计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200900082

一种基于图的文档关键词和摘要协同抽取方法研究

毛湘科[1,2,3]　黄少滨[1]　余秦勇[2,3]
1 哈尔滨工程大学计算机科学与技术学院　哈尔滨 150001
2 中电科大数据研究院有限公司　贵阳 550022
3 提升政府治理能力大数据应用技术国家工程实验室　贵阳 550022
(maotiamo@hrbeu.edu.cn)

摘　要　关键词提取和摘要抽取的目的都是从原文档中选择关键内容并对原文档的主要意思进行概括。评价关键词和摘要抽取质量的好坏主要看其能否对文档的主题进行良好的覆盖。在现有基于图模型的关键词提取和摘要抽取方法中,很少涉及到将关键词提取和摘要抽取任务协同进行的,而文中提出了一种基于图模型的方法进行关键词提取和摘要的协同抽取。该方法首先利用文档中词、主题和句子之间的6种关系,包括词和词、主题和主题、句子和句子、词和主题、主题和句子、词和句子,进行图的构建;然后利用文档中词和句子的统计特征对图中各顶点的先验重要性进行评价;接着采用迭代的方式对词和句子进行打分;最后根据词和句子的得分,得到关键词和摘要。为验证所提方法的效果,文中在中英文数据集上进行关键词提取和摘要抽取实验,发现该方法不管是在关键词提取还是摘要抽取任务上都取得了良好的效果。

关键词:关键词提取;摘要抽取;图模型;主题覆盖
中图法分类号　TP311.131

Graph Based Collaborative Extraction Method for Keywords and Summary from Documents

MAO Xiang-ke[1,2,3],HUANG Shao-bin[1] and YU Qin-yong[2,3]
1 College of Computer Science and Technology,Harbin Engineering University,Harbin 150001,China
2 CETC Big Data Research Institute Co.,Ltd.,Guiyang 550022,China
3 Big Data Application on Improving Governance Capabilities National Engineering Laboratory,Guiyang 550022,China

Abstract　The purpose of keywords extraction and summary extraction is to select key content from the original document to express the main meaning of the original document. The evaluation of keywords and summarization quality mainly depends on whether it can cover the main topics of the document. In the existing methods of keywords extraction and summary extraction based on graph models,it rarely involves the task of keywords extraction and summary extraction collaboratively. The article proposes a method based on a graph model for simultaneous keywords extraction and summary extraction. The method first uses the six relationships among words,topics,and sentences in the document,including words-words,topics-topics,sentences-sentences,words-topics,topics-sentences,words-sentences,to construct the graph;then uses the statistical characteristics of the words and sentences in the document to evaluate the prior importance of each vertex in the graph;next,it uses an iterative way to score words and sentences;finally,we get the final keywords and summary based on the scores of words and sentences. In order to verify the effectiveness of the proposed method,keywords extraction and summary extraction experiments are carried out on Chinese and English datasets. It is found that the proposed method achievs good results in both keywords extraction and summary extraction tasks.

Keywords　Keywords extraction,Extractive summarization,Graph model,Topic cover

1　引言

抽取型摘要的本质是从文档中选择重要的句子,关键词提取实际是从文档中选择重要的词或短语。通过对人工生成的关键词和摘要的观察,可以发现重要句子中通常包含有重要词语,包含重要词语的句子更有可能被选为摘要,同时关键词和摘要能对原文档的主题进行良好的覆盖。在对机器生成的摘要或关键词的质量进行评估时,一个非常重要的指标是

到稿日期:2020-09-10　返修日期:2021-03-10
基金项目:提升政府治理能力大数据应用技术国家工程实验室开放基金项目
This work was supported by the Big Data Application on Improving Governance Capabilities National Engineering Laboratory Open Fund Project.
通信作者:黄少滨(huangshaobin@hrbeu.edu.cn)

对原文档主题的覆盖性。因此，本文提出了一种候选关键词-主题-句子排序（Candidate-Keywords-Topic-Sentence Rank，CTSRank）的方法，利用词语、主题和句子之间存在的相互影响关系，对文档中的词语和句子进行协同排序，并选择出关键词和重要句子。

CTSRank 共分为 3 个阶段：第 1 个阶段是利用文档中的词、主题和句子进行图的构建。图的构建主要包括词图、主题图和句子图，以及图中各顶点的度量，其中文档中的主题是通过 K-Means 算法对提取的候选关键词聚类得到的。第 2 个阶段是对图中词、主题和句子的统计先验重要性进行计算。计算词语的先验重要性时，主要根据词语在文档中出现的位置；计算主题的先验重要性时，主要依靠主题中的候选关键词出现的位置、TF-IDF 值以及主题与文档的相似性 3 个指标；计算句子的先验重要性时，主要基于句子的位置、句子与文档整体的相似性、句子的 TF-ISF 值 3 个指标。在前两个阶段将文档表示成图之后，第 3 个阶段主要包括对词、主题、和句子的打分，以及根据打分的结果选择关键词和重要句子。对于重要句子的选择，将采用最大边缘相关算法[1]（Maximal Marginal Relevance，MMR）。对于关键词的选择，将根据主题和候选关键词的得分进行选取。CSTRank 算法对文本进行表示的方式如图 1 所示。

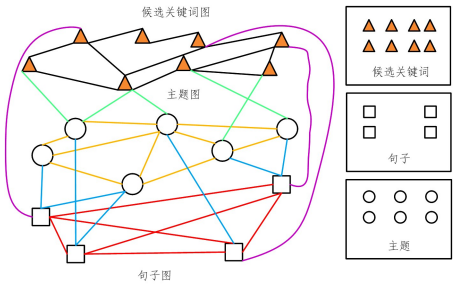


图 1　文档表示图

Fig. 1　Document representation

本文分别在中文和英文语料上对 CTSRrank 算法在关键词提取和摘要抽取问题上的效果进行评估。其中英文语料采用的是 DUC2001 数据集，中文语料采用的是来自中文期刊的学术文章，详细的信息将在实验部分进行介绍。通过将 CTSRank 在两个数据集上的实验结果与一些常见的关键词提取和摘要抽取算法进行对比可以发现，在相应的评价指标下，CTSRank 在关键词提取和摘要抽取任务上都取得了良好的效果 。

2　相关工作

基于图模型的自动文档摘要和关键词提取方法在网络上电子文档爆炸性增长的背景下越来越受到研究者们的关注。受到网页排序算法 PageRank[2] 的启发，两种经典的使用图模型进行文档建模的方法 TextRank[3] 和 LexRank[4] 被提出。其中 TextRank 在进行关键词提取时，建立文本图的方式为以词作为顶点，以共现关系建立边，且边的权重相等。LexRank 进行摘要抽取任务时，以句子作为顶点，句子间的相似性作为边的权重来建立文本图。现有的许多基于图模型的关键词提取或摘要抽取方法是由 TextRank 和 LexRank 改进而来的，主要包括对图中边的权重计算和顶点的初始权重的改进。下面对这两种方法的改进方法分别进行介绍。

对于 TextRank 中边的权重度量方法的改进方法如下。文献[5]针对 TextRank 中边权重为 1 的问题，提出 SingleRank 方法，将词语在滑动窗口中的共现次数作为边的权重，同时文献[5]提出了使用相邻文档的信息对图中边的权重进行修改的 ExpandRank 方法。还有一些方法是利用 WordNet、维基百科等知识库来计算边的权重[6-7]，或是利用词嵌入来衡量词语间的关系[8-9]。对于 TextRank 中顶点初始权重的改进，文献[10]提出了 TopicPageRank 方法，将得算法偏向于选择与隐含主题相关度高的词语作为关键词。文献[11]提出了 PositionRank 方法，其使用词语在文档中出现的所有位置信息度量词的初始权重。类似改变顶点初始权重的工作还包括文献[12-14]。

对于 LexRank 算法中边权重的改进方法如下。文献[15]提出了一种混合余弦、杰卡德等 4 种相似度的方法来度量边权重；文献[16]针对上下文敏感的词汇权重计算方法提出了句子间相似度计算方法。同时，还有些方法借助外部知识库[17-18]、词嵌入和句嵌入[19-21]的方式改进句子间相似度的度量。对于 LexRank 顶点初始权重的改进，文献[22]针对面向查询型摘要，使用句子和标题之间的相似性度量顶点初始权重，类似的工作还有文献[23]。对于通用型摘要，文献[24]提出了一种利用监督学习方法为句子赋予初始权重的方法。

有些研究工作对词和句子的协同排序进行了探索。文献[25]提出了一种以词、句为顶点的构图方式，综合词与词、句与句、词与句之间的关系对图中的词和句子进行排序。文献[26]基于重要句子包含重要词，且重要词组成重要句子的假设，提出了基于词和句协同排序的摘要抽取方法。但是上述方法并没有考虑到文档所包含的主题，本文在对文档进行构图时，融入了主题这一类顶点，使得在对词和句子进行排序时，能考虑各主题在文档中的重要性。

本文对原始的图模型方法进行了 3 方面的改进，分别是选取了 3 种不同粒度的顶点进行图的构建、改进边间的度量关系，以及更改了各顶点的初始权重。最后用迭代的方式对词和句子进行协同排序。

3　提出的方法

本节将详细介绍所提方法中文本图的构建方法，其作用在于将任意一个包含 n 个句子的文档 $d=\{s_1,s_2,\cdots,s_n\}$，使用图 $G=\{V,E\}$ 进行表示，其中 V 表示图中顶点集合，代表文档中的候选关键词、主题和句子；E 表示图中带权重的边集合，包括图中所有顶点之间边的关系。

3.1　词图的构建

词图的构建采用候选关键词作为顶点，候选关键词的选择方式可以参考文献[27]。从文档 d 中得到的候选关键词可

以表示为 $ck=\{c_1,c_2,\cdots,c_m\}$。边的权重计算方法如式(1)所示：

$$\boldsymbol{O}_{CC}(c_i,c_j)=\begin{cases}dist(c_i,c_j)*cosine(\boldsymbol{v}_{c_i},\boldsymbol{v}_{c_j}), & \text{if } i\neq j\\0, & \text{if } i=j\end{cases} \tag{1}$$

其中，$dist(c_i,c_j)$表示候选词c_i和c_j之间基于位置的距离，而$cosine(\boldsymbol{v}_{c_i},\boldsymbol{v}_{c_j})$表示基于语义的距离，$\boldsymbol{v}_{c_i}$和$\boldsymbol{v}_{c_j}$表示使用 Sentence-BERT[28]对$c_i$和$c_j$进行编码后的向量。$dist(c_i,c_j)$的计算方法如下：

$$dist(c_i,c_j)=\sum_{p_i\in pos(c_i)}\sum_{p_j\in pos(c_j)}\frac{1}{|p_i-p_j|} \tag{2}$$

其中，$pos(c_i)$表示c_i中词在文档中的所有位置。这里，将两个候选关键词向量化后的余弦相似度作为语义相似度。只有当两个候选关键词在文档中的位置相近、语义相似度高时，候选关键词之间边的权值才会大。

3.2 主题图的构建

主题的内容都是由候选关键词组成的，其中包括词或者短语。从目标文档中选择出候选关键词后，对各候选关键词采用 Sentence-BERT[28]模型进行向量化，并使用 K-Means 算法对候选关键词进行聚类。在完成对候选词的聚类后，得到k个不同的类$t=\{t_1,t_2,\cdots,t_k\}$，其中$t_i=\{c_i,c_j,\cdots,c_m\}$。最后使用这$k$个不同的类作为图中的顶点进行图模型的构建。在计算各顶点之间的关系时，CSTRank 融合了词语之间的语义关系和词语之间的相对位置关系。我们采用式(2)计算主题内两个候选关键词之间的相对位置距离，采用式(3)计算两个主题之间边的关系。

$$(\boldsymbol{O}_{TT})_{ij}=\begin{cases}\sum_{c_x\in t_i}\sum_{c_y\in t_j}\boldsymbol{O}_{CC}(c_x,c_y), & \text{if } t_i\neq t_j\\0, & \text{if } t_i=t_j\end{cases} \tag{3}$$

只有当两个主题c_i和c_j中的词语在文档中的相对位置差较小，且语义相似性较大时，主题c_i和c_j之间边的权重才会较大。

3.3 句子图的构建

句子图是以文档中的句子作为图中的顶点。边的权重计算包括两个步骤，先使用 Sentence-BERT 模型对句子进行向量化，然后采用余弦相似度计算边的权重。构建的句子图采用邻接矩阵的形式进行存储，如式(4)所示：

$$(\boldsymbol{O}_{SS})_{ij}=\begin{cases}cosine(\boldsymbol{s}_i,\boldsymbol{s}_j), & \text{if } i\neq j\\0, & \text{if } i=j\end{cases} \tag{4}$$

其中，$\boldsymbol{O}_{SS}$表示的是文档d中句子与句子之间的关系矩阵。

3.4 词、主题和句子间边的构建

在完成了词图、主题图和句子图的构建之后，需要将 3 个子图连接起来组成一个整体的关系图。该过程主要包括词图和主题图、词图和句子图、主题图和句子图中的顶点连接 3 部分。其顶点间的权重均采用将各顶点所表示的词、主题或句子向量化之后，计算它们之间的余弦相似度的方法来获得。如计算候选关键词顶点和句子顶点间边的关系时，将得到的句子和主题之间的所有关系采用矩阵的形式表示为：

$$(\boldsymbol{O}_{ST})_{ij}=(c_i,s_j),c_i\in c,s_j\in s \tag{5}$$

其中，$\boldsymbol{O}_{ST}$表示句子对主题的影响矩阵，主题对句子的影响矩阵表示为$\boldsymbol{O}_{TS}$，且$\boldsymbol{O}_{ST}^{\mathrm{T}}=\boldsymbol{O}_{TS}$。通过类似的方法可以获得主题和候选关键词的矩阵$\boldsymbol{O}_{TC}=\boldsymbol{O}_{CT}^{\mathrm{T}}$，以及句子和主题的关系矩阵$\boldsymbol{O}_{ST}=\boldsymbol{O}_{TS}^{\mathrm{T}}$。

在完成了将文档表示成图的步骤后，图中顶点的所有关系可以表示为：

$$\boldsymbol{O}=\begin{bmatrix}\boldsymbol{O}_{CC} & \boldsymbol{O}_{CT} & \boldsymbol{O}_{CS}\\\boldsymbol{O}_{TC} & \boldsymbol{O}_{TT} & \boldsymbol{O}_{TS}\\\boldsymbol{O}_{SC} & \boldsymbol{O}_{ST} & \boldsymbol{O}_{SS}\end{bmatrix} \tag{6}$$

3.5 顶点的先验重要性计算

在完成了用图结构表示文本之后，对图中包含的候选关键词、主题和句子 3 种类型顶点的先验重要性进行评估。

(1)词图顶点的先验重要性计算

对于词图中的顶点的先验重要性计算，本文基于出现在文档越靠前的词语就越重要的假设进行。计算的方式参考文献[11]中提出的方法，如式(7)所示：

$$pw(c_i)=\sum_{w\in c_i}\sum_{i\in pos(w)}\frac{1}{i} \tag{7}$$

其中，w表示候选关键词c_i中包含的单个词，$pos(w)$表示词语w出现在文档d中的所有位置。该计算方法不仅考虑了词语出现在文档中的位置，同时考虑了词语的词频，能够较好地对词语的先验重要性进行区分。

(2)主题图顶点的先验重要性计算

本文采用了 3 种不同的指标来计算主题图顶点的先验得分，包括主题图中词的位置、词的 TF-IDF 值、主题与整个文档的相似值。单个词语的位置得分采用式(7)计算。计算主题t_i基于位置的得分时采用式(8)进行计算。

$$pc(t_i)=\sum_{c_i\in t}pw(c_i) \tag{8}$$

在计算词语的 TF-IDF 值时，本文采用文献[24]中的式(2)计算得到$tc(t_i)$。在计算主题t_i和所有主题t的相似性时，分别对t_i和t中的词进行连接并向量化，然后使用余弦相似度进行计算得到$ct(t_i)$。对于 3 种不同指标的得分使用式(9)归一化后，对主题的先验重要性进行总体评价，其公式如式(10)所示：

$$normalize(x)=\frac{x}{\|x\|_1} \tag{9}$$

$$pt(t_i)=\phi*pc(t_i)+\mu*tc(t_i)+\pi*ct(t_i) \tag{10}$$

其中，$\phi+\mu+\pi=1$。在实验过程中，ϕ的值取为 0.5，μ的值取为 0.2，π的值取为 0.3。

(3)句子图顶点的先验重要性计算

本文一共设计了 3 种统计特征来对句子在文档中的先验重要性进行度量，包括了句子在文档中的位置、句子的 TF-ISF 值、句子与文档整体的相似性。对于句子在文档中的位置得分，采用文献[24]中的式(1)来计算，句子s_i的位置得分可以表示为$sp(s_i)$；基于 TF-ISF 的句子打分采用文献[24]中的式(2)和式(3)进行计算，句子s_i的 TF-ISF 值为$ti(s_i)$；对于句子和文档整体的相似性，本文采用 Sentence-BERT 模型分别对句子和文档进行向量化后，计算它们之间的余弦相似度，并使用$sd(s_i)$进行表示。完成了对d中所有句子 3 个值的计算后，对得到的 $\boldsymbol{sp}$，$\boldsymbol{ti}$ 和 $\boldsymbol{sd}$ 3 个向量分别进行归一化。

对于句子s_i基于统计的先验重要性计算公式如式(11)所示：

$$ps(s_i)=\phi * \boldsymbol{sp}(s_i)+\mu * \boldsymbol{ti}(s_i)+\pi * \boldsymbol{sd}(s_i) \quad (11)$$

其中，$\phi+\mu+\pi=1$。在实验过程中，ϕ 的值取为 0.5，μ 的值取为 0.2，π 的值取为 0.3。

3.6 协同提取关键词和摘要

本节的主要内容是依据前面获得的图对文档中的候选关键词、主题和句子进行打分，然后根据打分的结果，选出关键词或者关键句子。打分的过程可以表示如下：

$$\boldsymbol{u}^t=\alpha[\theta \boldsymbol{O}_{SS}\boldsymbol{u}^{t-1}+(1-\theta)\boldsymbol{pu}]+\beta \boldsymbol{O}_{ST}\boldsymbol{v}^{t-1}+\gamma \boldsymbol{O}_{SC}\boldsymbol{w}^{t-1} \quad (12)$$

$$\boldsymbol{u}^t=\frac{\boldsymbol{u}^t}{\|\boldsymbol{u}\|_1} \quad (13)$$

$$\boldsymbol{v}^t=\alpha[\theta \boldsymbol{O}_{TT}\boldsymbol{v}^{t-1}+(1-\theta)\boldsymbol{pv}]+\beta \boldsymbol{O}_{TS}\boldsymbol{u}^t+\gamma \boldsymbol{O}_{TC}\boldsymbol{w}^{t-1} \quad (14)$$

$$\boldsymbol{v}^t=\frac{\boldsymbol{v}^t}{\|\boldsymbol{v}\|_1} \quad (15)$$

$$\boldsymbol{w}^t=\alpha[\theta \boldsymbol{O}_{CC}\boldsymbol{w}^{t-1}+(1-\theta)\boldsymbol{pw}]+\beta \boldsymbol{O}_{TS}\boldsymbol{u}^t+\gamma \boldsymbol{O}_{CT}\boldsymbol{v}^t \quad (16)$$

$$\boldsymbol{w}^t=\frac{\boldsymbol{w}^t}{\|\boldsymbol{w}\|_1} \quad (17)$$

其中，$\boldsymbol{u}^t$，$\boldsymbol{v}^t$和$\boldsymbol{w}^t$分别表示句子、主题和候选关键词在第 t 次迭代时的得分。参数 α，β 和 γ 满足 $\alpha+\beta+\gamma=1$，它们分别控制句子、主题和候选关键词的最终得分，受句子、主题和候选关键词的比例的影响。参数 $\theta\in[0,1]$表示阻尼系数，决定了算法进行随机跳转的概率，在实验中将 θ 的值取为 0.7。经过不断地迭代，最终使得 $\boldsymbol{u}$，$\boldsymbol{v}$ 和 $\boldsymbol{w}$ 收敛，获得了图中各顶点的最终得分。

在得到候选关键词、主题和句子的得分后，从文档中选择句子组成摘要，以及从候选关键词中选择关键词。在选择句子组成摘要时，采用 MMR[1] 算法，以减少摘要内容之间的冗余。对于关键词的选取，若直接按照得分从高到低的原则选取前 k 个得分高的候选关键词作为关键词，那么选取的关键词中将包含许多意思相近的词或短语，这是由 CSTRank 方法的特点所决定的，也就是与得分高的词语距离相近的词语也偏向于得分高。为了提高选择的关键词的准确率，减少关键词间的冗余，在进行关键词选择时，本文综合考虑了主题和关键词的得分，优先选择得分高的候选关键词，但在每个主题内只选择一个关键词，按照此方法依次选择关键词，直到选择的关键词数量符合条件为止。

3.7 算法描述与分析

本文提出的协同提取关键词和抽取摘要的方法如算法 1 所示。

算法 1 CTSRank 算法

输入：文档 $d=\{s_1,s_2,\cdots,s_n\}$

输出：摘要 $S=\{s_a,s_b,\cdots,s_x\}$和关键词 $kw=\{ck_1,ck_2,\cdots,ck_k\}$

1. 对文档进行预处理，选择出候选关键词集合 $ck=\{ck_1,ck_2,\cdots,ck_m\}$。
2. T=K-Means(ck)
3. $\boldsymbol{O}_{SS}$=build_sentence_graph(d)
4. $\boldsymbol{O}_{TT}$=build_topic_graph(T)
5. $\boldsymbol{O}_{CC}$=build_candidate_graph(ck)
6. $\boldsymbol{O}_{ST}=\boldsymbol{O}_{TS}^{T}$=bstg(d,T)
7. $\boldsymbol{O}_{SC}=\boldsymbol{O}_{CS}^{T}$=bscg(d,ck)
8. $\boldsymbol{O}_{TC}=\boldsymbol{O}_{CT}^{T}$=btcg(T,ck)
9. ps,pt,pc=cal_prior_score(d,T,ck)。
10. $\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}$=Rank($\boldsymbol{O}$,ps,pt,pc,$\alpha$,$\beta$,$\gamma$,$\theta$,$\eta$)
11. 利用最大边缘相关算法获取摘要 S。
12. kw=TopK(T,C,v,w)

在算法 1 中，第 1，2 行主要用于获得文档 d 的 t 个主题；第 3－5 行分别得到了句子间、主题间和候选关键词之间的关系矩阵；第 6－8 行分别构建了句子、主题和候选关键词三者之间的关系矩阵；第 9 行计算了图中各顶点的先验重要性；第 10 行利用迭代的方式获取候选关键词、主题和句子的得分；第 12 行根据主题和候选关键词的得分选择 k 个关键词。

下面对算法 1 的时间复杂度进行分析。假设文档在经过预处理之后剩下 n 个句子、t 个主题和 m 个候选关键词，对词、主题和句子向量化后的维度为 e，表示有 e 个不同的属性。对复杂度较高的几个步骤进行分析可以得到：在第 2 步使用 K-Means 进行聚类时，时间复杂度为 $T_1=\boldsymbol{O}(I_1 tme)$，其中 I_1 表示进行迭代的次数。第 3－8 步主要进行图模型的主体构建，其总体的时间复杂度为 $T_2=\boldsymbol{O}(n^2e+t^2e+m^2e+nte+nme+tme)$。在第 10 步中，假设进行迭代的最大次数为 I_2，那么 $T_3=\boldsymbol{O}((n^2+t^2+m^2+2nt+2nm+2tm)I_2e)$。综上所述，算法 1 的总体复杂度为 $T=T_1+T_2+T_3$。

4 实验设计与分析

本节首先对实验中所使用的数据集进行介绍，接着介绍实验结果的评价标准及对比方法，然后对实验的结果进行分析，最后探索实验中使用的参数对摘要抽取和关键词提取的影响。

4.1 实验数据集

为了验证本文方法在关键词提取和摘要抽取任务上的效果，实验过程中采用的实验数据来自 DUC(Document Understanding Conferences) 的整个 DUC2001 数据集[1)]，以及实验室整理的与审计相关的学术论文数据集 CNP。DUC2001 数据集的语言为英文，而 SCNP 数据集的语言为中文。CNP 数据集共包含三万多篇中文论文文档，为了方便实验，本实验从 CNP 数据集中选择了 150 篇中文论文作为被提取关键词和抽取摘要的对象(SCNP 数据集)。为了缩短原文的长度，选择了引言、结论，以及每节的首段和末段重新组成文档。对于关键词提取任务，采用了原论文给出的关键词作为参考关键词，通过统计发现每个文档大致包含 3～5 个词，经统计一共获得了 656 个关键词。对于摘要抽取任务，采用了原论文提供的摘要作为参考摘要。对于算法所抽取的摘要，实验中定义的最大长度为 150 个字。

4.2 评价指标及对比方法

本文评估关键词提取的质量采用了精确率(Precision)、召回率(Recall)、F-1 值 3 个评价指标。评估摘要抽取的质量

1) https://wwwnlpir.nist.gov/projects/duc/data/2001_data.html

采用了 ROUGE[29]（Recall-Oriented Understudy for Gisting Evalua-tion）中的 ROUGE-1，ROUGE-2 和 ROUGE-L 指标。

为了验证所提方法在关键词提取和摘要抽取任务上的效果，本文在实验过程中分别使用了相关方法在两个数据集上进行了对比实验。关键词提取的方法包括 TF-IDF，TextRank[2]，SingleRank[5]，ExpandRank[5]，IRRank[25]，TopicRank[14]，PositionRank[11]。其中 IRRank 方法是一种同时进行关键词提取和摘要抽取的方法。摘要抽取的方法有 Lead，Random，LexRank[3]，IRRank[25] 和 Co-Rank[26]。其中 Lead 方法是选取前 n 个词作为摘要；Random 方法是随机选取句子，直到超过最大长度限制；Co-Rank 是一种基于重要词语更可能出现在重要句子中，而重要句子更有可能包含重要词语的设定而提出的摘要抽取方法。

4.3 实验及分析

实验目的如下：1）验证本文提出的方法与对比方法在两个数据集上的关键词提取和摘要抽取的效果；2）探索主要参数对关键词提取和摘要抽取效果的影响。

（1）实验总体对比与分析

关键词提取任务，所有方法都抽取 10 个关键词。摘要抽取任务，对 DUC2001 数据集，设置摘要最大长度为 100 个词；对于 SCNP 数据集，设置最大摘要长度为 150 个字。在 DUC2001 数据集和 SCNP 数据集上，关键词提取和摘要抽取的结果对比如表 1—表 4 所列。

表 1 DUC2001 数据集上的关键词提取

Table 1 Keywords extraction on DUC2001 dataset

方法	*Precision*	*Recall*	*F*-1 值
TF-IDF	0.198	0.245	0.219
TextRank	0.236	0.289	0.260
SingleRank	0.247	0.303	0.272
ExpandRank	0.288	0.354	0.317
TopicRank	0.268	0.331	0.296
PositionRank	0.277	0.343	0.306
IRRank	0.296	0.366	0.327
CTSRank	0.305	0.378	0.338

表 2 SCNP 数据集上的关键词提取

Table 2 Keywords extraction on SCNP dataset

方法	*Precision*	*Recall*	*F*-1 值
TF-IDF	0.149	0.341	0.208
TextRank	0.165	0.378	0.230
SingleRank	0.179	0.410	0.249
TopicRank	0.186	0.425	0.258
PositionRank	0.196	0.448	0.273
IRRank	0.201	0.459	0.279
Our-Rank	0.211	0.482	0.293

表 3 DUC2001 数据集上的摘要抽取

Table 3 Summarization extraction on DUC2001 dataset

方法	*ROUGE*-1	*ROUGE*-2	*ROUGE-L*
Lead	0.4401	0.19494	0.3317
Random	0.37824	0.12062	0.2549
LexRank	0.43604	0.19213	0.3125
IRRank	0.43685	0.18534	0.3184
Co-Rank	0.45048	0.20034	0.3403
CTSRank	0.45269	0.20460	0.3448

表 4 SCNP 数据集上的摘要抽取

Table 4 Summarization extraction on SCNP dataset

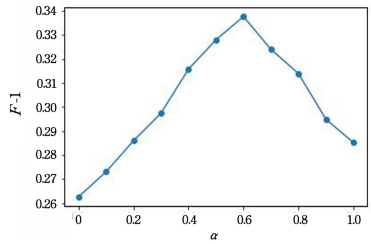
方法	*ROUGE*-1	*ROUGE*-2	*ROUGE-L*
Lead	0.18845	0.10989	0.14867
Random	0.13632	0.0788	0.10176
LexRank	0.18623	0.10735	0.14284
IRRank	0.19893	0.11693	0.15913
Co-Rank	0.20024	0.11745	0.16003
CTSRank	0.20132	0.11868	0.16204

从 4 个表中的结果可以发现，不管是在中文还是英文数据集上，本文提出的方法在摘要抽取和关键词提取任务上都优于其他对比方法。其中，使用 CTSRank 方法进行关键词提取时，其 *F*-1 值在 DUC2001 数据集上为 0.338，在 SCNP 数据集上为 0.293；在进行摘要抽取时，CTSRank 方法在 DUC2001 和 SCNP 数据集上获得的 ROUGE-1 值分别为 0.45269 和 0.20312。CTSRank 方法在两个数据集上都取得了良好的效果，表明本文提出的方法不依赖于语言类型，实际上，CTSRank 是完全无监督的，因此可以将其推广至任何语言的关键词提取和摘要抽取任务上。DUC2001 数据集属于新闻类型数据集，SCNP 属于论文数据集，这也验证了本文提出的方法对不同类型的文档具有较好的适应性。在关键词提取任务中，TextRank 和 SingleRank 仅仅采用了词语间的共现关系对词语进行打分；TopicRank 方法在提取关键词时，偏向于对文档所包含主题的覆盖；PositionRank 方法将词出现在文档中的位置信息作为图中顶点的先验重要性；IRRank 方法则利用词、句子之间的相互增强关系进行关键词提取。上述 5 种方法在进行关键词提取时都具有各自的优势，而本文提出的 CTSRank 方法将这些优势都融入同一个图中，不仅融入了词语之间的位置和语义关系，以及词、主题和句子的统计信息，还利用了不同粒度之间的相似性关系，因此 CTSRank 方法能够更好地对词语的重要性进行评价，特别是在选取关键词时，考虑了主题信息，使得选取的关键词能够很好地覆盖文档的主题信息。在摘要抽取任务中，Lead 方法较 Random 和 LexRank 方法取得了较好的结果，这表明句子出现在文档中的位置在摘要抽取任务中起着重要的指示性作用。Co-Rank 方法和 IRRank 方法均是基于重要句子包含重要词语，且重要词语组成重要句子的假设提出的，其同时对词和句子的重要性进行排序。本文提出的 CTSRank 利用了类似假设，同时融合了主题信息，使得在对句子进行评价时，选择的句子能够对原文档的内容具有更好的覆盖性。与 Co-Rank 和 IRRank 方法不同的是，CTSRank 使用的是完全图，能够更全面地利用词语、主题、句子之间的关系对句子的重要性进行评价。但是，CSTRank 方法存在模型的时间复杂度和空间复杂度均较高的缺陷。

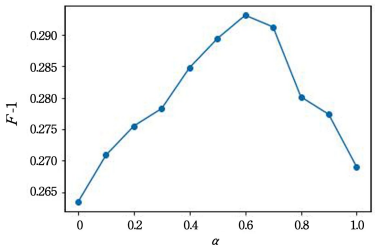
4.4 参数分析

下面对所提方法中使用的参数 α，β 和 γ 对关键词提取和摘要抽取的影响进行分析。α，β 和 γ 的关系满足 $\alpha+\beta+\gamma=1$，因此本节对 α 从 0 至 1，每隔 0.1 进行取值。在实验中，对 β 和 γ 的取值相等。图 2(a)和图 2(b)分别是随着 α 的变化，在 DUC2001 和 SCNP 数据集中进行关键词提取时 F-1 值变化的趋势图。图 3(a)和图 3(b)是随着 α 的变化，在 DUC2001

和 SCNP 数据集中进行摘要抽取时 ROUGE-1 变化的趋势图。



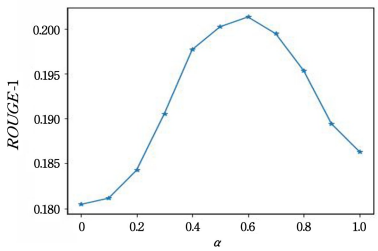
(a)DUC2001 dataset



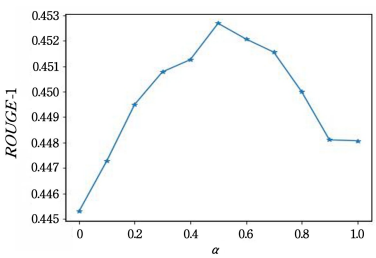
(b)SCNP dataset

图 2 *F*-1 值随着 α 的变化趋势

Fig. 2 *F*-1 values change with α



(a)DUC2001 dataset



(b)SCNP dataset

图 3 *ROUGE*-1 值随着 α 的变化趋势

Fig. 3 *ROUGE*-1 values change with α

当 α 取值为 1 时，对于关键词提取任务，CTSRank 没有融入主题和句子对候选关键词重要性的影响，候选关键词的得分完全是基于候选关键词之间关系和词语的位置特征确定的；对于摘要抽取任务，CTSRank 没有融合词和主题对句重要性的影响，句子的最终得分完全是基于句子之间的关系和句子的统计先验重要性确定的。当 α 的取值为 0 时，对于关键词提取任务，关键词的得分完全由主题和句子对其的评价而获得；对于摘要抽取任务，句子的得分完全由候选关键词和主题进行确定。可以发现，当 α 的取值为 1 时关键词提取和摘要抽取的效果都要优于 α 取值为 0 时的效果，可以猜测其原因分别是利用词之间的关系和统计特征对词的重要性评价要比利用句子和候选关键词的评价更为准确；利用句子之间的关系和统计特征对句子的评价要比利用候选关键词和主题对句子的评价更准确。

从图 2 和图 3 中可以发现，随着 α 值的增加，摘要和关键词的提取效果在相应的评价指标下都呈现上升的趋势，而且在 α 值为 0.5～0.7 的区间内都取得了最佳效果。这一现象反映了当利用词或句子的内部关系以及其各自的统计特征对词或句子进行评价占主要比例时，会使得关键词提取和摘要抽取的效果提升。当 α 值超过 0.7 后，效果呈现下降趋势，表明融入适当比例的主题和句子信息对关键词提取的效果有促进作用。同时，融入适当比例的主题和关键词信息对摘要提取的效果有促进作用。与此同时，还可以发现抽取的关键词和摘要的质量在评价指标下的变化趋势相似，这表明了 CTSRank 方法对文档建模的方式确实对词语和句子重要性评价有着促进效果。

结束语 本文提出了一种利用文档中词、主题和句子为顶点建立图对文本进行表示的方法，并且对图中的各顶点设计了先验重要性评价方法，然后利用了迭代的方式求解词、主题和句子的得分，最后根据得分同时选出了关键词和摘要。该方法是一种完全无监督学习方法，不需要依靠任何标注语料，其在中英文数据集上的关键词提取和摘要抽取任务都取得了不错的效果。在将来的工作中，我们将改进词语、主题和句子的表示方式，使得语义距离的计算更加准确。此外，我们还将简化算法，在保证算法效果的前提下，降低算法的时间复杂度。

参 考 文 献

[1] CARBONELL J, GOLDSTEIN J. The use of MMR, diversity-based reranking for reordering documents and producing summaries[C]// Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1998: 335-336.

[2] PAGE L, BRIN S, MOTWANI R, et al. The PageRank citation ranking: bringing order to the web[R]. Stanford InfoLab, 1999.

[3] MIHALCEA R, TARAU P. Textrank: bringing order into text [C]// Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. 2004: 404-411.

[4] ERKAN G, RADEV D R. Lexrank: graph-based lexical centrality as salience in text summarization[J]. Journal of Artificial Intelligence Research, 2004, 22: 457-479.

[5] WAN X, XIAO J. Exploiting neighborhood knowledge for single document summarization and keyphrase extraction[J]. ACM Transactions on Information Systems (TOIS), 2010, 28(2): 1-34.

[6] GOLLAPALLI S D, CARAGEA C. Extracting keyphrases from research papers using citation networks[C]// Twenty-Eighth AAAI Conference on Artificial Intelligence. 2014.

[7] YU Y, NG V. Wikirank: improving keyphrase extraction based on background knowledge[J]. arXiv: 1803.09000, 2018.

[8] WANG R, LIU W, MCDONALD C. Corpus-independent generic keyphrase extraction using word embedding vectors[C]// Software Engineering Research Conference. 2014: 1-8.

[9] WANG H, YE J, YU Z, et al. Unsupervised keyword extraction

methods based on a word graph network[J]. International Journal of Ambient Computing and Intelligence (IJACI), 2020, 11(2): 68-79.

[10] LIU Z, HUANG W, ZHENG Y, et al. Automatic keyphrase extraction via topic decomposition[C]// Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. 2010: 366-376.

[11] FLORESCU C, CARAGEA C. Positionrank: an unsupervised approach to keyphrase extraction from scholarly documents [C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017: 1105-1115.

[12] TENEVA N, CHENG W. Salience rank: efficient keyphrase extraction with topic modeling[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2017: 530-535.

[13] BISWAS S K, BORDOLOI M, SHREYA J. A graph based keyword extraction model using collective node weight[J]. Expert Systems with Applications, 2018, 97: 51-59.

[14] BOUGOUIN A, BOUDIN F, DAILLE B. Topicrank: graph-based topic ranking for keyphrase extraction[C]// International Joint Conference on Natural Language Processing (IJCNLP). 2013: 543-551.

[15] AL-KHASSAWNEH Y A, SALIM N, JARRAH M. Improving triangle-graph based text summarization using hybrid similarity function[J]. Indian Journal of Science and Technology, 2017, 10(8): 1-15.

[16] GOYAL P, BEHERA L, MCGINNITY T M. A context-based word indexing model for document summarization[J]. IEEE Transactions on Knowledge and Data Engineering, 2012, 25(8): 1693-1705.

[17] RAMESH A, SRINIVASA K G, PRAMOD N. SentenceRank—A graph based approach to summarize text[C]// The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014). IEEE, 2014: 177-182.

[18] SANKARASUBRAMANIAM Y, RAMANATHAN K, GHOSH S. Text summarization using Wikipedia[J]. Information Processing & Management, 2014, 50(3): 443-461.

[19] CHENGZHANG X, DAN L. Chinese text summarization algorithm based on word2vec[J]. Journal of Physics: Conference Series, 2018, 976(1): 012006.

[20] ROUANE O, BELHADEF H, BOUAKKAZ M. Word Embedding-Based Biomedical Text Summarization[C]// International Conference of Reliable Information and Communication Technology. Cham: Springer, 2019: 288-297.

[21] YANG K, AL-SABAHI K, XIANG Y, et al. An integrated graph model for document summarization[J]. Information, 2018, 9(9): 232.

[22] ERKAN G. Using biased random walks for focused summarization[C]// Proceedings of the 2006 Document Understanding Conference held at the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. 2006.

[23] OTTERBACHER J, ERKAN G, RADEV D. Using random walks for question-focused sentence retrieval[C]// Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. 2005: 915-922.

[24] MAO X, YANG H, HUANG S, et al. Extractive summarization using supervised and unsupervised learning[J]. Expert Systems with Applications, 2019, 133: 173-181.

[25] WAN X, YANG J, XIAO J. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction[C]// Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. 2007: 552-559.

[26] FANG C, MU D, DENG Z, et al. Word-sentence co-ranking for automatic extractive text summarization[J]. Expert Systems with Applications, 2017, 72: 189-195.

[27] MAO X, HUANG S, LI R, et al. Automatic Keywords Extraction Based on Co-Occurrence and Semantic Relationships Between Words[J]. IEEE Access, 2020, 8: 117528-117538.

[28] REIMERS N, GUREVYCH I. Sentence-bert: Sentence embeddings using siamese bert-networks[J]. arXiv: 1908. 10084, 2019.

[29] LIN C Y. Rouge: a package for automatic evaluation of summaries[C]// Text Summarization Branches Out. 2004: 74-81.

MAO Xiang-ke, born in 1992, Ph.D. His main research interests include natural language processing and machine learning.

HUANG Shao-bin, born in 1965, professor. His main research interests include data mining, natural language processing and machine learning.