

改进的增量词集频率主题词提取算法

刘兴林*

(五邑大学 计算机学院, 广东 江门 529020)

(*通信作者电子邮箱 jmxlliu@163.com)

摘要:为了解决基于增量词集频率的主题词提取算法不能提取合成词的问题,在原算法的基础上增加了文本预处理环节,即合成词识别。采用基于词性探测和词共有向图算法识别文本中的合成词,并对分词结果进行修正。生成候选主题词集时,考察每个词的出现位置,根据不同的出现位置赋予不同的权重;然后累加获得同一个词的总权重,并按权重从高到低生成候选主题词集。提取主题词时逐个考察候选主题词集中的每一个候选主题词,计算其对主题词集权重的增量,若增量小于给定阈值,则主题词提取算法结束;否则将该候选主题词加入主题词集。实验结果表明,该算法取得了较好的效果,所获得的主题词能更贴切地反映文档的主题内容,主题词满意度比原算法提高了5个百分点。

关键词:主题词;词共有向图;词位置权重;词集频率;知识获取

中图分类号:TP301.6;TP391.1 **文献标志码:**A

Improved algorithm of thematic term extraction based on increment term-set frequency from Chinese document

LIU Xinglin*

(School of Computer Science, Wuyi University, Jiangmen Guangdong 529020, China)

Abstract: In order to solve the problem that the thematic term extraction algorithm based on incremental term-set frequency cannot extract compound-words, this paper added text preprocessing, compound-word recognition, to the original algorithm. Compound-word recognition was based on part-of-speech detection and word co-occurrence directed graph, and corrected the results of segmentation. When generating thematic term candidate set, the position of each word was examined and determined its weight. And then, the total weight of the same word was accumulated, and a candidate set of thematic terms was generated by the weight from high to low. When this algorithm got a term from thematic term candidate set, the increment frequency was calculated. If the increment was less than a given threshold, the algorithm stopped; otherwise, the thematic term candidate was added into thematic term set. The experimental results show this algorithm achieves sound effects, the thematic terms acquired by this algorithm can more aptly reflect the main contents of the article, and the satisfaction of thematic term increased 5% than the original algorithm.

Key words: thematic term; word co-occurrence directed graph; word position weigh; term-set frequency; knowledge acquisition

0 引言

主题词能够帮助人们快速地了解整篇文档的主要内容,也可用于多方面的应用,如文摘、索引、标记、分类、聚类 and 检索等。目前主题词的提取方法主要有3类:基于词典、基于规则和基于统计的提取方法。3类方法各有优缺点,基于统计的提取方法是目前使用最为广泛的,也是研究得最为深入的主题词提取方法^[1-2]。

赵鹏等^[1]提出一种基于复杂网络特征的中文文档关键词抽取算法,该算法根据文档语言网络中单词节点的复杂网络特征值进行关键词抽取,利用复杂网络“小世界”特征,解决了在网络不连通的情况下,或者当网络中的某个节点及其连边删除后不再连通的情况下,则无法获得网络的平均最短路径,从而无法抽取关键词这个问题,而且时间复杂度也大大

降低了。

耿焕同等^[2]提出了一种基于词共现图的文档主题词抽取算法,以词频统计方法为基础,利用在词共现图形成的主题信息以及不同主题间的连接特征信息自动地提取文档中的主题词,从而找出一些非高频词且又对主题贡献大的词。

刘菲等^[3]利用关联规则挖掘文本主题词,有别于统计方法,通过其他文档的关键词对当前文档关键词起补充和引导作用,所提取的主题词能增加用户对文章的理解程度。有些文献根据词语在文档中语义联系^[4]和语义关系^[5]将文档表示成词汇链形式,并在此基础上抽取关键词。对中文新闻网页和学术期刊文献两种语料中的实验进行实验,该方法提取到的关键词质量明显得到提高。

石晶等^[6]根据文本词汇的概率分布,通过香农信息抽取体现主题的主题词,而 Anette 等^[7]则通过建立文档领域知识

辞典,进而提取关键词,效果提高了近30%。

Zhang等^[8]提出利用支持向量机(Support Vector Machine, SVM)方法来抽取文档关键词,实验结果评论关键词抽取准确率达到67.43%。

黄先珍等^[9]针对当前向量空间模型中特征项的选取与权重的计算分开导致特征项区分度下降的问题,提出一种基于统计与规则的关键词抽取方法,利用句法规则提取出基本短语,以取代词袋模型中的词,考虑特征项位置、分布及语法角色等信息,综合加权计算特征项权重以抽取关键词。刘晓明等^[10]则提出基于动态点阵匹配以识别二阶关键词,解决了噪声较大情况下识别率降低的问题。

上述文献所提出的主题词提取方法各有其优势和不足之处,存在的主要问题有两点:1)不能抽取到合成词;2)当候选主题词出现频率较平均时无法精确抽取主题词。本文基于统计,对文档进行预处理,识别文档中的合成词并修正分词结果,使得合成词有机会被抽取为主题词,同时从分析词位置角度出发,结合增量词集频率进行主题词提取,使得当候选主题词出现频率较平均时,仍然能提取到最合适的主题词。

1 算法设计与分析

1.1 合成词识别^[11]

文档中的词语分为原子词和合成词,合成词由多个原子词构成,且表达了一个完整的概念。当前的分词系统并未将这些合成词收录进词典,因而未能识别。

本文认为一个词串是合成词,必须满足以下3个条件:

- 1) 该词串由句子中 $L(L \geq 2)$ 个无间隔的原子词构成;
- 2) 该词串在文档中多次出现;
- 3) 在该词串的前面或后面加上其他原子词所形成的新词串出现的次数明显减少。

结合词性探测的方法扫描文档,进而从文档中获得词串,使用一个三元组记录词串出现的句子编号、句内起始位置和结束位置。

词共有向图标记为: $G: \langle V, E \rangle$, 其中 V 指文档中的原子词集, E 是由词对构成的集合, 边的起点对应词对的首词, 边的终点对应词对的末词。有向边的权是一个集合, 是词对在文档中共现的位置集, 每个元素是一个三元组 $\langle sno, start, end \rangle$, 标识词对所在的句子编号, 以及在句子中的起始和结束位置。

V 的元素用 $v_1, v_2, \dots, v_{|V|}$ 表示, e_{ij} 表示以 v_i 为起点, v_j 为终点的有向边, s_{ij} 表示边 e_{ij} 上的集合, $w_{ij} = |s_{ij}|$ 表示边 e_{ij} 的权重值, $p\langle v_i, \dots, v_j \rangle$ 表示由顶点词串 v_i, \dots, v_j 所对应的路径, $ps\langle v_i, \dots, v_j \rangle$ 表示对应路径上所有边的集合的交集, 也称为 $p\langle v_i, \dots, v_j \rangle$ 的集合, $len\langle v_i, \dots, v_j \rangle$ 表示路径 $p\langle v_i, \dots, v_j \rangle$ 的长度, $weight\langle v_i, \dots, v_j \rangle = |ps\langle v_i, \dots, v_j \rangle|$ 表示路径 $p\langle v_i, \dots, v_j \rangle$ 的权重值。定义一个类交集运算 \cap^* 用于词共有向图上边的集合交集运算, 如下所示。

$$X \cap^* Y = \{ \langle sno, start, end \rangle \mid \langle sno, start, mid \rangle \in X, \langle sno, mid, end \rangle \in Y \} \quad (1)$$

显然, $X \cap^* Y \neq Y \cap^* X$, 因此在进行类交集运算时, 必须保证左操作数的边(或路径)尾顶点是右操作数的边(或路径)头顶点。

如上所述, 在词共有向图中, 当路径 $p\langle v_i, \dots, v_j \rangle$ 所对

应的词串满足上述的3个条件, 则这个词串是合成词。

借鉴 Bellman-Ford 算法思想, 本文设计了求解词共有向图中多源点路径长度最长($\geq L$)并且权重值满足给定条件($\geq T$)的路径算法, 用于合成词识别。由于是长度优先, 所以总是将更长的合成词先识别出来, 这样的好处在于, 当一个合成词包含了另外一个合成词, 算法能先后识别出这两个合成词。若是权重值优先, 则只能识别出长度较短的合成词。

识别出的合成词词性标注格式为: 词性 + cw + Num, 其中 cw 为 Compound-word 的首字母, 表示该词为合成词, Num 为合成词的长度(即合成词中包含的原子词的个数), 如“知识经济/new2”“人文社会科学/new3”等, 同时对原文分词结果进行修正, 即用合成词替换原文对应的词串。

1.2 生成候选主题词集

候选主题词集的生成基于以下两个假设。

假设1 一个词在文档中出现的次数越多, 它成为主题词的概率就越大。

假设2 同一个词在文档的不同位置出现, 对该词是否成为主题词的影响是不一样的。

对于假设1, 这是显而易见的。一个词出现次数越多, 成为主题词的可能性也就越大, 但是当一篇文档的各个词的出现次数相对平均时, 显然各个词成为主题词的概率也是基本一致的, 那么如何生成候选主题词集呢? 针对这个问题, 本文提出了假设2, 即根据词出现的位置不同而赋予不同的权重。根据汉语行文习惯, 本文将词在文档的出现位置分为三类: 段序、句序、词序, 下面给出相关的定义及其取值, 取值如表1所示。

定义1 段序(po)表示词出现在文档的不同段落, 段序 = {首段, 末段, 其他}。

定义2 句序(so)表示词出现在段落中的不同句子, 句序 = {首句, 末句, 其他}。

定义3 词序(wo)表示词在句子里出现的顺序, 词序 = {首词, 末词, 其他}。

表1 段序、句序、词序值

| 位置 | 值 | 位置 | 值 | 位置 | 值 | 位置 | 值 |
|----|----|----|----|----|---|----|---|
| 首段 | 64 | 首词 | 16 | 末段 | 4 | 末词 | 1 |
| 首句 | 32 | 其他 | 8 | 末句 | 2 | | |

依上述分析, 一个词可能出现的位置共有 $| \text{段序} | \times | \text{句序} | \times | \text{词序} | = 27$ 种, 针对这27种情况, 分别给出不同的位置值和权重, 如表2所示。

显然, 一个词出现在文档中重要的位置, 它的权重也应该越大, 表1~2的位置值和权重的设置能较好地体现这个思想。

考虑到进行合成词识别后, 合成词与原子词的权重计算应有所区别, 因此词 t 的位置值 pv_t 的计算公式如下所示:

$$pv_t = (po + so + wo) \cdot \sqrt{Num} \quad (2)$$

其中 Num 为合成词的长度, 对于非合成词 $Num = 1$ 。

词 t 单次出现的权重 w_{ti} 计算公式如下所示:

$$w_{ti} = pv_{ti} / \sum_{i=1}^{27} pv_i \quad (3)$$

其中 $\sum_{i=1}^{27} pv_i$ 指27种位置值的总和。

词 t 的总权重 w_t 计算公式如下所示:

$$w_t = \sum_{i=1}^{|d|} t_{wi} \quad (4)$$

其中 $|t|$ 指词 t 的出现次数。

由此可得到一个按总权重从高至低排序的候选主题词集,记为 S_c 。考虑到候选主题词集中可能出现同义词的现象,则使用哈尔滨工业大学信息检索研究室同义词词林扩展版进行同义词合并。

表2 位置值和权重表

| 首段 | 首句 | 首词 | 其他 | 末段 | 末句 | 末词 | 位置值 | 权重 |
|----|----|----|----|----|----|----|------|--------|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 112 | 0.0870 |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 104 | 0.0808 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 97 | 0.0754 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 88 | 0.0684 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 82 | 0.0637 |
| 1 | 0 | 0 | 2 | 0 | 0 | 0 | 80 | 0.0622 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 74 | 0.0575 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 73 | 0.0567 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 67 | 0.0521 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 56 | 0.0435 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 52 | 0.0404 |
| 0 | 1 | 0 | 2 | 0 | 0 | 0 | 48 | 0.0373 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 44 | 0.0342 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 41 | 0.0319 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 37 | 0.0287 |
| 0 | 0 | 1 | 2 | 0 | 0 | 0 | 32 | 0.0249 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 28 | 0.0218 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 26 | 0.0202 |
| 0 | 0 | 0 | 3 | 0 | 0 | 0 | 24 | 0.0186 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 22 | 0.0171 |
| 0 | 0 | 0 | 2 | 1 | 0 | 0 | 20 | 0.0155 |
| 0 | 0 | 0 | 2 | 0 | 1 | 0 | 18 | 0.0140 |
| 0 | 0 | 0 | 2 | 0 | 0 | 1 | 17 | 0.0132 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 14 | 0.0109 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 13 | 0.0101 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 11 | 0.0085 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 7 | 0.0054 |
| 合计 | | | | | | | 1287 | 1.0000 |

1.3 提取主题词

对候选主题词集 S_c , 执行以下步骤:

- 1) 初始化阈值 φ , 将第1个词加入主题词集 S_t 。
- 2) 从第 $i (i \geq 2)$ 个词 t_i 开始, 计算 t_i 加入主题词集后对主题词集频率的增量 Δf_{ti} , 以及 t_i 在候选主题词集的频率 f_{ci} 。
- 3) 计算 $\Delta f = \Delta f_{ti} \cdot \alpha + f_{ci} \cdot \beta$ 。
- 4) 如果 $\Delta f > \varphi$, 则将该词加入主题词集, $i = i + 1$, 回到第2)步; 否则算法结束。

算法中 Δf_{ti} 的计算公式如下所示:

$$\Delta f_{ti} = \frac{|t_i|}{\sum_{j=1}^{|S_d|} t_j + |t_i|} \quad (5)$$

其中 $|t_i|$ 表示 t_i 的出现次数, $|t_j|$ 表示主题词集中各词的出现次数。

f_{ci} 的计算公式如下所示:

$$f_{ci} = \frac{|t_i|}{\sum_{j=1}^{|S_d|} |c_j|} \quad (6)$$

其中 $|c_j|$ 表示候选主题词中各词的出现次数。

算法中各参数 α, β 和阈值 φ 的初始值经训练后确定。

1.4 算法分析

合成词的识别解决了分词系统将一个具有完整意义的词切分成多个原子词,使主题词的提取不能识别合成词的问题得以解决。由于合成词的标注与其他词的标注有区别,因此在使用式(2)计算权重时,对于合成词进行了加权处理,使合成词权重的计算更加合理。

在已有的主题词提取算法中,大多数仅考察各词在文档中的出现频率,并将排在前 N 个词作为主题词,因此提取的主题词数是固定的。然而表达一篇文档的主要思想所需要的主题词个数是不确定的,有时需要增加一个词,有时需要减少一个词,才能更贴切地反映文档本意。本文算法提取到的主题词数量是动态变化的,而且由于词集频率增量的设置,使得文档所提取到的主题词数量基本在一个固定范围内。

与大多数其他算法不一样,本文算法不但计算单个词 t_i 的频率,而且计算 t_i 在候选主题词集的频率 f_{ci} , 以及 t_i 加入主题词集后对主题词集频率的增量 Δf_{ti} , 以加权平均的计算方式得到增量词集频率 Δf , 与给定的阈值 φ 进行比较,用于确定候选主题词 t_i 能否成为主题词。显然,在确定一个词 t_i 能否成为主题词时考虑了主题词集频率的增量 Δf_{ti} , 解决了候选主题词集中各候选词出现次数低,较平均时导致主题词提取不准确的问题。

2 实验结果分析与比较

2.1 参数 α, β 和阈值 φ 的确定

实验在复旦大学上海(国际)数据库研究中心自然语言处理(Natural Language Processing, NLP)小组提供的大约20 MB的文档集(包含1600篇政治经济类论文)上进行,随机选取了200篇文档用于参数的确定。

给定三组参数 α, β 值,即在不同阈值下着重考察提取主题词数量在5~10的文档数,所得结果如表3所示。

表3 不同参数下提取主题词数量在5~10的文档数

| 阈值 φ | $\alpha, \beta = (0.4, 0.6)$ | $\alpha, \beta = (0.5, 0.5)$ | $\alpha, \beta = (0.6, 0.4)$ |
|--------------|------------------------------|------------------------------|------------------------------|
| 0.030 | 153 | 118 | 149 |
| 0.035 | 176 | 165 | 153 |
| 0.040 | 176 | 186 | 172 |
| 0.045 | 183 | 194 | 176 |
| 0.050 | 179 | 183 | 184 |
| 0.055 | 186 | 178 | 191 |
| 0.060 | 167 | 180 | 185 |

从上述实验结果可知,参数 α 和 β 的值对实验结果的影响不太,而当 $\alpha = 0.5, \beta = 0.5, \varphi = 0.045$, 提取到主题词数量在5~10的文档篇数最多,覆盖面最广,达到了97%。

2.2 实验结果评价

选取参数 $\alpha = 0.5, \beta = 0.5, \varphi = 0.045$, 在复旦大学上海(国际)数据库研究中心 NLP 小组提供的大约20 MB的文本集(包含1600篇政治经济类论文)上随机选取的200篇文本(用于参数训练的文本)进行主题词提取,对所提取到的主题词进行满意度评价时,着重考察主题词数量在5~10的194篇文本,按“满意”“基本满意”“不满意”三个等次分析比较。其中“满意”指所提取到的主题词能够很好地表达文章的主

题思想,“基本满意”指提取到主题词有1~2个词不够准确或与该文章主题思想不符,“不满意”表示取到主题词有3个以上的词不能贴切地反映文档中心内容,而对其余10篇提取到主题词数不在5~10的文章则直接标记为不满意。满意度的评价是以人工方式进行的,即事先通过人工方式获得文本的主题词,作为参考主题词,然后将算法提取到的结果与参考主题词进行对比,对比过程借助程序来实现,并获得评价结果。表4是此次实验结果统计分析情况。

表4 实验结果满意度统计分析

| 满意情况 | 篇数 | 占总篇数的/% |
|------|-----|---------|
| 满意 | 148 | 74 |
| 基本满意 | 34 | 17 |
| 不满意 | 18 | 9 |

实验结果表明,对所提取到的主题词达到基本满意以上的文本篇数为182,占全部200篇的91%。

将本文算法应用于复旦大学上海(国际)数据库研究中心NLP小组提供文本集(包含1600篇政治经济类论文)除用于参数训练的200篇文本外的其余1400篇文本进行开放测试,算法运行在参数 $\alpha = 0.5, \beta = 0.5, \varphi = 0.045$ 下提取到主题词数在5~10的文本篇数为1302,达到了93%。对所提取到的主题词是否贴切表示文章主题思想进行评价比较,按“满意”“基本满意”“不满意”三个等次分析比较,统计分析结果如表5所示,实验数据显示总体满意度(基本满意以上)达到了87.42%,这个结果是比较理想的,比文献[12]算法满意度提高了5个百分点。

表5 开放测试结果统计分析

| 满意情况 | 篇数 | 占总篇数的/% |
|------|-----|---------|
| 满意 | 917 | 65.50 |
| 基本满意 | 307 | 21.92 |
| 不满意 | 176 | 12.58 |

2.3 与其他算法比较

本文算法提取到的文本主题词数量是不固定的,而其他大多数算法^[2-5]提取到的文本主题词数量是固定的,所以在一定程度上本文算法与其他算法难于进行比较。将本文算法应用于“人民日报中文版的文章作为语料库中一篇题目为‘温家宝总理在庆祝香港回归祖国六年酒会上的讲话’”^[2],将提取到的主题词与文献[2]算法及词频-逆向文件频率(Term Frequency-Inverse Document Frequency, TFIDF)算法提取结果进行比较,如表6所示。

表6 三种算法提取的主题词比较

| 序号 | 本文算法 | 文献[2]算法 | TFIDF 算法 |
|----|------|---------|----------|
| 1 | 香港 | 香港 | 香港 |
| 2 | 发展 | 繁荣 | 温家宝 |
| 3 | 中央政府 | 有利于 | 回归祖国 |
| 4 | 回归祖国 | 振兴 | 酒会 |
| 5 | 社会 | 回归祖国六周年 | 一国两制 |
| 6 | 同胞 | 稳定 | 中央政府 |
| 7 | 经济 | 团结 | 回归祖国六周年 |
| 8 | 一国两制 | 经济 | 同胞 |
| 9 | 繁荣 | 保持 | 总理 |
| 10 | 团结 | 社会 | 发展 |

表5所得结果是本文算法在参数 $\alpha = 0.5, \beta = 0.5, \varphi = 0.045$ 下提取到的,从主题词与文章主题思想切近程度而言,本文算法略差于文献[2]算法,但优于TFIDF算法。

3 结语

在对目前主要的主题词提取方法进行分析总结的基础上,改进文献[12]所提算法,增加文本预处理工作,加入合成词识别。实验结果表明,改进的算法取得了较理想的效果,对复旦大学上海(国际)数据库研究中心NLP小组提供文本集中的1400篇政治经济类论文提取的主题词总体满意度达到了87.42%。

该算法仍存在需要继续改进的方面:一是该算法较依赖于分词系统,分词的准确性直接影响到主题词的提取准确率,算法采用了中国科学院的ICTCLAS3.0,分词的准确性得到了一定的保证,尽管对分词的文本进行了处理,通过合成词的识别对文本进行了分词修正,但合成词识别的效率和准确率仍有待提高;二是候选主题词集的生成,需要采用更为有效的方法,词位置权重的方法虽然解决了词出现次数较平均情况下主题词提取出现的问题,但在语义上缺少分析和比较;三是需要更有效地解决提取的主题词中有同义或语义相近的词,尽管采用了哈尔滨工业大学同义词词林进行消同,但如何把握度的问题仍需提出更为合理的解决方法。

参考文献:

- [1] 赵鹏,蔡庆生,王清毅,等.一种基于复杂网络特征的中文文档关键词抽取算法[J].模式识别与人工智能,2007,20(6):817-831.
- [2] 耿焕同,蔡庆生,于琨,等.一种基于词共现图的文档主题词自动抽取方法[J].南京大学学报:自然科学版,2006,42(2):156-162.
- [3] 刘非,董萱菁,吴立德.利用关联规则挖掘文本主题词的方法[J].计算机工程,2008,34(7):81-83.
- [4] 胡学钢,李星华,谢飞,等.基于词汇链的中文新闻网页关键词抽取方法[J].模式识别与人工智能,2010,23(1):45-51.
- [5] 李芳芳,葛斌,毛星亮,等.基于语义关联的中文网页主题词提取方法研究[J].计算机应用研究,2011,28(1):105-107,123.
- [6] 石晶,李万龙.基于LDA模型的主题词抽取方法[J].计算机工程,2010,36(19):81-83.
- [7] ANETTE H, JUSSI K, ANNA J, et al. Automatic keyword extraction using domain knowledge[C]// CICLing 2001: Proceedings of the Second International Conference on Computational Linguistics and Intelligent Text Processing. Berlin: Springer-Verlag, 2001: 472-482.
- [8] ZHANG K, XU H, TANG J, et al. Keyword extraction using support vector machine[C]// Proceedings of WAIM 2006, LNCS 4016. Berlin: Springer-Verlag, 2006: 85-96.
- [9] 黄先珍,杨玉珍,刘培玉.信息过滤中基于统计与规则的关键词抽取研究[J].计算机工程,2012,38(2):57-59.
- [10] 刘晓明,冯晓荣,班超帆.基于动态点阵匹配算法的二阶关键词识别[J].吉林大学学报:工学版,2012,42(3):771-775.
- [11] 刘兴林,郑启伦,马千里.中文合成词识别及分词修正[J].计算机应用研究,2011,28(8):2905-2908.
- [12] 刘兴林,彭宏,马千里.基于增量词集频率的文本主题词提取算法研究[J].计算机应用研究,2010,27(9):3237-3238,3246.