利用整数线性规划自动抽取多样性关键短语

李珊珊 陈 黎 唐裕婷 王艺霖 于中华
(四川大学计算机学院 成都 610065)

摘 要 关键短语是文本信息的精简概括,能够代表文本的主题和核心观点。而关键短语的自动抽取更是自然语言处理和信息检索的重要任务之一。针对目前无监督方法自动抽取关键短语存在过度生成候选短语语义的问题,提出了一种将整数线性规划和短语语义相似度相结合的自动抽取算法。通过惩罚语义相似度高的候选短语实现目标函数的最大化,以此形成多样性的关键短语。实验利用 TextRank 和 TFIDF 算法在两种不同的语料集中分别产生候选短语,并利用提出的优化算法对候选短语的权值得分进行优化。最后将所提算法产生的优化结果与现有多个算法的结果进行了比较。实验结果表明,通过加入相似性度量的惩罚能够有效解决语义过度问题,并获取更多样的关键短语,其优化结果的 P,R 和 F 值均高于其他算法。

关键词 关键短语自动抽取,整数线性规划,语义过度生成,多样性
中图法分类号 TP309 **文献标识码** A

Automatic Extraction of Diversity Keyphrase by Utilizing Integer Liner Programming

LI Shan-shan CHEN Li TANG Yu-ting WANG Yi-lin YU Zhong-hua
(College of Computer Science, Sichuan University, Chengdu 610065, China)

Abstract Keyphrases are the concise summary of text information, which can represent the main topics and the core ideas of texts. And the automatic extraction of key phrases is one of the important tasks for natural language processing and information retrieval. Aiming at the existing problem caused by semantic over-generation on candidate phrases with unsupervised method, this paper proposed an algorithm for automatic extraction of keyphrase by using integer linear programming (ILP) and similarity of candidate phrases, in which candidate phrases with high sematic similarity are punished for maximizing the object function to obtain diversified keyphrases. TextRand and TFIDF algorithms are applied in the proposed method to create candidate phrases based on two different corpus sets and the proposed optimization algorithm is utilized to optimize the weight scores of candidate phrases. Finally, the results of the proposed optimization algorithm is compared with the ones of baseline methods, and the experimental results show that the proposed method can solve the semantic over-generation problem effectively by punishing candidate phrases with high semantic similarity. Moreover, the optimization algorithm can obtain more diverse keyphrases and the optimized results of P, R and F value outperform the ones of baseline methods.

Keywords Automatic keyphrase extraction, Integer liner programming, Semantic over-generation, Diversity

1 引言

关键短语是文本信息的精简概括,能够代表文本的主题和核心观点[1]。关键短语的自动抽取是在自由文本中自动抽取能够代表文本主题的短语,也是自然语言处理和信息检索的基本任务之一,关键短语自动抽取有助于文本摘要、文本分类、观点挖掘、文本索引、文本聚类等[2]下游任务。

目前利用无监督方法[3-6]进行关键短语抽取的研究主要是根据候选短语的得分进行排序,在计算候选短语得分时常对构成短语的各个词的得分进行累加,这样会导致短语中词本身的过度生成错误。词本身的过度生成问题指当利用模型计算得到词 A 的权值得分很高时,如果利用形成候选短语的各个词的权值得分累加和作为候选短语权值的得分,就会导致包含词 A 的所有候选短语的权值得分都很高,如与 A 组合形成的候选短语 AB,AC,AD 等的权值得分高,那么最终根据权值得分排名选取前几个候选短语形成关键短语集合时,会得到类似{AB,AC,AD,…}的关键短语集合。这个集合包含大量 A 词组成的关键短语,而真正需要的关键短语集合可能只需要一个包含 A 的关键短语 AB。因此,文献[13]提出利用整数线性解决短语中词本身过度生成的问题,但是在自动抽取的短语集合中仍然存在语义相似的关键短语。如果候选短语中存在大量的相似短语,则称这样的现象为语义过度生成问题。例如在同一个主题下的"Olympics games"和"Olympics"、"peer to peer"和"p2p"作为关键短语都被抽取出

本文受四川省科技支撑项目(2014GZ0063),四川省重点研发项目(2018GZ0182)资助。
李珊珊(1989-),女,硕士,主要研究方向为自然语言处理;**陈 黎**(1977-),女,讲师,主要研究方向为自然语言处理;**唐裕婷**(1994-),女,硕士生,主要研究方向为自然语言处理;**王艺霖**(1995-),男,硕士生,主要研究方向为自然语言处理;**于中华**(1967-),男,副教授,主要研究方向为自然语言处理,E-mail:yuzhonghua@scu.edu.cn(通信作者)。

来，但是这两个候选短语在语义层面所表达的含义几乎是一样的，这将不利于关键短语的多样性。

为此，本文受文献[13]的启发，提出了一种在整数线性规划中加入关键短语的相似性度量的关键短语自动抽取方法。该方法不仅可以减少关键短语集合中词本身的过度生成错误，同时还能解决候选短语的语义过度生成问题，从而获取多样性的关键短语。

2　相关工作

19 世纪 50 年代在 Luhn 提出自动标引后，关键短语自动抽取研究任务随之到来。关键短语自动抽取方法主要分为有监督方法和无监督方法[7]。有监督方法[8-11]是把关键短语自动抽取作为一个二分类问题，判断在文本中形成的候选短语是否为关键短语。虽然这类方法能有效利用文本特征并取得了较好的实验结果，但是训练分类器模型往往需要大量的标注语料，而这种带有标注的关键短语在大部分领域的语料中很少，这使得关键短语自动抽取的有监督方法具有一定局限性，因此学者们提出了关键短语自动抽取的无监督方法。无监督方法是利用算法对形成的候选短语计算得分，最终自动抽取得分前 N 的候选短语作为关键短语。

无监督方法最初大多数是基于图的方法，把文本中的词看作图中的结点，然后计算词的权值得分，最后将包含在候选短语里的所有词的权值得分相加作为候选短语的权值得分。Mihalcea 等[11]提出 TextRank 算法来计算每个词的权值，并使用候选短语中各个词的权值加和结果作为该候选短语最终的权值得分，选取权值得分前 n 的候选短语作为关键短语。Wan[12]对文献[11]进行改进，首先利用与待抽取文本相似的 k 个文本聚成簇，再对待抽取文本以及簇分别构造图的方法进行关键短语抽取，而图中边的权重不再是文献[11]中的词在文本中的共现次数，而是在簇中出现的次数。Florescu[6]对 PageRank 进行了改进，在文献[11]的基础上加入了位置信息，即在图中由一个结点跳转到另外一个结点的可能性不再是一个平均的概率，而是这个词在文本中出现的所有位置的倒数求和。

这些研究中将短语中包含的各个词的权值得分相加作为候选短语权值得分，会产生过度生成问题，即词本身的过度生成问题[12]。基于这个问题，文献[13]提出了利用整数线性规划解决关键短语自动抽取的词本身过度生成问题的模型。该模型首先计算每个词的权值得分，然后利用整数线性规划对目标函数进行优化，约束每个词的权值得分仅且只能被加到目标函数中一次或者零次，这种优化模型在一定程度上解决了文献[12]提出的词本身的过度生成问题。但是，文献[13]仍然没有考虑语义过度生成问题。为此，本文提出将整数线性规划和短语相似度量相结合的方法解决关键短语抽取中存在的词过度生成以及语义过度生成问题。

3　多样性短语的整数线性规划算法(DPILP)

语义过度生成问题，即自动抽取的关键短语集合中存在语义相似的关键短语。这种语义过度生成的问题会占据关键短语自动抽取任务整体错误的 8%到 12%。为了解决关键短语自动抽取任务中的语义过度生成问题，本文提出在整数线性规划目标函数里加入候选短语之间语义相似度的度量方法，使得自动抽取的关键短语语义上区别更大，能更好地概括文本的主题和内容。本文提出的 DPILP 算法首先计算文本中每个词的权值得分(分别使用 TextRank 和 TFIDF)，然后利用改进的整数线性规划模型进行优化，得到每篇文本的关键短语集合。

3.1　数据集和预处理

本文使用 Hulth2003 和 SemEval2010 作为语料集合。在预处理中，首先去掉文本中特殊的字符，将大写变为小写，然后分句、分词并进行词性标注。使用文献[16]提出的词性模板形成候选短语，词性模板为："(JJ|JJR|JJS|VBG|VBN)*(NN|NNS|NNP|NNPS|VBG)+"。

3.2　计算词权值

在计算词权值时，我们分别使用了两种不同的方法来获取词的权值得分。

3.2.1　TextRank

TextRank 是一种基于图模型的算法，图的结点是文本中的每个词，设定一定的窗口大小，认为出现在固定窗口大小的词是共现的，则这些共现作为词与词在图中连接的无向边，TextRank 算法通过迭代计算候选短语的权值，迭代公式如式(1)所示：

$$WS(V_i)=(1-d)+d\times\sum_{V_j\in In(V_i)}\frac{w_{ji}}{\sum_{V_k\in Out(V_j)}w_{jk}}WS(V_j) \quad (1)$$

其中，d 是衰减因子，设置为 0.85；V 代表文本中词的集合；V_i 是文本中第 i 个词；$In(V_i)$ 表示图中指向第 i 个词的结点(词)的集合；$Out(V_j)$ 表示以第 j 个词作为顶点的结点(词)集合；w_{jk} 表示词 V_k 与词 V_j 之间边的权重；$WS(V_i)$ 是文本中第 i 个词的权值。

3.2.2　TFIDF

在逆向文档频率(Inverse Document Frequency，IDF)中，如果训练语料中包含词 i 的文档越少，那么词 i 的 IDF 值越大，表示词 i 对文本之间的区别度就越大，则词 i 作为关键词的可能性就越大，IDF 的计算如式(2)所示：

$$idf_i=\log\frac{N}{df_i} \quad (2)$$

其中，N 为训练语料集中文档的总数；df_i 为包含词 i 的文档数。

词逆向文档频率(Term Frequency Inverse Document Frequence，TFIDF)既考虑了词 i 在第 j 个文档出现的频率，又考虑了词 i 的逆文档频率，其公式如式(3)所示：

$$W_{ij}=\frac{f_{ij}}{|d_j|}\times\log\frac{N}{df_i} \quad (3)$$

其中，f_{ij} 为词 i 在文档 d_j 中出现的次数；$|d_j|$ 为文档 d_j 包含词的总数；N 是训练语料集中文档的总数；df_i 为包含词 i 的文档数。

如果一个词在一个文档中出现频率很高而在整个语料集中包含该词的文档很少，则该词的 TFIDF 值较高，表示该词对于该文档较重要。TFIDF 使那些在整个语料集中具有区分度而在本身文档中出现频率高的词具有较高的得分。

3.3　计算候选短语相似度

DPILP 算法利用 word2vec 计算候选短语之间的相似度，以解决关键短语语义过度生成的问题。本文计算候选短语之间语义相似度的方法步骤如下：

(1)利用 word2vec 训练词向量模型。

(2)候选短语的向量表示:将候选短语包含词的词向量相加得到候选短语的向量表示。

(3)利用候选短语向量之间的夹角余弦值计算候选短语之间的语义相似度,如式(4)所示:

$$psim(p_1,p_2)=\frac{vecp_1 \cdot vecp_2}{\| vecp_1 \| \cdot \| vecp_2 \|} \tag{4}$$

其中,$vecp_1$ 是候选短语 p_1 的向量表示,$vecp_2$ 是候选短语 p_2 的向量表示。式(4)利用夹角余弦值计算候选短语 p_1 和候选短语 p_2 之间在语义层面的相似度。相似度得分越高,代表两个候选短语之间的语义相似度越大。本文为了抽取能够代表文本内容更多主题的关键短语,尽量选取语义相似度小或者在语义层面差别大的候选短语组合形成关键短语集合。

3.4 多样性短语的整数线性规划

本节利用整数线性规划模型,减少词本身的过度生成和短语之间语义的过度生成错误,目标函数和约束条件如下所示:

$$\max \sum_i w_i x_i-\lambda\sum_j\frac{(l_j-l)c_j}{1+substr_j}-\mu\sum_{m<n}y_{mn}rel_{mn} \tag{5}$$

subject to:

$$\sum_j c_j \leqslant N \tag{6}$$

$$c_j Occ_{ij} \leqslant x_i,\ \forall i,j \tag{7}$$

$$\sum c_j Occ_i j \geqslant x_i,\ \forall i,j \tag{8}$$

$$\begin{cases} c_m+c_n-1\leqslant y_{mn} \\ y_{mn}\leqslant c_m \\ y_{mn}\leqslant c_n \end{cases},\ \forall m<n \tag{9}$$

$$\begin{cases} x_i\in\{0,1\}, & \forall\ i \\ c_j,m,n\in\{0,1\}, & \forall j,m,n \\ y_{mn}\in\{0,1\}, & \forall m<n \end{cases} \tag{10}$$

其中,w_i 表示利用 TextRank 或者 TFIDF 计算得到的词 i 的权值得分;$x_i,c_j,c_m,c_n,y_{mn}\in\{0,1\}$;$\lambda,\mu$ 是超参数;l_j 是候选短语 j 包含词的个数;$substr_j$ 是候选短语 j 在其他候选短语中作为子字符串的个数;rel_{mn} 是候选短语 c_m 和候选短语 c_n 利用余弦相似度计算的语义相似度值;$Occ_{ij}\in\{0,1\}$为指示变量,当词 i 存在于候选短语 j 时,Occ_{ij} 为 1,当词 i 不在候选短语 j 中时,Occ_{ij} 为 0。

DPILP 模型中,x_i,c_j,c_m,c_n 以及 y_{mn} 是要学习的参数,目的是最大化目标函数。目标函数中的第一部分是把所有被选为关键短语的词的权值得分相加,因为最长的候选短语总能够使目标函数的第一个求和部分达到最大,因此引入目标函数的第二部分,目的是避免模型总是选择最长的候选短语,对包含超过两个词的候选短语进行惩罚,同时惩罚经常作为其他候选短语子字符串的候选短语。

第一个约束条件如式(6)所示,表示要抽取的关键短语个数,当 $N=5$ 时,目标函数最多自动抽取 5 个候选短语作为关键短语;第二个约束条件(见式(7))与第三个约束条件(见式(8))共同约束词 i 的权值在目标函数中仅且只能被加一次或者零次。如果自动抽取的关键短语集合中包含词 i,那么词 i 的权值只能加到目标函数中一次;如果自动抽取的关键短语集合中不包含词 i,那么词 i 的权值不被加到目标函数中。

为了解决语义过度生成的问题,本文在目标函数加入 $\mu\sum_{m<n}y_{mn}rel_{mn}$,当候选短语 m 与候选短语 n 同时被选为关键短语时,这个惩罚因子将起作用,惩罚项是候选短语 m 和候选短语 n 利用余弦相似度计算的语义相似度。当 $y_{mn}=1$ 时,表示候选短语 m 与候选短语 n 同时被选中,当候选短语 m 与候选短语 n 其中一个没被选中或者两个都没选中时,$y_{mn}=0$,y_{mn} 是由 c_m 与 c_n 共同约束的。下面分析式(9)中 3 个约束条件是如何约束 y_{mn} 的。

(1)当候选短语 m 与候选短语 n 都没被选中时,即 $c_m=0$ 并且 $c_n=0$,约束条件为:$0+0-1\leqslant y_{mn}$,$y_{mn}\leqslant 0$,$y_{mn}\leqslant 0$,则 $y_{mn}=0$。

(2)当候选短语 m 与候选短语 n 有一个被选中时,若 $c_m=1$ 并且 $c_n=0$,约束条件为:$1+0-1\leqslant y_{mn}$,$y_{mn}\leqslant 1$,$y_{mn}\leqslant 0$,则 $y_{mn}=0$。

(3)当候选短语 m 与候选短语 n 同时被选中时,即 $c_m=1$ 并且 $c_n=1$,约束条件为:$1+1-1\leqslant y_{mn}$,$y_{mn}\leqslant 1$,$y_{mn}\leqslant 1$,则 $y_{mn}=1$。

综上所述,可以通过约束条件在目标函数中惩罚语义相似的候选短语,使得能够 提出的关键短语算法既解决了词本身的过度生成问题,又避免了形成的关键短语集合中存在大量语义相似的关键短语。DPILP 模型自动抽取的关键短语集合能够代表文本的更多主题,在语义层面对文本有更好的概括性。

4 实验设计及分析

4.1 实验数据

Hulth2003[14]是用作关键短语自动抽取算法评价的英文数据集,它是从全球著名的科技文摘数据库 INSPEC 收集的语料库。Hulth2003 数据集包括 1000 篇含有关键短语的文本摘要训练集、500 篇含有关键短语的测试集以及 500 篇含有标注关键短语的验证集。该数据集已经被广泛应用在关键短语自动抽取研究任务中。本文利用两种无监督方法对文本中词的权值进行计算,使用 Hulth2003 训练集中的 1000 篇文本作为本文的实验数据。

SemEval2010[15]是从 ACM 数据图书馆收集的英文语料库,包括会议和研讨会论文 250 篇。该语料库包含 ACM 的 4 种文本类型:C2.4(分布式系统)、H3.3(信息搜索和检索)、I2.11(分布式人工智能)以及 J4(社会和行为科学),并且每篇文本都包含了作者标注的关键短语。SemEval2010 语料集被广泛应用在关键短语自动抽取研究任务中。本文使用 SemEval2010 数据集验证提出的整数线性规划模型的有效性。

4.2 实验设计

词权值得分使用两种无监督的方法:TextRank 和 TFIDF。本文设置 TextRank 窗口大小为 5,利用 TextRank 迭代公式计算词的权值得分。

本文采用了 Google 提供的词向量,该词向量是基于 1000 亿条新闻数据利用 word2vec 进行训练得到的,结果包含 300 万个词且每个词表示为一个 300 维向量。候选短语向量由候选短语包含的词的词向量相加得到。

在实验中利用 Matlab 进行整数线性规划时,有多个参数需要人工设置,包括整数线性规划目标函数中的 λ 和 μ。对于超参数的设置本文进行了多组实验,首先在整数线性优化

问题中令 μ 等于 0，不考虑语义过度生成问题，找到使得文献[13]的实验结果达到最好的 λ，然后在最好的 λ 的条件下，比较不同的 μ 对提出的模型 DPILP 实验结果的影响，并从不同 μ 的多个实验结果中找到最好的 μ 值，使得能够自动抽取最后的关键短语集合。

在利用模型得到每篇关键短语的集合后，因在英文语料中，同一个词可能表现为不同形态，为了消除由于这一因素使得模型抽取的关键短语与作者标注的关键短语不同，本文在得到模型抽取的关键短语集合之后，使用 NLTK 自然语言处理包对实验抽取的关键短语以及作者标注的关键短语进行取词根的后处理工作。

4.3　评价指标

在关键短语自动抽取研究任务中，通常采用精确率(P)、召回率(R)和 F 值作为模型实验结果的评价指标。在关键短语自动抽取任务中，精确率是指利用模型自动抽取的关键短语集合中正确的关键短语的比例；召回率是指所有作者标注的关键短语集合中，被模型正确抽取的关键短语的比例；F 值是对精确率和召回率的调和平均值，是对其他两个指标的综合考量。精确率、召回率、F 值 3 个指标的计算如式(11)—式(13)所示：

$$P=\frac{C_{\text{correct}}}{C_{\text{extract}}} \tag{11}$$

$$R=\frac{C_{\text{correct}}}{C_{\text{standard}}} \tag{12}$$

$$F=\frac{2\times P\times R}{P+R} \tag{13}$$

其中，C_{correct} 是模型抽取的正确的关键短语总数；C_{extract} 是模型自动抽取的关键短语的总数；C_{standard} 是人工标注的关键短语的总数。

4.4　实验对比方法

将 DPILP 算法分别在 Hulth2003 和 SemEval2010 数据集上进行实验，并与下面的 baseline 方法进行对比。

(1)SUM：利用 TextRank 和 TFIDF 方法计算得到每篇文档中词的权值得分，候选短语的权值得分即将候选短语包含的词的权值得分相加，最终选取每篇文档中权值得分前 N 的候选短语作为关键短语。

(2)AVG：利用 TextRank 和 TFIDF 方法计算得到每篇文本中词的权值得分，候选短语的权值得分即将候选短语包含的词的权值得分相加，再除以候选短语包含的词的个数，最终选取每篇文档中权值得分前 N 的候选短语作为关键短语。

(3) ILP2015：即文献[13]的方法，利用 TextRank 和 TFIDF 方法计算得到每篇文档中词的权值得分，然后利用减少词本身过度生成错误的整数线性规划模型，为每篇文档优化得到 N 个关键短语。

(4)DPILP：本文提出的减少语义过度生成错误的整数线性规划算法，为每篇文档优化得到 N 个关键短语。

4.5　实验结果与分析

4.5.1　TextRank 方法实验

首先利用 TextRank 方法计算出文本中词的权值得分，得到 SUM 和 AVG 方法的结果。实验中，ILP2015 方法在 $\lambda=0.04$ 时取得最好的实验结果，因此在整数线性规划模型 DPILP 中，令 $\lambda=0.04$ 作为固定的参数，然后 μ 分别设置为 0.02，0.04，0.06，0.08，0.1 进行多组实验，得到 DPILP 最优的结果。各种方法在两个数据集上自动抽取 5 个和 10 个关键短语的实验结果对比如表 1、表 2 所列。

表 1　TextRank 自动抽取 Top_5 关键短语的实验对比

(单位：%)

方法	Hulth2003			SemEval2010		
	P	*R*	*F*	*P*	*R*	*F*
SUM	23.30	12.50	16.27	10.00	7.38	8.50
AVG	26.56	14.38	18.66	18.72	10.14	13.15
ILP2015	30.23	17.41	22.10	22.06	11.72	15.31
DPILP	**34.14**	**19.02**	**24.43**	**24.45**	**13.42**	**17.33**

表 2　TextRank 自动抽取 Top_10 关键短语的实验对比

(单位：%)

方法	Hulth2003			SemEval2010		
	P	*R*	*F*	*P*	*R*	*F*
SUM	21.61	15.32	17.92	9.60	8.58	9.06
AVG	23.15	18.27	20.42	16.12	12.45	14.05
ILP2015	26.80	20.71	23.36	19.23	15.67	17.27
DPILP	**29.49**	**22.93**	**25.80**	**21.85**	**17.08**	**19.17**

实验结果表明，本文提出的模型 DPILP 在两个数据集上比 ILP2015、SUM 和 AVG 方法都有很大的提高。在自动抽取 5 个和 10 个关键短语实验中，本文方法分别在 μ 为 0.06 和0.04 时取得最好的结果，F 值有将近 3%的提高，验证了本文所提方法的有效性。

4.5.2　TFIDF 方法实验

同样利用 TFIDF 方法计算出文本中词的权值得分，得到 SUM 和 AVG 方法的结果。实验中，ILP2015 方法在 $\lambda=0.02$ 时取得最好的实验结果，因此在整数线性规划模型 DPILP 中，令 $\lambda=0.04$ 作为固定的参数，然后 μ 分别设置为 0.02，0.04，0.06，0.08，0.1 进行多组实验，得到 DPILP 模型最优的结果。各种方法在两个数据集上自动抽取 5 个和 10 个关键短语的实验结果对比如表 3、表 4 所列。

表 3　TFIDF 自动抽取 Top_5 关键短语的实验对比

(单位：%)

方法	Hulth2003			SemEval2010		
	P	*R*	*F*	*P*	*R*	*F*
SUM	23.83	12.72	16.59	11.23	7.91	9.28
AVG	27.36	14.57	19.01	19.68	10.45	13.65
ILP2015	32.22	18.14	23.21	24.37	12.23	16.29
DPILP	**35.85**	**19.92**	**25.61**	**27.13**	**13.91**	**18.39**

表 4　TFIDF 自动抽取 Top_10 个关键短语的实验对比

(单位：%)

方法	Hulth2003			SemEval2010		
	P	*R*	*F*	*P*	*R*	*F*
SUM	22.28	16.76	19.13	10.89	9.32	10.04
AVG	26.90	19.31	22.48	18.71	13.53	15.70
ILP2015	28.85	21.83	24.85	21.26	15.11	17.67
DPILP	**31.73**	**23.22**	**26.82**	**24.28**	**16.92**	**19.94**

实验结果同样表明，本文提出的模型 DPILP 在两个数据集上比 ILP2015、SUM 和 AVG 方法都有很大的提高。在自动抽取 5 个和 10 个关键短语实验中，本文方法分别在 μ 为 0.06 和 0.04 时取得最好的结果。

结束语　本文提出整数线性规划模型 DPILP，该模型既解决了词本身的过度生成问题，又解决了语义过度生成问题，

(下转第 70 页)