好的适用性，抽取性能优于 CETR、CETD 和 CEPR 算法。

关键词： 正文抽取；文本块密度；标签路径覆盖率；特征融合
中图分类号： TP391.1　　**文献标志码：** A　　**文章编号：** 1001-3695(2018)06-1645-06
doi:10.3969/j.issn.1001-3695.2018.06.010

Webpage content extraction via text block density and tag path coverage

Liu Pengcheng, Hu Jun, Wu Gongqing
(*School of Computer & Information*, *Hefei University of Technology*, *Hefei* 230009, *China*)

Abstract: Most Webpages contains the content information, as well as noisy information such as navigation, advertisements and disclaimer notices. To address this problem and improve the accuracy of Webpage extraction, this paper proposed a Webpage content extraction method via text block density and tag path coverage (CETD-TPC). Combining the advantages of Webpage text block density feature and tag path feature, this paper designed a new feature, TDTPC, which mixed the two features together. Then it extracted the best text block from a Webpage by using the TDTPC feature. Finally, it extracted contents from the content block. Without the manual processing and training, CETD-TPC is an effective solution to deal with the problems of noise block information filtering and short text extraction. Experimental results on CleanEval datasets and Web news pages randomly selected from well-known websites show that the CETD-TPC method has good applicability on different data sets and performs better than CETR, CETD and CEPR.

Key words: content extraction; text block density; tag path coverage; feature fusion

0　引言

互联网成为信息时代的标志，Web 逐渐成为很多应用的重要信息来源。《2016 年互联网趋势报告》(2016 Internet trends)[1]指出，全球互联网用户已达 30 亿，比上年增长 9%，互联网全球渗透率达到 42%。根据中国互联网信息中心(CNNIC)的调查显示[2]，互联网用户的主要行为之一就是阅读网络新闻，同时中国互联网用户阅读互联网新闻的普及率为 81.6%，其中移动互联网用户比率达到 78.9%。移动互联网的快速发展，对新闻媒体质量有着更高的要求，各种各样的 Web 新闻替代传统报纸，成为人们获取最新信息的重要载体。

Web 网页中除了主要内容外，还包括大量与主题内容无关的噪声信息，如导航信息、广告文字、推荐链接、版权申明等。Gibson 等人[3]研究表明噪声数据占网页整体数据的 40% ~ 50% 的规模，且预测仍在持续增长中。由此可见，网页中的这些噪声对 Web 信息检索、Web 内容的管理和分析、Web 信息聚合和推送等研究带来了巨大的挑战。因此，过滤网页中的噪声信息，抽取网页的正文内容，具有重要的研究价值和应用前景。

与此同时，随着互联网技术的不断发展，Web 网页具有多源、海量、异构等特点，这对 Web 信息抽取工作提出了巨大的挑战。随着 CSS 等技术的发展与广泛应用，不同网站的网页结构也趋向于复杂化和多样化，导致手工和基于规则构造的包装器技术很难适应不同的网页结构。而 Web 网站的显示风格、页面结构的不断变化，对基于视觉特征和基于模板的信息抽取技术也提出了挑战。

通过研究发现，Web 网页与其对应的 DOM 解析树的文本块密度和标签路径存在潜在联系。具体表现在：

a) Web 网页中正文部分集中且多在一个文本块中，多为长文本，含有少量的超链接文本，文本块密度比较大。

b) Web 网页中噪声部分分散，多为短文本，含有大量的超

收稿日期： 2017-01-13；**修回日期：** 2017-02-24　**基金项目：** 国家重点研发计划资助项目(2016YFB1000901)；国家自然科学基金资助项目(61273297,61229301,61673152)；国家教育部创新团队发展计划资助项目(IRT13059)；国家留学基金资助项目(201506695019)

作者简介： 刘鹏程(1991-)，男，安徽合肥人，硕士研究生，主要研究方向为 Web 数据集成、数据挖掘(lpc_hfut@126.com)；胡骏(1990-)，男，安徽合肥人，博士研究生，主要研究方向为 Web 数据集成、数据挖掘、机器学习；吴共庆(1975-)，男，安徽岳西人，副教授，博士，主要研究方向为 Web 智能、数据挖掘.

链接文本,文本块密度比较小。

c)Web 网页中正文部分标签路径与特征值比较大,噪声部分标签路径与特征值比较小。

根据网页正文集中在一个文本块的特点,抽取该文本块下的正文内容,可以完成抽取任务。虽然文本块密度特征可以解决文本块内短文本难以抽取的问题,但是却无法确定正文块的范围,导致部分文本块漏抽或者部分噪声块被误抽;而标签密度特征能够确定网页中正文块的范围,但由于特征设计缺陷,难以抽取正文块中的短文本。

本文定义了文本块密度特征和标签路径覆盖率特征以及两者融合的新特征,以区分网页的正文和噪声,有效地解决了文本块中短文本难以抽取和噪声块过滤问题。另外,提出了基于文本块密度和标签路径覆盖率的网页正文抽取算法。主要贡献如下:

a)定义了文本块密度特征和标签路径覆盖率特征。

b)设计了基于文本块密度和标签路径覆盖率融合的新特征。

c)基于文本块密度和标签路径覆盖率融合的新特征,设计了在线的网页正文抽取算法 CETD-TPC,实验验证该算法适用于海量、异构、动态变化的网页正文内容的抽取。

1 相关工作

Web 信息抽取(Web information extraction, WIE)最早由 Rahman 等人[4]在 2001 年提出,旨在将半结构化网页中的文本信息抽取出来,并提供更为结构化、语义更为清晰的形式。此后,自动内容抽取评测会议(Automatic Content Extraction, ACE)、文本理解会议(Document Understanding Conference, DUC)等与信息抽取相关的国际学术会议,推动了信息抽取在不同领域、不同语言中应用的发展[5]。

Web 信息抽取的核心是包装器(wrapper),按照包装器的生成方式可以将 Web 信息抽取技术分为手工抽取、半自动抽取和全自动抽取方法。按照抽取原理的不同,可以将 Web 信息抽取技术分为基于包装器、基于模板、基于视觉和基于统计的方法。

基于包装器的 Web 信息抽取技术中,手工构建包装器是一种最为简单、直接的抽取方法。基于手工构建的包装器有 Minerva[6]、W4F[7]和 XWRAP[8]等。这类方法的优点是能够解决特定网站的信息抽取问题,但缺点是包装器的构建工作耗费大量时间,难以推广,自动化程度不高,不同网站都需要重新构建特定的包装器。为了解决手工构建的包装器的缺陷,引用了数据挖掘、机器学习等方法自动分析网页结构、配置包装器规则。李汝君等人[9]利用自然语言处理方法进行实体识别,生成健康领域的信息抽取规则。孙东普等人[10]提出基于条件随机场的抽取方法,这类方法能够应用在同一领域的不同结构的网页中,提高了算法的适用性。

基于模板的 Web 信息抽取技术是假设网页使用相同或者相似的模板构建的,这类方法通过具有相同或者相似模板的网页,训练生成一个通用的模板结构,以此来完成网页的信息抽取工作。Bar-Yossef 等人[11]实现了基于模板的 Web 信息抽取算法。Yi 等人[12]在 DOM 树的基础上,定义了网页节点的信息熵,通过高信息熵值来过滤掉网页中的噪声信息。顾韵华等人[13]将领域本体应用在构建的模板上,生成抽取规则。邵堃等人[14]对网页信息进行领域识别,加载相应领域词库来完善信息抽取的匹配模式。基于模板方法的优点是可以针对特定网页,生成相应的模板进行信息抽取,抽取性能比较好。但是,不同结构的网页需要构建不同的模板,在网页抽取领域需要构建成千上万个模板,而且随着互联网技术的不断革新,网页的结构也在不断变化,已有的模板难以有效长久地工作,为了保证信息抽取的有效性,需要耗费大量人力对模板进行更新。

在 Web 网页中,一般情况下,网页的正文内容在网页的中间位置,噪声内容在网页的两端,其中正文内容的段落长短和背景颜色、字体的格式和颜色,这些视觉特征能够有效地区分网页的正文内容和噪声内容。基于视觉的 Web 信息抽取技术就是利用这些视觉特征对网页进行分块和信息抽取。Cai 等人[15]提出的 VIPS 分块算法,通过启发式规则利用网页中的视觉特征将网页分成不同的块。李伟男等人[16]基于 VIPS 算法,提出了改进的隐马尔可夫模型,实现 Web 信息抽取。Wang 等人[17]引用了一些新的视觉特征,使用机器学习的方法在一个网站进行训练建立抽取模型,在其他网站上也有很好的推广性。基于视觉的 Web 信息抽取的优点是面对许多表现形式单一、代码层次上区别很大的网页时,有很好的抽取性能,但是这种方法需要对网页进行渲染,相对于其他方法,需要占用更多的计算资源。

随着 Web 技术的发展,Web 网页的模板具有多样性,前面描述的 Web 网页信息抽取方法大多数适用于相同结构的网页,难以满足现在对 Web 信息抽取的需求,利用网页内容中的统计特征来进行信息抽取开始被深入研究。樊梦佳等人[18]基于规则和多种统计策略相融合的方法,提出了面向领域术语的信息抽取算法。Weninger 等人[19,20]提出并完善了 CETR 算法,该方法基于 HTML 代码的标签比特征值的大小来进行网页信息抽取。Sun 等人[21]提出了基于网页文本密度和特征的 Web 信息抽取算法 CETD,能够有效地抽取网页正文块。Wu 等人[22]提出了 CEPR 算法,该算法基于文本标签路径比特征及其扩展特征,通过阈值的设立来有效地区分网页中的正文内容和噪声内容。实验结果标明,在大多数情况下,CEPR 算法的性能要优于 CETR 算法。基于 CEPR 的方法虽然能够有效区分正文文本和噪声文本,但是存在正文块中短文本难以抽取的问题。

针对上述问题,本文设计的 CETD-TPC 算法利用 DOM 树节点的文本块密度特征值和标签路径特征,设计正文节点标签路径覆盖率,并提出了一种融合文本块密度和标签路径覆盖率的新特征,在此基础上,基于新特征实现了一种准确、快速的网页正文抽取方法。该方法不依赖于任何 HTML 启发信息,可应用于不同的网页结构中,通过在多个数据集和已有方法进行对比,验证算法具有良好的抽取性能。

2 文本块密度和标签路径覆盖率

2.1 DOM 树

DOM(document object model,文档对象模型)(https://www.w3.org/DOM/)是一种跨平台的且与语言无关的表示和处理 HTML 或 XML 文档的标准协议。DOM 不仅可以结构化显示 HTML 和 XML 文档,同时提供了处理该文档的方法。DOM 将文档解析为一个树型结构,对 HTML 文档进行添加、删除和修改等操作的处理,都是通过对 DOM 树的操作来实现。

2.2　文本块密度

观察图 1 的示例网页(http://www.bbc.com/news/health-38506735)(矩形框内为网页的正文内容)以及相关网页发现,Web 网页的正文内容具有文本长、内容集中和超链接少等特点,而噪声内容则具有文本简短、内容分散和超链接多等特点。

图 1　BBC 网页样例

定义 1　文本块、子树。设 T 为网页解析树,blk(v)是 T 上以节点 v 为根节点的子树,其中 v 为非文本节点。若 blk(v)不为空时,则称子树 blk(v)为文本块或子树。

研究发现可以用文本块下的文本长度、超链接长度与标签个数之间的关系来区分 Web 网页的正文内容和噪声信息。这些特征的描述如下:

a)文本块字符数(content number,CN):文本块所包含的文本字符数。通常情况下,正文文本块下的文本比较集中,文本字符长度会比较大;噪声文本块下的文本比较分散,文本字符长度会比较小。

b)文本块超链接字符数(link content number,LCN):文本块所包含的超链接字符数。正文文本块下的超链接文本比较少,噪声文本块下的超链接文本比较多。

c)文本块标签数(tag number,TN):文本块所包含的标签的个数。正文文本块下多为连续文本,标签个数少;噪声文本块下为分散文本,标签个数多。

d)文本块超链接标签数(link tag number,LTN):文本块所包含的超链接标签的个数。超链接标签下的文本多为噪声信息,正文文本块下含有的超链接标签个数少,噪声文本块下含有的超链接标签个数多。

定义 2　文本块密度(text block density,TBD)。设 v 为网页解析树 T 中的一个节点,blk(v)是以 v 为根节点的文本块。定义 v 的文本块密度 TBD(v)为节点 v 所有子节点为根的文本块中非链接文本字符数与非链接标签数比值之和,即

$$\mathrm{TBD}(v)=\sum_{v_i\in v.\,\mathrm{children}}\frac{\mathrm{CN}_{v_i}-\mathrm{LCN}_{v_i}+1}{\mathrm{TN}_{v_i}-\mathrm{LTN}_{v_i}+1} \tag{1}$$

其中:节点 v_i 为节点 v 的孩子节点;CN_{vi}为节点 v_i 的文本块字符数;LCN_{vi}为节点 v_i 的文本块超链接字符数;TN_{vi}为节点 v_i 的文本块标签数;LTN_{vi}为节点 v_i 的文本块超链接标签数;CN_{vi}-LCN_{vi}为节点 v_i 的文本块中非超链接文本数;TN_{vi}-LTN_{vi}为节点 v_i 的文本块中非超链标签数,为了避免分母为 0,在计算时分母加 1,保证公式的计算有意义。

图 2 为示例网页对应的 TBD 特征值的直方图。其中横坐标是网页从根节点遍历的文本块序列,纵坐标为相对应的文本块密度 TBD 值。观察可以发现 TBD 有效地将网页中的正文文本块和噪声文本块区分开。其中,TBD 特征值最大的文本块为网页的最佳正文文本块,即该文本块下的内容包含尽可能多的网页正文内容,且包含噪声信息比较少,该文本块下的文本内容通常是真正需要抽取的正文或者部分正文。

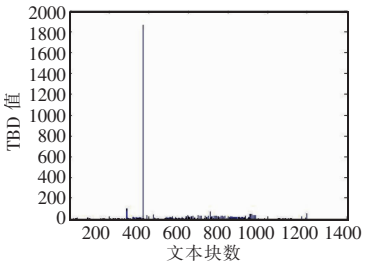

图 2　BBC 样例网页的 TBD 直方图

2.3　标签路径及标签路径覆盖率

网页的文本块密度特征解决了文本块内短文本的抽取问题,但是却难以确定抽取正文块的范围,保证网页信息抽取质量。如图 3、4 为 Hadoop 网站示例网页(http://hadoop.apache.org/docs/r2.6.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/HadoopArchives.html/)和对应的 TBD 直方图,其中 blk_1 是手工标记要抽取的网页正文文本块,blk_2 是 TBD 最大值抽取的文本块。此时根据最大文本块密度 TBD 值,仅仅抽取了网页中的部分正文。在这类网页中,仅仅靠文本块密度特征很难保证网页信息抽取的质量。

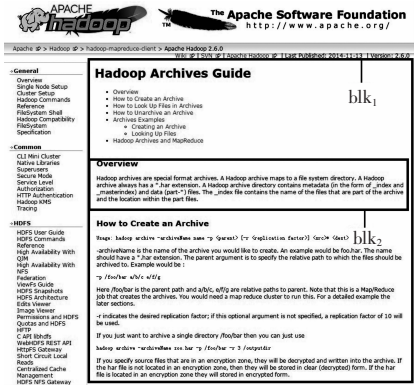

图 3　Hadoop 网站示例网页

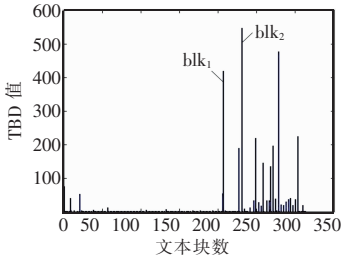

图 4　图 3 网页对应的 TBD 直方图

定义 3　标签路径。设 v_0 为网页解析树 T 中的根节点。$v_0v_1v_2\cdots v_k$ 为解析树 T 从 v_0 到达 v_k 的节点序列,tag(v)为节点 v 对应的标签。其中:$\mathrm{parent}(v_i)=v_{i-1}(1\leqslant i\leqslant k)$,$v_k=v$。节点 v 的标签路径 path(v)为根节点 root 到达 v 经过的节点标签序列,即

$$\mathrm{path}(v)=\mathrm{tag}(v_0)\mathrm{tag}(v_1)\cdots\mathrm{tag}(v_k) \tag{2}$$

记 v 为给定解析树 T 的节点,$c(v)$为节点 v 上的文本内容,length($c(v)$)为节点 v 上的文本长度,nodes(p)为路径 p 可

到达的节点集合。

定义 4 文本标签路径比。设 p 为网页解析树 T 的一个标签路径，定义 p 的文本标签路径比 TPR(p)为路径 p 可达到的文本节点的文本字符之和与 p 可到达的文本节点数的比值，即

$$\mathrm{TPR}(p)=\frac{\sum_{v\in \mathrm{nodes}(p)}\mathrm{length}(c(v))}{\mathrm{size}(\mathrm{nodes}(p))} \tag{3}$$

通常情况下，Web 网页的正文内容聚集，正文内容的标签路径变化小，正文内容的标签路径集合比较小；而噪声内容分散，噪声内容的标签路径变化大，噪声内容的标签路径集合比较大。Web 正文抽取转换为抽取文本块的标签路径集合尽可能覆盖正文内容的标签路径集合，尽可能不覆盖噪声文本的标签路径集合。

定义 5 标签路径覆盖。设 blk(v)为网页解析树 T 中的文本块，blk(v)中节点的标签路径集为 $P=\{p_1,p_2,\cdots,p_{n-1},p_n\}$，则称文本块 blk($v$)覆盖标签路径 $p(p\in P)$。

定义 6 标签路径覆盖率。设 blk(v)为网页解析树 T 中的文本块，blk(v)中节点的标签路径集为 P_1。给定一个已知标签路径集 P_2，则 blk(v)对标签路径集 P_2 的覆盖率 TPC(blk(v),P_2)为 P_1 和 P_2 标签路径交集可到达的节点集合与标签路径集 P_2 可到达的节点集合大小的比值，即

$$\mathrm{TPC}(\mathrm{blk}(v),P_2)=\frac{|\mathrm{containNodes}(P_1\cap P_2)|}{|\mathrm{containNodes}(P_2)|} \tag{4}$$

其中：containNodes(P)为标签路径集 P 能到达的节点集合。

定义 7 正文节点标签路径覆盖率(content nodes tag path coverage)。设 blk(v)为网页解析树 T 中的文本块，记 cntPaths(T)=$\{p_1,p_2,p_3,\cdots,p_m\}$为整个网页 T 中正文节点的标签路径集合(其中正文节点为网页正文内容所在的节点，即要抽取的节点)，则 blk(v)对正文节点标签路径集的覆盖率 CTPC(blk)为

$$\mathrm{CTPC}(\mathrm{blk},\mathrm{cntPaths}(T))=\mathrm{TPC}(\mathrm{blk},\mathrm{cntPaths}(T)) \tag{5}$$

正文节点标签覆盖率高时，该文本块包含的正文内容多，网页正文内容集中在该文本块中；覆盖率低时，该文本块包含的正文内容少，网页正文内容分散在其他文本块中。

3 基于文本块密度和标签路径覆盖率的网页正文抽取方法

文本块密度特征是将文本块里的文本内容抽取，不存在短文本遗漏问题，但从图 3 示例网页可以发现，该特征存在难以确定文本块范围的缺点。而标签路径特征可以有效地区分网页中正文节点和噪声节点，确定网页文本块的正文节点标签路径覆盖率，但通常情况下文本块越大，正文节点的标签路径覆盖率越大，标签路径覆盖率无法衡量不同文本块的好坏。

为了进一步观察 TBD 特征和 CTPC 特征之间的联系，图 5 为图 3 示例网页的 TBD 特征和 CTPC 特征之间联系的二维图。其中横坐标为 CTPC 特征值，纵坐标为同一文本块的 TBD 特征值；“o”为抽取该网页的最佳文本块，“x”为抽取该网页的其他文本块。观察该图可以发现，标注为“o”的最佳文本块的 TBD 特征值和 CTPC 特征值都不是最大，难以通过选取最大 TBD 特征值或最大 CTPC 特征值的方式抽取“o”所标记的文本块。

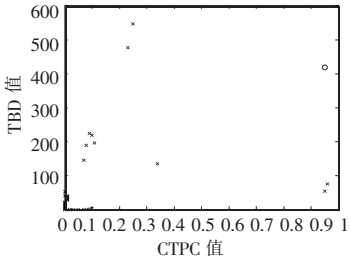

图 5 图 3 网页 CTPC 特征和 TBD 特征之间的联系

进一步观察发现，“o”所标记的最佳文本块的 TBD 特征值和 CTPC 特征都较高，虽然仅仅靠 TBD 特征或 CTPC 特征难以取得网页中的最佳文本块，但是利用 TBD 特征和 CTPC 特征的乘积能够有效地选择网页的最佳文本块，提高网页信息抽取的性能。因此，为了充分利用标签路径特征和文本块密度特征的优势，弥补各自的不足，将两者融合成一个性能更好的新特征 TDTPC，确定网页的最佳文本块。计算如式(6)所示。

$$\mathrm{TDTPC}(\mathrm{blk})=\mathrm{TBD}(\mathrm{blk})\times\mathrm{CTPC}(\mathrm{blk}) \tag{6}$$

图 6 给出了图 3 网页的 TDTPC 直方图(最佳文本块手工标记)，通过对比图 4 的 TBD 直方图可以发现，TDTPC 避免了单 TBD 特征选择的子树块不是最佳的问题，提高了网页信息抽取的质量。

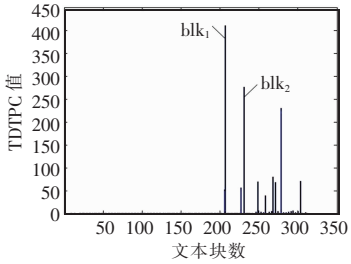

图 6 图 3 网页的 TDTPC 直方图

算法 CETD-TPC
输入：HTML 网页 *wp*。
输出：网页正文 content。
1 将网页 *wp* 解析为树 T；
2 TBD,TPR←calculateFeatures(T)；
3 for v in T do
4 if(v.TPR > τ) then
5 cntNodes.add(v)
6 cntPaths←getPath(cntNodes)；
7 for blk in T do
8 TDTPC(blk)←TBD(blk) * CTPC(blk,cntPaths)；
9 cnt$T$←argMax(blk.CETD-PRC)；
10 content←cnt$T$.text；
11 output content.

以上给出了基于文本块密度和标签路径覆盖的正文抽取算法 CETD-TPC。该算法第 1 步先将一个网页 *wp* 解析为树 T；第 2 步调用 calculateFeatures 函数，遍历解析树 T，计算文本块密度 TBD 特征值、文本标签路径比 TPR 特征值；第 3 ~ 5 步将 TPR 值大于阈值 τ 的节点加进正文节点集合 cntNodes 中(其中阈值 τ 采用文献[22]CEPR 算法的经验阈值)；第 6 步计算网页正文节点的标签路径集合 cntPaths；第 7、8 步根据式(6)计算文本块 blk 的 TDTPC 特征值；第 9、10 步获取最大 TDTPC 值的文本块 blk 为该网页的正文文本块，抽取该文本块内的内容；最后第 11 步，输出网页的正文内容。

4 实验

本章主要介绍实验所用的数据集和实验评价标准，以及对

比算法 CETD、CEPR、CETR 在数据集上的实验结果,并对实验结果进行分析。

实验的硬件环境为:Intel® Core™ i5-3470S CPU @2.90 GHz 2.90 GHz,8 GB RAM;

软件环境为:操作系统 Windows 7 旗舰版 x64;开发平台 JDK 1.8。

4.1 数据集与评价标准

本文实验中使用的数据集主要分为以下两类:

a)CleanEval 比赛数据集[23]。CleanEval 比赛数据集是为语言的发展、语言处理技术的研究和开发提供互联网上的数据集。CleanEval 数据集中有 971 个中文网页和 932 个英文网页,这些网页来自不同的网站,网页的结构与设计风格也是各种各样的。

b)News 数据集。该数据集分别来自 13 个不同的新闻网站的新闻网页。其中 BBC、Freep、NY Post、Reuters、Nytimes、Suntimes、Techweb、Tribune 也是对比实验 CETR 和 CEPR 所使用的数据集。其中原始数据集中每个网站只含有 50 个网页,数据集太小,单个网页实验的好坏影响最后的整个实验的结果。为了更好地验证算法的通用性和有效性,本文在原有数据集的基础上,每类网站增加了 250 个网页。此外,本文还增加了一个英文数据集(Yahoo!)和四个中文数据集(新浪、网易、人民网、新华网)。

本文采用精度 P(precision)、召回率 R(recall)和 F 值作为评价 Web 网页正文抽取的性能指标。抽取的结果和手工标注的结果均为字符串的集合。

$$P=\frac{|S_e\cap S_m|}{|S_e|},R=\frac{|S_e\cap S_m|}{|S_m|},F=\frac{2\times P\times R}{P+R}$$

其中:S_e 为 Web 网页正文抽取结果集合;S_m 为手工标注结果集合;$S_e\cap S_m$ 为抽取结果中应该被抽取的正文内容;精度 P 是衡量抽取结果中被正确抽取正文内容的比例;召回率 R 是衡量应该抽取结果中被正确抽取正文内容的比例;F 值则是衡量抽取性能的一个综合指标。

4.2 实验结果与分析

本文基于文本块密度和标签路径覆盖率的网页正文抽取方法记为 CETD-TPC。表 1 给出了 CETD-TPC 算法在不同数据集上的抽取性能。同时也给出了 CETR、CETD 和 CEPR 算法的实验结果,具体的算法实验对比分析在 4.3 节中进行详细说明。

从表 1 的实验结果发现在大部分情况下,CETD-TPC 具有良好的抽取性能,尤其是在 Nytimes、Reuters、新浪、人民网和新华网的数据集中,抽取的 F 值均在 94% 以上。但是在 Freep 网站数据集上,CETD-TPC 的抽取结果却比较差,其中 Freep 数据集中的精度比召回率低很多。通过观察该网站的网页结构可以发现,该类网页的噪声块和正文块无法再细分,这部分正文块下含有导航链接以及大量长文本的相关文章推荐链接等噪声信息,导致抽取结果中含有许多噪声信息,所以抽取结果中的召回率很高,但是精度却一般,导致最后的 F 值比较低。实验结果表明:在绝大部分数据集上,基于文本块密度和标签路径覆盖率的方法能够有效地区分 Web 网页的正文和噪声,具有良好的抽取性能。

4.3 对比算法

本文使用网页解析器 Jsoup 在 Java 环境下实现了本文的算法。同时为了验证该算法的抽取性能,对比了现在主流的 Web 信息抽取算法——CETR、CETD 和 CEPR 算法。其中 CETR 和 CEPR 算法采用的是笔者提供的源代码,阈值的设置均为文章中描述的经验阈值。CETD 算法源代码是用 QT C++ 实现的,因没有处理好自动编码识别,以及选用的解析器存在一些问题,在部分数据集上不能获得结果。为此对照 CETD 的算法描述,实现了 Java-Jsoup 版本。这些算法都是在线抽取 Web 网页正文,不需要特定地对网页进行人工标注训练模本。

表 1　对比实验结果　　/%

数据集	CETD			CEPR			CETR			CETD-TPC		
	Ave$P$	Ave$R$	Ave$F$	Ave$P$	Ave$R$	Ave$F$	Ave$P$	Ave$R$	Ave$F$	Ave$P$	Ave$R$	Ave$F$
CleanEval-ZH	85.09	90.19	87.57	92.14	70.93	80.16	89.42	87.20	**88.30**	87.67	87.94	87.80
CleanEval-EN	90.04	93.66	**91.82**	91.13	64.97	75.86	79.60	87.48	83.36	91.90	90.48	91.19
BBC	74.63	98.47	84.91	95.96	67.70	79.39	42.68	91.40	58.19	88.52	92.12	**90.28**
Freep	60.65	100.00	75.50	83.30	86.12	84.69	57.38	90.93	70.36	77.58	99.45	**87.17**
NY Post	74.29	95.39	83.53	90.11	70.95	79.39	70.82	97.95	82.20	93.40	89.91	**91.62**
Nytimes	97.24	98.03	**97.63**	98.53	77.55	86.79	60.29	97.68	74.56	98.30	90.56	94.27
Reuters	63.61	99.56	77.63	93.81	79.16	85.86	84.99	95.24	89.83	96.59	96.53	**96.56**
Suntimes	81.51	97.56	88.81	93.86	84.93	89.17	88.98	93.40	**91.14**	84.96	91.66	88.19
Techweb	63.17	99.45	77.27	90.08	81.22	85.42	60.11	92.17	72.76	82.76	91.57	**86.94**
Tribune	90.95	96.97	**93.87**	92.37	78.06	84.61	58.48	92.75	71.73	93.00	92.55	92.78
Yahoo!	76.66	97.75	85.93	85.93	68.82	76.43	72.67	94.22	82.06	96.38	87.44	**91.69**
新浪	81.98	98.47	89.47	98.86	77.41	86.83	59.09	98.94	73.99	96.09	96.46	**96.28**
人民网	69.79	99.91	82.17	91.08	89.78	90.43	77.04	97.89	86.23	93.61	94.84	**94.22**
网易	37.75	89.93	53.18	91.23	72.50	80.79	24.40	88.76	38.28	90.60	87.71	**89.13**
新华网	83.29	99.25	90.58	96.88	68.03	79.93	72.10	98.68	83.32	96.48	94.29	**95.37**
平均值	75.38	96.97	83.99	92.35	75.88	83.05	66.54	93.65	77.80	91.19	92.23	**91.57**

表 1 给出了 CETR、CETD 和 CEPR 的抽取实验结果。可以发现,CETR 和 CETD 算法在大部分数据集上都能取得较好的实验结果,精确度和 F 值低于 CETD-TPC 算法,召回率高于 CETD-TPC 算法,说明 CETR 和 CETD 算法倾向于将网页正文内容抽取完整,而无法保证抽取内容的准确率,导致抽取结果存在大量的噪声信息。

CEPR 算法的实验结果的精确度大于 CETD-TPC 算法,召回率和 F 值要小于 CETD-TPC 算法,说明 CEPR 方法倾向于保

证网页内容抽取的准确度,而不保证抽取正文内容的完整,导致网页正文内容抽取不全。

CETD-TPC 算法在大部分数据集的抽取性能要优于 CETR、CETD 和 CEPR,尤其是在数据集 NY Post、Yahoo! 和网易数据集上的平均抽取性能比排名第二的实验结果方法分别要高 8.09%、6.81% 和 8.34%。在所有的数据集上,平均抽取性能要高 7.58%。

在时间性能方面,设网页解析树中有 m 个节点,其中有 n 个文本节点($n<m$),非文本节点为 $m-n$。从本文中计算文本块密度、标签路径特征、正文节点标签路径、融合特征方法分析可知,这些的最坏时间复杂度分别为:$O(m-n)$、$O(n)$、$O(n)$、$O(m-n)$,则 CETD-TPC 算法的最坏时间复杂度为 $O((m-n)\times(O(m-n)+O(n)+O(n)+O(m-n)))=O(m^2)$。通过研究分析可知 CETR、CETD 和 CEPR 算法的时间复杂度为 $O(mn)$。在 CleanEval 比赛数据集和 News 数据集上的实验结果表明,CETD-TPC 方法平均抽取一个网页的时间为 166.36 ms,CETR、CETD 和 CEPR 算法平均抽取一个网页的时间分别为 92.02 ms、97.57 ms 和 114.51 ms,实验验证了 CETD-TPC 与 CETR、CETD 和 CEPR 相比,在时间性能方面差别不大,均能够有效地完成实时在线抽取任务。

综上所述可知,CETD-TPC 是一种高精度的实时在线抽取算法,抽取性能优于 CETR、CETD 和 CEPR 算法。

5 结束语

本文根据 Web 网页中的正文内容分布相对集中、格式单一、超链接文本信息少等特点,文本块密度特征难以确定正文块的范围、标签路径特征难以抽取短文本的问题,设计了文本块密度和标签路径覆盖融合的新特征,实现了基于新特征的网页正文抽取算法 CETD-TPC。在多个数据集的实验结果表明,CETD-TPC 是一种无须训练、通用的在线抽取算法,能针对多源、海量、异构的网页实现快速、准确和简便地完成抽取任务,在抽取精度、召回率和 F 值等方面均优于 CETR、CETD、CEPR 算法。

然而,本文算法还有改进空间。首先,在计算文本块密度时,本文仅考虑了文本、超链接特征,可能存在其他特征信息,如标点符号、句数等特征能够提高抽取性能;其次,本文仅仅使用了标签路径覆盖率来判断节点是否为正文节点,是否可以引入网页中其他特征、提高抽取算法的性能是进一步研究的方向。

参考文献:

[1] Mary M. 2016 Internet trends report[EB/OL]. [2016-07-01]. http://www.kpcb.com/blog/2016-internet-trends-report.

[2] CNNIC. 中国互联网络发展状况统计报告[R]. 北京:中国互联网中心,2016.

[3] Gibson D, Punera K, Tomkins A. The volume and evolution of Web page templates[C]//Proc of the 14th International Conference on World Wide Web. New York: ACM Press, 2005: 830-839.

[4] Rahman A F R, Alam H, Hartono R. Content extraction from HTML documents[C]//Proc of the 1st International Workshop on Web Document Analysis. Berlin: Springer, 2001: 1-4.

[5] 郭喜跃,何婷婷. 信息抽取研究综述[J]. 计算机科学,2015,42(2):14-17,38.

[6] Crescenzi V, Mecca G. Grammars have exceptions[J]. Information Systems, 1998, 23(8): 539-565.

[7] Sahuguet A, Azavant F. Building intelligent Web applications using lightweight wrappers[J]. Data & Knowledge Engineering, 2001, 36(3): 283-316.

[8] Liu Ling, Pu C, Han Wei. XWRAP: an XML-enabled wrapper construction system for Web information sources[C]//Proc of the 16th International Conference on Data Engineering. Piscataway, NJ: IEEE Press, 2000: 611-621.

[9] 李汝君,张俊,张晓民,等. 健康领域 Web 信息抽取[J]. 计算机应用,2016,36(1):163-170.

[10] 孙东普,朱鸣华,林鸿飞. 中文专利属性值对抽取技术及应用[J]. 计算机工程与科学,2016,38(4):800-806.

[11] Bar-Yossef Z, Rajagopalan S. Template detection via data mining and its applications[C]//Proc of the 11th International Conference on World Wide Web. New York: ACM Press, 2002: 580-591.

[12] Yi Lan, Liu Bing, Li Xiaoli. Eliminating noisy information in Web pages for data mining[C]//Proc of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2003: 296-305.

[13] 顾韵华,高原,高宝,等. 基于模板和领域本体的 Deep Web 信息抽取研究[J]. 计算机工程与设计,2014,35(1):327-332.

[14] 邵堃,杨春磊,钱立宾,等. 基于模式匹配的结构化信息抽取[J]. 模式识别与人工智能,2014,27(8):758-768.

[15] Cai Deng, Yu Shipeng, Wen Jirong, *et al*. VIPS: a vision-based page segmentation algorithm, MSR-TR-2003-79 [R]. Redmond: Microsoft, 2003.

[16] 李伟男,李书琴,景旭,等. 基于模拟退火算法和二阶 HMM 的 Web 信息抽取[J]. 计算机工程与设计,2014,35(4):1264-1268.

[17] Wang Junfeng, Chen Chun, Wang Can, *et al*. Can we learn a template-independent wrapper for news article extraction from a single training site? [C]//Proc of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2009: 1345-1354.

[18] 樊梦佳,段东圣,杜翠兰,等. 统计与规则相融合的领域术语抽取算法[J]. 计算机应用研究,2016,33(8):2282-2285,2306.

[19] Weninger T, Hsu W H. Text extraction from the Web via text-to-tag ratio[C]//Proc of the 19th International Workshop on Database and Expert Systems Applications. Washington DC: IEEE Computer Society, 2008: 23-28.

[20] Weninger T, Hsu W H, Han Jiawei. CETR: content extraction via tag ratios[C]//Proc of the 19th International Conference on World Wide Web. New York: ACM Press, 2010: 971-980.

[21] Sun Fei, Song Dandan, Liao Lejian. Dom based content extraction via text density[C]//Proc of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2011: 245-254.

[22] Wu Gongqing, Li Li, Hu Xuegang, *et al*. Web news extraction via path ratios[C]//Proc of the 22nd ACM International Conference on Information and Knowledge Management. New York: ACM Press, 2013: 2059-2068.

[23] Baroni M, Chantree F, Kilgarriff A, *et al*. CleanEval: a competition for cleaning Web pages[C]//Proc of International Conference on Language Resources and Evaluation. Marrakech: LREC, 2008.