文章编号：2096-1618(2018)02-0113-06

一种新的融合 BM25 与文本特征的新闻摘要算法

李 楠， 陶宏才
(西南交通大学信息科学与技术学院，四川 成都 611756)

摘要：提出一种融合 BM25 与文本特征的新闻摘要算法。首先使用 BM25 算法计算 TextRank 算法中的句子相似度，其次选择词频和句子位置作为文本特征，最后将文本特征的评分与 TextRank 的评分相加作为文本中句子的评分，对所有的句子按照评分降序排列，选择评分最高的几个句子作为摘要。使用 ROUGE 工具在 NLPCC2015 数据集上进行测试，结果表明该方法有较好的效果。

关 键 词：BM25；TextRank；词频；图排序；ROUGE
中图分类号：TP391 **文献标志码**：A
doi：10.16836/j.cnki.jcuit.2018.02.002

0 引言

随着社会发展进入互联网时代，人们获得信息的途径变得多种多样，同时越来越多的人依赖互联网获取所需要的信息。据 IDC 统计，在信息爆炸的时代，互联网数据已经跃升至 ZB 级别。在检索海量信息时，虽然搜索引擎可以筛选与检索条件相关的信息并且以标题列表的方式反馈给用户，但是搜索引擎提供的结果不够简洁和直接，用户仍需花大量时间去逐一浏览检索结果，降低了信息获取的效率。

文本摘要技术可以使用户在阅读网页全文之前快速了解网页的内容，继而决定是否需要阅读此网页全文，可以大大减少用户在检索信息时浏览网页所花费的时间，更高效地从万维网获取信息。自动文本摘要技术按所涉及的文本数量可以分为多文本摘要和单文本摘要，按产生摘要的形式则可以分为抽取式摘要和生成式摘要。抽取式摘要通过选取原文中最能概括全文信息的几个句子组成摘要，而生成式摘要则用新的句子描述原文的主要信息。目前，大多数的摘要方法都是基于抽取式方法。文中研究的是针对中文新闻的单文本抽取式摘要算法。

1 相关工作

自动文本摘要技术在 20 世纪 50 年代后期开始引起研究者的兴趣，随着研究的逐渐深入，研究者提出了多种自动文本摘要的算法，其中较为传统的文本摘要方法，是通过分析文章中的词频或者句子判断文本中最重要的元素，包括句子或词语。这种传统的文本摘要研究方式强调使用统计学模型来实现文本摘要，研究者基于语料库开发出了一些统计学模型，例如贝叶斯模型和隐马尔可夫模型，也有研究者在统计学模型的基础上融合了一些启发式的元素，例如关键字、句子的位置和长度、词频或标题。21 世纪，自动文本摘要技术开始广泛应用于网页文档，此时也出现了一些较新的自动摘要方法，例如基于图排序的方法，这种方法的基础是将文本表示成一个图，这样就考虑了文本的内在结构，而不是将文本作为简单的词语集合处理，因此在确定文章中的重要元素时这种方法能捕获和表达更丰富的信息，在构建图时，作为图中节点的文本片段可以是词组、句子或者段落。

Luhn[1]认为，文章中出现频率越高的单词和句子与文章主题的关联度越高，因此可以根据单词的词频和每个句子出现的频率给文本中的句子打分，并且选取得分最高的几个句子组成文章的摘要，该方法的提出标志着自动文本摘要技术的诞生。Baxendale 等[2]考虑了文本特征，将句子的位置作为评价句子重要性的参考依据，通过计算段落首句和末句中关键词出现的频率给句子打分，选取得分最高的几个句子作为摘要。Kupiec[3]提出一种基于分类模型的摘要方法，使用朴素贝叶斯分类算法判断一个句子是否应该抽取为摘要。Mihalcea R 等[4]提出基于图排序的 TextRank 算法，该算法改进自 LPage[5]提出的 PageRank 网页排序算法。TextRank 将文本中的句子作为图中的节点，句子之间的相似度作为边的权重，通过对图中的节点排序选择出最重要的几个句子作为文本的摘要。Federico Barrios 等[6]对 TextRank 算法的几种改进方式做

收稿日期：2018-01-13
基金项目：国家自然科学基金资助项目(61505168)

了对比,这几种改进方式均是针对句子间相似度的计算方式进行改进,并且在 DUC2002[7](document understanding conference)数据集上进行测试,得出的结论是基于 BM25 的相似度算法在 TextRank 中的应用效果最好。

中国 CCF 中文信息技术专委会曾在 2015 年组织过 NLPCC 评测,其中包括面向微博的新闻摘要任务。该专委会采用 ROUGE[8] 自动评价方法,并提供规模相对较大的样例数据和测试数据。此次 NLPCC 评测吸引了多支队伍参加,最终的评测结果表明 Liu M 等[9]提出的多特征融合算法取得了最好效果。此外,中国也有多种对 TextRank 算法改进的尝试。例如,针对科技文献,王子璇等[10]尝试将 WMD(word mover´s distance)[11]作为 TextRank 中计算句子相关度的方法,并且融合了科技文献的文本特征对句子进行评分,将评分最高的几个句子作为文献摘要。

综合来看,Federico Barrios 等提出的基于 BM25 的改进相似度算法在 TextRank 中应用虽然比经典的 TextRank 算法有较好的效果,但是该方法没有考虑文本特征,又由于新闻文本有较强的结构性和逻辑性,因此其在新闻摘要中的应用并没有取得最好的效果。Liu M 等提出的融合多特征的新闻摘要算法,选择了句子位置、句子长度、句子与标题的相似度和词频这 4 种文本特征作为给句子打分的依据,虽然在 NLPCC2015 的测评中取得了最好成绩[12],但是该方法仅仅考虑了文本特征而没有考虑语义层面的评价标准,因此具有一定的局限性。王子璇等提出的基于 WMD 语义相似度的 TextRank 改进算法融合了科技文献的文本特征,虽然在科技文献上有较好的效果,但是 WMD 的计算复杂度较高,且该算法存在一定局限性,并不适合新闻文本。

针对上述已有方法的不足提出了融合 BM25 与文本特征的新闻摘要算法,使用 ROUGE 测试工具和 NLPCC2015 数据集[13],测试对比了经典 TextRank 算法、基于 BM25 的改进 TextRank 算法(即文中算法)、基于 WMD 的改进 TextRank 算法和 Liu M 等提出的融合多特征的新闻摘要算法,测试结果表明提出的算法具有最好的效果。

2 融合 BM25 与文本特征的摘要算法

2.1 TextRank 与图模型

TextRank 算法的基本思想源于 Google 创始人 LPage 提出的 PageRank 算法,PageRank 算法广泛应用于搜索引擎的网页排序中;而 TextRank 算法则应用于基于有权图的文本排序。这种基于图的排序方法考虑了文本的内在结构,而不是将文本作为简单的词语集合处理,因此在确定重要概念时它能捕获和表达更丰富的信息,和 LDA[14]、HMM[15] 等模型相比,TextRank 的优点是简洁、高效,仅利用单篇文档本身的信息即可实现关键词提取、文本摘要,不需要事先对多篇文档进行学习训练,并且不受语言限制。

构建图时,选择的文本片段可以是词组、句子或者段落。考虑到内容丰富性和语法正确性之间的权衡,目前很多成功的系统主要使用句子构建图。根据此种构建方法,最重要的句子就是图中被连接最多的句子,它们被用来组成最终的摘要。TextRank 算法使用句子相似度确定句子之间的关系,也就是使用句子之间的相似度作为图中边的权重,句子间相似度的计算包括重叠词、余弦距离等多种计算方式。经典的 TextRank 算法中计算句子相似度的方法为

$$\mathrm{Sim}(S_i,S_j)=\frac{|\{w_k \mid w_k \in S_i \& w_k \in S_j\}|}{\log|S_i|+\log|S_j|} \quad (1)$$

其中,S_i 和 S_j 表示两个句子分词后各自得到的词向量,S_i 和 S_j 都由若干个词组成,$|S_i|$ 表示 S_i 中词的数量,$|\{w_k \mid w_k \in S_i \& w_k \in S_j\}|$ 表示既属于 S_i 也属于 S_j 的词的数量,通过公式(1)计算出的句子间相似度将作为图中边的权重,将用来计算图中节点的评分。

2.2 BM25 算法

经典的 TextRank 算法使用重叠词计算两个句子的相似度。为提升 TextRank 算法提取文本摘要的效果,有很多研究者尝试在 TextRank 中应用其他的相似度算法。例如,文献[6]对比包括 BM25 在内的 11 种相似度算法,使用 ROUGE 工具在 DUC2002 的数据集上测试了这些算法,得到的结果证明 BM25 在其对比的算法中有最好的效果。

BM25 算法最初由 Robertson 等于 1994 年提出并应用到信息检索领域,通常用来计算搜索相关性评分。BM25 的基本思想是,在信息检索过程中对查询语句 Q 进行解析,生成语素 q_i;然后,对每个搜索结果 D,计算每个语素 q_i 与 D 的相关性得分;最后,将所有语素相对于 D 的相关性得分进行加权求和,从而得到查询语句与 D 的相关性得分。BM25 算法的一般性公式为

$$\mathrm{Score}(Q,D)=\sum_{i}^{n} W_i \cdot R(q_i,D) \quad (2)$$

其中,Q 表示查询语句,q_i 表示 Q 解析之后的一

个语素。对中文来说,可以将对 Q 分词得到的结果作为 q_i ,D 表示一个文档。W_i 表示 q_i 的权重, $R(q_i,D)$ 表示 q_i 与文档 D 的相关性得分。应用在句子相似度的计算时,可以将文章中的一个句子作为 D,另一个句子作为 Q。

2.3　文本特征的选择

由于经典的 TextRank 算法没有考虑文本特征,而新闻文本又具有层次性和结构性强的特点,因此在使用 BM25 算法计算出新闻文本中每一句的 TextRank 评分之后,就加上了文本特征的评分。文献[9]提出的方法融合了词频、句子位置、句子与标题相似度和句子长度这 4 个特征来为新闻中的句子打分,4 个特征评分的比例为 1∶2∶1∶1,将评分最高的句子作为摘要,并且在 NLPCC2015 的数据集上取得了最好效果。

由于新闻标题具有简短、高度概括和吸引读者的特点,往往与新闻内容有所差别,而新闻句子长度也不能直接反映新闻的内容,因此文中选择了词频和句子位置作为文本特征。

2.4　新摘要算法:融合 BM25 与文本特征

提出的融合 BM25 与文本特征的摘要算法包括 3 个步骤。

首先使用 BM25 算法来计算文本中每个句子之间的相关度。在2.2节中已经初步介绍了 BM25 算法,应用在句子相似度的计算时,公式(2)中的 W_i 有多种表示方法,较常用的方法为 IDF,其计算过程如下:

$$\mathrm{IDF}(q_i)=\log\frac{N-n(q_i)+0.5}{n(q_i)+0.5} \tag{3}$$

其中 N 为文本中所有的句子数量, $n(q_i)$ 表示包含了 q_i 的句子数量。由公式(3)可以看出,包含 q_i 的句子越多,则 q_i 的权重越低。不过,当超过一半的句子都包含 q_i 时, $\mathrm{IDF}(q_i)$ 的计算结果是一个负数,这会影响后续的计算结果。因此,文献[6]提出了对计算 $\mathrm{IDF}(q_i)$ 的修正公式[6]:

$$\mathrm{IDF}(q_i)=\begin{cases}\log(N-n(q_i)+0.5)-\log(n(q_i)+0.5) & \\ \qquad\qquad n(q_i)\leqslant N/2 & \\ \varepsilon\cdot avgIDF \qquad n(q_i)>N/2 & \end{cases} \tag{4}$$

其中, $avgIDF$ 表示所有词的 IDF 值的平均值, ε 为调节参数。文献[6]通过实验证明了 ε 的最佳取值为0.25。BM25 中关于 $R(q_i,D)$ 的定义如下:

$$R(q_i,D)=\frac{f_i\cdot(k_1+1)}{f_i+K}\cdot\frac{qf_i\cdot(k_2+1)}{qf_i+k_2} \tag{5}$$

$$K=k_1\cdot(1-b+b\cdot\frac{dl}{avgdl}) \tag{6}$$

其中, k_1、k_2 和 b 均为调节参数,文献[6]设定 k_1 为1.2,b 为0.75, dl 为句子 Q 的长度, $avgdl$ 表示所有句子的平均长度, f_i 表示词 q_i 在文本中出现的频率, qf_i 表示词 q_i 在句子 Q 中出现的频率。绝大多数情况下,一个词只在一个句子中出现一次,即 $qf_i=1$。于是,BM25 算法的相关性公式可以总结为

$$\mathrm{Score}(Q,D)=\sum_i^n \mathrm{IDF}(q_i)\cdot\frac{f_i\cdot(k_1+1)}{f_i+k_1\cdot(1-b+b\cdot\frac{dl}{avgdl})} \tag{7}$$

通过公式(7),可以计算出句子 Q 和句子 D 之间的 BM25 相似度,使用公式(7)计算文本中每个句子之间的相似度,并且将该相似度作为 TextRank 算法中图中边的权重,该权重将用来计算图中节点的评分。

其次,使用 TextRank 算法将文章表示为有权图,计算每一个句子的 TextRank 评分。在构建有权图时,将文章中的句子作为图的节点,将上一步中计算出来的句子之间的相似度作为图中两节点之间的边的权重。计算节点评分时,需要为图中的节点指定初值,然后递归计算直至收敛,条件是图中任一节点的误差率小于给定的极限值即可达到收敛。节点的评分可通过公式(8)计算得到。

$$WS(V_i)=(1-d)+d\cdot\sum_{V_j\in In(V_i)}\frac{S_{ij}}{\sum_{V_k\in Out(V_j)}S_{jk}}WS(V_j) \tag{8}$$

其中, $WS(V_i)$ 是节点 V_i 的评分, $In(V_i)$ 代表指向 V_i 的节点集合, $Out(V_j)$ 代表 V_j 所指向的节点集合, d 是阻尼系数, 一般设置为0.85[10], $WS(V_j)$ 代表上次迭代 V_j 计算出的评分。通过公式(8)可以将一篇文本表示为有权图,并且对图中的所有句子计算评分。例如,一篇由 11 个句子组成的文本,其计算的结果可以由图 1 直观地展现出来。其中,节点的编号为句子的编号,节点的大小表示句子的评分高低,节点旁的数字表示该节点的评分。由图1可以看出,编号为

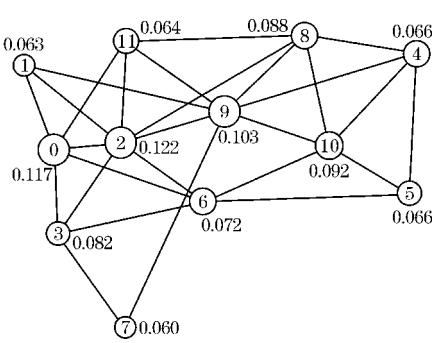

图 1　TextRank 计算结果示例

0、2、9 的 3 句为评分最高的句子,因此可以将这 3 句选作这篇文本的摘要。

最后,计算文章中每一个句子的文本特征评分,选择词频和句子位置作为文本特征,词频评分记为 TF,句子位置的评分记为 Pos。文献[9]使用词频和句子位置这两个文本特征,将两个特征评分的比例设置为 1∶2,在 NLPCC2015 的测试数据集上进行测试,也取得了较好的效果。根据文献[9]提出的方法,词频和句子位置的计算方法如公式(9)和公式(10)[9]所示。

$$tf(t_i) = \frac{m_i}{\sum_{j=1}^{n} m_j}$$

$$TF_i = \sum_{w \in sen_i} tf(w) \tag{9}$$

其中,n 表示文本中所有词的个数,m_i 表示词 t_i 在文本中出现的次数,$\sum_{j=1}^{n} m_j$ 表示文本中的所有词在文本中出现的次数之和,sen_i 表示编号为 i 的句子,TF_i 表示 sen_i 的词频评分,w 表示 sen_i 中的每一个词。

$$Pos_i = \frac{n - p_i + 1}{n} \tag{10}$$

其中,Pos_i 表示 sen_i 的句子位置评分,n 表示文本中句子的总数,p_i 表示 sen_i 在文本中的位置,即表示 sen_i 是文本中的第几个句子。由公式(10)可以看出,Pos_i 的取值最大为 1,随着句子位置增加依次递减,位置越靠前的句子得到的评分越高,位置越靠后的句子得到的评分越低。

文献[9]将 Pos_i 和 TF_i 的比例取为 2∶1,但是根据公式(9)可知,对一篇文章分词后,$\sum_{j=1}^{n} m_j$ 的值会远远大于 m_i,因此 TF_i 的计算结果远远小于 1。而 Pos_i 的最大值是 1,并且根据句子的位置依次递减。由此可知,大部分句子的 Pos_i 值会远大于 TF_i 值。因此,将这两个文本特征评分的比例设置为 2∶1 再相加时,使 TF_i 失去调节意义。所以将 Pos_i 和 TF_i 的比例取为 0.2∶1,将每个句子的词频评分和位置评分按照比例相加得到该句子的文本特征评分,记为 TFPos 评分。

经过上述 3 个步骤之后,对于文章中每一个句子,将其 TFPos 评分与 TextRank 评分相加,得到最终的句子评分记为 BM25TFPos。在得到文章中每一个句子的 BM25TFPos 评分之后,对所有句子按照 BM25TFPos 分数进行降序排列,选择评分最高的前 3 个句子作为这篇文章的摘要。获得一篇文章中每个句子的 BM25TFPos 分数的伪代码描述如下:

```
Input:Sentences 数组,每一个元素是一个句子
Output:Result 数组,每一个元素是一个句子的 BM25TFPos 分数
1:function GETBM25TFPOs(Sentences)
2:  N←length(Sentences)
3:  M←length(Sentences)
4:  TextRankScores[N]←0
5:  TF Scores[N]←0
6:  PosScores[N]←0
7:  Result[N]←0
8:  Graph[N]←0
9:  for i=0→N-1 do
10:   for i=0→M-1 do
11:     BM25Score←BM25(Sentences[i],Sentences[j])
12:     Graph[i][j]←BM25Score
13:     Graph[j][i]←BM25Score
14:   end for
15: end for
16: TextRankScores←TextRank(Graph)
17: T F Scores←CalcuteT F(Sentences)
18: PosScores←CalcutePos(Sentences)
19: for i=0→N-1 do
20:   Result[i]←TextRankScores[i]+T F Scores[i]+0.2 * Pos-
Scores[i]
21: end for
22: return Result
23:end function
```

3 实验分析

选择了经典 TextRank 算法、基于 BM25 的改进 TextRank 算法、基于 WMD 的改进 TextRank 算法和 Liu M 等提出的融合多特征的新闻摘要算法与提出的融合 BM25 与文本特征的摘要算法进行对比实验,为方便实验,将上述算法依次命名为 TextRank、BM25、WMD、WUST 和 BM25_TFPos。

实验使用的数据集是 NLPCC2015 的测试数据集,NLPCC2015 数据集包括 140 篇用于测试的新闻文本及其对应的人工撰写的摘要,和 250 篇用于评估的新闻数据。

实验使用 ROUGE1.5.5 作为自动测试工具,ROUGE 是基于摘要中 n 元词(n-gram)的共现信息来评价摘要,是一种面向召回率的评估方法。ROUGE 包括 ROUGE-1、ROUGE-2、ROUGE-3、ROUGE-L 等多种评价指标,本实验使用了 ROUGE-1、ROUGE-2、ROUGE-3、ROUGE-4、ROUGE-L 和 ROUGE-SU4 作为评价指标。由于 ROUGE 本身是用于英文摘要的评估,不支持中文字符,因此在进行实验时对 ROUGE 做

了轻微的修改,使其支持中文。ROUGE-*n* 的计算方法为

$$\text{ROUGE-}n=\frac{\sum_{S\in\{\text{Ref Summaries}\}}\sum_{n\text{-}gram\in S}Count_{match}(n\text{-}gram)}{\sum_{S\in\{\text{Ref Summaries}\}}\sum_{n\text{-}gram\in S}Count(n\text{-}gram)} \tag{11}$$

其中,*n-gram* 表示 *n* 元词,{Ref Summaries} 表示参考摘要,即事先获得的标准摘要,$Count_{match}(n\text{-}gram)$ 表示系统摘要和参考摘要中同时出现 *n-gram* 的个数,$Count(n\text{-}gram)$ 则表示参考摘要中出现的 *n-gram* 个数。

实验过程是,首先将一篇新闻文本分成句子,再使用 Jieba 分词工具对句子进行分词,并且去除停用词。然后依次使用上述 5 种算法,对这篇文本中的每一个句子计算评分,将评分最高的前 3 句作为这篇新闻的摘要。对数据集中的每一篇新闻都进行以上步骤,最后通过 ROUGE 工具计算所有摘要的平均 ROUGE 得分。5 种算法计算结果的对比如表 1 和图 2 所示。

表 1　ROUGE 评分的结果对比

算法	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4	ROUGE-L	ROUGE-SU4
BM25	0.52521	0.30217	0.21481	0.16675	0.49139	0.29612
TextRank	0.51499	0.31082	0.23172	0.18755	0.48221	0.30284
BM25_TFPos	0.55363	0.36408	0.28474	0.23685	0.5294	0.35342
WMD	0.40442	0.24458	0.18633	0.15297	0.38041	0.24105
WUST	0.54849	0.35867	0.2803	0.23312	0.52357	0.34808

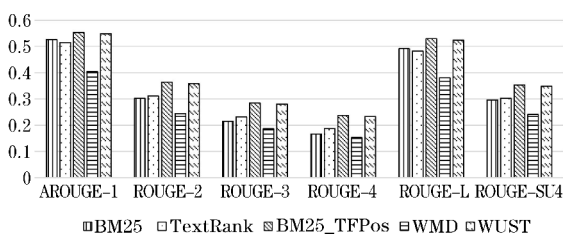

图 2　ROUGE 评分的柱状图对比

从实验结果的对比可以看出,提出的融合 BM25 与文本特征的摘要算法在 5 种算法的对比中,各项 ROUGE 评测值都高于其他算法,相较于在 NLPCC2015 中取得最好成绩的 WUST 算法也有所提高。由此可以证明,提出的算法在中文新闻的摘要提取方面有一定的优势,相较于其他算法有召回率高、算法复杂度低的特点。

4　结束语

提出一种融合 BM25 与文本特征的摘要算法,使用 BM25 计算 TextRank 中的句子相似度,同时融合词频和句子位置作为句子的评分依据。相较于其他使用词向量的方法,此方法有简单高效、计算复杂度低的特点;而与经典的 TextRank 算法相比,提出的方法融合了更多文本特征,并且在新闻摘要上取得了较好的效果。不过,方法在一定程度上依赖于对文本分句和分词的准确度,对文本分句和分词的准确性对 BM25、词频和句子位置的计算有一定影响,这些问题将在以后的工作中完善改进,以进一步提高新闻文本摘要的可读性,提高评估结果的召回率、准确率和 *F* 值。

参考文献:

[1] Luhn HP. The Automatic Creation of Literature Abstracts[J]. IBM Journal of Research Development, 1958,2(2):159-165.

[2] Baxendale P. Machine-made Index for Technical Literature-an Experiment[J]. IBM Journal of Research Development,1958,2(4):354-361.

[3] Kupiec J, Pedersen J, Chen F. A Trainable Document Summarizer[C]. ACM SIGIR. New York, USA,1995:68-73.

[4] Mihalcea, Rada, Tarau, et al. TextRank: Bringing Order into Texts[J]. Unt Scholarly Works,2004: 404-411.

[5] PAGE L. The PageRank Citation Ranking: Bringing Order to the Web, Online manuscript [J]. Stanford Digital Libraries Working Paper,1998,9(1):1-14.

[6] Barrios F, López F, Argerich L, et al. Variations of the Similarity Function of TextRank for Automated Summarization[C]. Argentine Symposium on Artificial Intelligence(ASAI)2015-44 JAIIO,2015:65-72.

[7] Document Understanding Conference: Duc 2002 guidelines[EB/OL]. http://www-nlpir. nist. gov/projects/duc/guidelines/2002. html,2002.

[8] Lin C Y. ROUGE: Recall-oriented understudy for gisting evaluation[J]. Text summarization branches out: Proceedings of the ACL-04 workshop,2004,

8:74-81.
[9] Liu M, Wang L, Nie L. Weibo-Oriented Chinese News Summarization via Multi-feature Combination [C]. Natural Language Processing and Chinese Computing-4th CCF Conference, 2015:581-589.
[10] 王子璇,乐小虬,何远标. 基于 WMD 语义相似度的 TextRank 改进算法识别论文核心主题句研究[J]. 现代图书情报技术,2017,1(4):1-8.
[11] Kusner M J, Sun Y, Kolkin N I, et al. From word embeddings to document distances[C]. International Conference on International Conference on Machine Learning. JMLR. org, 2015:957-966.
[12] Wan X, Zhang J, Wen S, et al. Overview of the NLPCC 2015 Shared Task: Weibo-Oriented Chinese News Summarization[M]. Natural Language Processing and Chinese Computing. Springer International Publishing, 2015.
[13] The 4th CCF Conference on Natural Language Processing & Chinese Computing[EB/OL]. http://tcci. ccf. org. cn/conference/2015/pages/page05_evadata. html, 2015.
[14] 张超,陈利,李琼. 一种 PST_LDA 中文文本相似度计算方法[J]. 计算机应用研究,2016,33(2):375-377.
[15] 孙师尧,妙全兴. 基于改进 SVM 和 HMM 的文本信息抽取算法[J]. 计算机应用与软件,2015,32(11):281-284.

A Novel News Summary Algorithm Combining BM25 and Text Features

LI Nan, TAO Hong-cai
(School of Information Science & Technology, Southwest Jiaotong University, Chengdu 611756, China)

Abstract: This paper presents a news summary algorithm that combines BM25 and text features. Firstly, we use the BM25 algorithm to calculate the sentence similarity in the TextRank algorithm, then select the word frequency and sentence position as the text features, and take the text feature score and the TextRank score as the final score of the sentence in the text. Finally, we sort all the sentences in descending order according to the final score, and select the sentences with the highest scores as the news summary. The test results on the dataset of NLPCC2015 using ROUGE tools show that this method has a better performance.

Keywords: BM25; TextRank; word frequency; graph sort; ROUGE