DOI:10.11721/cqnuj20190312

改进的关键词提取算法研究*

王 涛,李 明
(重庆师范大学 计算机与信息科学学院,重庆 401331)

摘要:【目的】针对词主题信息与词相似性信息对关键词提取的影响进行了研究,提出一种改进的 TextRank 关键词提取方法。【方法】首先,使用隐含狄利克雷分布(Latent Dirichlet allocation, LDA)主题模型对文档建模计算词主题信息;其次,使用 FastText 生成词向量,并计算词相似性矩阵;最后,融合词主题信息与词相似性信息的综合权重来优化 TextRank 词汇节点的初始权重,并进行词图模型的迭代运算与关键词提取。【结果】实验表明,改进方法的提取结果优于传统方法。【结论】证明了考虑词主题信息的全局性与词相似性信息的局部性能有效提高 TextRank 算法提取关键词的性能。

关键词:词向量;TextRank;图模型;LDA
中图分类号:TP391.1 **文献标志码:**A **文章编号:**1672-6693(2019)03-0098-07

关键词被认为是能够描述当前文档主题的一系列词语,被广泛运用于文本摘要、文本聚类、文本推荐中。在自然语言处理领域中,关键词提取为情感分析、知识图谱、舆情分析等热点问题提供了基础。文本关键词提取从是否需要对语料进行标记分为有监督、无监督两种类型。有监督[1-2]关键词提取将此看作二分类问题,需要提供已经标注好的训练语料,利用语料训练关键词提取模型,依据模型对需要提取的文档进行关键词提取;无监督提取无需对语料标注,通过对候选词集使用一定的算法机制将关键词按重要性排序,主流的方法包括基于词频统计 TF-IDF 模型[3]、基于主题 LDA 模型[4]、基于词汇图 TextRank 模型[5]这 3 种关键词抽取算法。

近年来,因无监督关键词提取方法适应性强,不少学者对上述方法进行改进以提高提取效果。基于词频统计特征的改进,牛萍等人[6]结合词位置特征和长度特征,并考虑兼类词的不同词性问题,来改进 TF-IDF 计算公式;黄磊等人[7]通过增加类内离散度来优化 IDF 计算词频的歧义问题,从而改进 TF-IDF 提取关键词的缺陷;基于主题模型的改进,刘啸剑等人[8-9]提出主题分布与统计特征相结合的方法对关键词进行抽取,还提出基于图与主题模型相结合的方法,利用 LDA 主题模型计算词与词之间的相似性,作为词与词之间的权重并构建一个带权无向词图,进行关键词提取;基于词汇图模型的改进,万小军等人[10]从文档集中计算与给定目标文档相近的文档集扩充词图信息量来辅助词图的构建;李晓等人[11]考虑词语语义相似度、位置、词频的重要性加权对 TextRank 算法改进;柳林青等人[12]将马尔科夫状态转移模型与 TextRank 中图节点相结合,使得词汇图中的节点边权由条件概率生成;以上研究将无监督关键词提取中 3 种算法单独或者组合改进来提升关键词的提取效果。

关键词提取受词语义与主题信息影响较大,方俊等人[13]通过消岐算法利用词义代替词丰富语义信息提高关键词提取算法性能;Wang 等人[14]首次将词向量引入候选词中来增加关键词的语义关系;宁建飞等人[15]利用 Word2vec 生成文档集中的词向量重构 TextRank 的概率转移矩阵;夏天等人[16]利用词向量聚类加权改进 TextRank;刘知远等人[17]将主题敏感的 PageRank 方法用于关键词提取中;顾益军等人[18]利用 LDA 对文档集候选关键词进行主题建模与影响力计算,来改进 TextRank 算法;使用 LDA 结合 TextRank 提取关键词未能考虑文档词的局部语义相关性,使用 Word2vec 结合 TextRank 提取关键词未能考虑文档词的全局主题信息,只考虑词主题或词相关性来抽取关键词,效果是不理想的。基于此,本文提出综合考虑词主题信息与词相似性信息对关键词提取算法 TextRank 进行改进。首先,利用 LDA 计算词在文档中的主题信息;其次,利用 FastText 训练生成词向量,并计算词向量间的相似度矩阵;最后,融合主题信息与词向量间相似性信息对 TextRank 关键词抽取

* **收稿日期:**2018-08-15 **修回日期:**2019-04-20 **网络出版时间:**2019-05-09 19:30
资助项目:重庆市教育委员会教改项目(No. 092055);重庆市教育委员会科技项目(No. kj098820)
第一作者简介:王涛,男,研究方向为文本挖掘、机器学习,E-mail:1379150080@qq.com;**通信作者:**李明,男,教授,E-mail:55613163@qq.com
网络出版地址:http://kns.cnki.net/kcms/detail/50.1165.N.20190509.1930.024.html

算法中的节点权重公式进行改进，从而构建概率转移矩阵进行词图模型的迭代计算与关键词的抽取。通过实验验证所提出方法的可行性，并与原有算法的效果比较。

1 相关理论

1.1 TextRank 原理

TextRank 算法应用于文本关键词提取时，将文档切分成单独的词，将每一个词看做一个节点，利用词共现关系建立词图模型，构建相应的概率转移矩阵，通过权重迭代计算每个词汇节点的得分，对得分进行排序，最终达到抽取关键词的目的。TextRank 算法的表达式为：

$$R(w_i)=(1-\lambda)+\lambda\times\sum_{w_j\in \mathrm{in}(w_i)}\frac{w_{ji}}{\sum\limits_{w_k\in \mathrm{out}(w_j)}w_{jk}}R(w_j), \tag{1}$$

其中，$\mathrm{in}(w_i)$是指向节点w_i的值，$\mathrm{out}(w_j)$是节点w_j指向的值，w_{ji}，w_{jk}是两点之间的边权，$R(w_i)$是节点w_i的权重，λ为阻尼系数。TextRank 算法抽取关键词步骤如下：

1）将给定的文本进行句子分割，得到 $L=[s_1,s_2,\cdots,s_n]$；

2）对于每个句子$s_i\in L$，进行分词、词性标注处理、过滤停用词，最后得到$s_i=[t_{i1},t_{i2},\cdots,t_{in}]$，$t_{in}\in s_i$为处理后的候选关键词；

3）构建词图 $G=(W,E)$，其中 w 为候选关键词节点集合，E 为候选关键词之间的边集合，候选关键词的共现关系决定边的有无，共现则有边，否则无；

4）根据上面公式，迭代传播候选关键词节点W_i的权重，直至收敛；

5）得到所有候选关键词节点W_i的权重，进行降序排列，将权重较大的词作为最终关键词。

1.2 LDA 主题模型

关键词提取可看作是将代表当前文档系列主题词汇提取出来，词语的重要性取决于对文档主题贡献度的内在关联关系，LDA 主题模型可用于挖掘大规模文档内部的隐含关系，LDA 是由 Bleide 等人[19]提出的包含文档-主题-词三层贝叶斯文档主题生成模型。在该模型中，每篇文档被表示为 K 个隐含主题的混合分布，而每个主题又是在 w 个词语上的多项分布，概率图如图 1 所示。

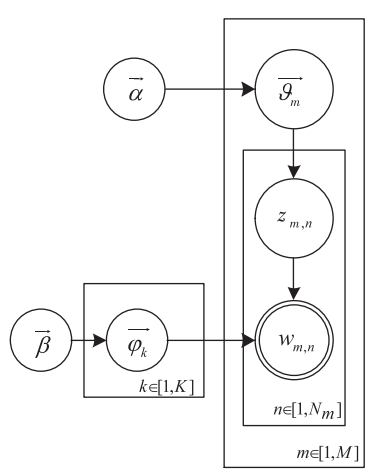


图 1　LDA 主题模型

Fig. 1　LDA theme model

在 LDA 主题模型中，ϑ 用来表示文档-主题的概率，φ 用来表示主题-词的概率分布，α，β 分别表示 ϑ，φ 服从的多项式 Dirichlet 分布的超参数，其中 w 是可被观测到词语。经过 LDA 主题建模后，文档 D 中的某个词汇 w，令 $T(w|D)$表示该词的在文档中的主题信息。假设文档 D 是由 K 个隐含主题构成，词汇 w 在主题 K 下所占概率越多，则词汇对主题 K 的贡献越大，如果该词 w 对相应的主题 K 在文档中 D 中出现的概率也越大，则可以说明主题 K 在文档中的作用也越大。因此，使用 LDA 主题模型来计算词的主题信息时，令φ_w^k表示词语 w 在主题 K 中的概率，ϑ_K^D表示文档中主题 K 的出现概率，那么词 w 的主题信息可表示为[18]：

$$T(w\mid D)=\sum_{i=1}^{t}(\vartheta_K^D\times\varphi_w^{k=i}), \tag{2}$$

其中，$\varphi_w^{k=i}$表示主题 K 含有词 w 的概率，词 w 在 TextRank 词汇图模型中代表节点。

在 LDA 模型中的参数 ϑ，φ 常使用吉布斯抽样(Gibbs sampling，GS)进行估算：

$$\vartheta_K^D=\frac{C_1(D,i)}{\sum\limits_{z=1}^{t}C_1(D,z)+t\times\alpha}, \tag{3}$$

$$\varphi_w^{k=i}=\frac{C_2(w,i)}{\sum\limits_{z=1}^{N}C_2(z,i)+N\times\beta}, \tag{4}$$

其中，$C_1(D,i)$表示主题 i 在文档 D 中出现的次数，$C_2(w,i)$表示训练语料(候选关键词)词 w 在主题 i 出现的次数。

α,β分别服从每个主题、词分布的超参数。N为候选关键词的个数,通过上面的描述即可计算词的主题信息,对LDA主题模型中的参数进行多次试验,根据经验将先验参数取值为$\alpha=0.14$,$\beta=0.01$,迭代次数为1 000次。

1.3 FastText 训练词向量

FastText是Facebook开源的一个计算词向量与文本快速分类工具。FastText训练词向量过程中是用连续字符的向量平均得到单词向量,FastText是在Word2vec的基础上提出一种有监督下的句子级别学习模型[20],其模型在训练得到词向量时类似于Word2vec中CBOW,由输入层、隐藏层和输出层构成。输入层输入一个词的序列(一段文本或一句话),序列中的词和词组组成特征向量,特征向量通过线性变换映射到中间隐藏层,隐藏层求得每个句子的词向量平均值;输出层实现负采样、层次softmax和softmax,其中负采样用来解决低频词和高频词被选中概率不均衡问题,层次softmax基于类别统计建立Huffman,确定一条从根节点到叶子节点的路径,计算其概率,然后利用随机梯度下降算法进行权值更新,以此来降低计算的复杂度。其中,加入n-gram特征,使得每个单词除了单词本身还被表示为多个字符级别的n-gram。例如,对于单词better,当n-gram中$n=3$时,该字符的n-gram就被表示为〈be,bet,ett,tte,er〉,FastText中每个Word可以产生多个字符n-gram,每个n-gram对应一个词向量,word的词向量是所有n-grams的词向量的和,使得语义信息更加丰富,FastText模型结构如图2所示。

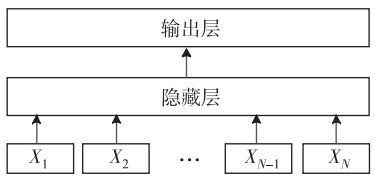


图2 FastText模型结构图

Fig. 2 FastText model structure diagram

2 改进方法与原理

2.1 改进的原理

在TextRank算法中,文档中的词语是通过共现关系来构建图模型,通过平均转移概率矩阵进行迭代计算每个词语权重,最终收敛后,将权重进行排序,选择权重较大的词语作为关键词。这样的做法很容易将在文档中出现频率高的词语抽取出来,但是一篇文档的关键词不仅只是出现频率高的词,有时频率高的词不一定是关键词。文档中的关键词应该是能概括文档主题的系列隐含词,不同主题下隐含词之间的语义相关性信息也是不同的,TextRank进行关键词的提取是利用词共现建立词汇图模型,词共现不仅受主题影响,受语义影响力也较大。因此,本文提出基于词主题信息和词相似性信息来改进TextRank算法。改进的算法主要分为3步:

1) 利用FastText对文档数据集进行训练,得到文档词向量,并计算词向量的相似性矩阵;

2) 考虑词主题信息引入LDA主题模型,计算词对主题的信息;

3) 融合词向量矩阵和词主题信息重构词节点权重计算公式,迭代计算至收敛,提取权重较大的词作为关键词。

本文提出基于词主题信息与词相关性信息改进TextRank算法的关键词提取,需构建关键词图,在构建词图前需对文档数据预处理,步骤如下:

1) 对于包含M个文档的集合D,对集合中的文本进行句子分割,得到句子集合$L=[s_1,s_2,\cdots,s_n]$。

2) 使用ICTCLAS分词工具对句子$s_i\in L$进行分词、去除停用词,进行词汇标注,保留名词、动词、形容词,进行词汇去重获取候选关键汇集$D_j=[w_1,w_2,\cdots,w_M]$。

3) 使用FastText对候选关键词集进行向量化表示,得到每个候选关键词的N维词向量$\boldsymbol{v}_{w_i}=(v_1,v_2,\cdots,v_N)$,用余弦函数计算词向量相似度为:

$$S(v_{w_i},v_{w_j})=\frac{v_{w_i}v_{w_j}}{\|v_{w_i}\|\cdot\|v_{w_j}\|}, \tag{5}$$

假设有N个候选关键词,最终得到候选关键词集$N\times N$基于词向量的相似度矩阵:

$$\boldsymbol{M}(S(v_{w_i},v_{w_j}))=\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nn} \end{bmatrix}, \tag{6}$$

其中,$\boldsymbol{M}(S(v_{w_i},v_{w_j}))$表示候选关键词的相似度矩阵。

4) 使用(2)式对文档中词汇的主题信息进行表示:

$$T(w_i \mid D_j)=\frac{C_1(D_j,i)}{\sum_{z=1}^{t}C_1(D_j,z)+t\times\alpha}\times\frac{C_2(w_i,i)}{\sum_{z=1}^{N}C_2(z,i)+N\times\beta}, \tag{7}$$

预处理结束后进行候选关键词图的构建，TextRank 的权重公式中，当前词汇结点的重要程度与有多少个相邻节点指向该节点有关，且相邻节点权重也影响当前节点，而词汇节点的权重为：

$$R(w_i)=(1-\lambda)\frac{1}{|V|}+\lambda\sum_{j:w_j\to w_i}\frac{e(w_j,w_i)}{O(w_j)}R(w_j), \tag{8}$$

其中，$R(w_i)$是词w_i的权重，$O(w_j)$为词w_i的出度，$e(w_j,w_i)$为$w_j\to w_i$为边上的权重，V 为节点集合，λ 是阻尼系数。

2.2 概率转移矩阵构建与关键词抽取

TextRank 算法在构建词图模型时，词汇节点间的权重通常取值为 1，通过相邻关系迭代来更新节点权重。本文改进算法融合词主题与词相似性信息进行概率转移矩阵构建，影响候选关键词节点权值的主要因素包括：

1）候选节点在图模型中的信息：生成的关键词排序依据候选节点在文档内部的重要性用 $R'(w_i)$表示，初始 $R'(w_i)=1$，随着迭代而改变。

2）候选节点的主题信息：本文在计算主题信息时使用 LDA 主题模型来计算，由 LDA 主题模型得到词主题信息 $T(w_i)$，对主题信息进行归一化计算，使用 $T'(w_j)$表示：

$$T'(w_i)=\sum_{j:w_j\to w_i}\frac{T(w_i)}{\sum_{j:w_i\to w_j}T(w_j)}R'(w_j), \tag{9}$$

3）候选关键词的语义信息：考虑文档语义主题信息使用 FastText 生成的词向量，根据余弦函数计算词向量间的相似度，词向量间相似度越大则词的语义相关性概率越高，对相似性关系信息进行归一化计算：

$$S'(v_{w_i},v_{w_j})=\sum_{j:w_j\to w_i}\frac{S(v_{w_i},v_{w_j})}{\sum_{j:w_i\to w_j}S(v_{w_i},v_{w_j})}R'(w_j), \tag{10}$$

综合上述 3 个因素对候选关键词节点重要性进行计算，对 TextRank 中原始节点权重计算公式(8)式进行改进，得到节点重要性计算为：

$$R(w_i)=\lambda\left(\alpha'T'(w_j)+\beta'S'(v_{w_i},v_{w_j})+\gamma'\sum_{j:w_j\to w_i}\frac{R'(w_j)}{O(w_j)}\right)+(1-\lambda)\frac{1}{|V|}, \tag{11}$$

在迭代计算之前，构建词汇间的概率转移矩阵为：

$$\boldsymbol{M}=\begin{bmatrix}w_{11}&w_{12}&\cdots&w_{1n}\\w_{21}&w_{22}&\cdots&w_{2n}\\\vdots&\vdots&\ddots&\vdots\\w_{n1}&w_{n2}&\cdots&w_{nn}\end{bmatrix}, \tag{12}$$

其中，矩阵中元素w_{ij}为当前节点w_j的影响力会转移到第 i 个词汇w_i的概率，可由(13)式计算得到：

$$w_{ij}=\alpha T'(w_j)+\beta S'(v_{w_i},v_{w_j})+\gamma\sum_{j:w_j\to w_i}\frac{1}{O(w_j)}, \tag{13}$$

根据(13)式可得到概率转移矩阵的所有元素值，进而计算每次迭代的结果，如(14)式所示：

$$\boldsymbol{B}_i=\lambda\boldsymbol{M}\boldsymbol{B}_{i-1}+(1-\lambda)\frac{\boldsymbol{e}}{n}, \tag{14}$$

上式中，每次候选节点的重要性分值记为向量$\boldsymbol{B}_i$，对于有 M 个候选关键词的集合文档 D，迭代计算的初始默认值均为$\frac{1}{M}$，则候选关键词的初始分值向量可表示为：

$$\boldsymbol{B}_0=\left(\frac{1}{M},\frac{1}{M},\cdots,\frac{1}{M}\right), \tag{15}$$

$\boldsymbol{e}$ 为所有分量为 1 的 K 维向量。当迭代中计算得到相邻两次结果的差异值当小于预设的阈值 0.000 1 时算法终止，及迭代过程以收敛，然后对所有节点的权值按降序排列，选取权重较大的词作为文档的关键词。

3 实验结果与分析

3.1 实验数据与评估标准

实验选取搜狗实验室数据集含国内、国际、体育、社会、娱乐等多个领域新闻数据共1.43 G使用FastText训练词向量。每个领域挑选10篇来作为测试集,共90篇测试语料,其余作为训练语料,在测试语料中,采用多组人工交叉标注每篇提取10个关键词,降低个人主观性对结果带来的偏差。使用关键词抽取常用的评价标准有准确率P(Precision)、召回率R(Recall)以及F值(F-measure),计算公式如下:

$$P=\frac{N_1}{N}\times 100\%, \tag{16}$$

$$R=\frac{N_1}{N_2}\times 100\%, \tag{17}$$

$$F=\frac{2\times P\times R}{P+R}\times 100\%, \tag{18}$$

其中,N_1代表正确提取的关键词集合,N代表提取的关键词集合,N_2人工标注的关键词集合。

3.2 试验参数调节

1) 对于改进的公式,参数取值不同可能会对关键词抽取的性能产生影响,因此对公式中参数权重因子α,β,γ分别进行取值研究,取500篇文本作为测试集,观察表1的实验结果。

表1 实验结果

Tab. 1 Experimental result

α	β	γ	准确率/%
0	0	1	33.42
1	0	0	27.23
0	1	0	30.75
0.33	0.33	0.34	42.24

从表1可以看出,不同的α,β,γ取值对关键词抽取的准确率有影响,当$\alpha=\beta=0,\gamma=1$时,即只使用原始TextRank算法进行关键词抽取,准确率较低;当$\beta=\gamma=0,\alpha=1$时,只考虑词主题信息来改变词图的权重,效果差于使用TextRank;当$\alpha=\gamma=0,\beta=1$,只考虑使用FastText生成词向量对词语加权的抽取算法,效果较好;当$\alpha=\gamma=0.33,\beta=0.34$时,即综合考虑关键词信息与词向量信息来对节点权重重构,即各参数重要性视为均等,可以看出,融合改进方法能更好地提升关键词抽取的准确率,因此,本文在实验中取$\alpha=\beta=\gamma=\frac{1}{3}$。

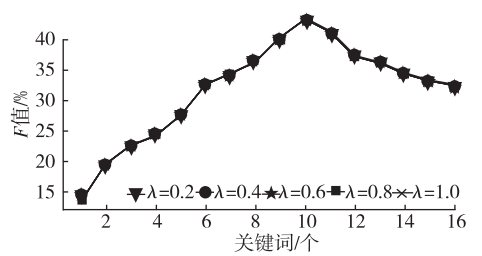


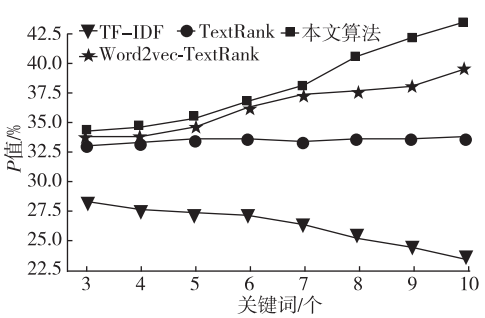
图3 阻尼系数对抽取结果的影响

Fig. 3 Influence of damping coefficient on extraction results

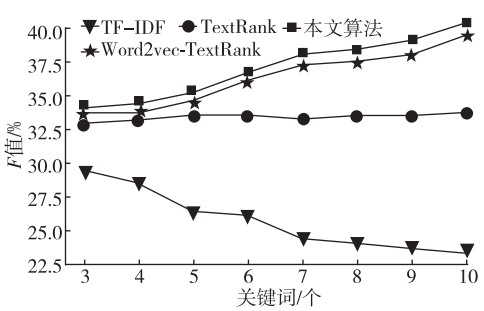
2) 使用TextRank抽取关键词时,根据经验常取阻尼系数$\lambda=0.85$,由于改进算法在重构权值时,对原始TextRank算法中的权值计算公式进行了改动,所以需要通过调节阻尼系数来观察对实验结果的影响。阻尼系数取值范围为$\lambda\in[0,1]$,实验分别取λ的值为0.2,0.4,0.6,0.8,1.0,并在各个阻尼系数下分别抽取关键词数K取值为1,2,3,4,5,…,16时,观察实验对F值的影响。如图3所示,从实验结果可以看出阻尼系数的不同取值对实验的影响不大,当关键词个数取值$K=10$时,其F值最高。

3.3 算法对比

1) 为了证明改进算法的性能,本文在相同的测试集,与基于TF-IDF中文关键词算法、基于TextRank关键词抽取算法以及文献[15]提出的Word2vec-TextRank关键词抽取算法进行实验对比,实验结果如图4和表2所示。



a P值的比较



b F值的比较

图4 几种算法的P值与F值的比较图

Fig. 4 Comparison graph of P and F values

由图 4 及表 2 可以看出，基于词频统计的 TF-IDF 算法随着抽取关键词的个数变化影响较大；而基于 TextRank 算法的关键词抽取效果受关键词个数影响不大；基于 Word2vec 与 TextRank 算法的关键词抽取效果比前两者更理想。本文考虑词主题与词相似性信息改进的 TextRank 算法在抽取关键词时相对于其他算法效果在 P 值、R 值、F 值有明显提高。

2）为了进一步验证本文算法提取关键词的性能，设定抽取的关键词为 10 个，分别在体育、社会、交通、环境、经济不同类别下，与上文中提到的算法进行比较，实验结果如图 5 所示。

从图 5 可以看出，本文提出的关键词提取算法比已有的关键词提取算法在各个类别上所表现的性能要好，这也证明了本文改进算法的有效性。

表 2　4 种算法的实验结果对比

Tab. 2　Experimental results of the four algorithms are compared

关键词/个	算法	P/%	R/%	F/%
5	TF-IDF	26.28	27.43	27.23
	TextRank	33.54	34.67	33.42
	Word2vec-TextRank	33.68	34.94	33.91
	本文算法	35.26	35.56	34.71
7	TF-IDF	24.32	24.92	24.54
	TextRank	33.35	33.35	33.17
	Word2vec-TextRank	37.63	38.61	38.53
	本文算法	38.54	39.14	40.32
10	TF-IDF	23.34	23.34	23.26
	TextRank	33.73	33.98	33.44
	Word2vec-TextRank	39.54	39.54	39.64
	本文算法	40.41	41.21	42.24

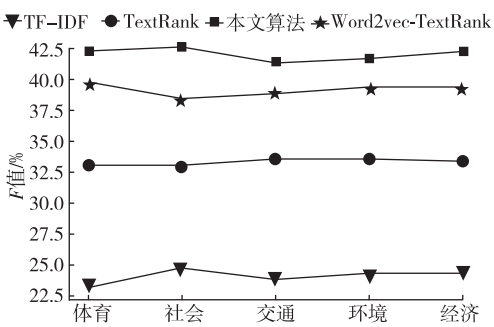


图 5　不同类别下关键词提取的实验结果

Fig. 5　Experimental results of keyword extraction under different categories

4 结束语

文档中的词主题与词相似性信息对关键词的抽取效果会产生直接的影响，使用 FastText 生成文档的词向量考虑文档上下文信息以获取更多的词语义信息。通过实验表明，本文所提出改进的 TextRank 算法能提升关键词抽取的准确性，将现有的词主题与词相关性融合多维信息从而提升对关键词提取的效果。同时，接下来的工作是研究与目标文档相关的外部知识库与训练词向量的文档语义信息对关键词提取效果的影响。

参考文献：

[1] HABIBI M，POPESCU-BELIS A. Keyword extraction and clustering for document recommendation in conversations [J]. IEEE/ACM Transactions on Audio Speech & Language Processing，2015，23(4)：746-759.

[2] XIE F，WU X，ZHU X. Efficient sequential pattern mining with wildcards for keyphrase extraction[J]. Knowledge-Based Systems，2017，115：27-39.

[3] LI J，FAN Q，ZHANG K. Keyword extraction based on TF/IDF for chinese news document[J]. Wuhan University Journal of Natural Sciences，2007，12(5)：917-921.

[4] BLEI D M，NG A Y，JORDAN M I. Latentdirichlet allocation[J]. J Machine Learning Research Archive，2003，3：993-1022.

[5] MIHALCEA R，TARAU P. TextRank：bringing order into texts[J]. Emnlp，2004：404-411.

[6] 牛萍，黄德根. TF-IDF 与规则相结合的中文关键词自动抽取研究[J]. 小型微型计算机系统，2016，37(4)：711-715.
NIU P，HUANG D G. Study on Chinese keyword automatic extraction based on TF-IDF and rules[J]. Journal of Chinese Mini-Micro Computer Systems，2016，37(4)：711-715.

[7] 黄磊，伍雁鹏，朱群峰. 关键词自动提取方法的研究与改进[J]. 计算机科学，2014，41(6)：204-207.
HUANG L，WU Y P，ZHU Q F. Research and improvement of automatic keyword extraction methods[J]. Computer Science，2014，41(6)：204-207.

[8] 刘啸剑，谢飞. 结合主题分布与统计特征的关键词抽取方法[J]. 计算机工程，2017，43(7)：217-222.
LIU X J，XIE F. Keyword extraction method combining subject distribution and statistical features[J]. Computer Engineering，2017，43(7)：217-222.

[9] 刘啸剑，谢飞，吴信东. 基于图和 LDA 主题模型的关键词抽取算法[J]. 情报学报，2016，35(6)：664-672.
LIU X J，XIE F，WU X D. Keywords extraction algorithm based on graph and LDA theme model[J]. Journal of the China Society for Scientific and Technical Information，2016，35(6)：664-672.

[10] WAN X J，XIAO J G. Single document keyphrase extractionusing neighborhood knowledge[C]//AAAI'08 Pro-

ceedings of the 23rd National Conference on Artificial Intelligence. Chicago, Illinois: AAAI Press, 2008: 855-860.

[11] 阿力甫·阿不都克里木,李晓. 基于 TextRank 算法和互信息相似度的维吾尔文关键词提取及文本分类[J]. 计算机科学,2016,43(12):36-40.
GHALIP A, LI X. Uighur keyword extraction and text classification based on textrank algorithm and mutual information similarity[J]. Computer Science, 2016, 43(12): 36-40.

[12] 柳林青,余瀚,费宁,等. 一种基于 TextRank 的单文本关键字提取算法[J]. 计算机应用研究,2018(3):705-710.
LIU L Q, YU W, FEN N, et al. A single text keyword extraction algorithm based on textrank[J]. Application Research of Computers, 2018(3): 705-710.

[13] 方俊,郭雷,王晓东. 基于语义的关键词提取算法[J]. 计算机科学,2008,35(6):148-151.
FANG J, GUO L, WANG X D. Semantic-based keyword extraction algorithm[J]. Computer Science, 2008, 35(6): 148-151.

[14] WANG R, LIU W, Mc DONALD C. Corpus-independent generic keyphrase extraction using word embedding vectors [EB/OL]. [2018-08-15]. https://www.ixueshu.com/document/6c41cd4df44fd223318947a18e7f9386.html.

[15] 宁建飞,刘降珍. 融合 Word2vec 与 TextRank 的关键词抽取研究[J]. 现代图书情报技术,2016,27(6):20-27.
NING J F, LIU J Z. Research on keyword extraction of Word2vec and TextRank[J]. New Technology of Library and Information Service, 2016, 27(6): 20-27.

[16] 夏天. 词向量聚类加权 TextRank 的关键词抽取[J]. 数据分析与知识发现,2017,1(2):28-34.
XIA T. Keyword extraction of word vector clustering-weighted TextRank[J]. Data Analysis and Knowledge Discovery, 2017, 1(2): 28-34.

[17] HAVELIWALA T H. Topic-sensitive page rank: a context-sensitive ranking algorithm for web search[J]. IEEE Transcations on Knowledge & Data Engineering, 2003, 15(4): 784-796

[18] 顾益军,夏天. 融合 LDA 与 TextRank 的关键词抽取研究[J]. 数据分析与知识发现,2014,30(z1):41-47.
GU Y J, XIA T. Research on keyword extraction of fusion LDA and TextRank[J]. Data Analysis and Knowledge Discovery, 2014, 30(z1): 41-47.

[19] BLEI D M, NG A Y, JORDAN M I. Latent Dirichlet allocation[J]. Journal of Machine Learning Research, 2003, 3: 993-1022.

[20] BOJANOWSKI P, GRAVE E, JOULIN A, et al. Enriching word vectors with subword information[J]. Transactions of the Association for Computational Linguistics, 2017, 1(5): 135-146.

Study on an Improved Keyword Extraction Algorithm

WANG Tao, LI Ming

(School of Computer and Information Sciences, Chongqing Normal University, Chongqing 401331, China)

Abstract: [**Purposes**]Aiming at the influence of word topic and word similarity on keyword extraction, an improved TextRank keyword extraction method is proposed. [**Methods**]First, by using Latent Dirichlet Allocation (Latent Dirichlet Allocation, LDA) word theme topic influence model to calculate the document model. Secondly, by employing FastText to generate word vectors and calculate word similarity matrices. Finally, by integrating the weight of word theme influence and word similarity influence to optimize the initial weight of vocabulary node in TextRank, iterative operation and keyword extraction of word graph model. [**Findings**]Experiments show that the extraction result of the improved method is better than the traditional method. [**Conclusions**]It is proved that the global influence of word topic and the local influence of word similarity can effectively improve the performance of TextRank algorithm in extracting keywords.

Keywords: word vector; TextRank; graph model; LDA

(责任编辑 许 甲)